

Analisis Performa Deteksi Objek Bergerak pada Algoritma *Background Subtraction* dan Algoritma *Frame Difference*

Dewi Anggraini Puspa Hapsari¹, Widya Khafa Nofa², Sugeng Santoso³

¹Program Studi Manajemen Universitas Gunadarma, ²Program Studi Sistem Informasi Universitas Gunadarma, ³Program Studi Teknik Informatika Universitas Raharja

E-mail: ¹dewi.anggraini.puspa@gmail.com, ²widyakhafa@gmail.com, ³sugeng.santoso@raharja.info

Abstrak

Penelitian ini bertujuan untuk membandingkan performa dari dua buah algoritma deteksi objek, yaitu algoritma *background subtraction* dan algoritma *frame difference*. Performa yang dibandingkan dari kedua algoritma ini adalah hasil akurasi deteksi objek dan waktu pemrosesan. Data diambil dari hasil akuisisi video kamera pengawasan. Langkah penelitian diawali dengan melakukan akuisisi video kemudian memecah menjadi *frame sequential* dan selanjutnya dilakukan deteksi objek pergerakan kamera. Hasil dari deteksi objek berupa sebuah *foreground mask* untuk masing-masing *frame sequential*. Dari analisis pengujian didapatkan untuk performa berdasarkan akurasi deteksi objek dan waktu pemrosesan dari dataset yang diambil menunjukkan jika algoritma *frame difference* lebih unggul dibandingkan algoritma *background subtraction*.

Kata Kunci— Deteksi Objek Bergerak, Algoritma *Background Subtraction*, Algoritma *Frame Difference*

Abstract

This study compares the performance of two object detection algorithms, namely the *background subtraction algorithm* and the *frame difference algorithm*. They reached the interpretation of these two algorithms due to object detection accuracy and processing time. The data is taken from the acquisition of video surveillance cameras. The research step begins with video acquisition, breaks it into sequential frames, and detects camera movement objects. The result of object detection is a *foreground mask* for each sequential frame. The test analysis obtained for performance based on object detection accuracy and processing time from the dataset taken shows that the *frame difference algorithm* is superior to the *background subtraction algorithm*.

Keywords— Object detection, *Background Subtraction Algorithm*, *Frame Difference Algorithm*

1. Pendahuluan

Sistem pengawasan (*video surveillance system*) mengidentifikasi area yang menarik dalam *scene* pada video, yaitu entitas yang bergerak melalui *scene* [1]. Berdasarkan KBBI, bergerak adalah peralihan tempat atau kedudukan, baik hanya sekali maupun berkali-kali. Jadi objek bergerak adalah objek yang berpindah dari tempat atau kedudukan atau tidak diam saja. Sedangkan didalam video, bergerak dapat didefinisikan sebagai perubahan nilai intensitas citra dimana obyek yang diam adalah latar atau *background* dari suatu *frame* pada video dan obyek bergerak adalah *foreground* [2].

Deteksi objek bergerak (*motion detection*) merupakan bagian penting dari sistem pengawasan, karena sistem selanjutnya menganalisis kegiatan dari benda bergerak tersebut. Kegiatan yang dianalisis akan mengklasifikasikan apakah objek bergerak tersebut adalah kendaraan, orang atau sekelompok orang.

Deteksi objek memungkinkan kamera pengawasan untuk melacak cuplikan *scene* dari lokasi tertentu. Metode ini secara akurat mengenali dan menempatkan beberapa contoh objek tertentu dalam sebuah video. Saat objek bergerak melalui *scene* tertentu atau melintasi *frame* tertentu, sistem menyimpan informasi.

Deteksi objek bergerak merupakan proses untuk dilakukan pemisahan antara *frame background* dan *frame foreground*. Pemisahan ini dilakukan untuk menghasilkan *foreground mask*. *Foreground mask* yang dihasilkan dijadikan tahap awal untuk melakukan pemrosesan video digital.

Beberapa penelitian telah dilakukan dalam deteksi pergerakan objek dan mengimplementasikan ke dalam algoritma deteksi yang berbeda. Algoritma yang umumnya digunakan adalah algoritma *background subtraction* [3] dan *algoritma frame difference* [4]. Algoritma *background subtraction* menggunakan teknik pengurangan (selisih) antara *frame* sekarang (f_i) dengan *frame background*-nya [5]. Sedangkan algoritma *frame difference* menggunakan teknik pengurangan (selisih) antara *frame* sekarang (f_i) dengan *frame* sebelumnya (f_{i-1}) [6].

Performa dari sebuah algoritma ditentukan dari akurasi deteksi objek dan waktu pemrosesan. Hasil dari perbandingan performa ini digunakan sebagai acuan untuk menerapkan algoritma deteksi objek pada dataset.

Berdasarkan uraian tersebut maka penelitian ini bertujuan untuk menganalisis performa perbandingan algoritma deteksi objek yang bergerak pada algoritma *background subtraction* dengan algoritma *frame difference*. Adapun performa yang akan dianalisis adalah hasil akurasi deteksi objek dan waktu pemrosesan algoritma tersebut yang diimplementasikan pada hasil akuisisi *video surveillance system*.

2. Metode Penelitian

Metode yang digunakan pada penelitian ini terdiri dari 3 (tiga) tahapan, yaitu:

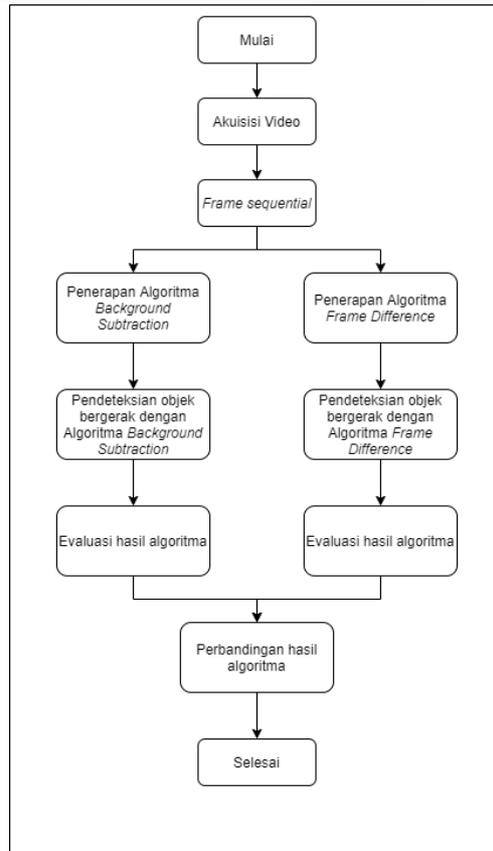
1. Studi literatur. Tujuan dari tahap ini adalah mendapatkan pijakan dan/atau fondasi untuk memperoleh landasan teori, kerangka berfikir dan alat penelitian yang akan digunakan untuk melakukan analisis perbandingan dari algoritma deteksi objek bergerak. Sumber dari literatur yang digunakan adalah jurnal, artikel ilmiah dan modul atau referensi lain yang terpercaya.
2. Desain dari algoritma. Tahap ini merupakan tahap desain dari algoritma deteksi objek bergerak yang dikembangkan menggunakan metode SDLC (*Software Development Life Cycle*) dengan model *Waterfall* yaitu: analisa, rancangan, pengkodean, dan implementasi. Penerapan metode SDLC dilakukan pada kedua algoritma.
3. Implementasi. Tahap ini dengan menggunakan perangkat lunak (*software*) MATLAB akan mengimplementasikan hasil desain dari algoritma yang telah dibuat. Selanjutnya dilakukan analisis pengujian hasil dari implementasi.

Berdasarkan dari studi pustaka dengan topik deteksi objek bergerak, algoritma *background subtraction* dan algoritma *frame difference* maka terbentuklah alur tahap penelitian untuk penelitian ini yang digambarkan pada Gambar 1.

Tahap penelitian diawali dengan melakukan akuisisi video. Akuisisi video bertujuan untuk mengumpulkan data video yang didapatkan dengan *capture* dengan menggunakan kamera pengawasan. Dalam melakukan proses akuisisi video, ada beberapa faktor yang perlu menjadi perhatian utama dalam prosesnya. Faktor tersebut adalah jenis alat akuisisi, resolusi kamera dan, teknik pencahayaan, teknik zooming (perbesar dan perkecil kamera) dan sudut pengambilan data [7].

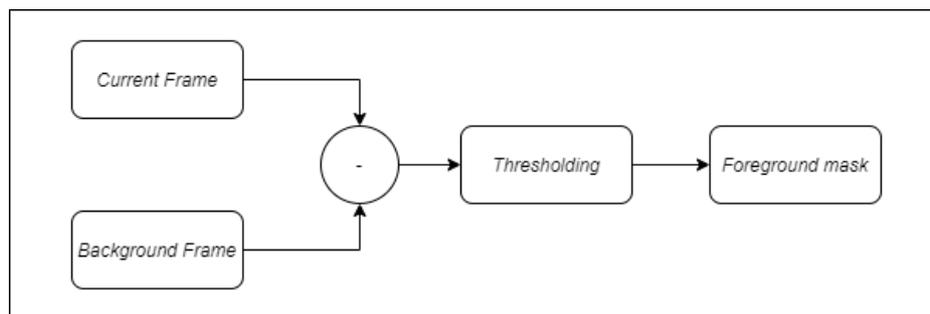
Dataset yang dihasilkan dari akuisisi video, selanjutnya akan dipecah terlebih dahulu menjadi *frame sequential*. Setiap *frame* yang dihasilkan hanya akan diambil komponen RGB saja dikarenakan komponen RGB memuat informasi dari dataset.

Setiap *frame* dengan komponen RGB akan diterapkan dua buah algoritma deteksi objek bergerak, yaitu algoritma *background subtraction* dan algoritma *frame difference*. Penerapan masing-masing algoritma pada *frame sequential* akan dilakukan di evaluasi. Evaluasi dilakukan dengan melihat pada dari hasil akurasi pendeteksian dan waktu pemrosesan algoritma saat diimplementasikan pada perangkat lunak MATLAB. Tahap akhir dari proses ini adalah melakukan analisis perbandingan dari hasil kedua algoritma tersebut.



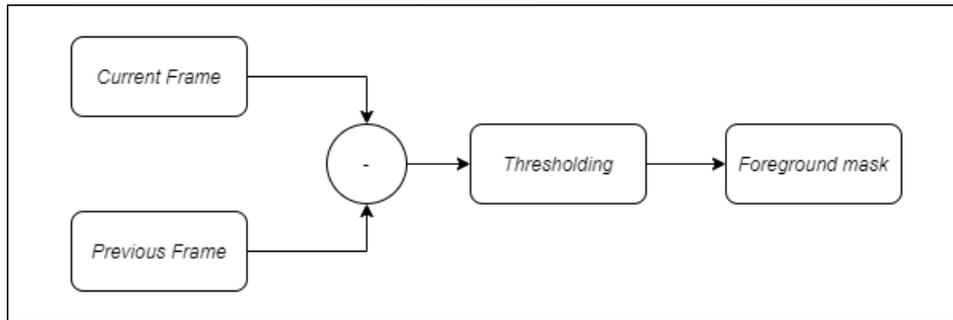
Gambar 1. Alur tahap penelitian

Algoritma *background subtraction* menggunakan teknik pengurangan (selisih) antara *frame* sekarang (f_i) dengan *frame background*-nya [5]. *Frame background* diambil dari *frame* awal pada data video. *Frame* ini yang akan disimpan terlebih dahulu untuk dijadikan pembandingan untuk setiap *frame* selanjutnya (f_i). *Foreground mask* dihasilkan setelah proses pengurangan dilakukan proses *thresholding*, dimana selisih *frame* yang diambil jika hasil selisih lebih besar dari *thresholding level*. Proses pengurangan dilakukan dengan mengurangi dengan *frame background* hingga *frame* habis. Gambar 2 merupakan gambar yang mengilustrasikan algoritma *background subtraction*.



Gambar 2. Algoritma *background subtraction*

Algoritma *frame difference* menggunakan teknik pengurangan (selisih) antara *frame* sekarang (f_i) dengan *frame* sebelumnya (f_{i-1}) [6]. Pada algoritma ini, selisih tidak dilakukan pada *frame* latar (*background*) melainkan pada *frame* sebelumnya sehingga tidak ada pendefinisian untuk *frame* awal.



Gambar 3. Algoritma *frame difference*

3. Hasil dan Pembahasan

3.1. Akuisisi Video

Akuisisi video merupakan proses atau kegiatan untuk melakukan pengambilan data video yang akan digunakan pada penelitian ini. Data video diambil dengan menggunakan kamera *fix outdoor* yang diletakkan didalam ruangan.



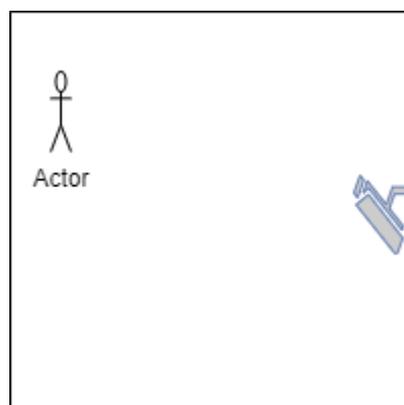
Gambar 4. Contoh kamera *fix indoor-outdoor*

Spesifikasi kamera yang digunakan seperti pada Tabel 1.

Tabel 1. Spesifikasi kamera yang digunakan

	Kamera
Jenis	Kamera <i>fix outdoor</i>
Resolusi Kamera	2 MP
Ukuran Citra	640 x 480
Frame rate	25 fps

Ketinggian kamera diletakkan setinggi 2 meter dari permukaan tanah. Ketinggian ini disesuaikan untuk mendapatkan *viewfinder* yang cukup luas. Pengambilan data dilakukan pada malam hari sehingga membutuhkan bantuan cahaya buatan (*lighting*). Tata letak kamera seperti terlihat pada Gambar 5.



Gambar 4. Tata letak pengambilan data

Objek yang digunakan dalam penelitian ini adalah manusia yang bergerak. Kegiatan pergerakan yang dilakukan adalah dengan berdiri, jalan, duduk dan berbaring. Kegiatan ini dilakukan secara berkesinambungan selama berdurasi 6 detik.

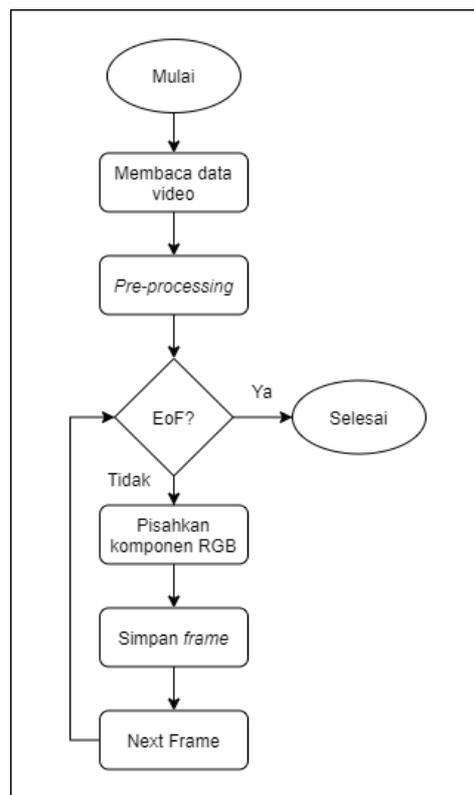
Spesifikasi perangkat keras (*hardware*) dan perangkat lunak (*software*) yang digunakan untuk melakukan implementasi dan pengujian tertuang pada Tabel 2.

Tabel 2. Spesifikasi perangkat keras (*hardware*) dan perangkat lunak (*software*) yang digunakan

Perangkat keras (<i>hardware</i>)		Perangkat lunak (<i>software</i>)
<i>Processor</i>	Intel(R) Core(TM) i7-11800H	MATLAB
<i>Memory</i>	16.0 GB	
<i>VGA</i>	Geforce RTX	

3.2. Pemisahan *frame*

Data video hasil akuisisi akan dipecah menjadi *frame* yang berdiri sendiri. Proses pemisahan diawali dengan membaca data video. Dari data video tersebut, dilakukan proses *pre-processing* yang merupakan proses untuk membaca informasi dari data video, yaitu panjang (*length*) *frame*. Setiap *frame* akan diambil komponen R, G dan B untuk diterapkan kedua algoritma. Gambar 4 menunjukkan alur proses pemisahan *frame* pada data video.



Gambar 4. Alur proses pemisahan *frame*

Gambar 5 menunjukkan *frame background* dari dataset. *Frame background* didapatkan pada awal *frame* dikarenakan kamera yang digunakan tidak bergerak atau *static*. *Frame background* ini menjadi *frame* latar dari video.



Gambar 5. *Frame background* dari dataset

Hasil pemisahan *frame* pada dataset adalah 120 *frame*. Setiap *frame* akan disimpan dalam format citra bmp. Contoh hasil pemisahan *frame* yang berhasil dilakukan terlihat pada Tabel 3.

Tabel 3. Contoh hasil pemisahan *frame*

Contoh hasil pemisahan <i>frame</i>		
 Frame ke-11	 Frame ke-12	 Frame ke-13
 Frame ke-14	 Frame ke-15	 Frame ke-16
 Frame ke-17	 Frame ke-18	 Frame ke-19

3.3. Algoritma *Background Subtraction*

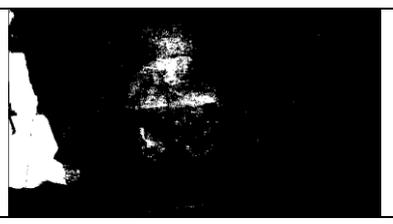
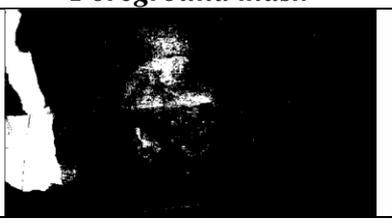
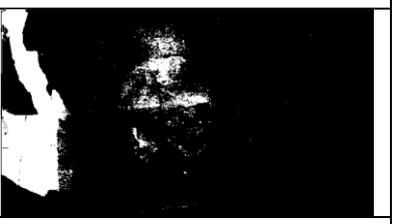
Frame dari data yang sudah dipecah akan diterapkan dengan algoritma *background subtraction*. Berdasarkan pada Gambar 2 yang menunjukkan langkah penerapan algoritma *background subtraction*, maka proses pertama adalah dengan membaca *frame* pertama sebagai *frame background*. Semua *frame* tersebut dirubah terlebih dahulu menjadi citra *grey-scale*. Selanjutnya dilakukan proses pembacaan *frame* berikutnya dimana *frame* tersebut akan dilakukan proses pengurangan (*subtraction*) dengan *frame*

background. Hasil pengurangan akan dilakukan pengecekan *thresholding* yang digunakan sebagai batas dari *frame* tersebut. Proses ini akan dilakukan hingga *frame* terakhir.

Contoh hasil dari penerapan algoritma *background subtraction* yang menghasilkan *foreground mask* diilustrasikan pada Tabel 4. Penerapan algoritma *background subtraction* dilakukan untuk 120 *frame* dari data video. *Thresholding level* dari data ini adalah 0.113725490196078.

Berdasarkan Tabel 4, terlihat jika algoritma *background subtraction* berhasil diterapkan hingga menghasilkan *foreground mask* untuk setiap *frame* yang telah dipisahkan. Akurasi penerapan algoritma masih kurang baik dikarenakan terdapat *noise* yang ditangkap oleh algoritma sebagai *foreground mask*. Waktu pemrosesan untuk implementasi algoritma ini adalah 1.888894 detik.

Tabel 4. Contoh *foreground mask* hasil algoritma *background subtraction*

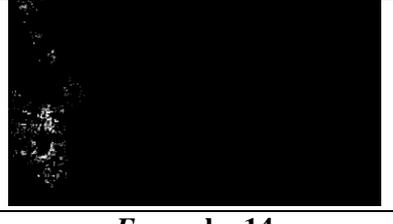
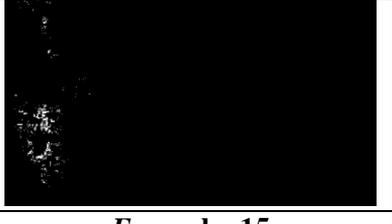
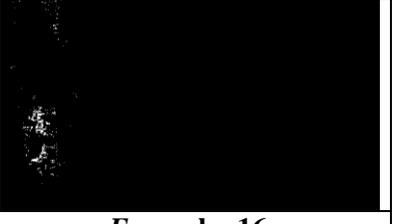
Frame dari data video		
		
Frame ke-11	Frame ke-12	Frame ke-13
		
Frame ke-14	Frame ke-15	Frame ke-16
Foreground mask		
		
Frame ke-11	Frame ke-12	Frame ke-13
		
Frame ke-14	Frame ke-15	Frame ke-16

3.4. Algoritma *Frame Difference*

Seperti halnya pada algoritma *background subtraction*, pada algoritma *frame difference* pun diterapkan setelah proses pemisahan *frame*. Berdasarkan Gambar 3 yang menunjukkan proses penerapan algoritma *frame difference* maka langkah awal adalah dengan membandingkan *frame* pertama dengan *frame* selanjutnya. Kemudian menghitung selisih dari kedua *frame* tersebut. Selisih

tersebut akan akan membentuk foreground mask menyesuaikan dengan *thresholding level* yang dibentuk 0.027450980392157.

Tabel 5. Contoh *foreground mask* hasil algoritma *frame difference*

Frame dari data video		
		
Frame ke-11	Frame ke-12	Frame ke-13
		
Frame ke-14	Frame ke-15	Frame ke-16
Foreground mask		
		
Frame ke-11	Frame ke-12	Frame ke-13
		
Frame ke-14	Frame ke-15	Frame ke-16

Berdasarkan Tabel 5, terlihat jika algoritma *frame difference* berhasil diterapkan hingga menghasilkan *foreground mask*. *Foreground mask* yang dihasilkan tepat mengikuti pada pergerakan objek. Tidak terdapat *noise* pada hasil *foreground mask*. Waktu pemrosesan untuk implementasi algoritma adalah 0.799775 detik.

3.5. Analisis Pengujian

Penerapan deteksi objek dengan menggunakan algoritma *background subtraction* dan algoritma *frame difference* berhasil dilakukan dan menghasilkan *foreground mask* seperti tertuang pada Tabel 4 dan Tabel 5. Tabel analisis pengujian antara kedua algoritma dengan melihat dari akurasi pendeteksian objek dan waktu pemrosesan tertuang pada Tabel 6.

Tabel 6. Analisis Pengujian

	Algoritma <i>Background Subtraction</i>	Algoritma <i>Frame Difference</i>
Deteksi objek bergerak	<ul style="list-style-type: none"> ✓ Akurasi deteksi objek kurang baik. ✓ Berhasil mendeteksi semua objek tetapi terdapat <i>noise</i> yang dianggap sebagai <i>foreground mask</i>. 	<ul style="list-style-type: none"> ✓ Akurasi deteksi objek baik. ✓ Berhasil mendeteksi semua objek dan tidak terdapat <i>noise</i>.
Thresholding level	0.113725490196078	0.027450980392157
Waktu pemrosesan	1.888894 detik	0.799775 detik.

4. Kesimpulan

Deteksi objek bergerak untuk menghasilkan *foreground mask* merupakan langkah awal dalam pemrosesan data video. *Foreground mask* digunakan untuk memisahkan area *background* dan *foreground*. Kedua algoritma yang digunakan berhasil diterapkan dan menghasilkan *foreground mask*.

Berdasarkan Tabel 6, performa algoritma *frame difference* lebih baik dibandingkan dengan performa dari algoritma *background subtraction*. Pada algoritma *background subtraction* masih terdapat *noise* yang dianggap sebagai *foreground mask*. Waktu pemrosesan yang diperlukan oleh algoritma *frame difference* dalam mengolah data video dengan jumlah 120 *frame* lebih cepat jika dibandingkan dengan algoritma *background subtraction*.

5. Saran

- Objek yang diteliti pada penelitian ini adalah satu orang objek yang bergerak sehingga dapat dimungkinkan jika melakukan dengan dataset menggunakan lebih dari satu objek atau kerumunan (*crowd*).
- Pada algoritma *background subtraction* masih terdapat *noise* yang ditangkap oleh algoritma sebagai *foreground mask* sehingga masih perlu ditambahkan *filter* untuk melakukan proses menghilangkan *noise* tersebut.
- Perbandingan performa berdasarkan hasil akurasi pendeteksian objek dan waktu pemrosesan, dimungkinkan untuk membandingkan kompleksitas dari algoritma.

Daftar Pustaka

- [1] L. A. Mushawwir and I. Supriana, “Deteksi dan Tracking Objek untuk Sistem Pengawasan Citra Bergerak,” *Konf. Nas. Inform. 2015 Deteksi*, vol. 2354–645X/, no. October, pp. 1–10, 2015.
- [2] P. Hartoto, J. T. Elektro, I. Teknologi, and S. Nopember, “Sistem Deteksi Kecepatan Kendaraan Bermotor pada Real Time Traffic Information System.”
- [3] W. Supriyatin, Y. Rafsyam, and Jonifan, “Analisis Pelacakan Objek Menggunakan Background Estimation Pada Kamera Diam Dan Bergerak,” *Orbith Maj. Ilm. Pengemb. Rekayasa dan Sos.*, vol. 13, no. 2, pp. 124–130, 2017.
- [4] R. Muhamad, T. Yulianti, S. R. Sulistiyanti, S. Purwiyanti, and F. X. A. Setyawan, “Deteksi Objek Bergerak Pada Video Bawah Air Menggunakan Metode Frame Differencing,” *J. EECCIS*, vol. 13, no. 2, pp. 100–104, 2019.
- [5] A. I. Iskandar, “Analisis Objek Bergerak Dengan Metode Background Subtraction,” *Celeb.*

- Eng. J.*, vol. 1, no. April, pp. 41–53, 2019.
- [6] S. Saluky, “Moving Object Detection on CCTV Surveillance Using the Frame Difference Method,” *ITEJ (Information Technol. Eng. Journals)*, vol. 4, no. 2, pp. 114–122, 2019, doi: 10.24235/itej.v4i2.52.
- [7] E. Tjin, *Kamera DSLR Itu Mudah!* Bukune, 2011.