

## Sistem Pengukur Kecepatan Kendaraan Berbasis Pengolahan Video

Satrio Sani Sadewo<sup>\*1</sup>, Raden Sumiharto<sup>2</sup>, Ika Candradewi<sup>3</sup>

<sup>1</sup>Prodi Elektronika dan Instrumentasi FMIPA, UGM Yogyakarta

<sup>2,3</sup>Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: <sup>\*</sup>[sani.rio@gmail.com](mailto:sani.rio@gmail.com), <sup>2</sup>[r\\_sumiharto@ugm.ac.id](mailto:r_sumiharto@ugm.ac.id), <sup>3</sup>[ika.candra.kc@gmail.com](mailto:ika.candra.kc@gmail.com)

### Abstrak

Sistem pengukur kecepatan kendaraan berbasis pengolahan video ini merupakan salah satu sistem yang memanfaatkan sistem pengolahan citra digital sebagai pendeteksi kendaraan dan mengukur kecepatannya. Metode yang digunakan dalam sistem ini adalah background subtraction dengan algoritma Gaussian Mixture Model (GMM). Background subtraction akan memisahkan antara background dengan objek yang dideteksi, yaitu kendaraan. Koordinat titik tengah objek dijadikan sebagai nilai perpindahan objek dalam satuan piksel. Jarak sesungguhnya diukur dalam satuan meter. Jarak pada citra dibatasi dengan region of interest (ROI) sebesar 160 piksel. Setelah diperoleh waktu perpindahan tiap pikselnya maka nilai piksel/detik dikonversikan menjadi km/jam.

Uji coba sistem dilakukan pada validasi kecepatan, pengukuran kecepatan, dan pengaruh intensitas cahaya. Proses validasi kecepatan menggunakan nilai kecepatan rata-rata 3 frame awal sebagai acuan untuk pengukuran kecepatan di frame berikutnya. Akurasi nilai kecepatan rata-rata 3 frame awal ini memberikan persentase eror 1,92 % - 15,75 % sedangkan ketika validasi tersebut dilakukan pada pembacaan keseluruhan frame video menghasilkan rentang eror 1,21 % - 21,37 %. Sistem bekerja dengan baik pada kondisi pagi, siang, dan sore hari dengan rentang intensitas cahaya 600-1900 lux, sedangkan pada malam hari dengan rentang intensitas cahaya 0-5 lux, sistem tidak bisa bekerja dengan baik.

**Kata kunci**— pengolahan video, pengukuran kecepatan, background subtraction, gaussian mixture model, region of interest

### Abstract

This system is implemented by digital image processing to detect the objects and measure the speed. This system using background subtraction method with Gaussian Mixture Model (GMM) algorithm. Background subtraction will separate background and detected objects. Coordinates of the objects midpoint used as the the object moving value in pixel. The actual distance also measured in meters where the distance is limited by region of interest (ROI). The ROI is 160 pixel. Having obtained the moving objects time from previous frame to current frame so the value of pixel/s can converted to km/h.

System testing the measurement validation, calculate the speed after being validated, and the influence of light intensity. The speed validation process uses average speed of early three frames speed as the reference for the speed measurement in the next frame. The average speed accuracy of 3 frames early gives a percentage error about 1,92% - 15,75%. When validation is performed on the entire reading frame of video, it produces an error range 1,21% - 21,37%. The system works well in the morning, afternoon, and evening conditions with light intensity about 600-1900 lux. While at night with 0-5 lux light intensity range, the system can't work properly.

**Keywords**— video processing, speed measurement, background subtraction, gaussian mixture model, region of interest

## 1. PENDAHULUAN

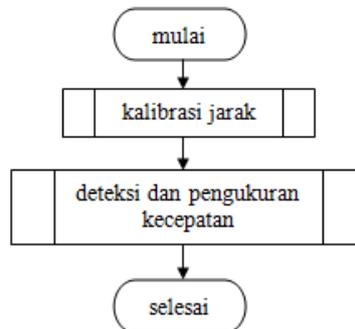
Salah satu faktor yang menyebabkan meningkatnya angka kecelakaan adalah pengemudi kendaraan yang memacu kendaraan dengan sangat cepat dan melebihi batas kecepatan yang ditentukan. Upaya yang telah dilakukan adalah dengan memasang kamera *CCTV* pada tempat-tempat strategis sekitar jalan [1]. Penggunaan kamera dalam metode ini dapat dimanfaatkan dengan menambahkan teknologi *image processing* di dalamnya. Adapun salah satu parameter yang dapat dipantau melalui sistem *monitoring* ini adalah kecepatan setiap kendaraan yang melewati jalan raya.

Teknologi *image processing* merupakan salah satu bidang riset yang berkembang pesat beberapa tahun belakangan ini. Kemunculan metode-metode pengenalan pola, bentuk benda, dan pengukuran berbasis pengolahan citra telah dimanfaatkan pada beragam aplikasi kehidupan. Untuk dapat menghitung kecepatan kendaraan yang melewati suatu jalan maka sistem harus mampu melakukan deteksi terhadap kendaraan yang terekam kamera. Berdasarkan latar belakang ini maka dalam penelitian ini akan diimplementasikan pengolahan citra digital pada sistem pendeteksi kecepatan kendaraan.

## 2. METODE PENELITIAN

### 2.1 Rancangan Sistem Secara Keseluruhan

Rancangan sistem secara keseluruhan merupakan bagian dalam rancangan untuk perencanaan tahap-tahap yang diperlukan untuk membuat program pendeteksi kecepatan kendaraan. Program pengukuran kecepatan kendaraan dirancang untuk dapat memproses frame dari suatu file video. Gambar 1 merupakan diagram alir rancangan sistem secara keseluruhan.



Gambar 1 Diagram alir rancangan sistem secara keseluruhan

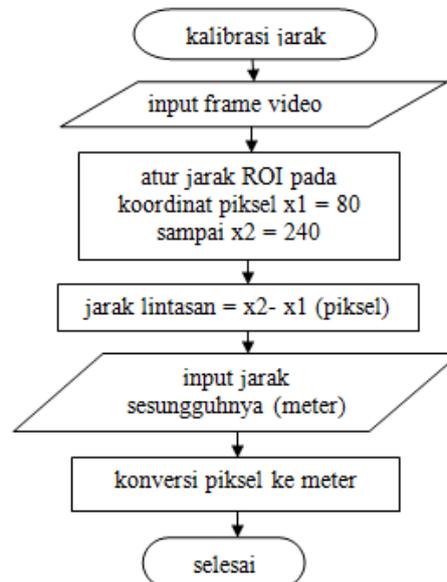
Sistem yang dibangun merupakan perangkat lunak yang mampu mengambil dan memasukkan video serta mendeteksi kendaraan yang melintas di jalan raya untuk kemudian mampu mengukur kecepatan dari kendaraan tersebut berdasarkan perubahan posisi koordinat kendaraan tiap *frame* per satuan waktu. Proses pengambilan dan memasukkan video sebagai citra yang akan diolah ini merupakan *preprocessing* citra. Selanjutnya pada proses deteksi kendaraan menggunakan metode *background segmentation*. Metode ini akan membedakan dan memisahkan antara latar belakang dengan objek yang bergerak [2]. Hasil dari metode ini berupa citra biner yang tertangkap kamera. Objek yang bergerak akan berwarna putih sedangkan latar belakangnya menjadi berwarna hitam. Langkah selanjutnya adalah menentukan nilai koordinat titik tengah dari hasil citra biner ini. Koordinat nilai (x,y) akan diambil sebagai referensi untuk menentukan posisi objek di setiap *frame*-nya. Perpindahan objek tiap *frame* digunakan sebagai acuan jarak dalam satuan piksel. Jarak dalam satuan piksel ini dikonversi ke dalam satuan meter dengan cara melakukan perbandingan jarak antara yang ditangkap oleh kamera dengan jarak sesungguhnya yang diukur secara manual dengan alat ukur. Setiap perpindahan *frame* tersebut dicari waktunya sehingga diperoleh kecepatan dari kendaraan yang melintas di depan kamera. Hasil dari sistem ini kemudian ditampilkan dalam bentuk GUI.

## 2. 2 Perancangan Logika Algoritma

Rancangan perangkat lunak dalam sistem pengukur terdapat dalam proses pengolahan citra. Inputan dari sistem ini merupakan video yang diambil menggunakan kamera yang ditempatkan tegak lurus dengan arah objek yang melewati lintasan jalan. Pada penelitian ini rancangan perangkat lunak dibagi menjadi empat bagian yaitu kalibrasi jarak, preprocessing citra, deteksi kendaraan, dan pengukuran kecepatan.

### 2. 2.1 Kalibrasi Jarak

Tahapan kalibrasi ini diawali dengan mengambil gambar lintasan dengan menggunakan kamera. Jarak lintasan dibatasi dengan *Region of Interest* (ROI) sebagai acuan titik mulai dan titik akhir proses pengukuran jarak. Panjang lintasan yang tertangkap kamera ini akan diukur terlebih dahulu secara manual dalam satuan meter. Setelah itu dilakukan pengukuran panjang piksel pada citra lintasan yang tertangkap kamera. Dari kedua hasil pengukuran ini maka akan dilakukan perhitungan kalibrasi jarak antara satuan piksel menjadi satuan meter. Gambar 2 merupakan diagram alir dari tahapan kalibrasi jarak lintasan.



Gambar 2 Diagram alir kalibrasi jarak

### 2.2.2 Preprocessing Citra

Langkah awal yang dilakukan pertama adalah pengambilan citra menggunakan kamera. Pada proses ini, kendaraan sebagai objek utama ditangkap oleh kamera dalam bentuk sebuah *frame*. Kemudian dilakukan proses *preprocessing* yaitu *frame* ini akan ditangkap secara berulang-ulang sehingga menghasilkan video. Hasil dari video ini diproses untuk dideteksi kendaraan yang tertangkap kamera. Data dari video ini digunakan sebagai inputan. Resolusi video dalam sistem ini ditampilkan dengan ukuran 320 x 240 piksel dengan pertimbangan supaya komputasi yang berjalan dapat bekerja lebih cepat. Pengaturan resolusi ini juga akan mengurangi penggunaan memori yang digunakan. Resolusi 320 x 240 piksel adalah ukuran yang cukup ideal, memang tidak terlalu kecil ataupun besar. Hal ini karena pada resolusi ini pengamatan yang dilakukan sebenarnya sudah cukup jelas.

### 2. 2.3 Deteksi Kendaraan dan Pengukuran Kecepatan

Tahap berikutnya adalah melakukan *filter gaussian* dengan menggunakan *Mixture Of Gaussian* (MOG). Hasil output dari *filter Gaussian* ini adalah citra biner dengan objek yang bergerak merupakan bagian dari *foreground* yang direpresentasikan dengan nilai piksel 1 (warna putih), sedangkan *background* akan direpresentasikan dengan nilai piksel 0 (warna hitam). Hasil yang diperoleh dari *filter gaussian* masih menyisakan beberapa *noise* sehingga perlu dihilangkan dengan proses morfologi citra [3]. Adapun metode morfologi citra yang digunakan adalah erosi dan dilasi. Erosi berfungsi untuk mengurangi *noise-noise* pada piksel bernilai 1 (warna putih) sedangkan dilasi digunakan untuk menambal lubang berwarna hitam di tengah objek yang terdeteksi [4].

Tahap berikutnya adalah dengan memberikan *bounding box* melalui dimensi objek yang diperoleh dari hasil deteksi. Tujuannya adalah untuk mengetahui koordinat piksel dari objek yang terdeteksi pada setiap *frame*. Nilai koordinat ini yang kemudian digunakan untuk menghitung perpindahan jarak antara satu *frame* dengan *frame* lainnya dalam satuan piksel [5]. Dalam melakukan pemberian *bounding box* mula – mula dihitung dahulu jumlah kontur / objek yang berwarna putih. Untuk meyakinkan bahwa kontur yang diseleksi adalah objek yang akan dideteksi serta untuk meminimalisir kemungkinan *noise* terdeteksi sebagai objek, maka diberikan batasan berupa luasan kontur  $> 2000$  piksel<sup>2</sup>. Kendaraan akan terdeteksi ketika telah memasuki batas wilayah ROI yang sudah ditentukan yaitu antara 80 – 240 piksel. Dengan demikian jarak lintasan dalam satuan piksel adalah 160 piksel. *Region of Interest* (ROI) ini digunakan untuk membatasi area frame yang digunakan sebagai tempat deteksi objek. Dalam penelitian ini ROI bisa digunakan sebagai penentu titik awal dan akhir perhitungan kecepatan, yaitu kondisi kendaraan mulai dan selesai dideteksi dan dihitung kecepataannya.

Koordinat titik tengah objek deteksi yang diperoleh pada setiap *frame* digunakan untuk mencari nilai perpindahan jarak antar *frame* dalam satuan piksel. Caranya dengan mencari selisih resultan dari koordinat titik pusat objek yang terdeteksi. Untuk hasil deteksi objek lebih dari satu maka akan dilakukan pengukuran jarak euclidean yaitu pencarian selisih jarak resultan piksel terdekat antara koordinat *frame* sekarang dengan koordinat *frame* sebelumnya [6]. Adapun untuk menghitung jarak euclidean dapat digunakan persamaan (1). Kemudian untuk menghitung kecepatan digunakan persamaan (2) dan (3). Diagram alir proses deteksi kendaraan dan pengukuran kecepatan ditunjukkan pada Gambar 3.

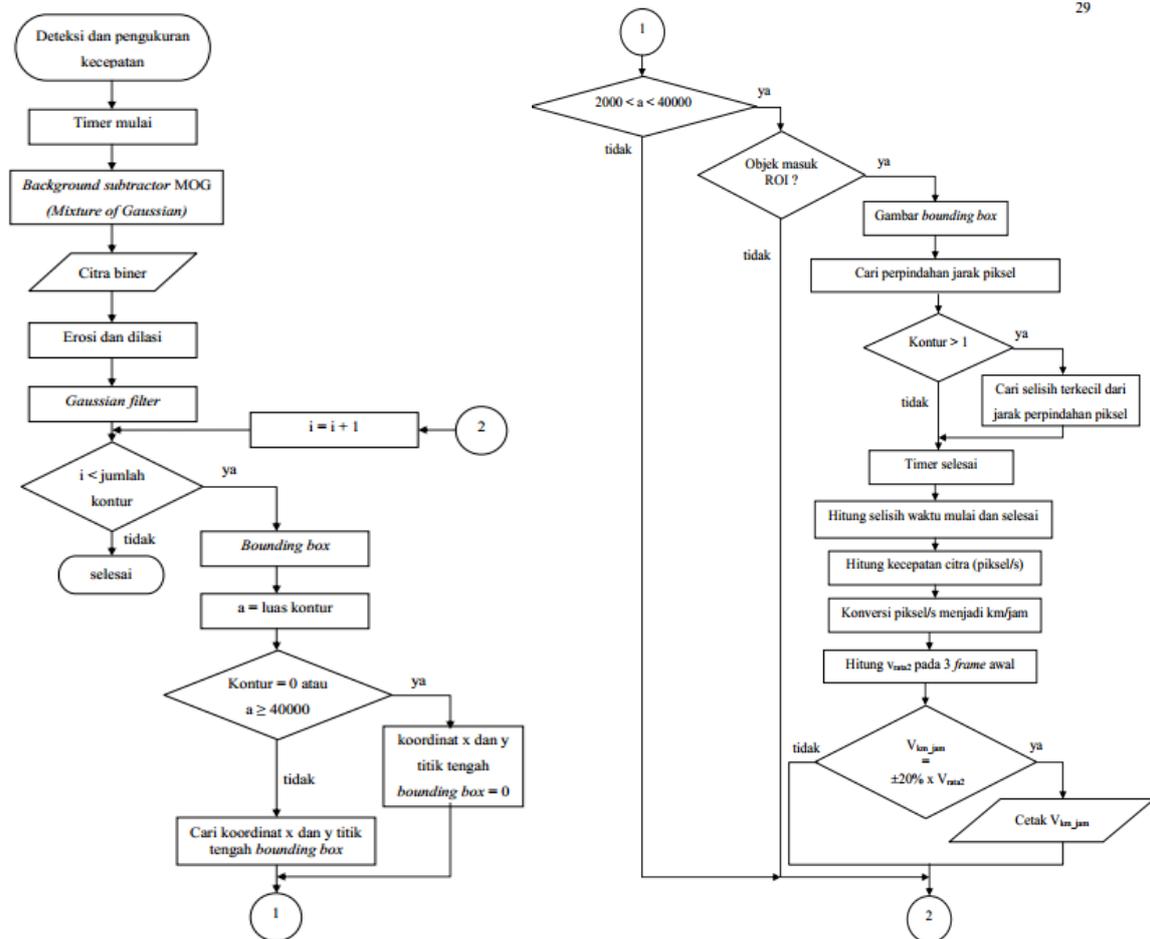
$$\text{jarak pixel} = \sqrt{(x_0 - x_t)^2 + (y_0 - y_t)^2} \quad (1)$$

$$\text{kecepatan citra} = \frac{\text{jarak pixel}}{\text{waktu}} \quad (2)$$

$$\text{kecepatan asli} = \frac{\text{jarak pixel} \times \text{lintasan asli}}{\text{waktu} \times \text{lintasan citra}} \quad (3)$$

dengan:

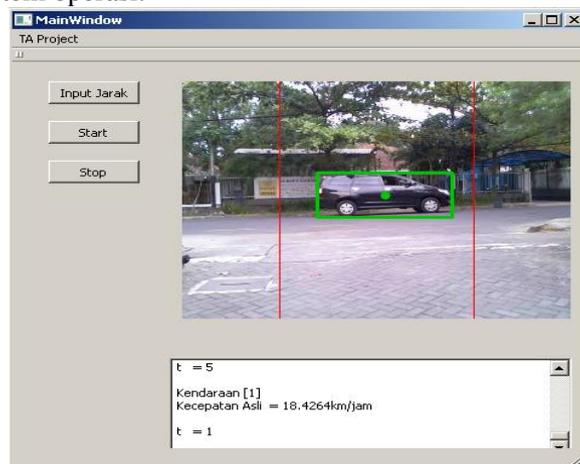
$x_0$	= koordinat x titik tengah pada <i>frame</i> sekarang (piksel)
$x_t$	= koordinat x titik tengah pada <i>frame</i> sebelumnya (piksel)
$y_0$	= koordinat y titik tengah pada <i>frame</i> sekarang (piksel)
$y_t$	= koordinat y titik tengah pada <i>frame</i> sebelumnya (piksel)
jarak piksel	= selisih resultan pada perpindahan <i>frame</i> (piksel)
waktu	= waktu tempuh setiap perpindahan <i>frame</i> (mikro detik)
lintasan asli	= panjang lintasan sesungguhnya (meter)
lintasan citra	= panjang lintasan pada citra (piksel)



Gambar 3 Diagram alir deteksi kendaraan dan pengukuran kecepatan

2. 2.4 Ussee Interface

Perangkat lunak yang digunakan untuk merancang GUI ini adalah Qt Creator versi 5.1.0. *User interface* digunakan sebagai media untuk menghubungkan antara pengguna / *user* berinteraksi dengan sistem operasi.



Gambar 4 Tampilan GUI

Pada GUI terdapat beberapa panel, diantaranya adalah panel tombol yang berisi tombol input jarak, tombol start dan tombol stop. Selain itu terdapat panel penampil video hasil deteksi

kendaraan dan panel penampil nilai hasil pengukuran kecepatan. Tampilan GUI terlihat seperti pada Gambar 4.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Pengujian Validasi Kecepatan

Pengujian ini dilakukan untuk mengetahui hasil pengukuran kecepatan kendaraan menggunakan pengolahan video. Pengujian ini dilakukan di depan gedung PKKH dan di jalan daerah selatan Masjid Kampus Universitas Gadjah Mada. Alat yang digunakan pada pengujian ini antara lain kamera, *tripod*, dan unit pemroses berupa laptop. Kamera diletakkan relatif tegak lurus ( $90^0$ ) terhadap posisi lintasan / jalan raya. Hal ini dilakukan untuk mempermudah proses perhitungan panjang lintasan jika posisi jalan yang tertangkap kamera berada pada posisi horisontal.

Variasi kecepatan yang diberikan adalah antara 10-60 km/jam dengan jangkauan 10 km/jam disetiap variasinya. Jarak lintasan yang terukur sepanjang 6 meter sedangkan jarak yang tertangkap pada citra sepanjang 160 piksel. Berdasarkan data hasil pembacaan kecepatan ternyata masih terdapat beberapa eror/kesalahan dalam pengukurannya. Kesalahan pengukuran diakibatkan oleh beberapa faktor diantaranya adalah hasil deteksi kendaraan yang buruk karena adanya *noise*/gangguan yang berasal dari pergerakan objek lain yang bukan kendaraan. Luas area dari objek yang terdeteksi akan berubah jika terdapat *noise*. Perubahan luas area ini berpengaruh terhadap koordinat titik tengah objek yang dijadikan sebagai acuan untuk menghitung jarak perpindahan objek pada citra. Untuk mengatasi terjadinya eror yang tinggi maka dilakukan pengamatan *trend*/perilaku dari data tersebut. Dari hasil pengamatan perilaku data ternyata didapatkan *trend* yang baik pada saat pengukuran rata-rata dari 3 data pertama. Data pengukuran dapat dilihat pada Tabel 1.

Tabel 1 Perbandingan kecepatan pada speedometer dengan hasil pengolahan video

Kecepatan pada speedometer (km/jam)	Rata-rata kecepatan 3 <i>frame</i> awal (km/jam)	% eror $\left( \frac{ v - \bar{v}_{3 \text{ frame awal}} }{v} \times 100\% \right)$
10	11,57535	15,75 %
20	21,00845	5,04 %
30	28,22017	5,93 %
40	39,23274	1,92 %
50	54,4249	8,85 %
60	57,91904	3,47 %

Dari data Tabel 1 dapat dilihat bahwa akurasi pembacaan rata-rata kecepatan pada 3 *frame* awal adalah dengan eror paling besar 15,75 % sedangkan eror paling kecil sebesar 1,92 %. Data rata-rata ini digunakan sebagai acuan kecepatan kendaraan yang akan ditampilkan pada *frame* berikutnya. Nilai kecepatan mobil yang akan ditampilkan pada GUI nanti adalah kecepatan yang memiliki toleransi rentang eror  $\pm 20\%$  terhadap nilai rata-rata 3 data *frame* pertama. Rentang toleransi tersebut tidak terlalu jauh dengan nilai eror tertinggi dari data Tabel 1 yaitu 15,75 %. Hasil data yang diperoleh dengan pendekatan ini dapat dilihat pada Tabel 2.

Tabel 2 Hasil pengukuran kecepatan rata-rata pada keseluruhan frame berdasarkan acuan kecepatan rata-rata pada 3 *frame* awal

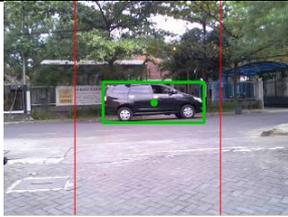
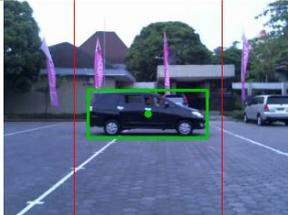
Kecepatan speedometer (km/jam)	Acuan rata-rata kecepatan 3 <i>frame</i> awal (km/jam)	Kecepatan rata-rata seluruh frame (km/jam)	% error
10	11,57535	12,13784	21,37%
20	21,00845	22,7657	13,83%
30	28,22017	31,69077	5,63%
40	39,23274	40,4856	1,21%
50	54,4249	52,764	5,53%

Data pengamatan yang didapatkan dari Tabel 2 menunjukkan keberhasilan sistem dalam mengukur kecepatan memberikan hasil yang cukup baik. Sistem memberikan nilai dengan eror terhadap kecepatan speedometer maksimal 21,37 % sedangkan eror terkecil adalah 1,21 %. Pengukuran pada kecepatan 60 km/jam tidak mendapatkan data pada *frame* berikutnya karena jumlah *frame* yang didapatkan pada saat deteksi kendaraan hanya mampu digunakan untuk menghitung rata-rata kecepatan 3 *frame* awal saja. Hal ini terjadi karena jarak lintasan sebenarnya yang tertangkap kamera kurang panjang untuk dapat mendeteksi kecepatan 60 km/jam.

### 3.2 Pengujian Pengaruh Intensitas Cahaya

Pengujian dilakukan dengan melakukan variasi waktu pengujian yang memiliki kondisi intensitas cahaya yang berbeda-beda. Lokasi pengujian semua dilakukan di luar ruangan. Waktu yang digunakan untuk pengujian ini adalah pada saat pagi, siang, sore, dan malam hari. Alat ukur yang digunakan untuk mengukur intensitas cahaya ini adalah LIGHT METER. Hasil pengujian pengaruh intensitas cahaya terhadap hasil deteksi kendaraan dan pengukuran kecepatan dapat dilihat pada Tabel 3.

Tabel 3 Hasil pengujian pengaruh intensitas cahaya

Waktu pengujian	Intensitas cahaya (lux)	Hasil deteksi	Parameter pengukuran kecepatan
Pagi (07.00 – 08.00 WIB)	1000 - 1040	 Terdeteksi dengan baik	Kecepatan terukur dengan baik ( $v_{rata2} = 31,69077$ km/jam pada sampel 30 km/jam)
Siang (11.00 – 12.00)	1800 - 1900	 Terdeteksi dengan baik	Kecepatan terukur dengan baik ( $v_{rata2} = 40,4856$ km/jam pada sampel 40 km/jam)
Sore (16.00 – 17.00 WIB)	600 - 800	 Terdeteksi dengan baik	Kecepatan terukur dengan baik ( $v_{rata2} = 22,7657$ km/jam pada sampel 20 km/jam)
Malam (18.30 – 19.00 WIB)	0 – 5	 Tidak bisa dideteksi	Kecepatan tidak dapat diukur

Pada kondisi pagi, siang, dan sore hari dengan rentang intensitas cahaya 600-1900 lux, sistem mampu melakukan deteksi objek dengan baik sekaligus melakukan pengukuran

kecepatan dengan cukup baik. Akan tetapi pada malam hari dengan rentang intensitas cahaya antara 0-5 lux, sistem tidak dapat melakukan deteksi dengan baik sehingga tidak dapat dilakukan pengukuran kecepatan. Selain karena kondisi cukup gelap, hal ini juga diakibatkan oleh kemampuan kamera pada malam hari yang tidak dapat menangkap citra dengan baik.

### 3.3 Pengujian Pengukuran Kecepatan Setelah Divalidasi

Pengujian ini dilakukan untuk mengetahui apakah hasil dari validasi tersebut benar atau tidak. Pada tahap pengujian ini menggunakan 2 buah sampel video dengan masing-masing 2 variasi kecepatan mobil. Jarak lintasan yang terukur sejauh 6,5 m. Hasil pengukuran oleh pengolahan citra dibandingkan dengan kecepatan speedometer mobil. Hasil dari pengujian ini dapat dilihat pada Tabel 4 dan 5.

Tabel 4 Hasil pengukuran kecepatan mobil pada sampel video 1

Kecepatan asli (km/jam)		Kecepatan deteksi citra (km/jam)	
Mobil 1	Mobil 2	Mobil 1	Mobil 2
50	30	47,18851	24,22732
		45,38866	29,0265
		46,5705	23,68268
		49,27372	24,08517
		48,68346	25,17558
		46,85201	24,3743
		38,17167	-
		43,36813	-
		43,23138	-
Kecepatan rata-rata		45,41422667	25,09525833
%eror		9,17 %	16,35 %

Tabel 5 Hasil pengukuran kecepatan mobil pada sampel video 2

Kecepatan asli (km/jam)		Kecepatan deteksi citra (km/jam)	
Mobil 1	Mobil 2	Mobil 1	Mobil 2
50	40	47,80651	39,2382
		47,73172	31,4471
		42,4667	43,39997
		52,8655	36,53443
		51,5964	41,1349
		46,34618	40,81785
		56,20496	31,63074
		53,46176	39,37803
		44,73389	42,14251
		-	40,14795
		-	41,29898
		-	42,03492
		-	42,23005
-	42,52226		
-	40,37267		
Kecepatan rata-rata		49,24595778	39,62203733
%eror		1,51 %	0,94 %

Data pengukuran yang diperoleh dari pengujian sampel video 1 menghasilkan persentase eror sebesar 9,17 % pada pengukuran kecepatan 50 km/jam dan 16,35 % pada pengukuran kecepatan 30 km/jam sedangkan pada sampel video 2 menghasilkan persentase eror sebesar 1,51 % pada pengukuran kecepatan 50 km/jam dan 0,94 % pada pengukuran kecepatan 40 km/jam. Perolehan data tersebut menunjukkan bahwa sistem dapat memberikan hasil pengukuran kecepatan dengan cukup baik. Faktor terbesar yang mengakibatkan eror adalah adanya *noise* pada *background* yaitu berupa gerakan suatu objek yang merubah luasan area dari kendaraan yang akan dideteksi sehingga mempengaruhi hasil pengukuran jarak perpindahan pada citra.

#### 4. KESIMPULAN

Dari hasil pengamatan, pengujian, dan analisis pada hasil yang diperoleh, kesimpulan yang dapat diambil adalah sebagai berikut.

1. Telah berhasil dibuat sistem pengukur kecepatan kendaraan berbasis pengolahan video.
2. Hasil validasi kecepatan diambil dari rata-rata kecepatan 3 *frame* awal dengan memberikan rentang eror 1,92 % - 15,75 % dengan masing-masing hasil pengukuran pada setiap kecepatan adalah sebagai berikut. Pada kecepatan 10 km/jam terukur 11,57 km/jam; pada kecepatan 20 km/jam terukur 21 km/jam; pada kecepatan 30 km/jam terukur 28,22 km/jam; pada kecepatan 40 km/jam terukur 39,23 km/jam; pada kecepatan 50 km/jam terukur 54,42 km/jam; dan pada kecepatan 60 km/jam terukur 57,92 km/jam.
3. Ketika validasi tersebut dilakukan pada pembacaan keseluruhan *frame* video menghasilkan rentang eror kecepatan rata-rata sebesar 1,21 % - 21,37 % dengan masing-masing eror pada setiap kecepatan adalah sebagai berikut. Pada kecepatan 10 km/jam terukur rata-rata 12,14 km/jam; pada kecepatan 20 km/jam terukur rata-rata 22,76 km/jam; pada kecepatan 30 km/jam terukur 31,7 km/jam; pada kecepatan 40 km/jam terukur 40,48 km/jam; dan pada kecepatan 50 km/jam terukur rata-rata 54,42 km/jam.
4. Sistem akan mendeteksi kendaraan dan mengukur kecepatan dengan baik pada saat pagi, siang, dan sore hari dengan jangkauan intensitas cahaya antara 600-1900 lux, sedangkan pada malam hari dengan jangkauan intensitas cahaya 0-5 lux objek tidak dapat terdeteksi dengan baik.
5. Akurasi keberhasilan pengukuran juga dipengaruhi oleh tingkat gangguan objek lain pada *background*.
6. Validasi sistem dapat diujikan pada lintasan dengan hasil pengujian pada sampel video 1 memberikan hasil eror 9,17 % dan 16,35 %. Hasil pengukuran pada kendaraan 1 dengan kecepatan 50 km/jam terukur 45,41 km/jam dan pada pengukuran kendaraan 2 dengan kecepatan 30 km/jam terukur 25,1 km/jam. Pada sampel video 2 memberikan hasil eror 1,51 % dan 0,94 %. Hasil pengukuran kendaraan 1 dengan kecepatan 50 km/jam terukur 49,25 km/jam dan pada pengukuran kendaraan 2 dengan kecepatan 40 km/jam terukur 39,62 km/jam

#### 5. SARAN

1. Sistem pengukuran kecepatan kendaraan berbasis pengolahan video ini dapat dikembangkan pada sistem lalu-lintas 2 arah.
2. Hendaknya bisa dikembangkan dengan metode lain sehingga hasil pengukuran kecepatan kendaraan menjadi lebih akurat dengan eror yang kecil.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberi dukungan baik secara moril maupun materiil terhadap penelitian ini.

## DAFTAR PUSTAKA

- [1]Raharjo, J., Susatio, E., dan Tirtoasmono, I. I., 2012, Perancangan dan Prototyping Sistem Pemantau Lalu Lintas Berbasis Video Processing dalam Mendukung Intelligent Transportation System, *Prosiding InSINas*, hal. 20-24, Bandung
- [2]Solichin, A., dan Harjoko, A., 2013, Metode *Background Subtraction* untuk Deteksi Objek Pejalan Kaki pada Lingkungan Statis, *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, hal. B-1 – B-6, Yogyakarta.
- [3] Zheng, Z., Wang, X., Zhou, G., dan Jiang, L., 2012, Vehicle Detection Based On Morphology From Highway Aerial Images, *IEEE International Conference on Signal and Image Processing Application pp. 5997-6000*, Sichuan, China.
- [4] Braedski, G. dan Kaehler, A., 2008, *Learning OpenCV*, Gravenstein Highway North : O'Reilly Media, Inc.
- [5] Awaludin, L., 2012, Pemrosesan Video Pendeteksi Kecepatan dan Ketinggian Aliran Lahar Dingin Pendukung Sistem Peringatan Dini, *Skripsi*, Jurusan Ilmu Komputer dan Elektronika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Gadjah Mada, Yogyakarta.
- [6] Rohman, M., 2011, Analisa Gerakan Tangan Manusia pada Video Digital, *Skripsi*, Jurusan Teknik Elektro, Fakultas teknik Industri, Institut Teknologi Sepuluh November, Surabaya.