

# **APLIKASI PENDETEKSI OBJEK BERGERAK PADAIMAGE SEQUENCE DENGAN METODE BACKGROUND SUBTRACTION**

*Nurrita Nafidha Putri*

*Teknik Informatika, Universitas Gunadarma  
nurritanafidhaputri@gmail.com*

## **Abstrak**

Saat ini kebutuhan sistem monitoring meningkat dengan pesat diberbagai sektor. Dengan berkembangnya kemajuan ilmu pengetahuan dan teknologi sekarang ini semakin banyak sistem monitoring diterapkan untuk tujuan peningkatan aspek keamanan dan produktivitas. Hampir di setiap ruang publik seperti mall, stasiun, lalu-lintas, taman, jalanan, kantor, bank, dan bandara terpasang sistem monitoring. Sebuah sistem monitoring berfungsi sebagai alat identifikasi dan pelengkap sistem pengawas keamanan. Salah satunya yang mendapat perhatian cukup serius adalah pengolahan citra digital. Di bidang ini banyak citra yang dapat diolah antara lain mencakup pada sebuah gambar dan video juga. Salah satu hal pengolahan citra pada video yang berkembang dengan pesat saat ini adalah sistem pendeteksian objek. Deteksi objek merupakan sebuah proses pendeteksian objek yang bergerak atau berpindah dalam suatu video yang telah dirubah kedalam bentuk image sequence yaitu potongan gambar yang terurut dari sebuah video. Tujuan dari penelitian ini untuk mendeteksi objek yang bergerak dari potongan gambar (image sequence) yang terurut dari sebuah video dengan memanfaatkan teknologi pengolahan citra menggunakan MATLAB R2012a. Diharapkan dapat mempermudah proses deteksi objek yang akan dideteksi.

**Kata kunci** : Background Substraction, Image Sequence, Objek Bergerak, ObjectDetection, Sistem Monitoring

## **DESIGNING AN APPLICATION OF MOVING OBJECT DETECTION ON IMAGE SEQUENCE USING BACKGROUND SUBTRACTION**

### **Abstract**

The current monitoring system increases rapidly in various sectors. Recently, the development of advanced science and technology applies monitoring systems for the purpose of increasing the safety and productivity. Almost every public spaces has applied monitoring system, such as supermarkets, stations, traffic, parks, streets, offices, banks, and airports. A monitoring system serves as a complementary means of identification and security surveillance systems. One of the system uses digital image processing. In this system, a lot of images are processed, including a picture and video as well. One of the video images thriving today is the object detection system. Object detection is a moving object detection process in a video that has been converted into the form of image sequences. This has pieces of image sequences from a video. The purpose of this study is to detect moving objects from pieces of the image (image sequences). This is done by utilizing image processing technology using MATLAB R2012a. This result is expected to simplify the process of detection of the objects.

**Keywords:** Background Substraction, Object Detection, Image Sequence, Moving Object, Monitoring System.

## PENDAHULUAN

Kebutuhan sistem monitoring saat ini meningkat dengan pesat di berbagai sektor, hampir di setiap ruang publik seperti *mall*, stasiun, lalu-lintas, taman, jalanan, kantor, bank, dan bandara terpasang sistem monitoring. Sebuah sistem monitoring berfungsi sebagai alat identifikasi dan pelengkap sistem pengawasan keamanan. Ruang kontrol bisa memiliki banyak layar monitor sistem monitoring untuk pengawasan keamanan yang efektif agar pengawasan ini mampu mendeteksi suatu rencana kejahatan diperlukan fitur teknologi sistem monitoring yang dapat membantu mendeteksi hal ini [5].

Berkembangnya kemajuan ilmu pengetahuan dan teknologisekarang ini semakin banyak sistem monitoring diterapkan untuk tujuan peningkatan aspek keamanan dan produktivitas. Penerapan monitoring selalu berdasarkan pada kebutuhan pengawasan secara berkala dan merekam segala aktivitas yang berlangsung di lokasi tersebut dengan harapan ketika terjadi suatu hal kritis atau penting maka dapat segera diketahui dan ditangani. Perkembangan teknologi sudah sangat pesat dalam mencakup banyak bidang. Salah satunya yang mendapat perhatian cukup serius adalah pengolahan citra digital. Di bidang ini tidak hanya citra pada sebuah gambar yang diproses melainkan mencakup video juga. Salah satu hal pemrosesan pada video yang berkembang dengan pesat saat ini adalah Pendeteksian objek atau *object detection* [2].

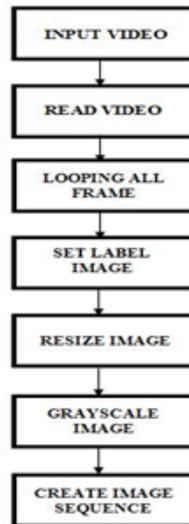
Pendeteksian objek merupakan proses mendeteksi objek yang bergerak dalam suatu video ataupun *image sequence*

yaitu potongan gambar yang terurut dari sebuah video. Pada tahun 1942 tercipta sebuah kreasi baru dari seorang insinyur asal Jerman, *Walter Bruch* dengan melakukan inovasi dari perangkat kamera dan perangkat televisi. Untuk pertama kalinya perangkat sistem teknologi CCTV digunakan. Nama CCTV sendiri adalah sebuah singkatan dari *Closed Circuit TeleVision* [3].

Pengolahan citra dapat diimplementasikan dalam kehidupan sehari-hari yaitu pada pelacakan objek yang bertujuan untuk mendeteksi atau melacak posisi dari objek yang bergerak dalam suatu *image sequence*. Dengan pendeteksian objek ini, diharapkan nantinya bisa memprediksikan dan menentukan posisi pada frame yang menunjukkan dimana objek berada. Manfaat dari pendeteksian objek yaitu mengidentifikasi perpindahan gerakan objek tersebut. Pemanfaatan teknologi *Computer Vision* dan *Image Processing* untuk melakukan pendeteksian objek bisa mempermudah manusia dalam mendeteksi objek, misalnya dibidang keamanan. Pelacakan objek bisa menggunakan metode *Background Subtraction*. Metode *Background Subtraction* dapat mendeteksi substraksi pada *background* dengan mengubah citra biner dan menentukan tingkat kepekaan perubahan piksel pada *background* [2].

## METODE PENELITIAN

Sebelum melakukan proses pendeteksian objek, video harus dirubah menjadi *image sequence*. Gambar 1 memperlihatkan tahapan video pada saat dirubah menjadi *image sequence*.



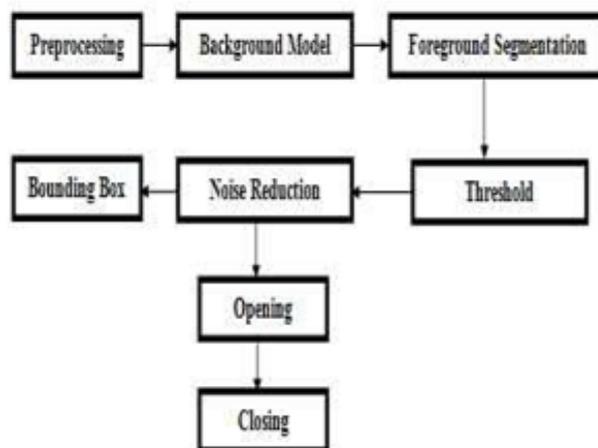
Gambar 1. Tahapan video menjadi *image sequence*

Gambar 1 dijelaskan untuk melakukan proses pendeteksian objek, video harus dirubah ke bentuk *image sequence* terlebih dahulu. Tahap pertama yang dilakukan untuk proses perubahan video menjadi *image sequence* diawali dengan penginputan video. Selanjutnya adalah proses *read video*, pada proses ini video akan dibaca tiap *frame*-nya. Setelah video dibaca, dilakukan *looping* terhadap semua *frame* pada video. Proses selanjutnya adalah memberikan label nama serta melakukan *resize*, dan *grayscale* terhadap video yang akan dirubah ke *image sequence*. Tahap terakhir adalah membuat *image sequence* dari video tersebut.

Jumlah *image sequence* dapat dihitung dari lamanya durasi video dikali *frame rate* video dengan:

$$JIS = Dv \times Fr \quad (1)$$

Dalam metodologi penelitian ini ada beberapa langkah-langkah umum dalam pelaksanaannya, yaitu proses *preprocessing*, *background model*, *foreground segmentation*, *threshold*, *noise reduction*, dan *bounding box* yang akan menghasilkan *output* data berupa hasil objek yang dideteksi. Tahapan gambaran pendeteksian objek secara umum dapat dilihat pada Gambar 2.



Gambar 2. Tahapan Pendeteksian Objek

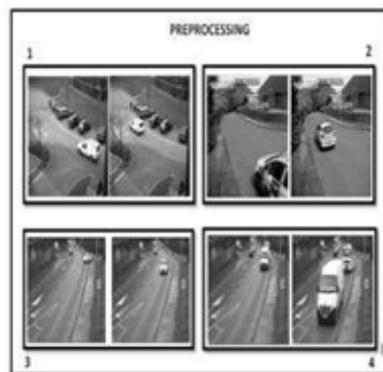
Pada Gambar 2 dijelaskan bahwa *bounding box* adalah proses terakhir dari alur pendeteksian objek yang bergerak pada *image sequence*. Berfungsi untuk menandai objek yang sudah dideteksi melalui 6 tahap yaitu *preprocessing*, *background model*, *foreground segmentation*, *threshold*, *noise reduction*, dan *bounding box*.

*Preprocessing* merupakan proses *image sequence* yang dapat dibaca dan dimengerti oleh komputer. Pada tahap ini sebelum melakukan *background modelling*, gambar yang akan digunakan terlebih dahulu harus dipanggil secara berurutan yang nantinya akan diproses dan didapatkan *background model*nya. Tampilan *image sequence* awal sebelum dilacak yang dipanggil dapat dilihat pada Gambar 3.

Langkah selanjutnya menentukan *background model* atau model latar. Model latar ini sangat penting dalam proses pelacakan objek, karena untuk

mendeteksi adanya gerakan perlu dipisahkan antara objek yang bergerak dan objek yang diam. Objek yang diam adalah latar atau *background* dari suatu *frame image*. Pepohonan, bangunan, manusia, tiang lampu, dan permukaan jalan merupakan bagian dari *background*. Dengan memisahkan latar dari *frame image* akan didapatkan objek yang bergerak atau *foreground*.

*Background model* diperoleh dengan cara mendapatkan nilai tengah dari semua *frame image* yang ada. Nilai tengah didapatkan dengan menggunakan fungsi median, median merupakan salah satu fungsi pada Matlab untuk pengoperasian matriks. Fungsi median akan mengembalikan nilai tengah dari *array*. Metode ini akan menghasilkan gambar latar yang merupakan pendekatan dengan cara mendapatkan nilai tengah dari semua *frame* yang ada. Berikut adalah tampilan model *background model* dapat dilihat pada Gambar 4.



Gambar 3. *Image Sequence* pada Tahap *Preprocessing*



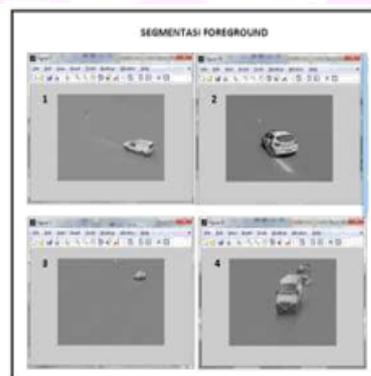
Gambar 4. *Background model*

Setelah memperoleh model latar, maka langkah selanjutnya yaitu *foreground segmentation*. Untuk mendapatkan *foreground* bisa dilakukan dengan cara mengurangi gambar awal dengan model latar. Proses ini dilakukan untuk mendapatkan *image foreground* yang merupakan tahap kedua dari metode *background subtraction*. Jika proses *foreground segmentation* berhasil maka akan didapatkan *image foreground* dari setiap frame yang diperlihatkan pada Gambar 5.

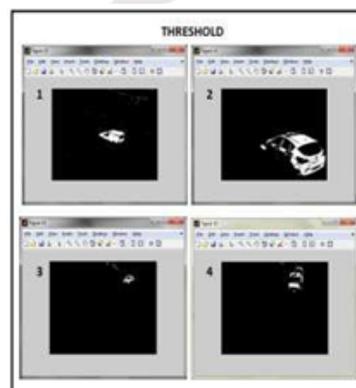
*Foreground* hasil segmentasi harus diubah ke citra biner agar dapat dilakukan proses *bounding box*. Setelah diperoleh hasil *foreground segmentation* maka dilakukan *thresholding* untuk memperoleh citrabiner. *Threshold* merupakan suatu prosedur untuk memisahkan bagian dari objek densitas rendah dengan yang tinggi. Pada tahap ini citra dari *foreground* diubah menjadi citra biner, biner sendiri dapat diartikan sebagai mengubah pixel dengan nilai intensitas pixel yang sama

atau lebih tinggi dari *threshold* menjadi pixel berwarna putih dan semua sisanya menjadi pixel warna hitam. Hasil *output* dari proses *threshold* diperlihatkan pada Gambar 6.

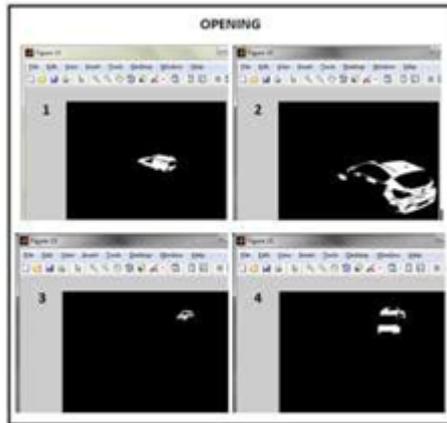
*Noise reduction* merupakan proses menghilangkan *noise* pada suatu *image*. *Noise* ini biasanya berbentuk titik-titik kecil, atau objek lain yang tidak diinginkan untuk diidentifikasi. Proses yang digunakan untuk menghilangkan *noise* menggunakan *morphological processing*. Operasi morfologi yang digunakan dalam penelitian ini yaitu proses *opening* dan *closing*. Langkah pertama yaitu dilakukan proses *opening* pada *binary image* untuk dihilangkan *noise*-nya. Proses *opening* ini berfungsi untuk menghilangkan objek-objek kecil yang mengganggu dan tidak ingin diidentifikasi dari *foreground image* yang telah diubah ke *binary image*. Gambar 7 memperlihatkan *noise* yang telah berhasil dihilangkan dari *image*.



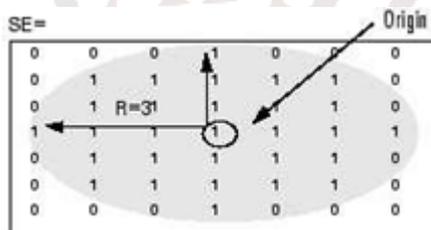
Gambar 5. Hasil Proses *Foreground Segmentation*



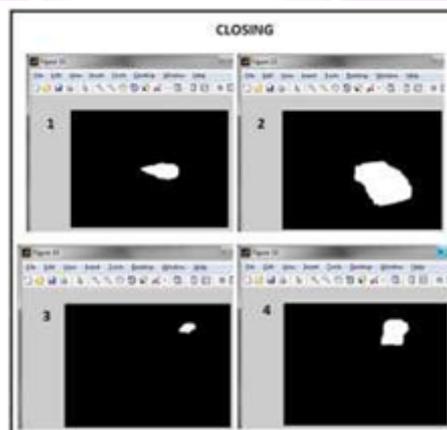
Gambar 6. Hasil proses *threshold*



Gambar 7. Hasil Pproses *Opening*



Gambar 8. *Strel Disk*



Gambar 9. Hasil dari proses *closing*

Langkah selanjutnya yaitu proses *closing*. Proses *closing* dilakukan untuk menghilangkan lubang kecil pada *image*. Agar objek yang ingin dilacak dapat diberikan *bounding box*, maka harus dilakukan proses *closing*, karena *bounding box* hanya akan bisa diimplementasikan jika objek yang terdapat pada *image* tidak terdapat lubang. Pada operasi *closing* ini

digunakan fungsi *strel disk*. *Strel disk* merupakan sebuah elemen yang terstruktur berbentuk cakram yang diperlihatkan pada Gambar 8. Gambar 9 memperlihatkan hasil citra yang telah dilakukan proses *closing*.

Proses terakhir dalam pendeteksian objek yaitu proses pemberian *boundingbox* pada objek yang akan

dideteksi. *Bounding box* merupakan persegi empat yang melingkupi suatu objek atau komponen yang terhubung dengan citra biner. *Bounding box* ini berfungsi untuk menandai objek yang sudah dideteksi. Proses *bounding box* ini menggunakan fungsi *regionprops*, *regionprops* merupakan salah satu *toolbox* pada matlab yang berfungsi untuk membuat *bounding box* pada citra. Ditunjukkan pada Gambar 10 *binary*

*image* yang sudah ditandai atau disensor oleh *bounding box*.

Untuk mempermudah pemahaman setiap program dalam sistem pendeteksian objek, dibuat diagram alur (*flowchart*) yang merepresentasikan proses aplikasi pendeteksian objek. Gambar 11 memperlihatkan *flowchart* yang merepresentasikan proses aplikasi pendeteksian objek.



Gambar 10. Hasil *Image* yang Telah Berhasil Disensor *Bounding Box*



Gambar 11. *Flowchart* PendeteksianObjek

Gambar 11 menjelaskan alur dari proses pendeteksian objek pada *image sequence* pada Matlab yang diawali dengan membaca *image sequence* ke dalam program. Kemudian dari *image sequence* tersebut didapatkan *background modelling*. Selanjutnya mendapatkan *foreground image* untuk membedakan antara *background* dan *foreground*. Proses ini sendiri digunakan untuk membedakan bagian yang menjadi *background modelling* dan bagian yang menjadi objek atau *foreground image* yang akan di proses lebih lanjut. Setelah proses membedakan antara *background* dan *foreground* selesai, dilakukan proses *opening* dan *closing* pada *image*.

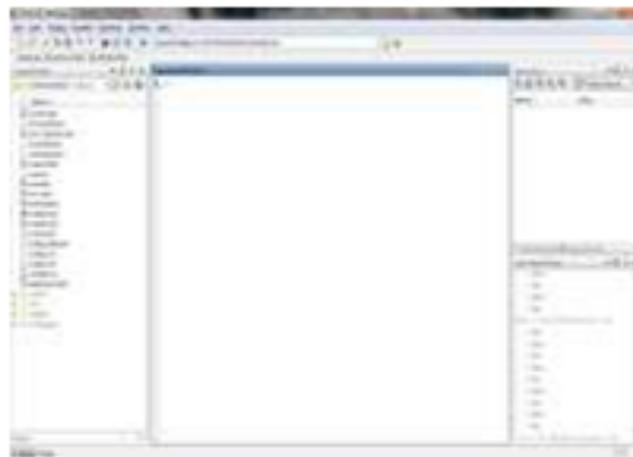
Selanjutnya dilakukan proses pembersihan citra dari *noise*. Apabila *noise* pada citra telah hilang dan bersih maka akan dilakukan langkah selanjutnya, tetapi jika masih terdapat *noise* pada citra maka akan kembali lagi ke proses *opening* dan *closing* pada *image*. Terakhir jika citra sudah bersih dari *noise* proses selanjutnya mendapatkan *bounding box*, yaitu pemberian *bounding box* pada objek. *Bounding box* merupakan persegi empat yang melingkupi suatu objek atau

komponen yang terhubung dengan *binary image* yang berfungsi untuk menandai objek yang sudah dideteksi.

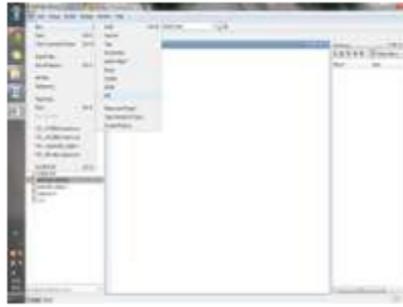
## HASIL DAN PEMBAHASAN

Pada bab ini akan membahas tentang langkah-langkah pembuatan antarmuka dan mengimplementasikannya sesuai dengan rancangan yang telah dibuat sebelumnya serta uji coba penelitian yang dilakukan. Untuk membuat tampilan antarmuka di Matlab dan menjalankan aplikasi dapat dilakukan dengan menggunakan GUI (*Graphical User Interface*). Setelah membuat fungsi-fungsi yang diperlukan, berikutnya adalah mengimplementasikannya sesuai dengan rancangan yang telah dibuat sebelumnya. Langkah selanjutnya adalah memilih menu *file* kemudian *new* dan pilih GUI seperti Gambar 13.

Jendela utama merupakan yang muncul pertama kali pada saat aplikasi ini dijalankan, jendela ini merupakan pusat dari kegiatan yang dilakukan pada aplikasi ini, rancangan GUI diperlihatkan pada Gambar 14.



Gambar 12. Kotak Dialog pada Matlab R2012a



Gambar 13. Pilihan Menu GUI pada Matlab



Gambar 14. Rancangan GUIJendela Utama

Dari Gambar 14 terlihat bahwa pada jendela utama ini terdapat 8 buah komponen yang terdiri dari 2 teks yaitu untuk menampilkan judul dan menampilkan teks hak cipta dari penulis, 1 buah axes untuk menampilkan citra *image sequence*, dan 4 button untuk proses pendeteksian objek serta 1 buah button *help* untuk menampilkan tata cara penggunaan aplikasi.

Aplikasi pendeteksi objek yang bergerak ini dilakukan beberapa ujian *testing* pada beberapa *sample* video yang didapat dari sumber video berikut:

- <https://www.youtube.com/watch?v=JLT1LHvmU7k>
- <https://www.youtube.com/watch?v=dk1cwXteAg8>

Data *testing* tersebut nanti yang akan digunakan sebagai parameter data untuk mendeteksi objek yang bergerak pada

saat aplikasi selesai di *compile*. Pengujian dilakukan untuk melihat apakah setiap objek yang bergerak pada *image sequence* dapat terdeteksi oleh *bounding box* saat aplikasi berjalan.

Setelah mengalami serangkaian perbaikan kode sumber program sesuai dengan kebutuhan dan layak untuk dijalankan secara *runtime*. Pertama adalah proses menjalankan video yang akan dipilih sebagai contoh pemrosesan secara GUI dengan cara menekan tombol *Play Video*. Pilih video yang akan dijalankan atau dilihat, maka akan tampil jendela baru yang menampilkan video yang dipilih.

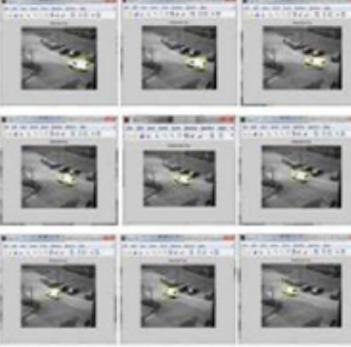
Sebagai contoh berikut ini salah satu hasil dari *image sequence* yang telah didapatkan dari video pada folder '1' dapat dilihat pada tabel 1.

Tabel 1. Hasil Image Sequence

Tabel 4.1 Hasil *image sequence*

Nama image	Image sequence
image001.jpg	

Tabel 4.2 Hasil deteksi objek bergerak

Video	Hasil Deteksi Objek
	
Keterangan	Objek mobil berhasil ditandai dengan bounding box secara keseluruhan

## SIMPULAN DAN SARAN

Kesimpulan dari penelitian ini adalah pendeteksian objek berhasil dilakukan pada objek yang bergerak dari *image sequence* menggunakan metode *background subtraction* pada sebuah video. *Image sequence* yang digunakan merupakan *image* tanpa warna atau *grayscale image* yang terurut. Objek yang dideteksi berupa kendaraan mobil yang bergerak. Dari keempat uji coba yang dilakukan, ada 3 video yang berhasil ditandai secara keseluruhan oleh *bounding box* tetapi *bounding box* belum dapat menandai objek mobil secara keseluruhan yaitu pada video ke-3, namun dari seluruh objek tersebut telah berhasil didapatkan area objeknya dan berhasil dideteksi.

Pengembangan dari aplikasi iniantara lain meliputi citra yang digunakan dapat memakai citra berwarna atau RGB, dapat menghitung jumlah objek yang bergerak sesuai dengan kriteria tertentu. Selain itu dapat mendeteksi lebih dari satu objek yang bergerak. Serta tiap objek yang terdeteksi dapat ditandai oleh *boundingbox*.

## DAFTAR PUSTAKA

[1] Abdul, Kadir, dan Susanto. 2013. Teori dan Aplikasi Pengolahan Citra, Andi.

[2] Darshak, Thakore, Himani, Parekh, dan Udesang, Jaliya. 2014. A Survey on Object Detection and Tracking Methods. International Journal of Innovative Research in Computer and Communication Engineering.

[3] Desire, Sidibe. 2013. Multi-sensors Fusion and Tracking Module an Introduction to Visual Tracking Methods. MSC VIBOT/Computer Vision.

[4] Eko, Prasetyo. 2011. Pengolahan Citra Digital dan Aplikasinya Menggunakan Matlab.

[4] Hitesh, Patel dan Darshak, Thakore. 2013. Moving Object Tracking Using Kalman Filer. International Journal of Computer Science and Mobile Computing.

[5] Kurniawan, Dwi, Gunawan, Ariyanto, dan Dedi, Ary. 2009. Motion Detection Using Opencv With Background Subtraction and Frame Differencing Technique. Universitas Muhammadiyah Surakarta.

[6] Mallikarjuna, Satyanarayana. Tracking System Using Approximate Median Filter, Kalman Filter and Dynamic Template Matching. I.J. Intelligent System and Applications. Massimo, Piccardi. 2004. Background subtraction techniques: a review. IEEE International Conference on Systems, Man and Cybernetics.

- [7] Sianipar, R.H. 2013. Pemrograman Matlab Dalam Contoh dan Penerapan. Informatika Bandung.
- [8] Tan, Yoong, Khong, Kow, dan Teo. 2011. Kalman Filtering Based Object Tracking in Surveillance Video System, Proceedings of the 3rd CUTSE Internasional Conference.
- [9] Vadivu, Tharanidevi dan Sethupathy. 2013. Moving Object Tracking Distance and Velocity Determination Based on Background Subtraction Algorithm. IOSR journal of Electronics and Communication Engineering.
- [10] <http://thesis.binus.ac.id/asli/Bab2/2008-1-00278-MTIF>. 2014, tanggal akses 02 Agustus 2015.
- [11] <https://www.youtube.com/watch?v=JLT1LHvmU7k> tanggal akses 05 Juli 2015.
- [12] <https://www.youtube.com/watch?v=dk1cwXteAg8> tanggal akses 05 Juli 2015
- [13] [www.mathworks.com](http://www.mathworks.com) tanggal akses 05 Juli 2015.

