

**ESTIMASI KECEPATAN OBYEK BERGERAK BERBASIS
VIDEO PROCESSING MENGGUNAKAN METODE *OPTICAL
FLOW HORN-SCHUNCK***

SKRIPSI

Disusun untuk melengkapi syarat-syarat guna memperoleh gelar Sarjana Sains



**Diajukan Oleh:
NOVIA ANGELITA
3225122054**

**PROGRAM STUDI FISIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI JAKARTA
2016**

PERSETUJUAN PANITIA UJIAN SKRIPSI

Estimasi Kecepatan Obyek Bergerak Berbasis *Video Processing* Menggunakan
Metode *Optical Flow Horn-Schunck*

Nama : Novia Angelita

No. Reg : 3225122054

	Nama	Tanda tangan	Tanggal
Penanggung Jawab			
Dekan	: Prof. Dr. Suyono, M.Si NIP. 19671218 199303 1 005
Wakil Penanggung Jawab			
Pembantu Dekan I	: Dr. Muktiningsih, M.Si NIP. 19640511 198903 2 001
Ketua	: Dr. Anggara Budi Susila, M.Si NIP. 19601001 199203 1001
Sekretaris	: Riser Fahdiran, M.Si NIP. 19830717 200912 1008
Anggota			
Pembimbing I	: Dr. Bambang Heru Iswanto, M.Si NIP. 19680401 199403 1 002
Pembimbing II	: Dr. Mutia Delina, M.Si NIP. 19801119 200801 2 007
Penguji Ahli	: Dr. Sunaryo, M.Si NIP. 19550303 198703 1 002

Dinyatakan lulus ujian skripsi tanggal 27 Juli 2016

SURAT PERNYATAAN KEASLIAN SKRIPSI

Dengan ini, saya yang bertandatangan di bawah ini, mahasiswa Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta:

Nama : Novia Angelita

No. Reg : 3225122054

Program Studi : Fisika

Menyatakan bahwa skripsi yang saya buat dengan judul **“Estimasi Kecepatan Obyek Bergerak Berbasis *Video Processing* Menggunakan Metode *Optical Flow Horn-Schunck*”**, adalah:

1. Dibuat dan diselesaikan oleh saya sendiri, berdasarkan data yang diperoleh dari hasil penelitian pada bulan Maret – Juni 2016.
2. Bukan merupakan duplikat skripsi yang pernah dibuat oleh orang lain atau jiplakan karya tulis orang lain dan bukan terjemahan karya tulis orang lain.

Pernyataan ini saya buat dengan sesungguhnya dan saya bersedia menanggung segala akibat yang timbul jika pernyataan saya ini tidak benar.

Jakarta, Juli 2016

Yang membuat pernyataan

(Materai 6000)

Novia Angelita

MOTTO DAN PERSEMBAHAN

MOTTO

“Just be who actually you are, no matter what people think about you”

*“Learning gives creativity,
Creativity leads to thinking,
Thinking provides knowledge,
Knowledge makes you great.”*

- Abdul Kalam

*“A man without education, is
like a building without
foundation”*

“The best revenge is bettering yourself” – Zacky Vengeance

PERSEMBAHAN

*Seluruh kerja keras dalam penyelesaian karya ini saya
persembahkan untuk:*

*“Kedua orang tua tercinta, Ibu Maryam dan Bapak Herman
Sagata. Mama dan Papa telah berjuang dalam mengurus
dan mendidik puteri sulung ini sehingga bisa menyelesaikan
pendidikan tinggi. Mama dan Papa yang selalu
mengorbankan apapun demi pendidikan anak-anaknya.
Terima kasih kepada Mama dan Papa atas kasih sayang dan
perhatian tak terhingga yang sedari dulu telah diberikan,
semoga apa yang Pipi kerjakan ini dapat membuat Mama
dan Papa bangga...”*

ABSTRAK

Novia Angelita. Estimasi Kecepatan Obyek Bergerak Berbasis *Video Processing* Menggunakan Metode *Optical Flow Horn-Schunck*. Skripsi. Jakarta: Program Studi Fisika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta, 2016.

Berkendara melewati ambang batas kecepatan yang ditentukan merupakan salah satu penyebab dalam kecelakaan lalu lintas. Untuk mengurangi angka kecelakaan, dalam penelitian ini telah dibuat sebuah sistem untuk mengestimasi kecepatan obyek bergerak secara otomatis yang berbasis *video processing*. Penelitian ini mensimulasikan *video surveillance system* yang dikombinasikan dengan *computer vision* untuk menginterpretasikan informasi yang terdapat pada obyek yang diamati. Metode yang digunakan dalam penelitian ini adalah *Foreground Detector* berbasis *Gaussian Mixture Models* (GMM) untuk deteksi obyek bergerak dan *Optical Flow Horn-Schunck* untuk estimasi kecepatannya. Penelitian ini merekam sepeda motor yang bergerak dengan variasi kecepatan konstan, yaitu 20 km/jam, 40 km/jam dan 60 km/jam, masing-masing diulangi sebanyak 5 kali. Hasil estimasi kecepatan secara komputasi kemudian dibandingkan dengan kecepatan aktual sepeda motor. Perbandingan tersebut menunjukkan bahwa rata-rata akurasi adalah 85,2% dengan akurasi tertinggi 94,1% pada 60 km/jam dan akurasi terendah 79,8% pada 40 km/jam. Penelitian ini menunjukkan bahwa metode *Optical Flow Horn-Schunck* dapat diaplikasikan untuk mengestimasi kecepatan obyek bergerak secara otomatis tanpa mengeluarkan biaya tinggi.

Kata kunci: *kecepatan obyek bergerak, Optical Flow, Horn-Schunck*

ABSTRACT

Novia Angelita. *Estimating Moving Object Speed Based on Video Processing Using Optical Flow Horn-Schunck. Minithesis. Jakarta: Study Program of Physics, Faculty of Mathematics and Natural Sciences, State University of Jakarta, 2016.*

Driving past the speed limit specified is one of the common causes in traffic accidents. To reduce the number of accidents, in this research we have created a system to estimate a moving object speed based on video processing. This research simulated video surveillance system that was combined to computer vision to interpret the information about the observed object. The methods used in this research were Foreground Detector based Gaussian Mixture Models (GMM) to detect a moving object and Optical Flow Horn-Schunck to estimate its speed. This research recorded a moving motorcycle with various speed of 20 km/h, 40 km/h and 60 km/h, each speed was repeated five times. The results of the speed estimation computationally were then compared to the actual speed of the motorcycle. The comparison showed that the average accuracy was 85.2% with the highest accuracy was 94.1% at the speed of 60 km/h and the lowest accuracy was 79.8% at the speed of 40 km/h. This research showed that the Optical Flow Horn-Schunck method can be applied to estimate moving object speed automatically without high-cost.

Keywords: *moving object speed, Optical Flow, Horn-Schunck*

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Allah SWT yang telah senantiasa memberikan rahmat dan nikmat-Nya sehingga Skripsi yang berjudul **“Estimasi Kecepatan Obyek Bergerak Berbasis *Video Processing* Menggunakan Metode *Optical Flow Horn-Schunck*”** dapat diselesaikan dengan baik.

Penulis menyadari bahwa terselesainya penyusunan proposal ini tidak terlepas dari bantuan dan dukungan dari berbagai pihak. Dalam kesempatan ini, penulis ingin menyampaikan rasa terima kasih kepada:

1. Bapak Dr.rer.nat. Bambang Heru Iswanto, M.Si selaku Pembimbing I atas segala saran, bimbingan, serta motivasi yang membangun kepada penulis untuk menyelesaikan penyusunan skripsi ini.
2. Ibu Dr. Mutia Delina, M.Si selaku Pembimbing II atas saran, bimbingan dan motivasi yang membangun.
3. Ibu Dr. Widyaningrum Indrasari, M.Si selaku ketua program studi Fisika Universitas Negeri Jakarta.
4. Kedua orang tua serta keluarga atas doa dan motivasi yang selalu diberikan kepada penulis selama penyusunan skripsi ini.
5. Intan Ayu Rembulan, sebagai sahabat yang telah meluangkan waktunya untuk membantu dalam pengambilan data yang dibutuhkan dalam penulisan skripsi ini. Terima kasih pula atas saran dan motivasi yang selalu diberikan kepada penulis.
6. Peny Amalia dan Hanifah Widya Putri, sebagai sahabat yang telah meluangkan waktu dan memberikan sebagian ide-ide cemerlangnya untuk penyempurnaan penyusunan skripsi ini.
7. Martalia Andayani, Anisa Agustianingsih, Sabrina Tamimi, Desy Rakhmawati dan Ayu Mutia yang selalu memberikan doa, dukungan dan motivasi bagi penulis untuk menyusun skripsi serta telah melewati suka dan duka bersama dalam ikatan persahabatan The Angels Sisters.

8. Tanti Jatiningrum, Devi Maharani, Ade Febriansyah dan Nadya Andarisna selaku sahabat yang selalu memberikan doa, dukungan dan motivasi dalam penyusunan skripsi.
9. Intan Khaerani, Ulfah Zuhaeriah, Endah Dwi Cahyani, Indra Permana, Izrul Supriyadi, Elnoordiansyah, Alfareza Lazuardy, dan seluruh teman-teman prodi Fisika angkatan 2012 sebagai teman seperjuangan semasa kuliah yang selalu memberikan bantuan, saran dan motivasi untuk selalu semangat dalam penyusunan skripsi.
10. Serta semua pihak yang telah membantu, yang tidak dapat disebutkan satu persatu.

Penulis menyadari keterbatasan ilmu pengetahuan dan pengalaman dalam menyusun skripsi ini, untuk itu segala saran dan kritik yang membangun senantiasa penulis harapkan sebagai proses pembelajaran untuk masa yang akan datang.

Jakarta, Juli 2016

Novia Angelita

DAFTAR ISI

MOTTO DAN PERSEMBAHAN	i
ABSTRAK	ii
ABSTRACT	iii
KATA PENGANTAR	iv
DAFTAR ISI.....	vi
DAFTAR TABEL.....	x
DAFTAR LAMPIRAN.....	xi
BAB I PENDAHULUAN	1
A. Latar Belakang	1
B. Identifikasi Masalah	3
C. Pembatasan Masalah	3
D. Perumusan Masalah	3
E. Ruang Lingkup Penelitian.....	3
F. Tujuan Penelitian	4
G. Manfaat Penelitian	4
BAB II KAJIAN PUSTAKA.....	5
A. <i>Video Processing</i>	5
1. <i>Video Digital</i>	5
2. <i>Format Video Digital</i>	6
B. <i>Computer Vision</i>	7
C. Deteksi Obyek Bergerak	8
D. <i>Optical Flow</i>	8
E. <i>Metode Horn-Schunck</i>	9
BAB III METODOLOGI PENELITIAN	11
A. Tujuan Operasional	11
B. Tempat dan Waktu Penelitian	11
C. Metode Penelitian.....	11

D. Alat dan Bahan Penelitian.....	11
E. Prosedur Penelitian.....	12
1. Perancangan Perangkat Keras	12
2. Perancangan Perangkat Lunak.....	13
3. Perancangan Pengujian.....	16
F. Diagram Alir Penelitian	17
BAB IV HASIL DAN PEMBAHASAN	18
A. Eksperimen.....	18
B. Hasil Eksperimen dan Pembahasan	18
1. Deteksi Obyek Bergerak.....	18
2. Estimasi Kecepatan Obyek Bergerak	21
BAB V PENUTUP.....	30
A. Kesimpulan	30
B. Saran.....	30
DAFTAR PUSTAKA	32
LAMPIRAN.....	34
DAFTAR RIWAYAT HIDUP.....	61

DAFTAR GAMBAR

Gambar 2.1.	Proses terbentuknya <i>video digital</i>	5
Gambar 2.2.	<i>Sampling</i> video dalam dimensi spasial, temporal, vertikal dan horizontal [8].	6
Gambar 2.3.	Format resolusi spasial pada <i>video digital</i> [8].	6
Gambar 2.4.	Komputer menerjemahkan gambar yang kita lihat sebagai serangkaian angka [12]	7
Gambar 2.5.	<i>Frame</i> pada (a) t_1 , (b) $t_1+\Delta t$ dan (c) pola <i>optical flow</i> -nya.....	9
Gambar 3.1.	Blok diagram perancangan perangkat keras	12
Gambar 3.2.	<i>Flowchart</i> program deteksi obyek bergerak dengan <i>Foreground Detector</i>	13
Gambar 3.3.	<i>Flowchart</i> program mengestimasi kecepatan dengan <i>Optical Flow Horn-Schunck</i>	14
Gambar 3.4.	<i>Flowchart</i> keseluruhan program gabungan	15
Gambar 3.5.	Diagram alir penelitian	17
Gambar 4.1.	Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 dengan <i>minimum blob area</i> 500 pada kecepatan 20 km/jam	19
Gambar 4.2.	Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 dengan <i>minimum blob area</i> 1000 – 2000 pada kecepatan 20 km/jam	19
Gambar 4.3.	Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 dengan <i>minimum blob area</i> 2500 pada kecepatan 20 km/jam	20
Gambar 4.4.	Grafik perbandingan hasil kecepatan estimasi vs kecepatan aktual	22
Gambar 4.5.	Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan pada 40 km/jam	23
Gambar 4.6.	Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan pada 20 km/jam	24
Gambar 4.7.	Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan pada 60 km/jam	24
Gambar 4.8.	Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 eksperimen kedua pada 40 km/jam	26

Gambar 4.9. Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 eksperimen keempat pada 40 km/jam	26
Gambar 4.10. Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 eksperimen pertama pada 60 km/jam	26
Gambar 4.11. Grafik MRE kecepatan estimasi vs kecepatan aktual	28
Gambar 4.12. Grafik standar deviasi kecepatan estimasi vs kecepatan aktual	29

DAFTAR TABEL

Tabel 4.1. Karakteristik video yang diujikan.....	18
Tabel 4.2. Hasil deteksi obyek bergerak berdasarkan nilai minimum blob area	21
Tabel 4.3. Data hasil estimasi kecepatan dengan metode Optical Flow Horn-Schunck	25

DAFTAR LAMPIRAN

Lampiran 1.	Program yang digunakan dalam penelitian.....	34
Lampiran 1.1.	Program deteksi obyek bergerak pada MATLAB.....	34
Lampiran 1.2.	Program estimasi kecepatan obyek bergerak pada MATLAB.....	35
Lampiran 1.3.	Program keseluruhan gabungan deteksi dan estimasi kecepatan obyek bergerak.....	36
Lampiran 2.	Hasil deteksi obyek bergerak.....	38
Lampiran 2.1.	Deteksi obyek bergerak pada kecepatan 20 km/jam.....	38
Lampiran 2.2.	Deteksi obyek bergerak pada kecepatan 40 km/jam.....	39
Lampiran 2.3.	Deteksi obyek bergerak pada kecepatan 60 km/jam.....	41
Lampiran 3.	Hasil <i>bounding box cropping</i> pada obyek bergerak yang terdeteksi.....	43
Lampiran 3.1.	<i>Bounding box cropping</i> pada <i>frame</i> awal kecepatan 20 km/jam....	43
Lampiran 3.2.	<i>Bounding box cropping</i> pada <i>frame</i> awal kecepatan 40 km/jam...	44
Lampiran 3.3.	<i>Bounding box cropping</i> pada <i>frame</i> awal kecepatan 60 km/jam....	45
Lampiran 4.	Data hasil estimasi kecepatan obyek bergerak.....	46
Lampiran 4.1.	Estimasi kecepatan obyek bergerak pada 20 km/jam.....	46
Lampiran 4.2.	Estimasi kecepatan obyek bergerak pada 40 km/jam.....	51
Lampiran 4.3.	Estimasi kecepatan obyek bergerak pada 60 km/jam.....	56

BAB I

PENDAHULUAN

A. Latar Belakang

Pelanggaran lalu lintas seringkali terjadi di Indonesia, mulai dari tidak mengenakan atribut berkendara yang sesuai, menerobos lampu merah, hingga tidak memiliki Surat Ijin Mengemudi (SIM). Pelanggaran lalu lintas tersebut dapat ditangani oleh pihak yang berwajib karena masih dapat diperiksa dan dilihat secara langsung. Namun terdapat pula pelanggaran lalu lintas yang tidak dapat diperiksa secara langsung dan memerlukan alat bantu untuk memeriksanya, yaitu berkendara melewati ambang batas kecepatan yang telah ditentukan.

Berkendara melewati ambang batas kecepatan yang telah ditentukan dapat menimbulkan kecelakaan lalu lintas. Menurut Badan Pusat Statistik (BPS), angka kecelakaan di Indonesia hingga saat ini masih cukup tinggi. Pada tahun 2014 terjadi kecelakaan lalu lintas sebanyak 95.906 kasus [1]. Untuk mengurangi angka kecelakaan dalam berkendara, tentunya harus ada alat bantu yang digunakan untuk mengawasi kendaraan yang melaju dengan kecepatan tinggi. Salah satu cara untuk mengawasi kendaraan yang dapat dilakukan adalah dengan menggunakan *video surveillance system*.

Sistem pengamanan yang menggunakan *video camera* selama ini umumnya kurang praktis, karena pengguna dituntut untuk melakukan pengamatan secara terus-menerus pada obyek. *Video camera* hanya dapat menangkap suatu obyek tanpa adanya informasi kecepatan yang dimunculkan terkait obyek tersebut. Hal ini menjadi tidak efektif karena pihak yang berwenang akan sulit untuk mengetahui apakah kecepatan kendaraan tersebut masih dalam kecepatan normal atau telah melewati ambang batas kecepatan yang diperbolehkan. Maka dari itu dibuat sebuah inovasi teknologi berbasis *video processing* yang dapat mengestimasi kecepatan kendaraan yang sedang terekam.

Salah satu teknik berbasis *video processing* yang dapat menginterpretasikan informasi pada obyek yang diamati ialah *Computer Vision*. *Computer Vision* merupakan suatu bidang ilmu komputer yang mempelajari konten-konten yang

terdapat pada citra [2]. *Sub-domain* dari *Computer Vision* terdiri dari rekonstruksi citra, deteksi kejadian, pelacakan video, pengenalan obyek, pembelajaran, indeksasi, estimasi gerakan, dan restorasi citra [3].

Pada penelitian sebelumnya, Bambang dkk, telah mengimplementasikan pengolahan citra digital untuk menghitung kecepatan obyek bergerak dengan menggunakan metode *Frame Difference* dan memperoleh tingkat akurasi rata-rata tertinggi untuk semua kondisi yang diperoleh adalah 95,92% [4]. Danny dkk, juga melakukan eksperimen dengan menghitung kecepatan menggunakan *Computer Vision*. Dari hasil percobaan yang mereka peroleh, disimpulkan bahwa *Computer Vision* tidak dapat bekerja pada kondisi gelap dan pada pencahayaan yang terlalu tinggi juga akan menyebabkan *noise* [5].

Sama halnya dengan penelitian Danny dkk, Satrio dkk, juga menyimpulkan bahwa sistemnya tidak bisa bekerja dengan baik pada malam hari dengan rentang intensitas cahaya 0 sampai 5 lux. Dalam penelitiannya, mereka menggunakan metode *background subtraction* dengan algoritma *Gaussian Mixture Model* (GMM) yang akan memisahkan antara *background* dengan obyek yang akan dideteksi. Pada percobaan ini diperoleh persentase *error* keseluruhan dengan rentang 1,21% sampai dengan 21,37% [6].

Penelitian lainnya dalam menghitung kecepatan kendaraan dilakukan oleh Bonanalia dkk, dengan menggunakan metode *Optical Flow Lucas-Kanade*. *Optical Flow* merupakan suatu gerakan yang terlihat karena adanya pola *image brightness* dengan mengasumsikan bahwa *brightness* tiap citra dari waktu ke waktu bernilai tetap, perpindahan piksel pada citra dari satu *frame* ke *frame* berikutnya tidak terlalu jauh, dan arah pergerakan piksel yang berdekatan adalah sama. Besarnya nilai *Mean Absolute Error* (MAE) rata-rata untuk semua kondisi yang diperoleh adalah 2,2 [7].

Pada penelitian kami dilakukan pembuatan program untuk mengestimasi kecepatan kendaraan yang terekam oleh *video camera*, dimana estimasi kecepatannya menggunakan metode *Optical Flow Horn-Schunck*. Adapun pengujian sistem ini dilakukan dengan beberapa perbedaan kecepatan obyek uji yaitu dengan variasi antara 20 km/jam hingga 60 km/jam.

B. Identifikasi Masalah

Berdasarkan latar belakang masalah yang telah disampaikan, permasalahan-permasalahan yang dapat diidentifikasi adalah sebagai berikut:

- a. Bagaimana proses estimasi kecepatan obyek bergerak pada video menggunakan metode *Optical Flow Horn-Schunck*?
- b. Bagaimana efektivitas yang dimiliki oleh sistem estimasi kecepatan obyek bergerak pada video?
- c. Apa format video yang digunakan dalam penelitian ini?
- d. Apa obyek yang diujikan dalam penelitian ini?
- e. Bagaimana kondisi tempat pengambilan video obyek uji?
- f. Berapa variasi kecepatan yang digunakan dalam pengambilan video obyek uji?

C. Pembatasan Masalah

Pada penelitian ini permasalahan akan dibatasi pada penggunaan metode *Optical Flow Horn-Schunck* untuk mengestimasi kecepatan obyek bergerak pada video.

D. Perumusan Masalah

Perumusan masalah dalam penelitian ini adalah: “Apakah metode *Optical Flow Horn-Schunck* dapat digunakan untuk menentukan kecepatan obyek bergerak pada video?”

E. Ruang Lingkup Penelitian

Agar pembahasan dapat dilakukan dengan lebih cermat, maka ruang lingkup yang akan dibahas pada penelitian ini adalah:

- a. Input sistem berupa video dengan format file (*.MOV*).
- b. Kecepatan obyek bervariasi antara 20 km/jam hingga 60 km/jam.
- c. Hanya terdapat satu obyek uji pada video.
- d. Obyek uji bergerak hanya satu arah.
- e. Obyek uji berupa sepeda motor.
- f. Metode deteksi obyek yang digunakan adalah *Foreground Detector* berbasis *Gaussian Mixture Models* (GMM).

- g. Metode estimasi kecepatan yang digunakan adalah *Optical Flow Horn-Schunck*.

F. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah sebagai berikut:

- a. Mendeteksi obyek bergerak pada video.
- b. Mengestimasi kecepatan obyek bergerak pada video.
- c. Mengevaluasi efektivitas metode *Optical Flow Horn-Schunck* dalam menentukan kecepatan obyek bergerak pada video.

G. Manfaat Penelitian

Manfaat yang dapat diambil dari penelitian ini adalah:

- a. Memprediksi kecepatan obyek bergerak.
- b. Sebagai langkah awal untuk membantu dalam meminimalisir angka kecelakaan lalu lintas melalui pemantauan kecepatan kendaraan dengan *video camera*.
- c. Sebagai bahan perbandingan untuk disempurnakan pada penelitian berikutnya.

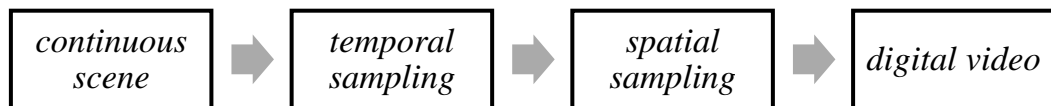
BAB II

KAJIAN PUSTAKA

A. Video Processing

1. Video Digital

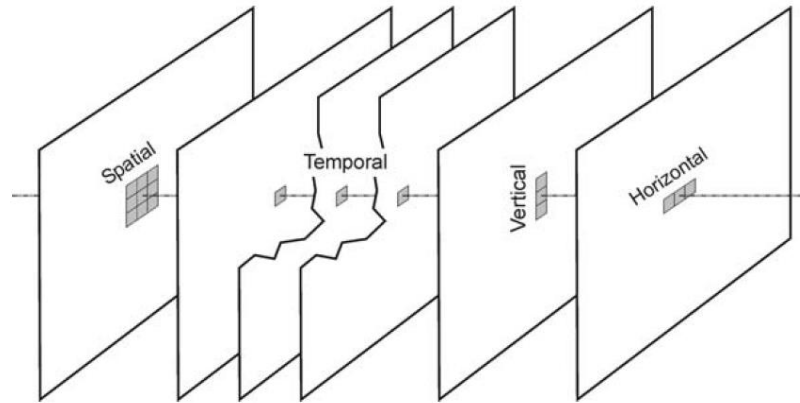
Pada dasarnya, video merupakan deretan citra yang berbeda pada waktu yang berurutan [8]. *Video digital* dapat diperoleh dengan cara mencuplik *raster scan* (yang memerlukan konversi analog-digital) atau secara langsung menggunakan *video camera* digital. Kamera digital mencuplik gambar adegan dari waktu ke waktu, menghasilkan *frame-frame* diskrit. Untuk menghasilkan gerakan yang halus, *frame* pada *video digital* diambil dan ditampilkan pada *frame rate* tertentu [9].



Gambar 2.1. Proses terbentuknya *video digital*

Temporal Sampling

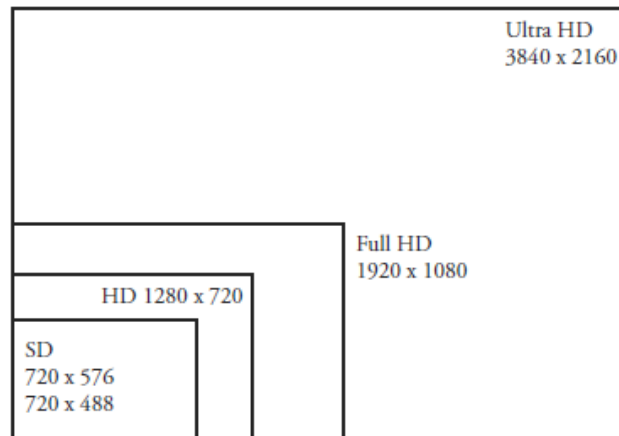
Video digital terdiri dari urutan *frame*. Jika pergantian *frame* pada layar dilakukan dengan cepat, kita akan melihat gambaran yang bergerak. Hal itu terjadi karena mata kita tidak dapat melihat pergantian *frame* yang terjadi dengan sangat cepat sebagai *frame-frame* yang terpisah [10]. Jika beda waktu antar urutan *frame* terlalu jauh, kita akan melihat gerakan yang tersentak-sentak (tidak halus). Pada praktiknya, kebanyakan format video menggunakan temporal sampling sebesar 25 *frame* per detik atau lebih [10].



Gambar 2.2. *Sampling* video dalam dimensi spasial, temporal, vertikal dan horizontal [8].

Spatial Sampling

Resolusi horizontal dan vertikal dari *video digital* berkaitan erat dengan jumlah piksel persatuan jarak. Jumlah piksel per baris dan jumlah baris per-*frame* menunjukkan kualitas video [8], seperti ditunjukkan oleh Gambar 2.3.



Gambar 2.3. Format resolusi spasial pada *video digital* [8].

2. Format Video Digital

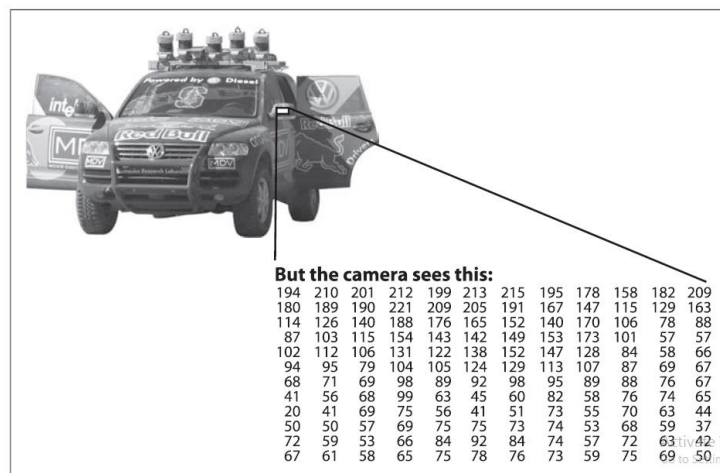
Sebuah *video digital* terdiri dari sekumpulan *frame* yang mana *frame-frame* tersebut dikonversi menjadi sebuah *file* komputer dan dapat dioperasikan menggunakan sebuah perangkat lunak *multimedia player* [11].

Berdasarkan jenis kompresi dari *file video digital* tersebut, terdapat format-format *video digital* yang dapat digunakan sesuai dengan keunikannya masing-masing. Beberapa format *video digital* yang umumnya dijumpai diantaranya [10]:

1. AVI (*Audio Video Interleave*)
2. MPEG (*Motion Picture Expert Group*)
3. RVMB (*Real Media Variable Bitrate*)
4. MKV (*Matroska Video*)
5. WMV (*Windows Media Video*)
6. FLV (*Flash Video*)

B. Computer Vision

Computer Vision bertujuan untuk meniru kemampuan penglihatan manusia dengan cara memproses dan memahami konten-konten dalam citra dan *video digital* [8]. *Computer Vision* mampu mengubah data citra dan *video digital* menjadi suatu representasi baru, yang mungkin berarti mengubah warna RGB menjadi *grayscale* atau menghapus urutan gambar pada gerakan *video*. *Sub-domain* dari *Computer Vision* terdiri dari rekonstruksi citra, deteksi kejadian, pelacakan *video*, pengenalan obyek, pembelajaran, indeksasi, estimasi gerakan, dan restorasi citra [3]. Pada *Computer Vision*, komputer hanya menerima data berupa angka dari kamera. Apa yang “dilihat” oleh komputer tak sama dengan apa yang dilihat oleh mata kita. Jika kita melihat kaca spion mobil, maka yang dilihat oleh komputer hanyalah serangkaian angka, seperti yang ditunjukkan pada Gambar 2.4 [12].



Gambar 2.4. Komputer menerjemahkan gambar yang kita lihat sebagai serangkaian angka [12]

Computer Vision dapat diaplikasikan dalam berbagai bidang ilmu, mulai dari sains, teknologi, hingga bidang industri. Dalam bidang sains, *Computer Vision* digunakan untuk dapat mengekstraksi informasi dari citra, seperti citra dari rangkaian video atau data multi dimensi dari pemindai medis. Pada bidang teknologi, *Computer Vision* digunakan dalam proses kontrol *robotic*, sistem navigasi, dan sebagainya. Sedangkan dalam bidang industri, *Computer Vision* digunakan untuk mesin industri sebagai pemeriksa botol maupun kemasan lainnya selama produksi, dan masih banyak lagi aplikasi *Computer Vision* lainnya [13].

C. Deteksi Obyek Bergerak

Deteksi obyek merupakan hal penting yang dilakukan dalam *video processing*, biasanya dilakukan pada *video surveillance system*. Deteksi obyek bergerak dalam video dapat dilakukan dengan beberapa metode, diantaranya dengan ekstraksi warna, *background subtraction*, *foreground detector* dan *blob analysis*. Pada penelitian ini, metode yang digunakan untuk mendeteksi obyek adalah *foreground detector* dan *blob analysis*.

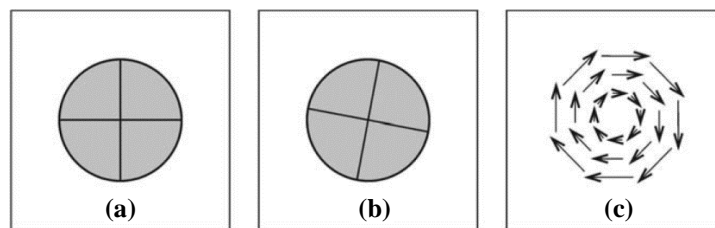
Foreground detector merupakan salah satu metode berbasis *Gaussian Mixture Models* (GMM), metode ini memiliki kemiripan dengan metode *background subtraction* dimana *foreground* dari video dipisahkan dari *background*-nya. Namun dalam *background subtraction* diperlukan *background* acuan yang kemudian dilakukan operasi pengurangan dengan *current frame*, sedangkan pada *foreground detector* tidak diperlukan *background* acuan karena dilakukan secara otomatis menggunakan *vision.ForegroundDetector* yang merupakan seperangkat *Computer Vision System Toolbox* pada MATLAB. *Output* dari algoritma tersebut berupa deretan citra biner (*frame*), dimana piksel pada *background* didefinisikan sebagai 0 dan obyek bagian depan (*foreground*) didefinisikan sebagai 1 [9].

D. Optical Flow

Optical Flow merupakan suatu gerakan yang terlihat karena adanya pola *image brightness* dengan mengasumsikan bahwa *brightness* tiap citra dari waktu ke waktu bernilai konstan, perpindahan piksel pada citra dari satu *frame* ke *frame*

berikutnya tidak terlalu jauh, dan arah pergerakan piksel yang berdekatan adalah sama [7]. Dengan menentukan *optical flow* tiap *frame* pada *video*, kita dapat menghitung kecepatan obyek yang bergerak dalam video.

Optical Flow digunakan dalam *Computer Vision* untuk mencirikan dan mengukur gerakan obyek pada *video stream*, seringkali digunakan untuk deteksi obyek *motion-based* dan pelacakan sistem [14]. *Optical Flow* dapat ditentukan dengan menghitung variasi nilai intensitas citra dari waktu ke waktu. Hasilnya merupakan pola medan vektor, lebih dikenal sebagai medan *optical flow*, berisi vektor gerakan untuk masing-masing *support region* pada *frame* [9]. Contoh dari dua *frame* yang saling berurutan dan hasil pola *optical flow*-nya ditunjukkan oleh Gambar 2.5.



Gambar 2.5. *Frame* pada (a) t_1 , (b) $t_1 + \Delta t$ dan (c) pola *optical flow*-nya

Prinsip *Optical Flow* dapat ditentukan dengan menggunakan banyak metode untuk menghitung kecepatan obyek yang bergerak, metode yang umum digunakan adalah:

1. Metode *Lucas-Kanade*
2. Metode *Horn-Schunck*

Metode ini digunakan dalam penelitian.

E. Metode *Horn-Schunck*

Metode *Horn-Schunck* merupakan metode yang dapat digunakan untuk mendapat hasil pola *optical flow*. Metode ini pertama kali digugus oleh Berthold K. P. Hold dan Brian G. Schunck pada tahun 1981. Metode *Horn-Schunck* merupakan metode pertama yang mendeteksi obyek menggunakan intensitas cahaya [12]. Metode ini seringkali digunakan dalam pelacakan obyek dan dapat diimplementasikan dalam penelitian ini untuk mendeteksi sepeda motor bergerak dengan melacak vektor gerakan sepeda motor tersebut [13]. Metode ini memiliki

keuntungan karena menghasilkan vektor gerakan dengan tingkat kepadatan yang tinggi [13], namun juga memiliki kelemahan yaitu kurang sensitif terhadap *noise* [15]. Metode *Horn-Schunck* bergantung pada iterasi penyelesaian persamaan diferensial [13]:

$$\frac{\partial}{\partial x} \frac{\partial u}{\partial x} - \frac{1}{a^2} I_x (I_x u + I_y v + I_t) = 0 \quad (2.1)$$

$$\frac{\partial}{\partial y} \frac{\partial v}{\partial y} - \frac{1}{a^2} I_y (I_x u + I_y v + I_t) = 0 \quad (2.2)$$

Dengan u dan v merupakan kecepatan pada (x, y) , sehingga persamaan (2.3) dan (2.4) di bawah ini merupakan rumus pelacakan *Horn-Schunck* untuk mengestimasi pergerakan dan kecepatan dari obyek [15].

$$u^{k+1} = \bar{u}^k - \frac{I_x(I_x \bar{u}^k + I_y \bar{v}^k + I_t)}{a^2 + I_x^2 + I_y^2} \quad (2.3)$$

$$v^{k+1} = \bar{v}^k - \frac{I_y(I_x \bar{u}^k + I_y \bar{v}^k + I_t)}{a^2 + I_x^2 + I_y^2} \quad (2.4)$$

Dimana I_x , I_y , dan I_t merupakan turunan dari intensitas citra, u merupakan *Optical Flow* horizontal, v merupakan *Optical Flow* vertikal, $[u^k v^k]$ merupakan perkiraan kecepatan sebuah piksel pada (x, y) , dan $[\bar{u}^k \bar{v}^k]$ merupakan rata-rata kecepatan pada titik-titik sebelumnya [16]. Sedangkan a merupakan konstanta pemberat koefisien, dikenal sebagai konstanta regularisasi. Semakin besar nilai a , semakin halus alur vektor gerakan [13].

BAB III

METODOLOGI PENELITIAN

A. Tujuan Operasional

Penelitian ini bertujuan untuk memperoleh data estimasi kecepatan obyek bergerak yang terekam pada video yang merupakan langkah awal dalam meminimalisir angka kecelakaan lalu lintas melalui pemantauan kecepatan kendaraan dengan *video camera*.

B. Tempat dan Waktu Penelitian

Penelitian ini dilakukan di jalan Arabika, Pondok Kopi dan kawasan Banjir Kanal Timur, Jakarta Timur pada bulan Maret – Juni 2016.

C. Metode Penelitian

Metode yang kami gunakan dalam penelitian ini adalah metode eksperimen. Dasar teori yang digunakan dalam penelitian ini yaitu *Optical Flow* yang digunakan untuk mengestimasi nilai kecepatan suatu obyek berdasarkan pola *brightness* yang dimiliki oleh video. Kecepatan obyek bergerak dihitung menggunakan persamaan dari metode *Horn-Schunck* yang membutuhkan nilai-nilai intensitas cahaya pada gambar untuk menentukan nilai kecepatan suatu obyek.

D. Alat dan Bahan Penelitian

Alat dan bahan yang digunakan dalam penelitian ini berupa perangkat keras dan perangkat lunak.

- a. Perangkat keras yang digunakan antara lain:
 1. Komputer
 2. Kamera digital
- b. Perangkat lunak yang digunakan adalah MATLAB R2015b.

E. Prosedur Penelitian

1. Perancangan Perangkat Keras

Spesifikasi lokasi obyek pengambilan citra adalah sebagai berikut:

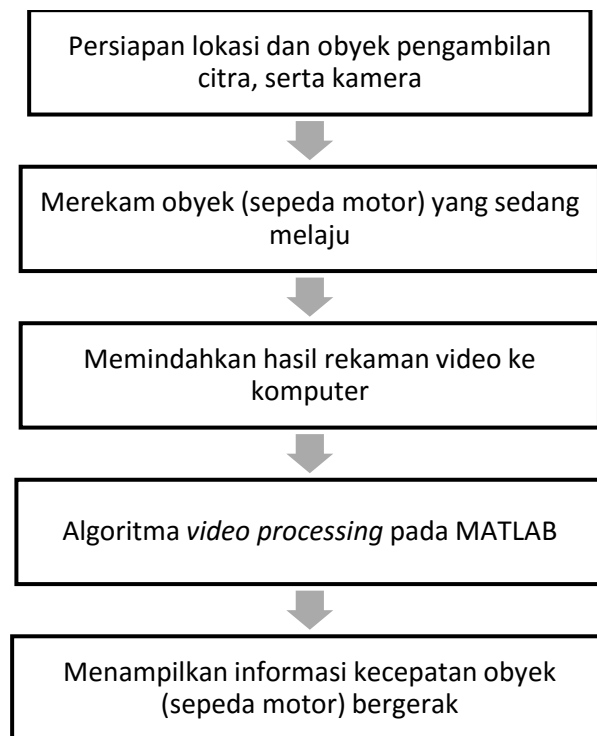
1. Memiliki jalur lurus
2. Dalam keadaan sepi; hanya ada sebuah obyek pengambilan citra (sepeda motor) yang bergerak

Setelah lokasi obyek pengambilan citra diasumsikan seperti spesifikasi di atas, maka pengambilan citra dapat dilakukan dengan menggunakan kamera digital.

Spesifikasi kamera digital yang digunakan adalah sebagai berikut:

1. Merk dan Tipe : CANON EOS 550D
2. *Digital zooming* : $3\times$ *optical zoom*
3. Resolusi video : 1920×1080
4. Format video output : MPEG-4 (.MOV)

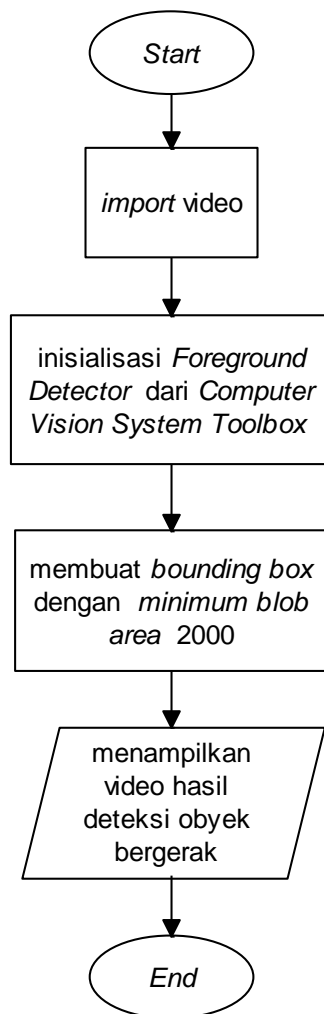
Setelah dilakukan pengambilan citra, hasil rekaman video dapat langsung diolah di komputer menggunakan algoritma yang telah dibuat. Blok diagram untuk perancangan perangkat keras dapat dilihat pada Gambar 3.1.



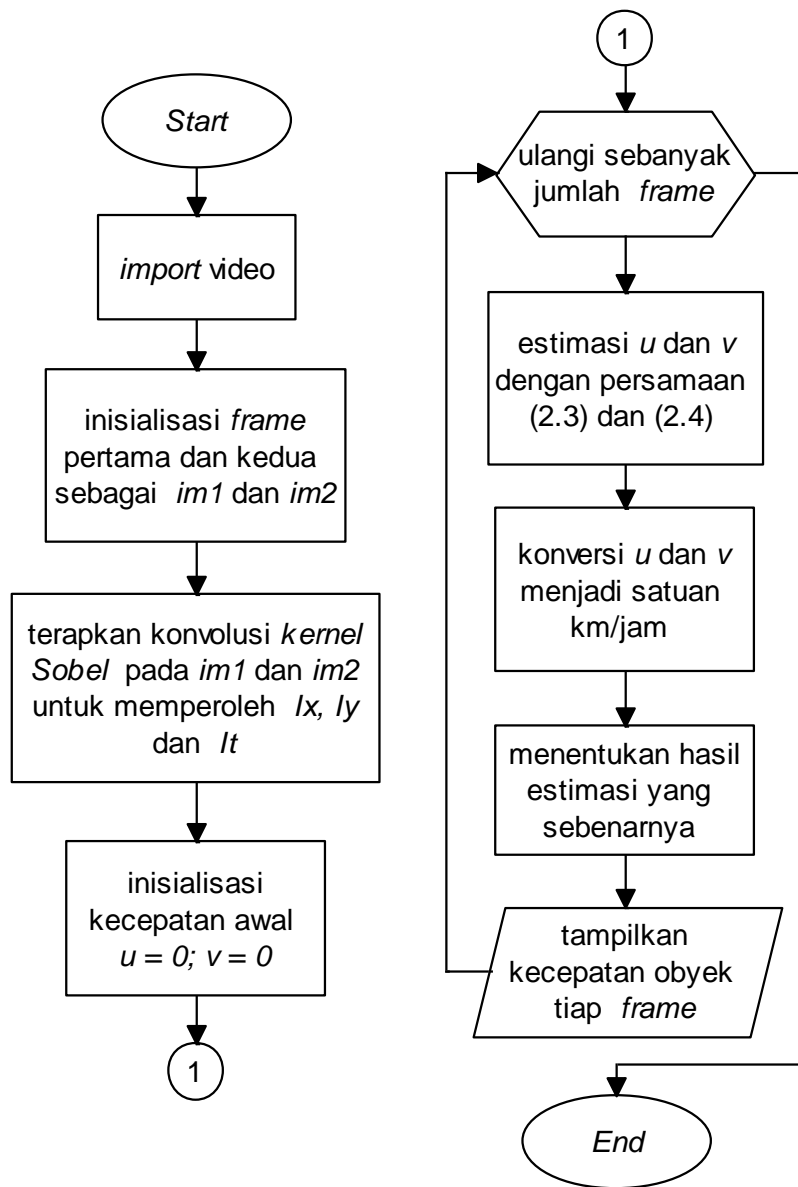
Gambar 3.1. Blok diagram perancangan perangkat keras

2. Perancangan Perangkat Lunak

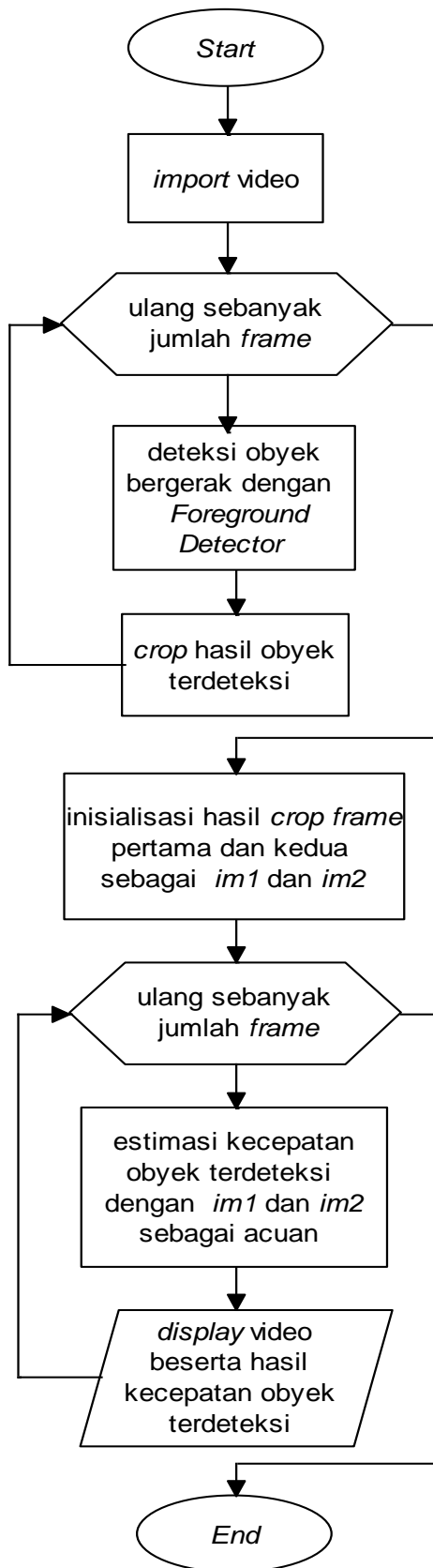
Perangkat lunak yang digunakan pada penelitian ini adalah MATLAB R2015b. Untuk menghitung kecepatan obyek (sepeda motor) yang bergerak, sebelumnya dilakukan deteksi obyek bergerak dengan menggunakan metode *Foreground Detector* berbasis *Gaussian Mixture Models (GMM)*. Kemudian digunakan metode *Optical Flow Horn-Schunck* yang akan menghitung nilai estimasi kecepatan obyek bergerak dan dibandingkan dengan kecepatan aktualnya pada *speedometer*. Berikut ini adalah beberapa *flowchart* program yang dibutuhkan dalam penelitian ini.



Gambar 3.2. Flowchart program deteksi obyek bergerak dengan *Foreground Detector*



Gambar 3.3. Flowchart program mengestimasi kecepatan dengan *Optical Flow Horn-Schunck*



Gambar 3.4. Flowchart keseluruhan program gabungan

3. Perancangan Pengujian

Pengujian dilakukan setelah hasil akhir kecepatan obyek bergerak berhasil diperoleh dari *video processing* dengan metode *Optical Flow Horn-Schunck*. Pengujian dilakukan dengan cara melakukan perbandingan antara hasil akhir penelitian dengan kecepatan kendaraan yang tertera pada *speedometer*. Pengujian dilakukan untuk mengevaluasi efektivitas metode *Optical Flow Horn-Schunck* dalam mengestimasi kecepatan obyek bergerak pada video. Efektivitas dari metode yang digunakan pada metode ini akan dibuktikan dengan tingkat akurasi dan presisi yang tinggi sesuai dengan persamaan berikut ini.

$$MRE = \frac{1}{n} \sum_i^n \frac{|t_i - e_i|}{t_i} \quad (3.1)$$

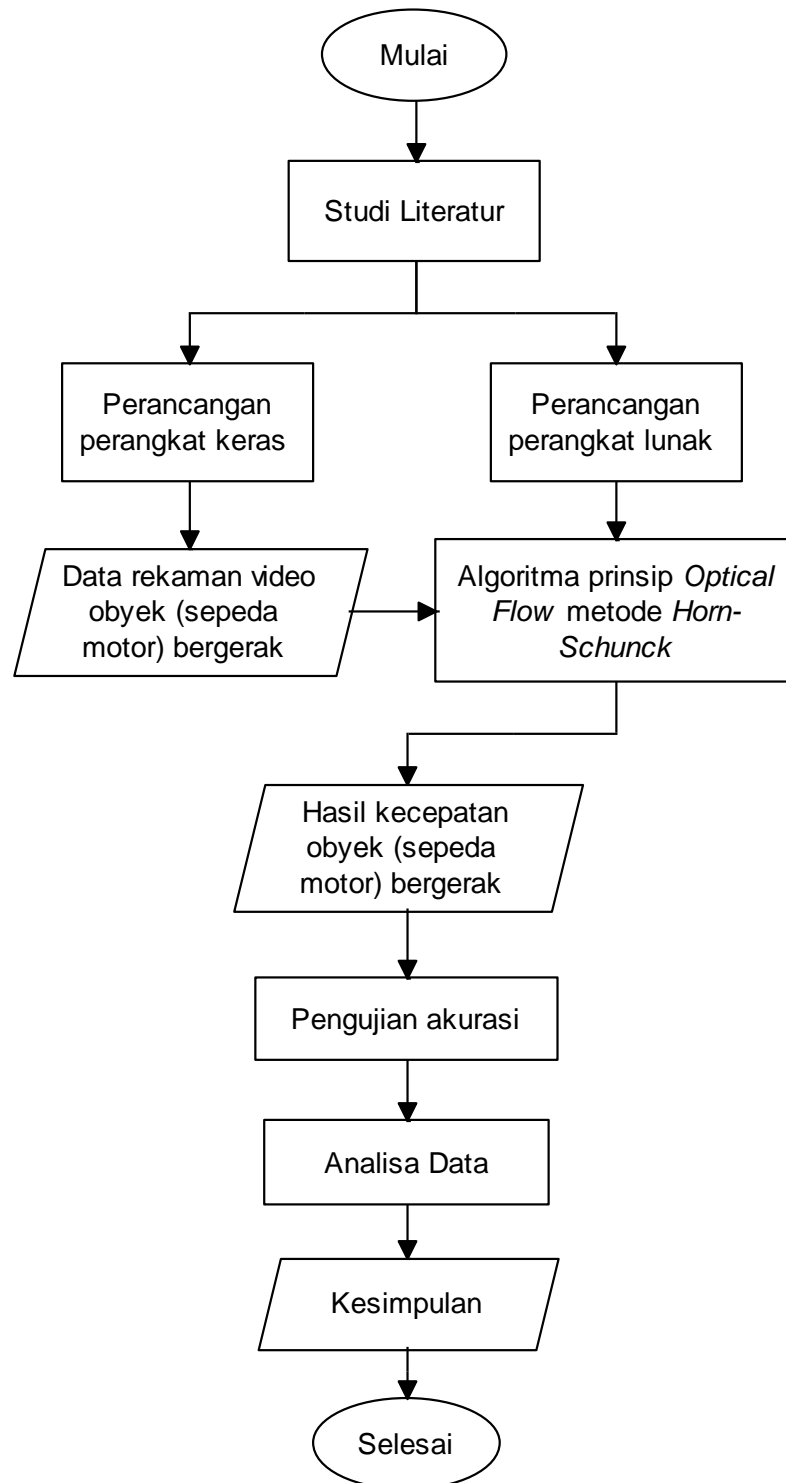
$$\sigma = \sqrt{\frac{\sum (e - \bar{e})^2}{n-1}} \quad (3.2)$$

Dimana t merupakan nilai teoritis yang sudah ditetapkan dan e merupakan nilai eksperimental yang diperoleh dalam penelitian. *Mean Relative Error* (MRE) menunjukkan rata-rata besarnya kesalahan baku dari eksperimen yang dilakukan. Kesalahan baku berbanding terbalik dengan tingkat akurasi dari eksperimen sehingga semakin besar nilai MRE maka semakin kecil akurasi yang dimiliki oleh data eksperimental. Sementara itu, Standar deviasi (σ) berbanding terbalik dengan tingkat presisi dari eksperimen sehingga semakin besar nilai σ maka semakin kecil tingkat presisi yang dimiliki oleh data eksperimental.

Sementara itu, nilai *error* yang dimiliki sistem juga ditunjukkan oleh nilai *Mean Absolute Error* (MAE) yang merupakan rata-rata dari selisih antara nilai teoritis dan nilai eksperimental. MAE merupakan perhitungan yang digunakan untuk menghitung seberapa tepat nilai eksperimental jika dibandingkan dengan nilai teoritis. Adapun perhitungan yang dilakukan adalah sebagai berikut:

$$MAE = \frac{1}{n} \sum_i^n |t_i - e_i| \quad (3.3)$$

F. Diagram Alir Penelitian



Gambar 3.5. Diagram alir penelitian

BAB IV

HASIL DAN PEMBAHASAN

A. Eksperimen

Pengambilan video dilakukan dengan menggunakan kamera SLR CANON EOS 550D, bertempat di jalan Arabika, Pondok Kopi dan kawasan Banjir Kanal Timur. Pengambilan video dilakukan sebanyak 15 kali dengan 5 kali repetisi untuk masing-masing obyek berkecepatan konstan sebesar 20 km/jam, 40 km/jam dan 60 km/jam. Pengambilan video dilakukan pada siang hari namun pada cuaca berawan. Setiap rekaman diambil dengan jumlah obyek bergerak sebanyak 1 buah, yaitu sepeda motor. Setelah rekaman video diperoleh, rekaman dipindahkan ke komputer dan diproses lebih lanjut dengan algoritma deteksi obyek bergerak dan estimasi kecepatan obyek bergerak. Tabel 4.1. menampilkan karakteristik video yang diproses dengan masing-masing algoritma yang diujikan.

Tabel 4.1. Karakteristik video yang diujikan

<i>Format</i>	MPEG-4 (.MOV)
<i>Video resolution</i>	1920 × 1080 <i>pixels</i>
<i>Matrix dimension</i>	640 × 480
<i>Color model</i>	RGB
<i>Frame rate</i>	50 <i>frames per second</i>

B. Hasil Eksperimen dan Pembahasan

1. Deteksi Obyek Bergerak

Deteksi obyek bergerak dilakukan dengan menggunakan metode *Foreground Detector* berbasis *Gaussian Mixture Models* (GMM). Untuk melakukan deteksi obyek bergerak, digunakan perangkat lunak MATLAB 2015b yang memiliki *Computer Vision System Toolbox* yang memudahkan dalam deteksi *foreground* dengan menggunakan *syntax* “*vision.ForegroundDetector*”. Dengan menggunakan algoritma yang sesuai dengan Gambar 3.2., pengambilan video di jalan Arabika, Pondok Kopi dengan kecepatan sepeda motor konstan sebesar 20

km/jam menggunakan beberapa variasi *minimum blob area* memperoleh hasil sebagai berikut:



Gambar 4.1. Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 dengan *minimum blob area* 500 pada kecepatan 20 km/jam



Gambar 4.2. Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 dengan *minimum blob area* 1000 – 2000 pada kecepatan 20 km/jam

Gambar 4.1. dan 4.2. menunjukkan video yang sama yang menunjukkan obyek dengan kecepatan yang sama, yaitu pada kecepatan konstan 20 km/jam telah diaplikasikan nilai *minimum blob area* yang bervariasi antara 500 hingga 2500. Dapat dilihat bahwa pada rentang *minimum blob area* 1000 hingga 2000, obyek yang bergerak dapat terdeteksi dengan sempurna yaitu dengan *bounding box* yang mengelilingi sekitar sepeda motor yang bergerak, baik ketika sepeda motor berada di kejauhan dan berukuran lebih kecil pada *frame* awal video maupun ketika sepeda motor berukuran besar pada *frame* akhir video. Sementara pada penggunaan *minimum blob area* 500 dapat dilihat bahwa *bounding box* tidak hanya mengelilingi

obyek yang sedang bergerak, melainkan juga mendeteksi obyek yang tidak bergerak, yaitu pagar sebagai obyek bergerak. Hal ini dapat dikatakan bahwa deteksi obyek bergerak yang dilakukan kurang akurat pada penggunaan *minimum blob area* 500.



Gambar 4.3. Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 dengan *minimum blob area* 2500 pada kecepatan 20 km/jam

Sedangkan pada penggunaan *minimum blob area* 2500 seperti pada gambar di atas, dapat dilihat bahwa tidak ada *bounding box* yang terbentuk walaupun terdapat sepeda motor yang sedang bergerak pada *frame* awal video dan pada *frame* akhir video *bounding box* kembali terbentuk sebagai pendeteksi sepeda motor yang bergerak.

Ternyata perbedaan variasi nilai *minimum blob area* sangat mempengaruhi dalam deteksi obyek yang bergerak dalam sebuah video. Dari ketiga hasil yang dijabarkan di atas, semakin kecil nilai *minimum blob area* yang digunakan maka akan semakin banyak *bounding box* yang akan terbentuk. Hal ini dikarenakan program berusaha untuk mendeteksi obyek bergerak dengan lebih teliti walaupun yang terdeteksi bukanlah obyek yang sebenarnya sedang bergerak secara kasat mata. Hal ini dapat dibandingkan dengan penggunaan *minimum blob area* 2500 yang kurang teliti dalam mendeteksi obyek bergerak yang berada di kejauhan (kecil). Sedangkan pada penggunaan *minimum blob area* 1000 hingga 2000 diperoleh pendeteksian obyek bergerak secara sempurna, hal ini berarti kita menggunakan batasan *minimum blob area* yang sesuai, tidak terlalu besar maupun terlalu kecil untuk mendeteksi obyek bergerak secara sempurna.

Tabel 4.2. Hasil deteksi obyek bergerak berdasarkan nilai *minimum blob area*

No.	<i>Minimum blob area</i>	Hasil deteksi obyek
1.	500	Obyek tak bergerak ikut terdeteksi
2.	1000	Sempurna
3.	1500	Sempurna
4.	2000	Sempurna
5.	2500	Tak terdeteksi saat berukuran kecil

2. Estimasi Kecepatan Obyek Bergerak

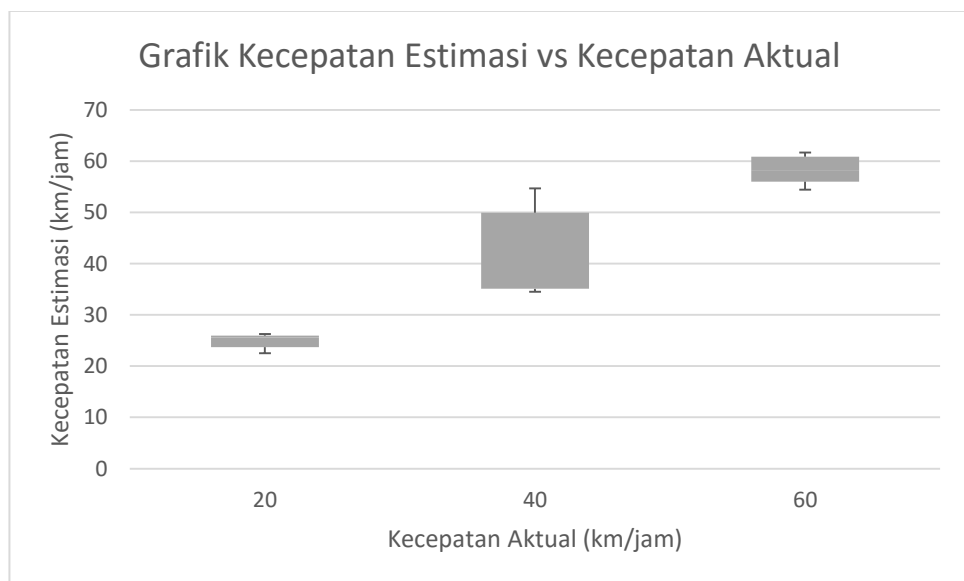
Estimasi kecepatan obyek bergerak dilakukan dengan menggunakan metode *Optical Flow Horn-Schunck*. Estimasi kecepatan obyek bergerak dilakukan tepat setelah deteksi obyek bergerak menggunakan perangkat lunak yang sama, yaitu MATLAB 2015b. Sesuai dengan Gambar 3.4, setelah melakukan obyek bergerak terdeteksi dilakukan *bounding box cropping* terlebih dahulu pada setiap *frame* video agar hasil kecepatan komputasi yang diperoleh tidak terlalu terganggu oleh *noise*.

Algoritma pemrograman yang digunakan dalam estimasi kecepatan obyek bergerak sesuai dengan *flowchart* pada Gambar 3.3. dan 3.4. adalah sebagai berikut:

1. Mulai.
2. *Import* video.
3. Melakukan pengulangan sebanyak jumlah *frame*:
 - a. Deteksi obyek bergerak dengan metode *Foreground Detector*.
 - b. Melakukan *bounding box cropping* pada hasil obyek yang terdeteksi.
4. Menginisialisasi variabel *im1* dan *im2* sebagai hasil *cropping frame* pertama dan kedua untuk dijadikan gambar acuan dalam estimasi kecepatan.
5. Menerapkan konvolusi kernel *Sobel* pada *im1* dan *im2* untuk memperoleh *Ix*, *Iy* dan *It* yang dibutuhkan pada persamaan (2.3) dan (2.4).
6. Menginisialisasi kecepatan awal obyek secara horizontal $u = 0$ dan vertikal $v = 0$.

7. Melakukan pengulangan sebanyak jumlah *frame*:
 - a. Mengestimasi nilai u dan v dengan persamaan (2.3) dan (2.4)
 - b. Mengkonversi satuan u dan v menjadi km/jam.
 - c. Menentukan kecepatan hasil estimasi yang sebenarnya.
 - d. Menampilkan video beserta hasil kecepatan obyek yang terdeteksi tiap *frame*.
8. Selesai.

Dari algoritma pemrograman di atas, hasil estimasi kecepatan obyek bergerak menggunakan metode *Optical Flow Horn-Schunck* ditunjukkan oleh grafik berikut:



Gambar 4.4. Grafik perbandingan hasil kecepatan estimasi vs kecepatan aktual

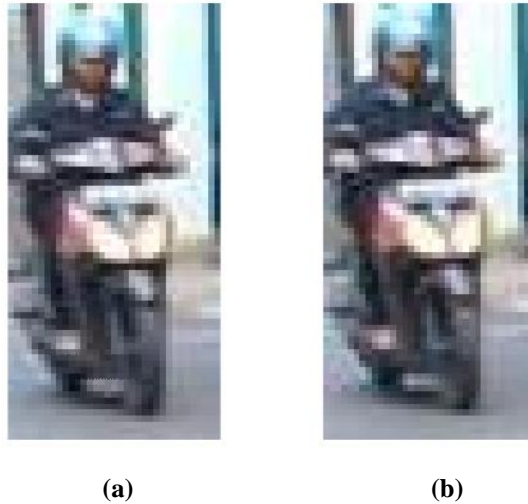
Grafik di atas menunjukkan ringkasan distribusi data hasil estimasi kecepatan menggunakan metode *Optical Flow Horn-Schunck*. Sebaran distribusi data terluas ditunjukkan pada kecepatan aktual 40 km/jam. Kecepatan aktual yang dimaksud dalam penelitian ini ialah kecepatan yang terbaca pada *speedometer* saat sepeda motor sedang dikendarai. Dari grafik *box plot* di atas, dapat kita lihat bahwa hasil kecepatan estimasi untuk tiap-tiap kecepatan aktual tidak tepat sesuai dengan yang ditampilkan pada *speedometer*, terdapat *error* yang dipengaruhi berbagai macam *noise* sehingga hasil yang diperoleh menggunakan perhitungan komputasi metode *Optical Flow Horn-Schunck* tidak 100% sama dengan kecepatan aktualnya.

Dari grafik yang ditampilkan di atas, kita dapat melihat secara kasat mata bahwa estimasi kecepatan secara komputasi menggunakan metode *Optical Flow Horn-Schunck* pada kecepatan aktual 40 km/jam memiliki distribusi data yang cenderung lebih luas dibandingkan dengan pada kecepatan lainnya, hal ini dapat dipengaruhi oleh *bounding box cropping* yang kurang sesuai dengan obyek yang sesungguhnya sedang bergerak sehingga banyak *noise* yang mengganggu variabel-variabel penting dalam estimasi kecepatan secara komputasi. Contoh *bounding box cropping* yang tidak sesuai dapat dilihat di bawah ini.

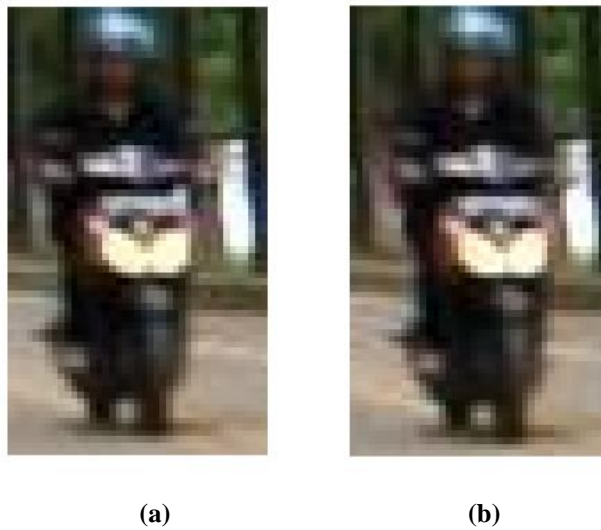


Gambar 4.5. Hasil *bounding box cropping* pada (a) t dan (b) $t+1$ sebagai *frame* sampel untuk estimasi kecepatan secara komputasi pada 40 km/jam

Gambar 4.5. menunjukkan contoh *bounding box cropping* yang kurang sesuai pada 40 km/jam, yaitu hasil deteksi lebih luas dari yang seharusnya sehingga menyebabkan banyak *noise* dari *background* yang ikut dianggap sebagai obyek yang bergerak dan diestimasi pula kecepatannya. Oleh sebab itu, pada 40 km/jam *error* yang dimiliki hasil estimasi secara komputasi lebih tinggi dibandingkan dengan pada kecepatan lainnya. *Bounding box cropping* mempengaruhi estimasi kecepatan komputasi dari obyek yang bergerak, hal tersebut dapat dibuktikan dan dibandingkan dengan kedua gambar di bawah ini.



Gambar 4.6. Hasil *bounding box cropping* pada (a) t dan (b) $t+1$ sebagai *frame* sampel untuk estimasi kecepatan secara komputasi pada 20 km/jam



Gambar 4.7. Hasil *bounding box cropping* pada t dan $t+1$ sebagai *frame* sampel untuk estimasi kecepatan secara komputasi pada 60 km/jam

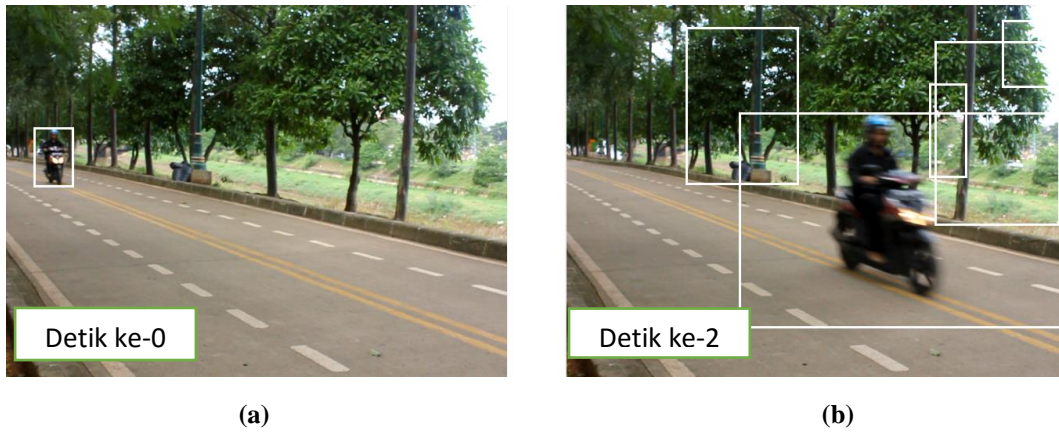
Gambar 4.6 dan 4.7. menunjukkan contoh *bounding box cropping* yang lebih sesuai daripada Gambar 4.5, hal tersebut terjadi karena area obyek bergerak yang terdeteksi tidak terlalu luas seperti pada Gambar 4.5 sehingga tidak terlalu banyak *noise* dari *background* yang mempengaruhi variabel-variabel perhitungan dan membuat sebaran data distribusi yang diperoleh dari estimasi kecepatan secara komputasi menjadi lebih sempit.

Selain ditunjukkan oleh grafik pada Gambar 4.4, data hasil estimasi kecepatan secara komputasi menggunakan metode *Optical Flow Horn-Schunck* juga ditunjukkan oleh Tabel 4.3.

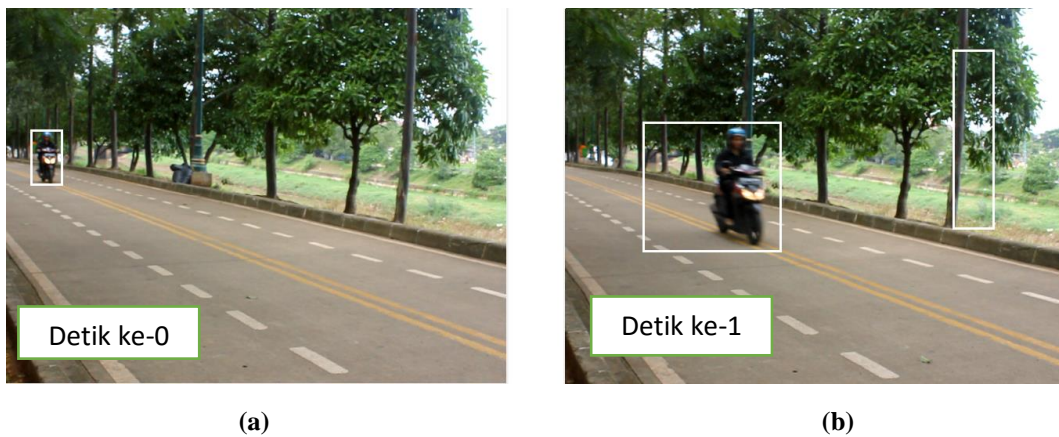
Tabel 4.3. Data hasil estimasi kecepatan dengan metode *Optical Flow Horn-Schunck*

Kecepatan Aktual	Eksperimen	Kecepatan Estimasi (km/jam)	Rata-rata	MAE	MRE	σ
20 km/jam	1	22.52705	23.651	2.527048	0.126352	0.35023
	2	25.138244		5.138244	0.256912	0.346675
	3	24.79785		4.79785	0.239892	0.291389
	4	21.37727		1.377265	0.068863	0.201018
	5	24.41629		4.416294	0.220815	0.344859
40 km/jam	1	33.94338	41.289	6.056615	0.151415	0.70163
	2	54.11349		14.11349	0.352837	0.534758
	3	34.51002		5.489982	0.13725	0.856755
	4	49.3423		9.3423	0.233557	0.896152
	5	34.54023		5.459774	0.136494	0.883707
60 km/jam	1	52.79652	56.632	7.20348	0.120058	1.514025
	2	59.26022		1.139262	0.018988	1.762092
	3	54.40725		5.592751	0.093213	1.720687
	4	56.65237		3.34763	0.055794	1.789593
	5	60.04412		0.605861	0.010098	0.843879

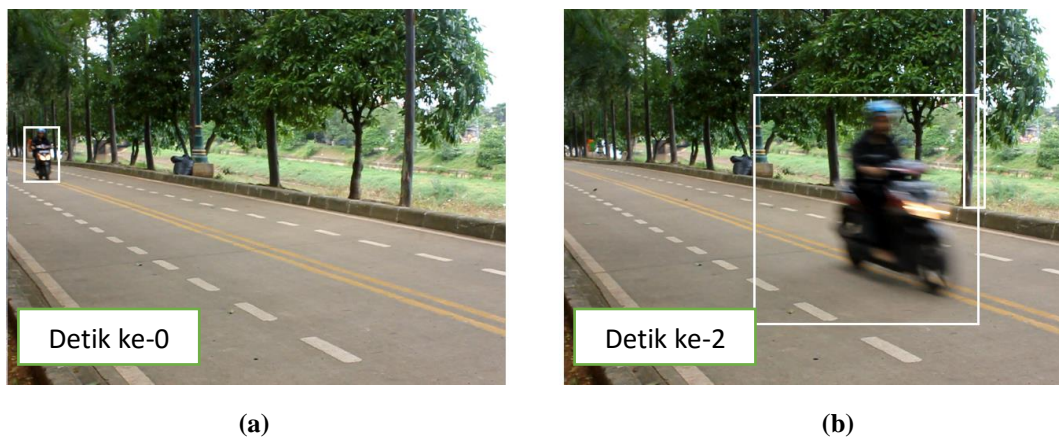
Pada tabel di atas, bagian yang diberi warna abu-abu merupakan hasil kecepatan estimasi dengan metode *Optical Flow Horn-Schunck* yang memiliki selisih paling besar dengan kecepatan aktualnya. Angka-angka tersebut memiliki peranan pada Gambar 4.4. sehingga grafik *box plot* kecepatan estimasi terhadap kecepatan aktual memiliki sebaran distribusi yang luas, terutama pada kecepatan aktual 40 km/jam. Selisih yang besar antara kecepatan estimasi dan kecepatan aktual menandakan bahwa *error* yang dimiliki pun besar seperti yang ditunjukkan pada kolom *Mean Absolute Error* (MAE). *Error* yang besar pada eksperimen yang diberi warna abu-abu tersebut dapat disebabkan oleh deteksi obyek yang tidak tepat pada tahap sebelumnya, yaitu dengan hasil deteksi sebagai berikut:



Gambar 4.8. Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 eksperimen kedua pada 40 km/jam



Gambar 4.9. Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 eksperimen keempat pada 40 km/jam

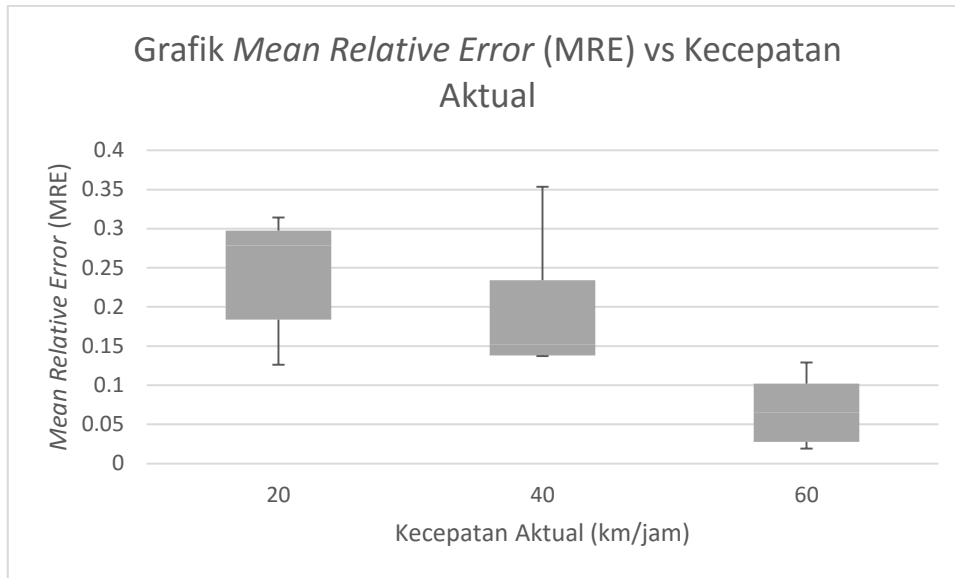


Gambar 4.10. Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 eksperimen pertama pada 60 km/jam

Ketiga gambar di atas menunjukkan hasil deteksi obyek yang gagal dan menyebabkan *error* yang tinggi diperoleh pada proses estimasi kecepatan. Ketiga gambar tersebut menampilkan bahwa terdapat lebih dari satu obyek yang terdeteksi pada bagian akhir video, sementara dengan mata telanjang pun kita dapat melihat bahwa hanya terdapat sebuah obyek yang bergerak. Sedangkan *bounding box* lainnya yang seharusnya hanya mendeteksi obyek bergerak, terlihat mendeteksi obyek yang diam. Banyaknya obyek yang salah terdeteksi pada video membuat program estimasi kecepatan bingung dalam memprediksi kecepatan obyek, sehingga diperoleh nilai MAE yang besar pada eksperimen kedua dan keempat ketika kecepatan aktual bernilai 40 km/jam serta pada eksperimen pertama saat kecepatan aktual bernilai 60 km/jam. Dari tabel dan ketiga gambar di atas, dapat disimpulkan bahwa hasil deteksi obyek mempengaruhi nilai kecepatan estimasi. Semakin tidak tepat hasil deteksi yang diperoleh, maka akan semakin tinggi nilai MAE.

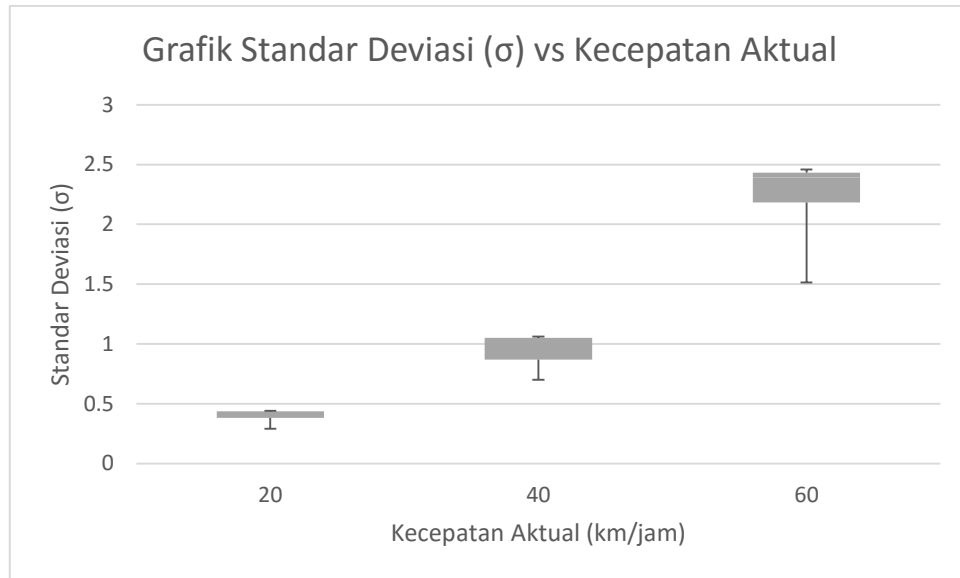
Sementara itu, efektivitas metode yang digunakan pada penelitian ini ditunjukkan oleh tingkat akurasi dan presisi yang secara berturut-turut berbanding terbalik dengan nilai *Mean Relative Error* (MRE) dan Standar Deviasi (σ). Dari data pada Tabel 4.3., diperoleh rata-rata MRE untuk semua kecepatan dalam persen adalah 14,8% dengan rata-rata MRE pada 20 km/jam adalah 18,2%, rata-rata MRE pada 40 km/jam adalah 20,2% dan rata-rata MRE pada 60 km/jam adalah 5,9%. Dengan berbanding terbalik dari persen MRE, maka tingkat akurasi untuk semua kecepatan adalah 85,2% dengan tingkat akurasi pada 20 km/jam sebesar 81,8%, tingkat akurasi pada 40 km/jam sebesar 79,8% dan tingkat akurasi pada 60 km/jam sebesar 94,1%.

Dari data pada tabel 4.3, diperoleh pula rata-rata σ untuk semua kecepatan adalah 0,869 dengan rata-rata σ pada 20 km/jam sebesar 0,307, rata-rata σ pada 40 km/jam sebesar 0,775 dan rata-rata σ pada 60 km/jam sebesar 1,526. Hasil tersebut menunjukkan bahwa semakin tinggi kecepatan, semakin besar nilai Standar Deviasi (σ) dari hasil kecepatan estimasi. Eksperimen dilakukan sebanyak 5 kali untuk tiap-tiap kecepatan aktual dan memperoleh nilai MRE dan σ dari kecepatan estimasi yang telah ditunjukkan oleh Tabel 4.3. dan ditampilkan kembali pada grafik di bawah ini.



Gambar 4.11. Grafik MRE kecepatan estimasi terhadap kecepatan aktual

Gambar 4.11. menunjukkan nilai *Mean Relative Error* (MRE) dari hasil estimasi kecepatan secara komputasi menggunakan metode *Optical Flow Horn-Schunck*. Grafik di atas menunjukkan bahwa distribusi *error* terhadap kecepatan aktual bersifat linear, yaitu distribusi *error* terbesar dimiliki pada kecepatan aktual 20 km/jam dan distribusi *error* terkecil dimiliki pada kecepatan aktual 60 km/jam. Hal tersebut menunjukkan bahwa semakin tinggi kecepatan aktual maka semakin kecil distribusi *error* yang dimiliki pada hasil kecepatan estimasi. Karena *error* berbanding terbalik terhadap tingkat akurasi, ini berarti bahwa semakin tinggi kecepatan aktual yang dimiliki obyek bergerak maka semakin tinggi tingkat akurasi pada sistem estimasi kecepatan obyek bergerak menggunakan metode *Optical Flow Horn-Schunck*. Hal ini juga berarti bahwa besarnya kecepatan aktual berbanding lurus dengan tingkat akurasi yang dimiliki sistem estimasi kecepatan obyek bergerak menggunakan metode *Optical Flow Horn-Schunck*. Akan tetapi, hasil ini tidak sesuai dengan hasil rata-rata MRE untuk tiap-tiap kecepatan aktual pada Tabel 4.3. Hasil rata-rata MRE tidak menunjukkan keterkaitan antara besarnya kecepatan dengan besarnya MRE. Hal tersebut terjadi karena pada kecepatan 40 km/jam terdapat *error* yang sangat besar dalam beberapa percobaan. Besarnya *error* tersebut dikarenakan hasil deteksi obyek yang tidak tepat sehingga hasil rata-rata menjadi tidak sesuai dengan grafik *box plot* diatas yang menunjukkan sebaran distribusi MRE.



Gambar 4.12. Grafik standar deviasi kecepatan estimasi terhadap kecepatan aktual

Gambar 4.12. menunjukkan nilai Standar Deviasi (σ) dari hasil estimasi kecepatan secara komputasi menggunakan metode *Optical Flow Horn-Schunck*. Grafik di atas menunjukkan bahwa distribusi standar deviasi terhadap kecepatan aktual bersifat linear, yaitu distribusi standar deviasi terkecil dimiliki pada kecepatan aktual 20 km/jam dan terbesar dimiliki pada kecepatan aktual 60 km/jam. Hal tersebut menunjukkan bahwa semakin tinggi kecepatan aktual maka semakin besar pula nilai standar deviasi yang dimiliki pada hasil kecepatan estimasi. Karena standar deviasi berbanding terbalik terhadap tingkat presisi, ini berarti bahwa semakin tinggi kecepatan aktual yang dimiliki obyek bergerak maka semakin rendah tingkat presisi pada sistem estimasi kecepatan obyek bergerak menggunakan metode *Optical Flow Horn-Schunck*. Hal ini juga berarti bahwa besarnya kecepatan aktual berbanding terbalik dengan tingkat presisi yang dimiliki sistem estimasi kecepatan obyek bergerak menggunakan metode *Optical Flow Horn-Schunck*. Hasil ini sesuai dengan nilai rata-rata σ untuk tiap-tiap kecepatan aktual pada Tabel 4.3. Hasil rata-rata σ dari Tabel 4.3. juga menunjukkan adanya keterkaitan antara besarnya kecepatan dengan rata-rata σ yang saling berbanding lurus.

BAB V

PENUTUP

A. Kesimpulan

Dari penelitian dan evaluasi yang telah dilakukan, dapat disimpulkan bahwa:

1. Dalam deteksi obyek bergerak menggunakan metode *Foreground Detector* berbasis *Gaussian Mixture Models* (GMM) diperlukan nilai ambang batas *minimum blob area* dengan rentang 1000 hingga 2000 untuk memperoleh hasil yang sempurna.
2. Estimasi kecepatan obyek bergerak secara komputasi menggunakan metode *Optical Flow Horn-Schunck* memerlukan *bounding box cropping* dari hasil deteksi terlebih dahulu untuk mengurangi *noise* dari *background video*.
3. Hasil kecepatan estimasi menunjukkan bahwa tingkat akurasi dari metode *Optical Flow Horn-Schunck* pada semua kondisi kecepatan adalah 85,2%.
4. Pada sistem estimasi kecepatan obyek bergerak menggunakan metode *Optical Flow Horn-Schunck*, semakin tinggi kecepatan aktual maka semakin tinggi pula tingkat akurasi yang dimiliki sistem.
5. Pada sistem estimasi kecepatan obyek bergerak menggunakan metode *Optical Flow Horn-Schunck*, semakin tinggi kecepatan aktual maka semakin rendah tingkat presisi yang dimiliki sistem.

B. Saran

Adapun saran-saran yang dapat dipertimbangkan untuk penelitian lebih lanjut adalah sebagai berikut:

1. Dalam perekaman video, gunakan beberapa kondisi intensitas cahaya untuk mengetahui pengaruh intensitas cahaya dalam tingkat akurasi deteksi obyek dan perhitungan kecepatan.
2. Sebaiknya digunakan beberapa jenis variasi kecepatan obyek bergerak dalam video, sehingga selanjutnya dapat dianalisis pengaruh hubungan

kecepatan aktual obyek terhadap efektivitas metode estimasi yang digunakan.

3. Sebaiknya dikembangkan dengan percobaan deteksi dan perhitungan kecepatan pada beberapa obyek uji sekaligus dalam sebuah video.
4. Sebaiknya dikembangkan dengan mengkombinasikan metode ini dengan video secara *real-time* agar dapat langsung diaplikasikan pada sistem video pengawas.
5. Sebaiknya menggunakan pembanding yang lebih akurat sebagai acuan kecepatan aktual karena *speedometer* masih memiliki kekurangan jika digunakan sebagai pembanding.
6. Sebaiknya menggunakan metode deteksi obyek yang lebih teliti sehingga dalam perhitungan kecepatannya hanya terdapat sedikit *noise* yang diperoleh dari *background*. Akan lebih baik jika obyek dapat terdeteksi berdasarkan bentuk konturnya.

DAFTAR PUSTAKA

- [1] BPS. 2014. *Jumlah Kecelakaan, Korban Mati, Luka Berat, Luka Ringan, dan Kerugian Materi yang Diderita Tahun 1992-2014*. <http://www.bps.go.id/linkTableDinamis/view/id/1134>, diakses pada 29 Februari 2016.
- [2] Elysia. 2015. *Application of Computer Vision for Vehicle Tracking in Micro-Simulation*. Jakarta: Binus University Graduate Program.
- [3] Shapiro, Linda G., Stockman, George C. 2001. *Computer Vision*. New Jersey: Prentice Hall.
- [4] Hermanto, Bambang., Usman, Koredianto., Tirtoasmoro, Iwan Itut. 2009. *Implementasi Perhitungan Kecepatan Objek Bergerak Berbasis Webcam dan Pengolahan Citra Digital*. Bandung: IT Telkom.
- [5] Wahyudi, Danny Agus., Kartowisastro, Iman H. 2011. *Menghitung Kecepatan Menggunakan Computer Vision*. Jurnal Teknik Komputer Vol. 19, No. 2, Agustus 2011: 89 – 101.
- [6] Sadewo, Satrio Sani., Sumiharto, Raden., Candradewi, Ika. 2015. *Sistem Pengukur Kecepatan Kendaraan Berbasis Pengolahan Video*. IJEIS Vol. 5, No. 2, Oktober 2015, pp. 177 – 186.
- [7] Panjaitan, Bonanalia B. W., Usman, Koredianto., Susatio, Eko. 2010. *Implementasi Perhitungan Kecepatan Kendaraan Menggunakan Metode Optical Flow Berbasis Video Processing*. Bandung: IT Telkom.
- [8] Tekalp, A. Murat. 2015. *Digital Video Processing Second Edition*. Massachusetts: Pearson Education, Inc.
- [9] Marques, Oge. 2011. *Practical Image and Video Processing Using MATLAB*. Florida: Wiley & Sons, Inc.

- [10] Lubis, Nuzul Sakinah. 2015. *Analisis Perbandingan Kompresi File Video dengan Motion Picture Expert Group-4 dan Flash Video dengan Menggunakan Algoritma Huffman*. Medan: Universitas Sumatera Utara.
- [11] Chandra K, Ian. 2003. *Utility Audio/Video*. Jakarta: PT. Alex Media Komputindo.
- [12] Arioputra, Dimas. 2012. *Analisa Perbandingan Akurasi Metode Optical Flow dan Gaussian Mixture Model untuk Sistem Pemantau Lalu Lintas Berbasis Computer Vision*. Depok: Universitas Indonesia.
- [13] Bradsky, Gary., Kaehler, Adrian. 2008. *Learning OpenCV: Computer Vision with the OpenCV Library*. Sebastopol: O'Reilly Media Inc.
- [14] Trambadia, Dhara., Varnagar, Chintan., Kathiriya,. 2015. *Moving Object Detection and Tracking Using Hybrid Approach in Real Time to Improve Accuracy*. International Journal of Innovative Research in Computer and Communication Engineering Vol. 3, Issue 4, April 2015.
- [15] Horn, Berthold K.P., Schunck, Brian G. 1980. *Determining Optical Flow*. Massachusetts Institute of Technology Artificial Intelligence Laboratory.
- [16] Baron, J. L., Fleet, D. J., Beauchemin, S. S. 1994. *Performance of Optical Flow Techniques*. IJCV 12:1, pp. 43 – 77.

LAMPIRAN

Lampiran 1. Program yang digunakan dalam penelitian

Lampiran 1.1. Program deteksi obyek bergerak pada MATLAB

```
videoReader = vision.VideoFileReader('20-1.MOV');
detector = vision.ForegroundDetector(...
    'NumGaussians', 3, 'NumTrainingFrames', 50);

blob = vision.BlobAnalysis(...
    'CentroidOutputPort', false,
    'AreaOutputPort', false, ...
    'BoundingBoxOutputPort', true, ...
    'MinimumBlobAreaSource', 'Property',
    'MinimumBlobArea', 1000);

shapeInserter=vision.ShapeInserter('LineWidth', 3, '
BorderColor', 'White');

videoPlayer = vision.VideoPlayer();
while ~isDone(videoReader)
    frame = step(videoReader);
    fgMask = step(detector, frame);
    bbox = step(blob, fgMask);
    out = step(shapeInserter, frame, bbox);
    step(videoPlayer, out);
end

release(videoPlayer);
release(videoReader);
```


Lampiran 1.3. Program keseluruhan gabungan deteksi dan estimasi kecepatan obyek bergerak

```
videoReader = vision.VideoFileReader('20-1.MOV');
detector = vision.ForegroundDetector(...
    'NumGaussians',3,'NumTrainingFrames', 50);

blob = vision.BlobAnalysis(...
    'CentroidOutputPort',false,
    'AreaOutputPort', false, ...
    'BoundingBoxOutputPort', true, ...
    'MinimumBlobAreaSource','Property',
    'MaximumCount',1,'MinimumBlobArea',1000);

shapeInserter=vision.ShapeInserter('LineWidth',3,'
BorderColor','White');

videoPlayer = vision.VideoPlayer();
obj = VideoReader('20-1.MOV');

vid = read(obj);
frames = obj.NumberOfFrames;
z=0;
for x = 1 : frames
    frame = step(videoReader);
    fgMask = step(detector, frame);
    bbox = step(blob, fgMask);
    if isempty(bbox)
        z=z+1;
        continue
    else
        out = step(shapeInserter, frame, bbox);
        imwrite(vid(:,:,x),strcat('frame-
            ',num2str(x),'.jpg'));
        I=imread(strcat('frame-
            ',num2str(x),'.jpg'));
        cr=imcrop(I,bbox);
        imwrite(cr,strcat('crop-
            ',num2str(x),'.jpg'));
    end
end

z=z+1;
im1=imread(strcat('crop-',num2str(z),'.jpg'));
im2=imread(strcat('crop-',num2str(z+1),'.jpg'));

while size(im1,1)~=size(im2,1)
```


Lampiran 2. Hasil deteksi obyek bergerak

Lampiran 2.1. Deteksi obyek bergerak pada kecepatan 20 km/jam



Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video pertama



Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video kedua



Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video ketiga

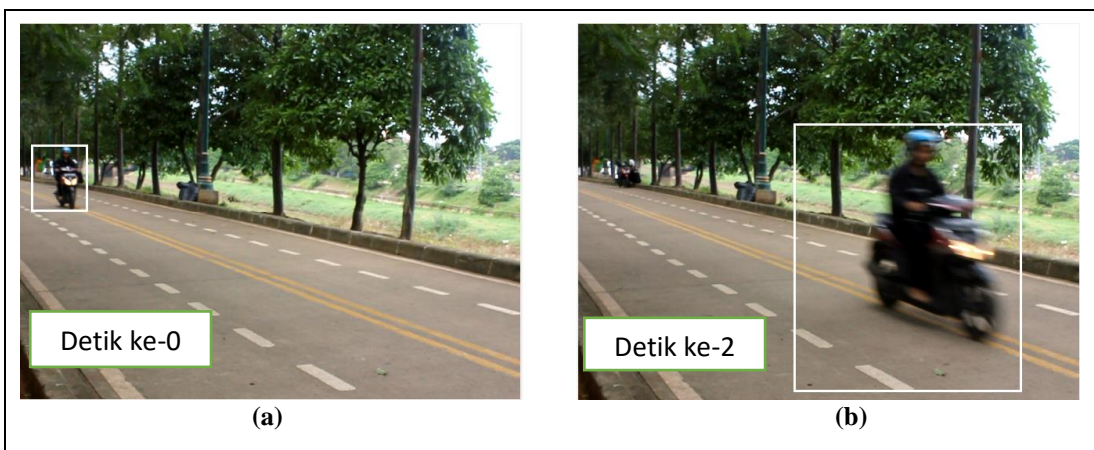


Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video keempat

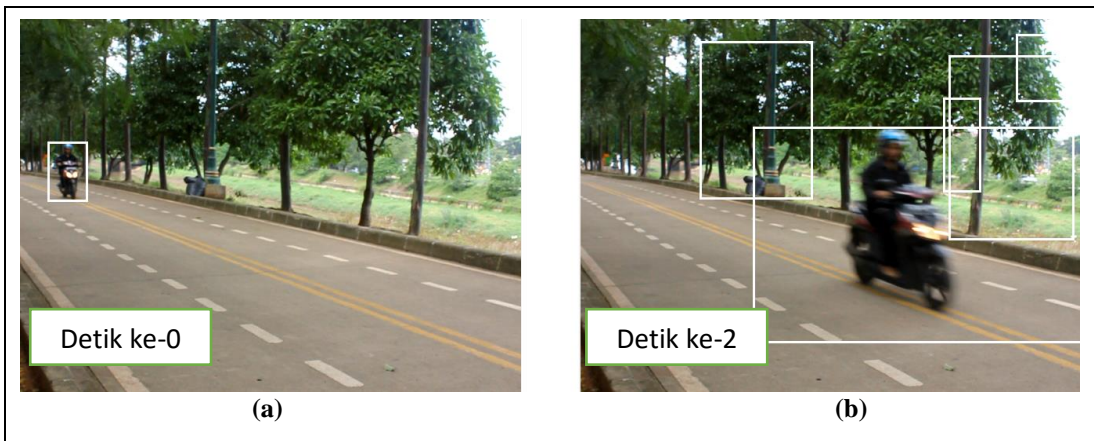


Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video kelima

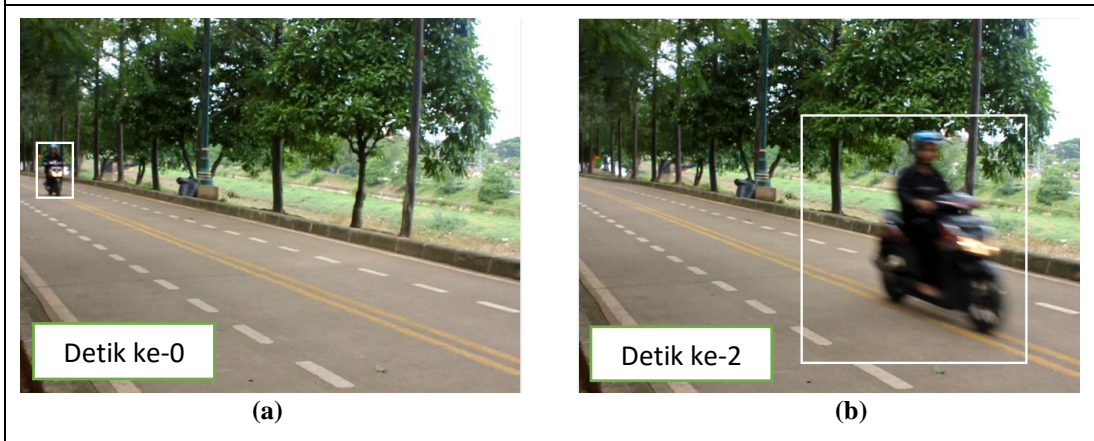
Lampiran 2.2. Deteksi obyek bergerak pada kecepatan 40 km/jam



Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video pertama



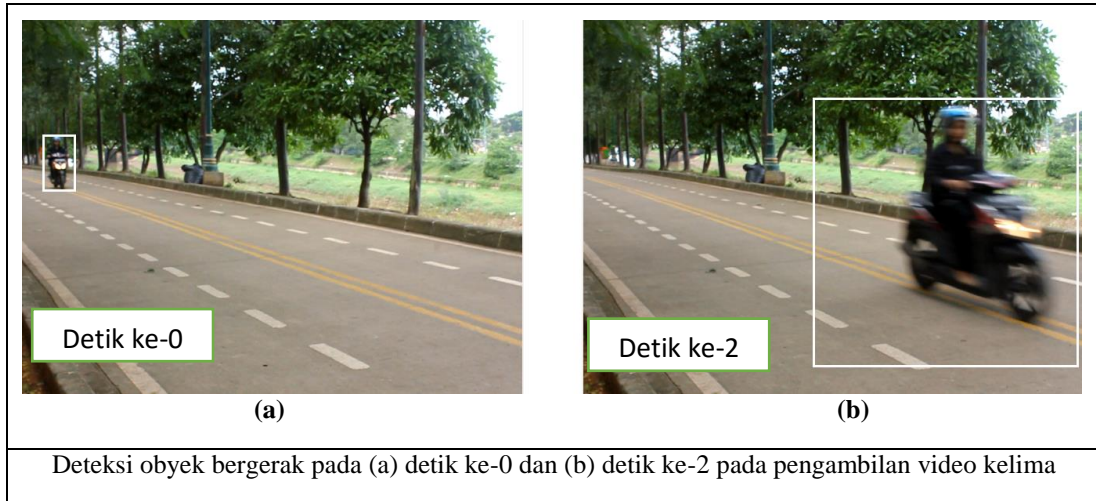
Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video kedua



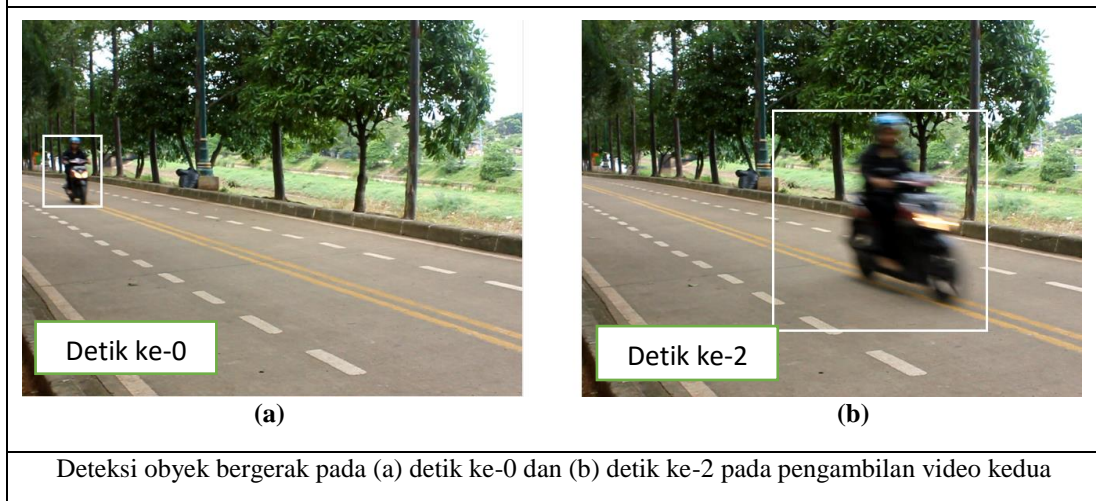
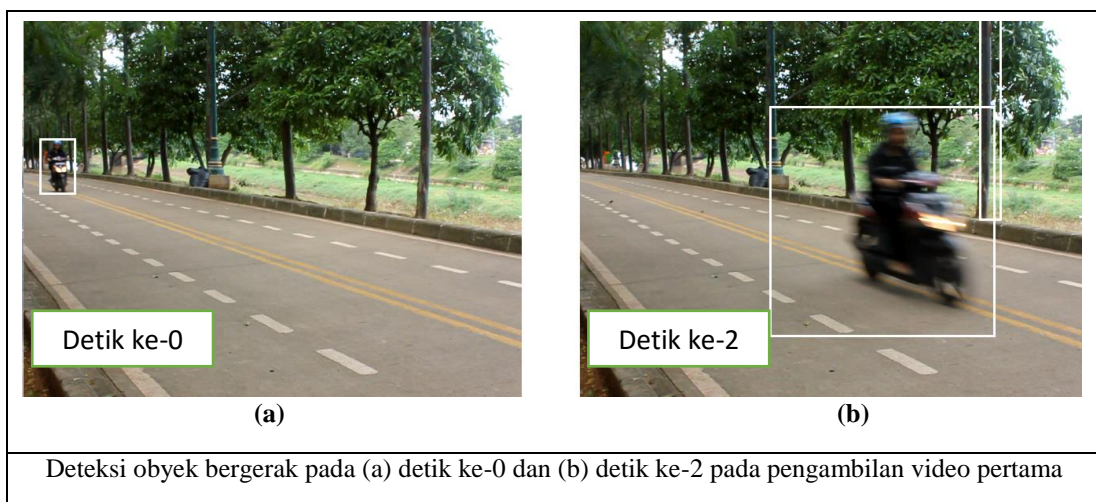
Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video ketiga

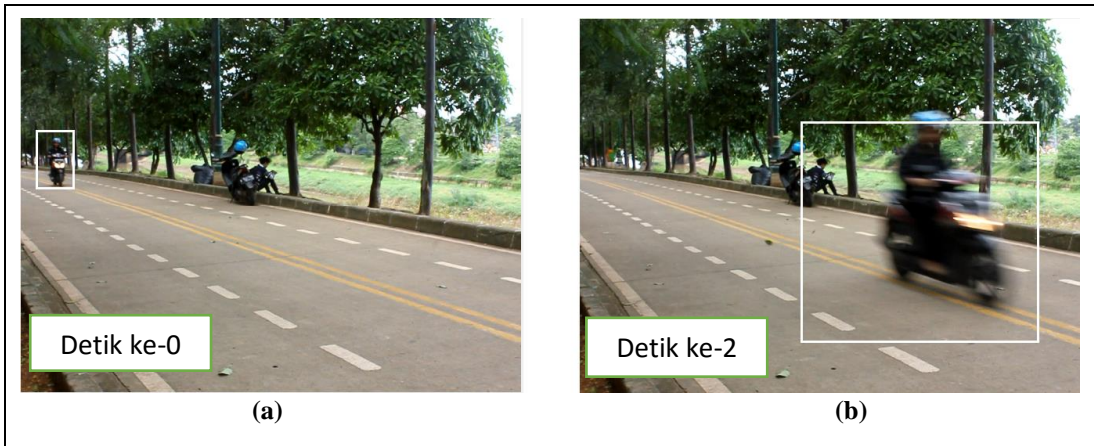


Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video keempat

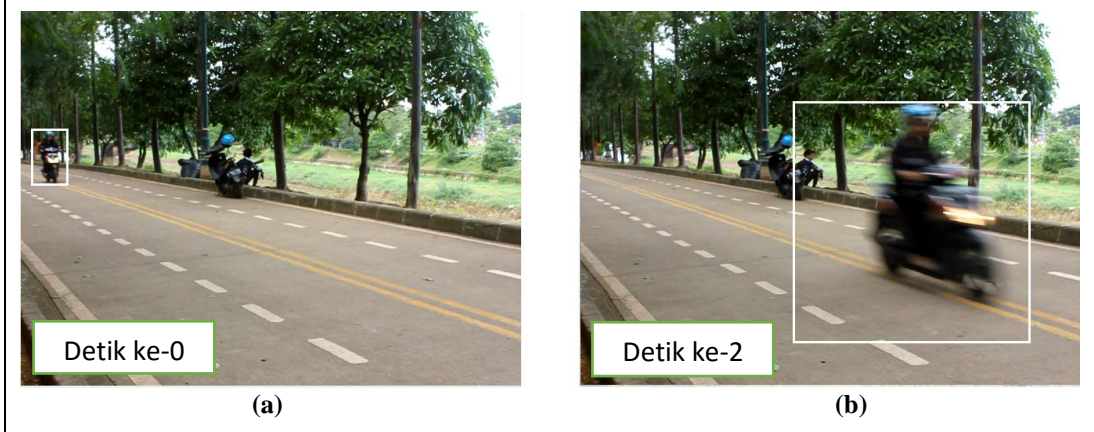


Lampiran 2.3. Deteksi obyek bergerak pada kecepatan 60 km/jam

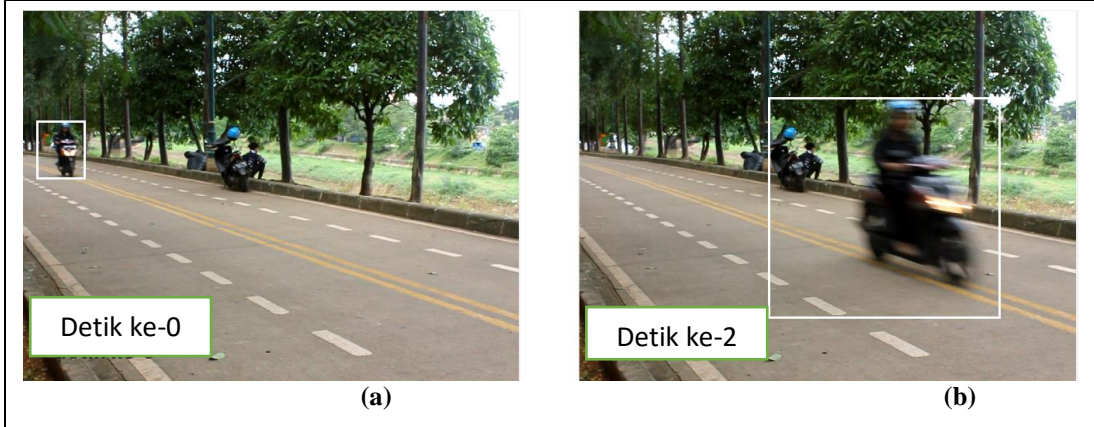




Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video ketiga








Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video keempat



Deteksi obyek bergerak pada (a) detik ke-0 dan (b) detik ke-2 pada pengambilan video kelima

Lampiran 3. Hasil *bounding box cropping* pada obyek bergerak yang terdeteksi

Lampiran 3.1. *Bounding box cropping* pada *frame* awal kecepatan 20 km/jam

 <p style="text-align: center;">(a) (b)</p>	 <p style="text-align: center;">(a) (b)</p>
<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen pertama</p>	<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen kedua</p>
 <p style="text-align: center;">(a) (b)</p>	 <p style="text-align: center;">(a) (b)</p>
<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen ketiga</p>	<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen keempat</p>
 <p style="text-align: center;">(a) (b)</p>	
<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen kelima</p>	

Lampiran 3.2. Bounding box cropping pada frame awal kecepatan 40 km/jam

 <p style="text-align: center;">(a) (b)</p>	 <p style="text-align: center;">(a) (b)</p>
<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen pertama</p>	<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen kedua</p>
 <p style="text-align: center;">(a) (b)</p>	 <p style="text-align: center;">(a) (b)</p>
<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen ketiga</p>	<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen keempat</p>
 <p style="text-align: center;">(a) (b)</p>	
<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame</i> sampel untuk estimasi kecepatan secara komputasi pada eksperimen kelima</p>	

Lampiran 3.3. Bounding box cropping pada frame awal kecepatan 60 km/jam

 <p style="text-align: center;">(a) (b)</p>	 <p style="text-align: center;">(a) (b)</p>
<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame sampel</i> untuk estimasi kecepatan secara komputasi pada eksperimen pertama</p>	<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame sampel</i> untuk estimasi kecepatan secara komputasi pada eksperimen kedua</p>
 <p style="text-align: center;">(a) (b)</p>	 <p style="text-align: center;">(a) (b)</p>
<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame sampel</i> untuk estimasi kecepatan secara komputasi pada eksperimen ketiga</p>	<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame sampel</i> untuk estimasi kecepatan secara komputasi pada eksperimen keempat</p>
 <p style="text-align: center;">(a) (b)</p>	
<p>Hasil <i>bounding box cropping</i> pada (a) t dan (b) $t+1$ sebagai <i>frame sampel</i> untuk estimasi kecepatan secara komputasi pada eksperimen kelima</p>	

Lampiran 4. Data hasil estimasi kecepatan obyek bergerak

Lampiran 4.1. Estimasi kecepatan obyek bergerak pada 20 km/jam

• **Eksperimen pertama**

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	21.093472	26	22.491456	51	22.685147	76	22.739104
2	21.279278	27	22.50478	52	22.688929	77	22.740315
3	21.419967	28	22.519571	53	22.692524	78	22.741478
4	21.551331	29	22.53594	54	22.695944	79	22.742593
5	21.648424	30	22.547222	55	22.699199	80	22.743664
6	21.728773	31	22.557561	56	22.702298	81	22.744691
7	21.817834	32	22.567035	57	22.70525	82	22.745678
8	21.881464	33	22.577068	58	22.708064	83	22.746625
9	21.955418	34	22.587725	59	22.709382	84	22.747535
10	22.000854	35	22.596351	60	22.711942	85	22.74841
11	22.054233	36	22.605714	61	22.714384	86	22.74925
12	22.103011	37	22.613145	62	22.716716	87	22.750057
13	22.148101	38	22.618686	63	22.718944	88	22.750832
14	22.182299	39	22.626454	64	22.721072	89	22.751578
15	22.226231	40	22.633769	65	22.723106	90	22.752295
16	22.255136	41	22.639303	66	22.723685	91	22.752984
17	22.28685	42	22.644446	67	22.725545	92	22.753647
18	22.317789	43	22.649226	68	22.727325	93	22.754285
19	22.34564	44	22.655028	69	22.729028	94	22.754898
20	22.37206	45	22.660514	70	22.730658	95	22.755488
21	22.393305	46	22.665706	71	22.73222	96	22.756056
22	22.416342	47	22.670623	72	22.733716	97	22.756602
23	22.437356	48	22.675282	73	22.735149	98	22.757127
24	22.45789	49	22.676975	74	22.736523	99	22.757633
25	22.475423	50	22.681167	75	22.737841	100	22.75812
						Rata-rata	22.52704764

- Eksperimen kedua

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	23.607255	26	25.153232	51	25.295409	76	25.320428
2	23.802902	27	25.166565	52	25.296471	77	25.320856
3	23.972757	28	25.176053	53	25.29866	78	25.321257
4	24.108529	29	25.18822	54	25.300699	79	25.321632
5	24.236379	30	25.198036	55	25.3026	80	25.321984
6	24.344717	31	25.204318	56	25.304372	81	25.322314
7	24.441944	32	25.213566	57	25.304735	82	25.322622
8	24.52832	33	25.220721	58	25.306277	83	25.322912
9	24.601001	34	25.227134	59	25.307715	84	25.323183
10	24.673159	35	25.2303	60	25.309057	85	25.323437
11	24.727309	36	25.236701	61	25.31031	86	25.323676
12	24.778543	37	25.243813	62	25.31148	87	25.323899
13	24.825392	38	25.250398	63	25.312572	88	25.324109
14	24.863695	39	25.256497	64	25.312304	89	25.324305
15	24.901747	40	25.260863	65	25.313257	90	25.324489
16	24.941465	41	25.266105	66	25.314148	91	25.324662
17	24.975764	42	25.267111	67	25.314981	92	25.324824
18	25.003836	43	25.271629	68	25.31576	93	25.324976
19	25.028647	44	25.275824	69	25.316488	94	25.325119
20	25.050533	45	25.27972	70	25.317169	95	25.325253
21	25.067263	46	25.283341	71	25.317806	96	25.325379
22	25.091871	47	25.285419	72	25.318402	97	25.325497
23	25.109253	48	25.28855	73	25.31896	98	25.325607
24	25.120895	49	25.291463	74	25.319482	99	25.325711
25	25.139765	50	25.292886	75	25.31997	100	25.325809
						Rata-rata	25.1382443

- Eksperimen ketiga

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	23.489787	26	24.794173	51	24.931264	76	24.951572
2	23.66132	27	24.806775	52	24.933444	77	24.952448
3	23.821336	28	24.822955	53	24.936644	78	24.953282
4	23.938164	29	24.835566	54	24.939663	79	24.952864
5	24.057277	30	24.8447	55	24.940089	80	24.953619
6	24.148944	31	24.851656	56	24.941568	81	24.953126
7	24.226069	32	24.858942	57	24.942898	82	24.952599
8	24.298584	33	24.865407	58	24.944089	83	24.953251
9	24.360285	34	24.87354	59	24.946362	84	24.952661
10	24.411782	35	24.882183	60	24.9473	85	24.950831
11	24.45212	36	24.888968	61	24.944486	86	24.951396
12	24.492173	37	24.892734	62	24.9452	87	24.951934
13	24.536452	38	24.895947	63	24.947023	88	24.952448
14	24.575002	39	24.897449	64	24.945115	89	24.951727
15	24.60374	40	24.903323	65	24.94554	90	24.952195
16	24.628097	41	24.906332	66	24.94588	91	24.952642
17	24.654579	42	24.910141	67	24.946139	92	24.953069
18	24.678818	43	24.911145	68	24.947533	93	24.953476
19	24.696348	44	24.915433	69	24.947644	94	24.953865
20	24.715951	45	24.915755	70	24.948899	95	24.954237
21	24.733077	46	24.919404	71	24.95009	96	24.954593
22	24.743161	47	24.922763	72	24.950009	97	24.954932
23	24.760884	48	24.925854	73	24.951083	98	24.955257
24	24.771966	49	24.927483	74	24.950892	99	24.955568
25	24.782615	50	24.928877	75	24.95065	100	24.955865
						Rata-rata	24.79784997

- Eksperimen keempat

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	20.380711	26	21.42314	51	21.46561	76	21.457265
2	20.53653	27	21.43136	52	21.467068	77	21.455124
3	20.662644	28	21.434924	53	21.465957	78	21.455432
4	20.762728	29	21.441357	54	21.464759	79	21.455724
5	20.857434	30	21.443316	55	21.46595	80	21.456002
6	20.930911	31	21.442113	56	21.465829	81	21.456266
7	20.999501	32	21.444006	57	21.4644	82	21.456516
8	21.059431	33	21.446571	58	21.465377	83	21.456754
9	21.112589	34	21.448629	59	21.465057	84	21.45698
10	21.155585	35	21.44775	60	21.463449	85	21.457195
11	21.190924	36	21.450163	61	21.46426	86	21.4574
12	21.220911	37	21.454663	62	21.463785	87	21.456362
13	21.251367	38	21.455099	63	21.460798	88	21.456548
14	21.278129	39	21.457691	64	21.461471	89	21.45426
15	21.299362	40	21.461226	65	21.462105	90	21.45443
16	21.324238	41	21.463254	66	21.461467	91	21.454592
17	21.33342	42	21.463802	67	21.459563	92	21.454746
18	21.348329	43	21.464131	68	21.458861	93	21.454894
19	21.365591	44	21.461788	69	21.45813	94	21.455035
20	21.379302	45	21.46173	70	21.457372	95	21.455169
21	21.390961	46	21.461498	71	21.457821	96	21.455297
22	21.395849	47	21.46358	72	21.457012	97	21.45542
23	21.404085	48	21.465514	73	21.457415	98	21.455537
24	21.409628	49	21.464841	74	21.457795	99	21.455649
25	21.416345	50	21.465282	75	21.456923	100	21.455757
						Rata-rata	21.37726521

- Eksperimen kelima

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	22.95352	26	24.417907	51	24.576762	76	24.606641
2	23.127765	27	24.433224	52	24.578242	77	24.607226
3	23.278041	28	24.443114	53	24.580941	78	24.60637
4	23.403543	29	24.456061	54	24.583469	79	24.606887
5	23.518647	30	24.465174	55	24.585836	80	24.607372
6	23.624633	31	24.476122	56	24.58665	81	24.607828
7	23.717865	32	24.486188	57	24.587325	82	24.608257
8	23.791335	33	24.495445	58	24.589273	83	24.60866
9	23.860553	34	24.503954	59	24.5911	84	24.609039
10	23.923071	35	24.508976	60	24.592812	85	24.609396
11	23.980227	36	24.513365	61	24.594417	86	24.609731
12	24.031765	37	24.51717	62	24.595923	87	24.610046
13	24.078544	38	24.52323	63	24.597335	88	24.608938
14	24.121351	39	24.525987	64	24.598659	89	24.609217
15	24.15805	40	24.532477	65	24.599902	90	24.60948
16	24.196095	41	24.538531	66	24.599664	91	24.609727
17	24.222115	42	24.544182	67	24.600759	92	24.609959
18	24.253105	43	24.549457	68	24.601786	93	24.610177
19	24.275566	44	24.551582	69	24.602751	94	24.610383
20	24.305057	45	24.556187	70	24.603656	95	24.610577
21	24.332133	46	24.560491	71	24.603102	96	24.610759
22	24.351473	47	24.564514	72	24.603901	97	24.61093
23	24.36329	48	24.568275	73	24.604652	98	24.611092
24	24.383076	49	24.570391	74	24.605357	99	24.611244
25	24.401233	50	24.573682	75	24.606019	100	24.611387
						Rata-rata	24.41629355

Lampiran 4.2. Estimasi kecepatan obyek bergerak pada 40 km/jam

• Eksperimen pertama

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	31.001898	26	33.935077	51	34.290239	76	34.316155
2	31.357041	27	33.961267	52	34.299628	77	34.311393
3	31.6289	28	33.992506	53	34.30431	78	34.311892
4	31.878131	29	34.018067	54	34.302997	79	34.313583
5	32.116013	30	34.04634	55	34.31337	80	34.315115
6	32.302794	31	34.073454	56	34.3164	81	34.316493
7	32.498072	32	34.089967	57	34.321669	82	34.319083
8	32.655969	33	34.108428	58	34.315608	83	34.318818
9	32.790038	34	34.113869	59	34.315888	84	34.319777
10	32.931461	35	34.128368	60	34.310342	85	34.321965
11	33.062782	36	34.150652	61	34.3139	86	34.321315
12	33.165223	37	34.1631	62	34.314356	87	34.323263
13	33.247476	38	34.176739	63	34.315825	88	34.321026
14	33.310522	39	34.191665	64	34.316955	89	34.320039
15	33.397344	40	34.202511	65	34.316441	90	34.323019
16	33.470541	41	34.214829	66	34.321016	91	34.325897
17	33.554117	42	34.223234	67	34.322576	92	34.325966
18	33.600959	43	34.233258	68	34.325219	93	34.321875
19	33.658505	44	34.239514	69	34.32352	94	34.323122
20	33.72062	45	34.24616	70	34.322933	95	34.324285
21	33.770106	46	34.25734	71	34.320745	96	34.325365
22	33.822422	47	34.267648	72	34.30886	97	34.326366
23	33.842564	48	34.2785	73	34.313025	98	34.32729
24	33.880076	49	34.284487	74	34.312905	99	34.329498
25	33.909431	50	34.287023	75	34.311228	100	34.328921
						Rata-rata	33.94338484

- Eksperimen kedua

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	51.59844	26	54.114399	51	54.363708	76	54.411775
2	52.014116	27	54.127721	52	54.366664	77	54.409429
3	52.290024	28	54.150278	53	54.364006	78	54.393407
4	52.591848	29	54.148548	54	54.369389	79	54.394263
5	52.811422	30	54.160213	55	54.374373	80	54.395062
6	52.94075	31	54.192225	56	54.378989	81	54.395806
7	53.101893	32	54.187334	57	54.379934	82	54.396501
8	53.191377	33	54.171613	58	54.383897	83	54.397148
9	53.331723	34	54.175185	59	54.38757	84	54.397753
10	53.391077	35	54.19496	60	54.387645	85	54.398317
11	53.523953	36	54.217875	61	54.390804	86	54.398844
12	53.631837	37	54.227119	62	54.393736	87	54.399335
13	53.69081	38	54.246484	63	54.396456	88	54.399795
14	53.772158	39	54.260983	64	54.398982	89	54.400224
15	53.79766	40	54.277394	65	54.397995	90	54.400624
16	53.812615	41	54.292511	66	54.400174	91	54.400999
17	53.853869	42	54.306441	67	54.402199	92	54.401349
18	53.897262	43	54.319284	68	54.404081	93	54.401676
19	53.938389	44	54.327799	69	54.402499	94	54.401982
20	54.005091	45	54.338729	70	54.404126	95	54.402268
21	53.979059	46	54.342159	71	54.40564	96	54.402536
22	54.009274	47	54.342995	72	54.407049	97	54.402786
23	54.045753	48	54.348268	73	54.40836	98	54.403021
24	54.058639	49	54.349562	74	54.40958	99	54.40324
25	54.092493	50	54.356912	75	54.410716	100	54.403445
						Rata-rata	54.1134868

- Eksperimen ketiga

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	31.160143	26	34.473251	51	34.901694	76	35.013154
2	31.460067	27	34.501092	52	34.91319	77	35.016509
3	31.765518	28	34.541049	53	34.916972	78	35.017934
4	32.012095	29	34.572477	54	34.923752	79	35.021
5	32.273241	30	34.590452	55	34.93	80	35.023931
6	32.473798	31	34.618056	56	34.93219	81	35.01783
7	32.699992	32	34.641172	57	34.939244	82	35.020514
8	32.897323	33	34.672553	58	34.944076	83	35.021302
9	33.047773	34	34.699991	59	34.950267	84	35.02376
10	33.211203	35	34.716732	60	34.956064	85	35.026113
11	33.369124	36	34.72628	61	34.959711	86	35.028365
12	33.463989	37	34.751621	62	34.964785	87	35.030522
13	33.59783	38	34.766473	63	34.967746	88	35.030805
14	33.704153	39	34.790478	64	34.973953	89	35.032784
15	33.793137	40	34.805979	65	34.978082	90	35.034679
16	33.895691	41	34.818405	66	34.98371	91	35.036496
17	33.962087	42	34.831386	67	34.987291	92	35.036456
18	34.0389	43	34.836362	68	34.99062	93	35.036344
19	34.086762	44	34.852477	69	34.993709	94	35.037946
20	34.146501	45	34.856934	70	34.998352	95	35.039483
21	34.206656	46	34.864023	71	35.002779	96	35.040957
22	34.259099	47	34.868467	72	35.007001	97	35.042371
23	34.321842	48	34.873881	73	35.009249	98	35.043728
24	34.369159	49	34.883861	74	35.013095	99	35.045031
25	34.424167	50	34.894895	75	35.013206	100	35.044499
						Rata-rata	34.51001848

- Eksperimen keempat

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	45.757107	26	49.21584	51	49.742143	76	49.885725
2	46.156526	27	49.243544	52	49.764821	77	49.886284
3	46.475143	28	49.290797	53	49.772449	78	49.895582
4	46.803958	29	49.333608	54	49.781114	79	49.898632
5	47.065355	30	49.387703	55	49.787421	80	49.902588
6	47.294888	31	49.4387	56	49.793829	81	49.905907
7	47.504189	32	49.465434	57	49.793345	82	49.905646
8	47.698936	33	49.490296	58	49.80332	83	49.910724
9	47.872305	34	49.484748	59	49.805537	84	49.91664
10	47.987031	35	49.517228	60	49.815016	85	49.914114
11	48.144574	36	49.541759	61	49.814121	86	49.917097
12	48.250448	37	49.571172	62	49.826294	87	49.913906
13	48.388034	38	49.591818	63	49.817847	88	49.911578
14	48.476202	39	49.618922	64	49.824781	89	49.91585
15	48.564794	40	49.628289	65	49.820714	90	49.916389
16	48.666803	41	49.640554	66	49.830826	91	49.920102
17	48.733833	42	49.670298	67	49.836811	92	49.924744
18	48.806042	43	49.667847	68	49.85027	93	49.92555
19	48.899771	44	49.686208	69	49.853924	94	49.923836
20	48.960988	45	49.681892	70	49.858153	95	49.92998
21	49.01338	46	49.698663	71	49.861781	96	49.937092
22	49.052561	47	49.711473	72	49.866014	97	49.940534
23	49.110002	48	49.719084	73	49.868519	98	49.947218
24	49.136922	49	49.722814	74	49.880854	99	49.945597
25	49.188257	50	49.726305	75	49.876615	100	49.939116
						Rata-rata	49.34229995

- Eksperimen kelima

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	31.162016	26	34.417747	51	34.939566	76	35.096561
2	31.504885	27	34.465537	52	34.944009	77	35.09942
3	31.78387	28	34.505175	53	34.953995	78	35.103713
4	32.042035	29	34.549457	54	34.96013	79	35.106209
5	32.297537	30	34.567168	55	34.968902	80	35.108536
6	32.506231	31	34.604088	56	34.977115	81	35.112321
7	32.688002	32	34.62261	57	34.983187	82	35.11595
8	32.851685	33	34.659416	58	34.991986	83	35.111635
9	33.028656	34	34.689232	59	35.001926	84	35.113354
10	33.167584	35	34.704525	60	35.0098	85	35.116558
11	33.294278	36	34.735164	61	35.018862	86	35.118012
12	33.421281	37	34.754566	62	35.025902	87	35.120962
13	33.53705	38	34.767491	63	35.034174	88	35.122174
14	33.635226	39	34.783492	64	35.040463	89	35.124893
15	33.720672	40	34.798236	65	35.048021	90	35.127503
16	33.825621	41	34.805377	66	35.055247	91	35.130009
17	33.91683	42	34.819057	67	35.062159	92	35.132416
18	33.990238	43	34.836143	68	35.068771	93	35.134728
19	34.055961	44	34.852107	69	35.075097	94	35.135326
20	34.122589	45	34.867014	70	35.079533	95	35.135838
21	34.160788	46	34.882538	71	35.07591	96	35.137887
22	34.226329	47	34.898736	72	35.08146	97	35.138235
23	34.285017	48	34.914052	73	35.085156	98	35.140128
24	34.338876	49	34.928536	74	35.087009	99	35.140325
25	34.372473	50	34.934422	75	35.091887	100	35.142073
						Rata-rata	34.54022649

Lampiran 4.3. Estimasi kecepatan obyek bergerak pada 60 km/jam

• Eksperimen pertama

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	47.145288	26	52.55122	51	53.477106	76	53.763335
2	47.694601	27	52.625883	52	53.498525	77	53.767041
3	48.155419	28	52.684876	53	53.518864	78	53.774831
4	48.600905	29	52.735771	54	53.538179	79	53.780123
5	48.974339	30	52.791279	55	53.556527	80	53.787327
6	49.322107	31	52.8373	56	53.569578	81	53.794257
7	49.637034	32	52.893634	57	53.583956	82	53.800924
8	49.945475	33	52.950501	58	53.589618	83	53.805139
9	50.181454	34	53.001671	59	53.604582	84	53.804714
10	50.400552	35	53.049551	60	53.620998	85	53.810658
11	50.615589	36	53.098701	61	53.63671	86	53.81638
12	50.804274	37	53.126963	62	53.649559	87	53.82189
13	50.990121	38	53.168358	63	53.659578	88	53.827197
14	51.18788	39	53.209413	64	53.668991	89	53.832309
15	51.338744	40	53.242379	65	53.67783	90	53.837235
16	51.498155	41	53.270941	66	53.692709	91	53.839779
17	51.647556	42	53.293858	67	53.704871	92	53.844353
18	51.773689	43	53.312615	68	53.710839	93	53.848761
19	51.888261	44	53.347533	69	53.722028	94	53.850809
20	51.987607	45	53.368451	70	53.730568	95	53.854906
21	52.113536	46	53.38691	71	53.740875	96	53.858857
22	52.203764	47	53.395935	72	53.746374	97	53.860465
23	52.29779	48	53.420165	73	53.749284	98	53.859737
24	52.37977	49	53.440852	74	53.756218	99	53.863282
25	52.468404	50	53.466818	75	53.757102	100	53.8623
						Rata-rata	52.7965197

- Eksperimen kedua

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	52.893483	26	58.799358	51	60.072736	76	60.426906
2	53.395844	27	58.898079	52	60.092942	77	60.43456
3	53.929097	28	58.973434	53	60.102965	78	60.447073
4	54.423009	29	59.056589	54	60.139494	79	60.457372
5	54.876554	30	59.136503	55	60.160322	80	60.467185
6	55.300149	31	59.208247	56	60.188511	81	60.47653
7	55.68254	32	59.275664	57	60.208527	82	60.487135
8	55.964234	33	59.334006	58	60.222962	83	60.4956
9	56.269109	34	59.40893	59	60.236443	84	60.50536
10	56.537062	35	59.466	60	60.251007	85	60.505915
11	56.723914	36	59.535607	61	60.260747	86	60.50609
12	56.934477	37	59.585237	62	60.266288	87	60.512997
13	57.130454	38	59.632374	63	60.28812	88	60.51758
14	57.321987	39	59.663328	64	60.303292	89	60.52721
15	57.514611	40	59.70099	65	60.318088	90	60.532835
16	57.676367	41	59.733418	66	60.333696	91	60.538403
17	57.842841	42	59.785067	67	60.339093	92	60.547089
18	57.997114	43	59.825411	68	60.358136	93	60.553774
19	58.125928	44	59.864416	69	60.372711	94	60.561889
20	58.225082	45	59.895777	70	60.384831	95	60.569735
21	58.344866	46	59.937545	71	60.396229	96	60.575609
22	58.420329	47	59.952779	72	60.402972	97	60.577813
23	58.532589	48	59.98093	73	60.409319	98	60.579522
24	58.626949	49	60.012412	74	60.41503	99	60.584675
25	58.709544	50	60.043652	75	60.415013	100	60.589605
						Rata-rata	59.26021822

• Eksperimen ketiga

<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	47.905015
2	48.576625
3	49.143328
4	49.637919
5	50.044414
6	50.419427
7	50.838288
8	51.20386
9	51.498849
10	51.789255
11	52.01451
12	52.236041
13	52.413377
14	52.621518
15	52.759663
16	52.959541
17	53.087654
18	53.237578
19	53.370804
20	53.486616
21	53.580147
22	53.684232
23	53.782476
24	53.890253
25	53.988243

<i>Frame</i>	Kecepatan Estimasi (km/jam)
26	54.073271
27	54.148683
28	54.224274
29	54.30876
30	54.391405
31	54.459352
32	54.524288
33	54.582811
34	54.637294
35	54.686114
36	54.729569
37	54.784207
38	54.810161
39	54.844846
40	54.880946
41	54.926287
42	54.956334
43	54.988344
44	55.018727
45	55.036172
46	55.061748
47	55.098481
48	55.116078
49	55.138423
50	55.156076

<i>Frame</i>	Kecepatan Estimasi (km/jam)
51	55.183429
52	55.209158
53	55.231459
54	55.246643
55	55.262297
56	55.270999
57	55.289841
58	55.310342
59	55.328829
60	55.350991
61	55.374071
62	55.390472
63	55.411706
64	55.426356
65	55.437379
66	55.437166
67	55.44386
68	55.452653
69	55.467405
70	55.481516
71	55.489283
72	55.506042
73	55.50604
74	55.510359
75	55.516968

<i>Frame</i>	Kecepatan Estimasi (km/jam)
76	55.527807
77	55.538182
78	55.539569
79	55.543336
80	55.54862
81	55.555434
82	55.558047
83	55.56222
84	55.561339
85	55.562937
86	55.568057
87	55.576707
88	55.583153
89	55.591237
90	55.592431
91	55.599994
92	55.601566
93	55.602022
94	55.598423
95	55.603144
96	55.607655
97	55.598564
98	55.604601
99	55.606622
100	55.60567
Rata-rata	54.40724885

- Eksperimen keempat

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	50.18482	26	56.198414	51	57.467884	76	57.865121
2	50.832042	27	56.280063	52	57.501699	77	57.87636
3	51.422342	28	56.368421	53	57.530205	78	57.880987
4	51.879472	29	56.454375	54	57.554681	79	57.88929
5	52.263344	30	56.521981	55	57.567293	80	57.897077
6	52.63675	31	56.593899	56	57.581997	81	57.902546
7	52.916326	32	56.673805	57	57.607936	82	57.907542
8	53.256787	33	56.730885	58	57.634346	83	57.909128
9	53.550654	34	56.793346	59	57.658242	84	57.92109
10	53.764443	35	56.845205	60	57.680334	85	57.916891
11	54.017604	36	56.905483	61	57.689151	86	57.928018
12	54.222749	37	56.963507	62	57.704683	87	57.930889
13	54.435742	38	57.015252	63	57.708263	88	57.941253
14	54.626027	39	57.069405	64	57.727681	89	57.938609
15	54.82038	40	57.115882	65	57.743711	90	57.946429
16	54.990654	41	57.167615	66	57.758799	91	57.951527
17	55.13518	42	57.195769	67	57.769912	92	57.95869
18	55.287226	43	57.224535	68	57.787459	93	57.965545
19	55.425686	44	57.269251	69	57.793916	94	57.969717
20	55.546297	45	57.301716	70	57.808023	95	57.975987
21	55.680604	46	57.339562	71	57.805636	96	57.979598
22	55.791641	47	57.36843	72	57.820075	97	57.985323
23	55.920111	48	57.39504	73	57.831996	98	57.990794
24	56.012385	49	57.418337	74	57.849291	99	57.996018
25	56.107235	50	57.436502	75	57.857517	100	57.998623
						Rata-rata	56.65236963

- Eksperimen kelima

<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)	<i>Frame</i>	Kecepatan Estimasi (km/jam)
1	56.518185	26	59.974064	51	60.402298	76	60.545005
2	56.904766	27	60.015158	52	60.40989	77	60.545241
3	57.32825	28	60.053187	53	60.412049	78	60.545249
4	57.637786	29	60.088208	54	60.422779	79	60.549929
5	57.926269	30	60.123599	55	60.430464	80	60.549513
6	58.106774	31	60.145376	56	60.439673	81	60.539112
7	58.339838	32	60.185126	57	60.445923	82	60.543189
8	58.477609	33	60.218317	58	60.453784	83	60.544813
9	58.685028	34	60.233823	59	60.463651	84	60.548541
10	58.792811	35	60.26066	60	60.470304	85	60.552108
11	58.935597	36	60.27527	61	60.479028	86	60.555519
12	59.058436	37	60.301999	62	60.489499	87	60.558783
13	59.1927	38	60.314327	63	60.494576	88	60.554738
14	59.313823	39	60.326672	64	60.499176	89	60.557718
15	59.397568	40	60.356156	65	60.50822	90	60.555679
16	59.502294	41	60.34257	66	60.516843	91	60.558408
17	59.577268	42	60.329649	67	60.525066	92	60.56102
18	59.618111	43	60.339098	68	60.530638	93	60.56352
19	59.647781	44	60.337523	69	60.533219	94	60.565914
20	59.703603	45	60.343875	70	60.535453	95	60.568207
21	59.758455	46	60.348783	71	60.535072	96	60.570403
22	59.823096	47	60.361823	72	60.531755	97	60.572506
23	59.854238	48	60.363844	73	60.533028	98	60.574521
24	59.901422	49	60.379369	74	60.534021	99	60.57645
25	59.93701	50	60.383997	75	60.539639	100	60.578299
						Rata-rata	60.04411624

DAFTAR RIWAYAT HIDUP



Novia Angelita merupakan puteri sulung dari Ibu Maryam dan Bapak Herman Sagata yang lahir di Jakarta, 21 November 1994.

Pendidikan formal dimulai dari SDN Mekarsari 01 Tambun Selatan (2000 – 2006), SMP Tamansiswa 1 Jakarta (2006 – 2009), SMAN 5 Jakarta (2009 – 2012) dan Program Studi Fisika FMIPA Universitas Negeri Jakarta (September 2012 – Agustus 2016). Semasa kuliah pernah menjadi asisten laboratorium Fisika Modern dan Fisika Komputasi, peserta olimpiade OSN-Pertamina dan menjadi pemakalah Seminar Nasional Fisika 2016 dalam bidang komputasi di Universitas Negeri Jakarta. Pernah bekerja *part time* sebagai pengajar *private* untuk mata pelajaran Matematika, Fisika dan Bahasa Inggris untuk SD, SMP dan SMA. Pernah bekerjasama dengan Pusat Teknologi dan Data Penginderaan Jauh Lembaga Penerbangan dan Antariksa Nasional (LAPAN) untuk melaksanakan Praktik Kerja Lapangan pada tahun 2015.

Di luar bidang akademik, penulis pernah aktif dalam organisasi BEM Jurusan Fisika UNJ sebagai Staf Department Komunikasi dan Informasi pada tahun 2013-2014 serta menjadi Koordinator Humas untuk acara Pekan Ilmiah Fisika 2014.