

PENERAPAN METODE *FRAME DIFFERENCE* PADA SISTEM DETEKSI GERAKAN BERBASIS ANDROID

Dodik Rismawan Affrudin

Teknik Informatika, Institut Teknologi Nasional Malang
dodikra8@gmail.com

ABSTRAK

Banyaknya tindakan kriminalitas seperti pencurian dan perampokan pada setiap rumah. Hal tersebut membuat pemilik rumah tidak nyaman, selain itu tindakan kriminalitas pencurian dan perampokan sangat banyak yang pelakunya sulit ditemukan dikarenakan kurang kuatnya bukti yang ada. Tindakan kriminalitas tersebut dapat terjadi dikarenakan tingkat keamanannya yang lemah. Di era digital ini memang seharusnya ada sistem yang setidaknya dapat membantu tingkat keamanan dalam rumah. Sehingga pemilik rumah tidak was – was ketika dia tidak berada di rumah.

Sistem yang dibangun pada penelitian ini merupakan perangkat lunak pendeteksi gerakan berbasis desktop dengan menggunakan Visual Studio 2010 dan mobile dengan menggunakan Eclipse Luna yang dapat dioperasikan di sistem operasi Windows beserta aplikasi androidnya yang akan dioperasikan di smartphone. Sistem pendeteksi gerakan ini berjalan dengan model Client Server dimana untuk aplikasi yang bertindak sebagai servernya adalah aplikasi desktopnya kemudian untuk client adalah aplikasi yang berada pada smartphone android.

Hasil dari penelitian ini berupa perangkat lunak yang dapat membantu pihak berwajib dalam penyelidikan kasus pencurian, dimana aplikasi ini sudah dapat menangkap adanya sebuah pergerakan yang lewat pada kamera, serta aplikasi sudah mampu di integrasikan ke android dengan berupa notifikasi pemberitahuan kepada pemilik rumah dengan presentase keberhasilan 95%.

Kata kunci : *Motion Detection, Frame Difference, Background Substraction, Komputer Vision, Metode Motion Detection, Aforge.Net*

1. PENDAHULUAN

1.1. Latar Belakang

Sejauh ini, dari sisi keamanan yang dimiliki oleh sistem CCTV ternyata belum dapat mendeteksi adanya pergerakan yang telah tertangkap kamera yang kemudian data tersebut dikirimkan ke sisi aplikasi lain sebagai informasi. Berdasarkan wawancara kepada ibu Sri Utami selaku warga perumahan Riverside Kota Malang, sistem keamanan rumah yang memanfaatkan teknologi CCTV ternyata masih banyak kekurangan dari segi dokumentasi datanya. Dimana di era digital ini memang seharusnya ada sistem yang setidaknya dapat membantu tingkat keamanan. Sehingga dari segi dokumentasi data lebih efektif dan efisien. Penjelasan dari segi efektif pada penelitian ini adalah pada segi penyimpanan datanya tidak hanya disimpan pada satu tempat penyimpanan sehingga dokumentasi data lebih aman, misalnya ketika ada seseorang yang sengaja merusak CCTV tetapi data tetap ada dan tidak terhapus.

Salah satu implementasi sistem keamanan adalah menggunakan metode *Frame Difference* untuk mendeteksi adanya sebuah pergerakan. Metode ini dipakai karena sangat sensitif terhadap pergerakan. Dengan metode ini, maka akan mampu mendeteksi

gerakan yang lebih sensitif dari sistem yang dimiliki CCTV, dari permasalahan yang timbul, membuka mata penulis untuk terjun dan memberi gagasan dan implementasi dalam membongkar satu demi satu permasalahan itu, dan menurut penulis permasalahan itu timbul tidak lain dari kurangnya cara, kurangnya ide untuk menangani sistem yang menurut penulis kurang lengkap.

Oleh karena itu, penulis membuat sebuah program sistem keamanan dengan mendeteksi pergerakan yang ditangkap oleh kamera. Program yang direncanakan dapat merekam atau menangkap kejadian yang terjadi ketika pemilik rumah sedang tidak berada di rumah. Program ini merekam atau menangkap gerakan orang yang masuk ke dalam rumah seseorang yang hasil rekaman tersebut akan menjadi dokumentasi data pemilik rumah. .

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka permasalahan yang dibahas dalam penelitian ini adalah:

1. Bagaimana cara menerapkan metode *frame difference* pada *Csharp.Net* ?

2. Bagaimana proses implementasi metode *frame difference* untuk deteksi gerakan pada *Csharp.Net* ?

1.3. Batasan Masalah

Batasan masalah pada penelitian ini adalah :

1. Kelemahan dari sistem pada penelitian ini adalah pada kondisi cahaya yang minim sistem tidak dapat menemukan adanya pergerakan.
2. Program mengharuskan pemakaian *internet*.
3. Program hanya berjalan pada sistem operasi minimal *Windows XP* untuk *server*nya dan untuk *client*nya hanya dapat berjalan pada sistem operasi minimal *Android Ice Cream Sandwich* (API versi 14) .
4. Sistem pada penelitian ini belum dapat membedakan pergerakan manusia, hewan, atau benda.

1.4. Tujuan

Tujuan dari penelitian ini adalah sebagai berikut:

1. Dapat diterapkannya metode *frame difference* pada *CSharp.NET*.
2. Dapat diterapkannya system deteksi gerakan dengan metode *Frame Difference*

2. TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Kajian tentang deteksi termasuk dalam kajian *Computer Vision* jika pendeteksian dilakukan dengan menggunakan citra. Kamera merupakan salah satu media yang digunakan untuk menangkap citra suatu objek. *IP Camera* merupakan salah satu perangkat yang digunakan untuk mengetahui kondisi suatu area. *IP Camera* memiliki kemampuan untuk berkomunikasi langsung dengan jaringan internet. Oleh karena itu, dengan mengakses alamat *IP* dari sebuah *IP Camera*, maka pengguna bisa melihat area atau objek yang dipantau melalui *web browser*.

Pada umumnya, *IP Camera* telah dilengkapi dengan kemampuan untuk mendeteksi gerak. Pendeteksian gerak dikomunikasikan dengan menggunakan *protocol FTP*. Sehingga, setiap citra objek yang dianggap bergerak, dikirimkan ke *FTP server* melalui *protocol FTP*.

Proses ini tergolong mudah karena hanya membutuhkan beberapa pengaturan sederhana di sistem *IP Camera*. Akan tetapi pemanfaatan *FTP* ini tidak memungkinkan dilakukan pengaturan pola pemantauan dan deteksi gerak yang disesuaikan dengan kondisi area atau objek yang dipantau. Keadaan ini terjadi karena pendeteksian yang akan dirancang untuk keperluan pemantauan dan keamanan.

2.2. Motion Detection

Motion Detection adalah proses mendeteksi perubahan posisi dari suatu objek relatif terhadap sekitarnya atau perubahan lingkungan relatif terhadap suatu objek (S. Mishra, at.all, 2011). *Motion detection* dapat dicapai dengan metode mekanik dan elektronik.

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 (D. A. Migliore, at.all, 2006). Kemudian membandingkan warna yang terdapat pada *frame* saat ini dengan *frame* sebelumnya. Apabila terdapat perbedaan warna, maka objek tersebut terdeteksi sebagai gerakan (K. Suganya at.all, 2013).

2.3. Frame Difference

Metode ini dilakukan dengan membandingkan *frame - frame* citra yang ditangkap sesuai dengan urutan waktu. Penelitian lain yang dilakukan oleh Kenchannavar dkk, menjelaskan tentang algoritma yang diterapkan dalam metode *background subtraction* dan *frame differences* (D. A. Migliore, at.all, 2006). Penelitian yang dilakukan dengan menerapkan konsep SAD. SAD merupakan singkatan dari Sum of Absolute Difference. SAD inilah yang digunakan untuk menyatakan ada atau tidaknya pergerakan suatu pasang citra. Menurut penelitian Rosin dan Ellis, Algoritma *frame differences* dilakukan melalui 3 tahap.

Tahap pertama adalah proses penentuan citra referensi atau citra latar (*background*) (H. H. Kenchannavar, at.all, 2013). Tahap kedua adalah proses operasi aritmatika *subtraction*, dan yang terakhir pengaturan ambang batas (*threshold*). *Threshold* merupakan bagian penting untuk penentuan ketelitian dari pendeteksian gerak. Menurut penelitian, nilai *threshold* berbanding terbalik dengan jumlah noise yang dideteksi setelah dilakukan proses perbandingan (W. K. Pratt, 2001). Semakin kecil nilai *threshold* suatu algoritma *frame differences*, maka semakin besar jumlah noise yang terdeteksi. Penelitian yang dilakukan oleh Rosin dan Ellis melakukan pengaturan *threshold* dengan menggunakan *algorithm automatic threshold* (R. J. Radke, at.all, 2005)..

3. METODE PENELITIAN

3.1. Deskripsi Sistem

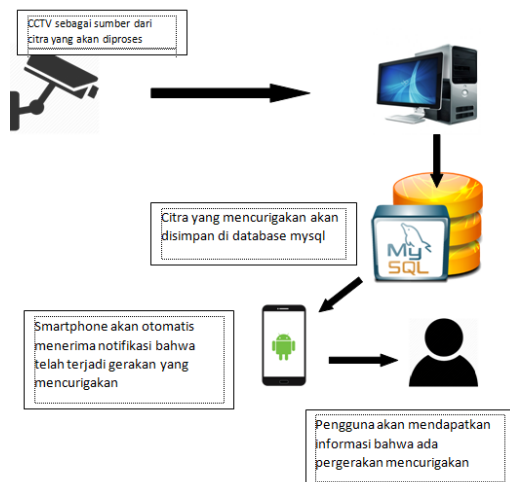
Sistem yang dibangun pada penelitian ini merupakan perangkat lunak pendeteksi gerakan berbasis *desktop* dan *mobile* yang dapat dioperasikan di sistem operasi *Windows* beserta aplikasi androidnya yang akan dioperasikan di *smartphone*. Sistem pendeteksi gerakan ini berjalan dengan model *Client Server* dimana untuk aplikasi yang bertindak sebagai *server*nya adalah aplikasi desktopnya

kemudian untuk *client* adalah aplikasi yang berada pada *smartphone android*.

Aplikasi ini dapat mendapatkan gambar yang dideteksi sebagai pergerakan secara real time dengan bantuan kamera yang tersambung dengan *server*, kemudian gambar yang telah dideteksi akan dikirimkan langsung ke aplikasi *smartphone*. Sistem yang ada pada *smartphone* akan memberikan Informasi berupa notifikasi hasil deteksi gerakan yang berupa gambar dari pergerakan sebuah objek yang ditangkap oleh *server*. Setiap pergerakan yang terdeteksi akan di simpan pada *computer server* dan *web service* yang sudah disediakan untuk perantara komunikasi antar *server* dan *client*.

3.2. Perancangan

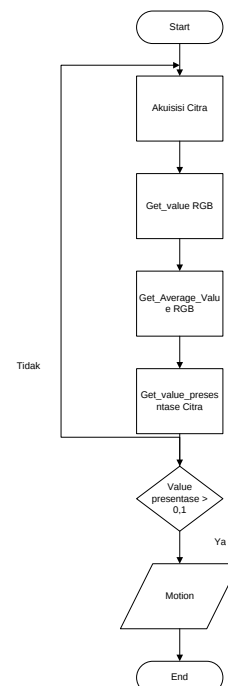
Teknologi yang akan dikembangkan nantinya akan menjadi dua aplikasi, dimana ada yang berperan sebagai pengirim data *motion* kemudian ada yang berperan sebagai penerima data. Seperti yang ditunjukkan pada Gambar 3.1.



Gambar 3.1 Teknologi yang hendak dikembangkan

3.3. Flowchart

Berikut merupakan *flowchart* dari *frame difference*, seperti yang ditunjukkan pada Gambar 3.2.



Gambar 3.2. Frame Difference

4. HASIL DAN PEMBAHASAN

4.1. Tampilan Login Server

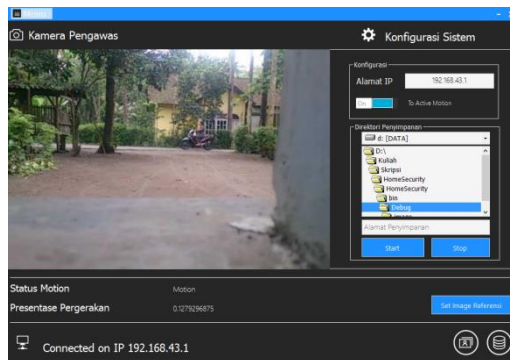
Pada bagian ini merupakan *interface* untuk *login* pengguna untuk melakukan proses pengaktifan sistem pendeteksi gerakan seperti yang ditunjukkan pada Gambar 4.1.

The screenshot shows a login window with a blue background. It contains a 'Username' field with a user icon, a 'Password' field with a lock icon and three dots for visibility, and an orange 'Login' button at the bottom.

Gambar 4.1. Login Server

4.2. Tampilan Menu Utama Server

Pada bagian ini merupakan *interface main menu system* dimana *menu* inilah yang bertugas memanajemen aktifitas dan fitur yang disediakan oleh sistem beserta berperan sebagai pengirim *motion* seperti yang ditunjukkan pada Gambar 4.2.



Gambar 4.1 Main Menu

4.3. Tampilan Record data Motion

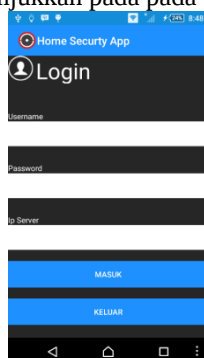
Pada bagian ini merupakan *interface* dari *history* pergerakan yang telah ditangkap oleh sistem pendeteksi pergerakan seperti yang ditunjukkan pada Gambar 4.3

ID	MOTION	PRESENTASE PERGERAKAN	WAKTU
23	Motion	1,06966145833333	19/08/2016 9:07
24	Motion	1,66731770833333	19/08/2016 9:07
26	Motion	2,00260416666667	19/08/2016 9:34
27	Motion	13,0755208333333	19/08/2016 9:34
29	Motion	15,8994140625	19/08/2016 9:41
30	Motion	15,974609375	19/08/2016 9:41
32	Motion	1,16145833333333	19/08/2016 9:45
33	Motion	3,47395833333333	19/08/2016 9:45
34	Motion	7,47786458333333	19/08/2016 9:45
35	Motion	12,0071614583333	19/08/2016 9:45
36	Motion	18,6162109375	19/08/2016 9:45

Gambar 4.3 History Motion

4.4. Tampilan Login Smartphone

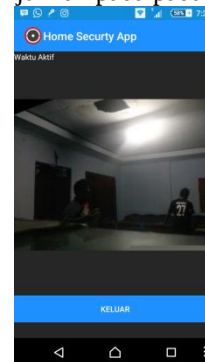
Pada bagian ini merupakan *interface* dari sisi *smartphone* yang dimana pengguna harus *login* aplikasi yang terletak pada *smartphone* jga untuk proses penerimaan data yang dikirim oleh *server* seperti yang ditunjukkan pada pada Gambar 4.4.



Gambar 4.4. Login Smartphone

4.5. Tampilan Main Smartphone

Pada bagian ini merupakan *interface* dari *main menu* yang ada di *smartphone* berfungsi sebagai melihat data terakhir yang ditangkap oleh *server*. seperti yang ditunjukkan pada pada Gambar 4.5.



Gambar 4.5 Last Motion

4.6. Pengujian Fungsionalitas

Berikut merupakan tabel pengujian fungsionalitas sistem pendeteksi pergerakan.

Tabel 4.1. Pengujian Fungsionalitas

Fungsi	Sukses	Gagal
Login	✓	
Connect IP Camera	✓	
Autocapture image when any motion	✓	
Get last image that capture system	✓	
Save image to local	✓	
Save image to server	✓	

4.7. Pengujian Sensifitas Gerakan Terdeteksi

Berikut merupakan tabel pengujian sensifitas sistem pendeteksi pergerakan dapat menangkap adanya pergerakan.

Tabel 4.2. Pengujian Sensifitas Gerakan Terdeteksi

Presentase (%)	Sukses	Gagal
0.1 – 1.0	✓	
1.1 – 10.0	✓	
11.0 – 20.0	✓	
21.0 – tak hingga		✓

4.8. Pengujian Jarak Gerakan Terdeteksi

Berikut merupakan tabel pengujian jarak dari sistem pendeteksi pergerakan dapat menangkap adanya pergerakan.

Tabel 4.3. Pengujian Jarak Gerakan Terdeteksi

Jarak (meter)	Sukses	Gagal
1 – 4	✓	
5 – 8	✓	
9 – 12	✓	
13 – 16	✓	
17 – 20	✓	
21 – ke atas		✓

4.9. Studi Kelayakan

Berikut merupakan tabel pengujian studi kelayakan aplikasi dari sistem pendeteksi pergerakan yang didapatkan dari kuisioner beberapa pengguna.

Tabel 4.4. Pengujian Studi Kelayakan Aplikasi

	Sesuai /Mudah	Kurang Sesuai / Lumayan	Tidak Sesuai / Sulit
Apakah <i>interface</i> aplikasi layak untuk digunakan ?	✓		
Apakah <i>interface</i> aplikasi mudah untuk dioperasikan ?	✓		
Apakah aplikasi sudah sesuai untuk mendeteksi adanya pergerakan ?	✓		
Apakah <i>interface</i> aplikasi menarik ?	✓		
Apakah aplikasi dapat membantu keamanan ?	✓		

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari hasil pengujian yang telah dilakukan didapatkan beberapa kesimpulan yaitu:

1. Dari hasil pengujian fungsionalitas sistem terlihat bahwa semua fungsi aplikasi dari sisi *client* maupun *server* dapat berjalan seperti yang diharapkan tanpa *error*.
2. Dari hasil pengujian sensitifitas penangkapan gambar berada diantara *range* hasil presentase 0.1 hingga 20 berdampak pada seberapa lama aplikasi mendeteksi adanya pergerakan pada *video* rekaman.
3. Dari hasil pengujian jarak pengambilan *video ip camera* terhadap sensitifitas aplikasi mendeteksi gerak yang terlihat bahwa sistem mampu mendeteksi dengan jarak maksimal 20 meter.

5.2. Saran

Aplikasi sistem pendeksi pergerakan ini pun masih memiliki beberapa keterbatasan, sehingga untuk itu penulis menyarankan untuk pengembangan aplikasi sistem pendeteksi pergerakan selanjutnya agar :

1. Tidak hanya menangkap gambar yang telah terdeteksi oleh *system*, akan tetapi dapat juga melakukan perekaman *video* secara *live*.

2. Proses deteksi gerakan dapat berjalan sesuai yang diharapkan di keadaan cahaya yang minim.
3. Di dalam sistem pendeteksi pergerakan ini dapat membedakan manusia atau benda dengan lebih spesifik.
4. Sistem dapat melakukan penjadwalan otomatis untuk mendeteksi pergerakan.

DAFTAR PUSTAKA

- [1] S. Mishra, P. Mishra, N. K. Chaudhary, P. Asthana, "A Novel Comprehensive Method for Real Time Video Motion Detection Surveillance," *International Journal of Scientific & Engineering Reserch* Volume 2, Issue 4, April-2011.
- [2] D. A. Migliore, M. Matteucci and M. Naccari, "Revaluation of Frame Difference in Fast and Robust Motion Detection, " - *Proceeding VSSN '06 Proceedings of the 4th ACM International Workshop on Video Surveillance and Sensor Networks*, 2006.
- [3] Aforge.net, "AForge Framework", 2012. Tersediadi: <http://www.aforogenet.com/aforge/framework>. (Dikutip 25 Frebuari 2014)
- [4] Spacek, L.A., Edge detection and motion detection. *Image and vision computing*, 1986. 4(1): p. 43-56.
- [5] H. H. Kenchannavar, G. S. Patkar and U. P. Kulkarni, "Simulink Model for Frame Difference and Background Subtraction Comparision in Visual Sensor Network," on 2010 The 3rd International Conference on Machine Vision (ICMV 2010).
- [6] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: a systematic survey," *IEEE Transactions on Image Processing*, vol. 14, no. 3, pp. 294–307, 2005.
- [7] K. Suganya Devi, N. Malmurugan, M. Manikandan, Object Motion Detection in Video Frames Using Background Frame Matching, *International Journal of Computer Trends and Technology*, Vol.4, Issue 6, pp 1928-1931, June 2013.
- [8] W. K. Pratt, *Digital Image Processing: PIKS Inside*, Third Edition 2001 . New York: John Wiley & Sons, pp. 82-83.