

PENGUKURAN KECEPATAN KENDARAAN SECARA REALTIME BERBASIS ANDROID

Rendi Gustina¹, Widyadi Setiawan², Ngurah Indra ER³

Abstract— Intelligent transport systems in the android operating system is able to bring about change is very important in human life. Objectives and benefits in this thesis is to obtain an application program measuring the speed of vehicles in real time based on Android using the Frame Difference and Bounding Box wearing reference distance, the vehicle speed of 10 km / h, 20 km / h and 30 km / h and distance 6 meters. After getting the average speed of vehicles, will be calculated deviation measurements of vehicle speed and the actual results compared to the average. For the measurement of the speed of vehicles put on the vehicle speed measurement applications in realtime based on android gained an average of 20.572 km / h with a maximum deviation 0.572 km / h and an average of 29.688 km / h with minimal deviation 0.312 km / h. For the measurement of the speed of vehicles put on the analysis of vehicle speed measurement in digital video recording obtained by an average of 31.454 km / h with a maximum deviation 1,454 km / h and an average of 10.112 km / h with minimal deviation 0.112 km / h.

Intisari — Sistem transportasi cerdas dalam sistem operasi android mampu membawa perubahan yang sangat penting dalam kehidupan manusia. Tujuan dan manfaat dalam pengukuran kecepatan kendaraan ini adalah untuk mendapatkan program aplikasi pengukuran kecepatan kendaraan secara realtime berbasis android menggunakan metode Frame Difference dan Bounding Box memakai jarak referensi, dengan kecepatan kendaraan 10 km/jam, 20 km/jam, dan 30 km/jam serta jarak 6 meter. Setelah mendapatkan hasil rata-rata kecepatan kendaraan, akan dihitung penyimpangan pengukuran kecepatan kendaraan dari hasil sesungguhnya dan dibandingkan rata-ratanya. Untuk hasil pengukuran kecepatan kendaraan memakai aplikasi pengukuran kecepatan kendaraan secara realtime berbasis android diperoleh rata-rata 20,572 km/jam dengan penyimpangan maksimal 0,572 km/jam dan rata-rata 29,688 km/jam dengan penyimpangan minimal 0,312 km/jam. Untuk hasil pengukuran kecepatan kendaraan memakai analisis pengukuran laju kendaraan pada rekaman video digital diperoleh rata-rata 31,454 km/jam dengan penyimpangan maksimal 1,454 km/jam dan rata-rata 10,112 km/jam dengan penyimpangan minimal 0,112 km/jam.

Kata Kunci — Android, kecepatan kendaraan, Frame Difference, Bounding Box

I. PENDAHULUAN

Pesatnya perkembangan teknologi saat ini menjadi umpan bagi para ahli untuk mencetuskan terobosan-terobosan baru

¹Mahasiwa, Jurusan Teknik Elektro dan Komputer Fakultas Teknik Universitas Udayana, (e-mail: rendirustandi17@gmail.com)

²Dosen, Jurusan Teknik Elektro dan Komputer Fakultas Teknik Universitas Udayana (e-mail: widyadi@ unud.ac.id)

³Dosen, Jurusan Teknik Elektro dan Komputer Fakultas Teknik Universitas Udayana (e-mail: indra@ unud.ac.id)

Rendi Gustina: Pengukuran Kecepatan Kendaraan Secara ...

berbasis teknologi canggih. Terobosan ini diciptakan guna mempercepat dan mempermudah kinerja manusia dalam melakukan sesuatu dalam berbagai bidang. Salah satunya dalam bidang sistem transportasi cerdas (Intelligent Transport Systems).

Saat ini laju kendaraan di kota-kota besar seperti di Bali khususnya di kota Denpasar mulai terasa kurang terkontrol, ini disebabkan oleh ruas jalan yang tersedia tidak sebanding dengan peningkatan jumlah transportasi saat ini.

Pada penelitian ini menggunakan aplikasi berbasis android untuk mengukur kecepatan kendaraan sehingga mendapatkan hasil visualisasi kecepatan lalu lintas. Dengan diterapkannya ITS di Indonesia maka setiap pengguna jalan akan dapat mengaksesnya secara real time dan mudah. Hanya dengan menggunakan handphone maka setiap orang bisa mengetahui kondisi jalan apakah sedang mengalami macet atau lancar.

Dalam penelitian ini penulis menggunakan citra grayscale untuk menyederhanakan warna citra selanjutnya dilakukan proses segmentasi menggunakan frame difference. Selanjutnya proses feature extraction untuk menentukan titik koordinat dari objek yang akan deteksi agar proses bounding box dapat dilakukan. Pada penelitian ini akan dibandingkan hasil kecepatan kendaraan sesungguhnya dengan hasil pengukuran kecepatan kendaraan memakai jarak referensi.

II. FRAME DIFFERENCE

Proses yang digunakan untuk memperlihatkan objek yang bergerak adalah dengan menggunakan frame difference. Tetapi frame yang akan dibandingkan tidak menggunakan frame yang berurutan. Hal ini karena penggunaan frame yang berurutan memiliki kekurangan, yaitu hasil frame difference nya berupa objek berlubang dan hanya tepian dari obyek yang terlihat. Selain itu jika suatu obyek yang bergerak kemudian berhenti maka akan dianggap sebagai background. Proses frame difference pada tiap-tiap frame video dengan background dilakukan dengan format Hue Saturation Value (HSV). Penggunaan saturation dan value memiliki kelebihan pada deteksi perubahan intensitas cahaya dan tidak banyak bergantung pada perubahan warna.

Dengan demikian, dalam metode frame difference, objek bergerak yang diambil sesuai dengan perbedaan antara dua atau tiga frame terus menerus. Metode ini merupakan metode yang paling sederhana dan langsung berhubungan dengan perberubah dalam objek bergerak dalam video dapat cepat terdeteksi, dengan algoritmanya sebagai berikut:

1. Baca video input.
2. Baca seluruh frame dari video input.
3. For (i = (frame awal +1) sampai dengan frame akhir)
 - a) Hitung selisih nilai pixel antar 2 yang frame berdekatan (frame (i)-frame (i-1)).
 - b) Hitung nilai absolutnya.

p-ISSN:1693 – 2951; e-ISSN: 2503-2372

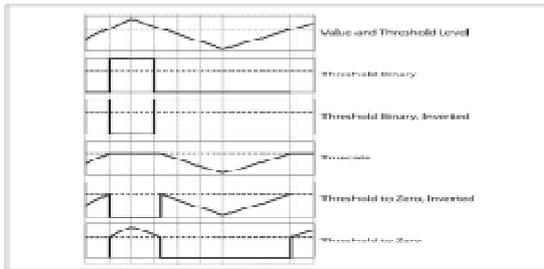


c) Simpan hasil pada frame baru.

Metode *frame difference* lebih memberikan atau menyesuaikan *edge* atau tepi di setiap pixel yang mengalami perbedaan selisih pixelnya, sehingga setiap nilai pixel antara 2 frame yang berdekatan akan dimunculkan pada frame yang baru, dimana nilai *zero* adalah nilai hitam dan *non zero* adalah nilai putih. Nilai *non zero* inilah yang merupakan *edge* dari frame yang baru dari hasil *frame difference*.

III. THRESHOLD

Threshold merupakan teknik yang sederhana dan efektif untuk segmentasi citra dengan melakukan pengolahan piksel pada suatu citra atau menghilangkan beberapa piksel dan juga mempertahankan beberapa piksel sehingga menghasilkan suatu citra baru hasil sortir piksel yang telah dilakukan. Dengan dilakukannya komputasi *threshold* maka dapat dengan mudah mendapatkan *edge* atau tepi dari suatu citra. *Threshold* dilakukan agar mempermudah dalam proses identifikasi ataupun perbandingan dari dua atau lebih citra. Tipe-tipe *threshold* dapat dilihat pada gambar 1 Terdapat 5 tipe *threshold*, yaitu *binary threshold*, *binary inverted*, *truncate*, *threshold to zero* dan *threshold to zero inverted*.



Gambar 1: Tipe-tipe *Threshold*

Binary threshold merupakan citra yang telah melalui proses pemisahan piksel-piksel berdasarkan derajat keabuan yang dimiliki. Pembentukan citra biner memerlukan nilai batas keabuan yang akan digunakan sebagai nilai patokan. Piksel dengan derajat keabuan lebih besar dari nilai batas akan diberi nilai 1 sebagai warna putih dan sebaliknya piksel dengan derajat keabuan lebih kecil dari nilai batas akan diberi nilai 0 sebagai warna hitam. Warna putih biasanya digunakan untuk warna *foreground* sedangkan warna hitam adalah warna *background*. Konversi *grayscale* ke biner dapat dilihat pada gambar 2. Setiap piksel mempunyai nilai warna dengan nilai 0 (hitam) dan 1 (putih).



Gambar 2 : Konversi *Grayscale* ke Biner

IV. BOUNDING BOX

Dalam metode *bounding box* gambar divisualisasikan melalui dimensi objek yang dihasilkan dari gambar sebelumnya yaitu dengan menggunakan hasil *output frame difference*. Namun, terlebih dahulu melakukan proses pengambilan ekstraksi ciri benda yang akan diolah untuk menentukan setiap koordinat dari suatu objek menggunakan *feature extraction* agar proses *bounding box* dapat dilakukan. *Scanning* dilakukan untuk mencari batas-batas nilai *edge* berupa nilai *non zero* disetiap *frame* yang dimulai dari baris atas sampai baris bawah setiap *frame*, kolom kiri sampai dengan kolom kanan setiap *frame*.

Dalam hal ini manfaat *bounding* digunakan untuk menetapkan objek *box* yang diidentifikasi agar lebih jelas, dimana *feature extraction* menggambarkan nilai intensitas warna piksel tersebut. Nilai-nilai intensitas terdiri dari nilai atas, bawah, kiri, dan kanan disetiap *frame* menggunakan dimensi nilai kotak pembatas persegi panjang yang diplot dalam batas-batas nilai yang dihasilkan. Kekurangan dari penggunaan *bounding box*, yaitu penggunaan memori yang tidak sedikit.

V. STUDI KASUS

A. Tempat dan Waktu Penelitian

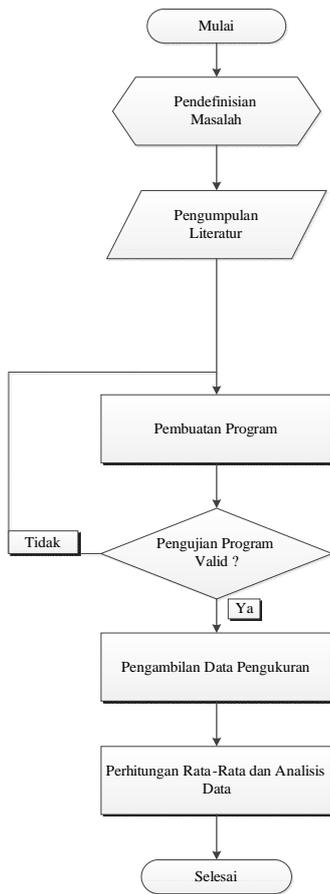
Penelitian dilakukan di Jalan Raya Tantular Barat Renon Denpasar Bali. Waktu pelaksanaan penelitian selama 1 bulan. Penelitian dimulai dari tanggal 01 Desember 2015 sampai tanggal 30 Desember 2015.

B. Data

Jenis data yang digunakan dalam penelitian ini adalah data kuantitatif. Data kuantitatif merupakan data yang berupa angka-angka yang diperoleh dari data pengukuran laju kendaraan 10 km/jam, 20 km/jam, dan 30 km/jam memakai jarak 6 meter Sehingga data tersebut dapat digunakan untuk pengukuran kecepatan kendaraan secara *real time* pada program ini.

C. Alur Analisis

Berikut merupakan flowchart alur analisis pada program dapat dilihat pada gambar 3



Gambar 3 : Flowchart Alur Analisis

VI. HASIL DAN PEMBAHASAN

Pada penelitian ini dibahas mengenai pengukuran kecepatan kendaraan pada aplikasi Pengukuran Kecepatan Kendaraan Secara Realtime Berbasis Android dengan kecepatan kendaraan yang diujikan 10 km/jam, 20 km/jam, dan 30 km/jam memakai jarak referensi 6 meter. Sebelum melakukan pengujian yang perlu dipersiapkan untuk pembuatan aplikasi Pengukuran Kecepatan Kendaraan Secara Realtime Berbasis Android adalah perangkat keras dan perangkat lunak untuk perangkat keras Handphone Samsung Galaxy S4 dan Notebook Asus A46CB dan untuk perangkat lunak Eclipse Integrated Development Environment versi Luna, Java Development Kit 8, Android Software Development Kit 5.0 atau API level 21, Sistem Operasi Windows 7 Ultimate Service Pack I dan Android Development Tools Plug-in. Saat melakukan pengambilan data menggunakan kamera Handphone, background atau latar belakang yang digunakan hanya menggunakan latar belakang tanpa ada benda yang bergerak kecuali satu objek yang bergerak. Permasalahan yang dibahas dimulai dari hasil pengukuran kecepatan kendaraan dibandingkan dengan kecepatan kendaraan sesungguhnya. Setelah mendapatkan hasil pengukuran kecepatan kendaraan akan dibandingkan dengan hasil penyimpangan pengukuran laju kendaraan dari

Rendi Gustina: Pengukuran Kecepatan Kendaraan Secara ...

hasil sesungguhnya dengan hasil simulasi laju kendaraan (Adi Setiawan 2009). Untuk hasil pengukuran kecepatan kendaraan beserta simpangan dan analisis rata-rata pengukuran kecepatan dapat dilihat pada tabel I, tabel II, tabel III, tabel IV, tabel V, tabel VI dan gambar 4.

TABEL I
HASIL PENGUKURAN KECEPATAN KENDARAAN PADA APLIKASI PENGUKURAN KECEPATAN KENDARAAN SECARA REALTIME BERBASIS ANDROID DARI SAMPEL PERTAMA SAMPAI KELIMA

Sampel	Jarak (m)	Hasil Pengukuran Kecepatan Kendaraan		
		10 km/jam	20 km/jam	30 km/jam
1.	6	10,64	18,76	28,28
2.	6	9,95	22,25	30,67
3.	6	10,31	20,29	29,78
4.	6	11,45	22,36	30,69
5.	6	9,84	19,20	29,02
Rata-rata		10,438	20,572	29,688

TABEL II
SIMPANGAN PENGUKURAN KECEPATAN KENDARAAN APLIKASI PENGUKURAN KECEPATAN KENDARAAN SECARA REALTIME BERBASIS ANDROID DARI SAMPEL PERTAMA SAMPAI KELIMA

Sampel	Jarak (m)	Penyimpangan Pengukuran kecepatan Kendaraan		
		10 km/jam	20 km/jam	30 km/jam
1.	6	0,64	1,24	1,72
2.	6	0,05	2,25	0,67
3.	6	0,31	0,29	0,22
4.	6	1,45	2,36	0,69
5.	6	0,16	0,80	0,98
Rata-rata		0,438	0,572	0,312

TABEL III
HASIL PENGUKURAN KECEPATAN KENDARAAN PADA ANALISIS PENGUKURAN LAJU KENDARAAN PADA REKAMAN VIDEO DIGITAL DARI SAMPEL PERTAMA SAMPAI KELIMA

Sampel	Jarak (m)	Hasil Pengukuran Kecepatan Kendaraan		
		10 km/jam	20 km/jam	30 km/jam
1.	6	10	21,6	31,76
2.	6	10	20	30
3.	6	10	22,5	31,76
4.	6	10,18	20,76	33,75
5.	6	10,38	20	30
Rata-rata		10,112	20,972	31,454

TABEL IV
SIMPANGAN PENGUKURAN KECEPATAN KENDARAAN PADA ANALISIS PENGUKURAN LAJU KENDARAAN PADA REKAMAN VIDEO DIGITAL DARI SAMPEL PERTAMA SAMPAI KELIMA

Sampel	Jarak (m)	Penyimpangan Pengukuran kecepatan Kendaraan		
		10 km/jam	20 km/jam	30 km/jam
1.	6	0	1,6	1,76
2.	6	0	0	0
3.	6	0	2,5	1,76
4.	6	0,18	0,76	3,75
5.	6	0,38	0	0
Rata-rata		0,112	0,972	1,454

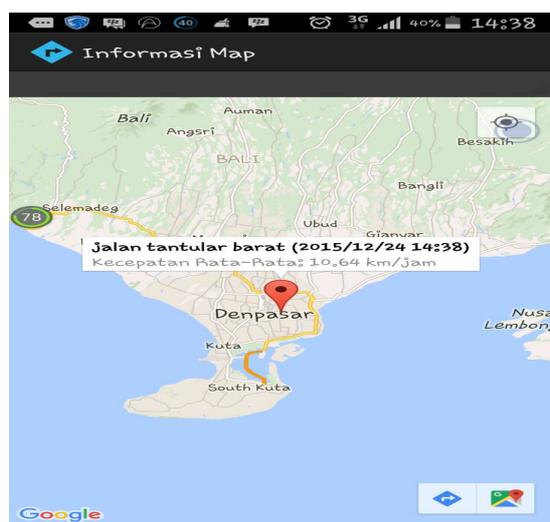


TABEL V
PERBANDINGAN HASIL RATA-RATA PENGUKURAN KECEPATAN KENDARAAN

Perbandingan Hasil Rata-Rata Pengukuran Kecepatan Kendaraan		
Kecepatan	Pengukuran Kecepatan Kendaraan Secara <i>Realtime</i> Berbasis <i>Android</i>	Analisis Pengukuran Laju Kendaraan Pada Rekaman Video Digital
10 km/jam	10,438	10,112
20 km/jam	20,572	20,972
30 km/jam	29,688	31,454

TABEL VI
SIMPANGAN HASIL RATA-RATA SIMPANGAN PENGUKURAN KECEPATAN KENDARAAN

Perbandingan Hasil Rata-Rata Simpangan Pengukuran Kecepatan Kendaraan		
Kecepatan	Pengukuran Kecepatan Kendaraan Secara <i>Realtime</i> Berbasis <i>Android</i>	Analisis Pengukuran Laju Kendaraan Pada Rekaman Video Digital
10 km/jam	0,438	0,112
20 km/jam	0,572	0,972
30 km/jam	0,312	1,454



Gambar 4: Tampilan Hasil Pengukuran kecepatan Kendaraan Secara *Realtime* Berbasis *Android*

VII. KESIMPULAN

- Pada penelitian ini telah berhasil dirancang aplikasi pengukuran kecepatan kendaraan secara realtime berbasis android dengan menggunakan metode *frame difference* dan metode *bounding box* pada program aplikasi berbasis android.
- Hasil pengukuran kecepatan kendaraan berdasarkan jarak referensi didapatkan perbandingan hasil rata-rata kecepatan kendaraan dari pengukuran kecepatan kendaraan yang bervariasi, namun cukup mewakili estimasi kecepatan kendaraan sesungguhnya. Untuk hasil pengukuran kecepatan kendaraan memakai aplikasi pengukuran kecepatan kendaraan secara realtime berbasis android diperoleh rata-rata 20,572 km/jam dengan penyimpangan maksimal 0,572 km/jam dan rata-rata 29,688 km/jam dengan penyimpangan minimal 0,312 km/jam. Untuk hasil pengukuran kecepatan kendaraan memakai analisis pengukuran laju kendaraan pada rekaman video digital diperoleh rata-rata 31,454 km/jam dengan penyimpangan maksimal 1,454 km/jam dan rata-rata 10,112 km/jam dengan penyimpangan minimal 0,112 km/jam.

REFERENSI

- [1] Peddireddi, L. 2008. Object Tracking and Velocity Determination using TMS320C6416T DSK. Alpen Adria: University Klagenfurt Faculty fur Technische Wissenschaften.
- [2] Setiawan, A. (2009). Analisis Pengukuran Laju Kendaraan Pada Rekaman Video Digital. Teknik Elektro Universitas Udayana, Denpasar.
- [3] Marques, O. 2011. Practical Image And Video Processing Using Matlab IEEE PRESS. Florida: A John Wiley & Sons, Inc, Publication.
- [4] Karisma, H. 2011. Implementasi Handtracking Pada Kontrol Mouse Pointer Menggunakan Algoritma Pyramidal Lucas-Kanade. Bandung: Universitas Komputer Indonesia.