

**PERANCANGAN DAN IMPLEMENTASI SISTEM ANTI TABRAKAN: SISTEM
PENDETEKSI MOBIL DAN PERGERAKAN MOBIL PADA SIANG HARI
UNIVERSITAS TELKOM**

*DESIGN AND IMPLEMENTATION OF ANTI CRASH SYSTEM: CAR
DETECTION AND CAR MOVEMENT SYSTEM AT NOON
TELKOM UNIVERSITY*

Aditya Reza Pahlawan M.¹, Agus Virgono, Ir., M.T.², Andrew Brian Osmond, S.T., M.T.³

¹²³Prodi S1 Sistem Komputer, Fakultas Teknik Elektro, Universitas Telkom University
Jalan Telekomunikasi 1, Dayeuhkolot, Bandung 40257, Indonesia

¹adityarezapahlawan@gmail.com ²avirgono@telkomuniversity.ac.id

³abosmond@telkomuniversity.ac.id

Abstrak

Berdasarkan data dari Badan Pusat Statistik (BPS) di Indonesia pada tahun 2012, jumlah kecelakaan kendaraan bermotor di Indonesia adalah 117.949 atau meningkat sebanyak 9.253 dari tahun 2011. Maka dari itu dibutuhkan sebuah sistem untuk meminimalisasi tingkat kecelakaan di jalan raya. Sistem anti tabrakan dibuat untuk meminimalisasi kecelakaan dengan cara memberikan peringatan dini kepada pengemudi sesaat sebelum tabrakan terjadi. Pengemudi diharapkan mempunyai respon terhadap sistem sehingga kecelakaan dapat dihindari. Sistem anti tabrakan terdiri dari berbagai macam jenis seperti sistem pendeteksi mobil dan pergerakan mobil, sistem penghitung jarak antar mobil, sistem pendeteksi rambu lalu lintas, dan sebagainya. Sistem yang dihasilkan merupakan sebuah sistem pendeteksi mobil dan pergerakan mobil dimana sistem akan bekerja pada siang hari. Sistem pendeteksi mobil dan pergerakan mobil dibuat dengan menggunakan bahasa C# dan *library* Emgu CV. Sistem pendeteksi mobil dan pergerakan mobil menghasilkan *output* berupa informasi jumlah mobil yang berada di area tidak aman beserta pergerakan mobil dengan tingkat akurasi sebesar 83,90%. Sistem ini menggunakan metode deteksi tepi Canny dan *Haar Cascade Classifier* dalam menentukan jarak aman beserta mobil dan gerakan mobil, dengan tingkat akurasi keberhasilan sistem membentuk area aman sebesar 75,47%.

Kata Kunci: Sistem anti tabrakan, Sistem pendeteksi mobil dan pergerakan mobil, *Haar Cascade Classifier*, dan deteksi tepi Canny

Abstract

Based on data from Agency of Statistics Central Indonesia in 2012, number of motor vehicle accidents in Indonesia is 117.949 or increasing as many as 9.253 from 2011. Therefore we need a system to decrease rate of road accidents. Anti crash system is made to minimize accidents by giving early warning to the driver shortly before the collision occurs. The driver is expected to have a response to the system so that accidents can be avoided. Anti crash system is composed of various types such as car detection system and its movement, the counter distance system between car to car, detection system of traffic sign and etc. The resulting system is a car detection system and its movement that can work at noon. This System is built using the C # language and libraries Emgu CV. A car detection system and car movement system will produce the output that gives information about the numbers of cars where are in unsafe area and cars movement depends on the information with accuracy rate is 83.90%. This system uses Canny edge detection method and Haar Cascade Classifier to determine a safe distance between car to car and its motion, with the accuracy of success system create safe area is 75.47%.

Keywords: anti crash system, car detector of system, Haar Cascade Classifier and Canny edge detection

1. Pendahuluan

Sistem anti tabrakan dibutuhkan untuk meminimalisasi terjadinya kecelakaan. Sistem anti tabrakan akan memberikan *output* kepada pengemudi mengenai keadaan yang sedang berlangsung di jalan raya. Sistem anti tabrakan berfungsi untuk memperingatkan pengemudi sesaat sebelum kecelakaan terjadi dan diharapkan dapat

menghindarkan pengemudi dari kecelakaan. Salah satu penerapan dari sistem anti tabrakan adalah sistem pendeteksi mobil dan pergerakan mobil.

Pada tugas akhir ini akan dilakukan penerapan teknologi analisis video yang berisi sistem informasi untuk memberikan suatu informasi berupa deteksi mobil dan pergerakan mobil. Sistem informasi pada tugas akhir ini akan berisi tentang pergerakan kendaraan yang ada di depan mobil pengemudi dan berguna efektif pada siang hari dengan kondisi cuaca cerah. Sistem informasi pendeteksi mobil dan pergerakan mobil diharapkan dapat mengurangi tingkat kecelakaan mobil.

2. Dasar Teori

2.1. Citra dan Citra Digital^{[4][7]}

Secara harfiah, citra adalah gambar pada bidang dua dimensi. Sedangkan jika ditinjau dari sudut pandang matematis, citra merupakan fungsi menerus (*continue*) dari intensitas cahaya pada bidang dua dimensi. Citra sebagai keluaran dari suatu sistem perekaman data dapat bersifat optik, analog, dan digital.

Citra digital merupakan kumpulan dari beberapa *pixel* atau piksel (*picture element*), setiap piksel memiliki nilai (*value*) yang menunjukkan intensitas keabuan pada piksel tersebut. Resolusi merupakan ukuran dari sebuah citra yang dinyatakan dalam piksel x piksel. Semakin tinggi resolusi dan kedalaman bit suatu citra, maka semakin baik kualitas suatu citra. Kedalaman bit menentukan jumlah bit yang digunakan untuk merepresentasikan tiap piksel pada sebuah *frame*, dan dinyatakan dengan bit/piksel.

2.2. Computer Vision^[3]

Computer vision dapat didefinisikan setara dengan pengertian pengolahan citra yang dikaitkan dengan akuisisi citra, pemrosesan, klasifikasi, pengakuan, dan pencakupan keseluruhan, pengambilan keputusan yang diikuti dengan pengidentifikasian citra. Ballard dan Brown (1982) mendefinisikan *Computervision* sebagai sebagai suatu kegiatan awal pengotomatisan dan pengintegrasian suatu pemrosesan dan representasi sebagai suatu persepsi visual dengan tahap-tahap tertentu.^[8]

2.3. Algoritma Viola-Jones^[8]

Metode Viola-Jones merupakan salah satu metode yang paling banyak digunakan untuk mendeteksi objek. Metode Viola-Jones merupakan algoritma pendeteksi objek yang terdapat dalam *library* Emgu CV yang efisien, sehingga tidak memerlukan waktu lama dalam melakukan proses pendeteksian objek. Proses pendeteksian objek dilakukan dengan mengklasifikasikan sebuah *image* setelah sebelumnya sebuah *classifier* dibentuk dari data training.^[7]

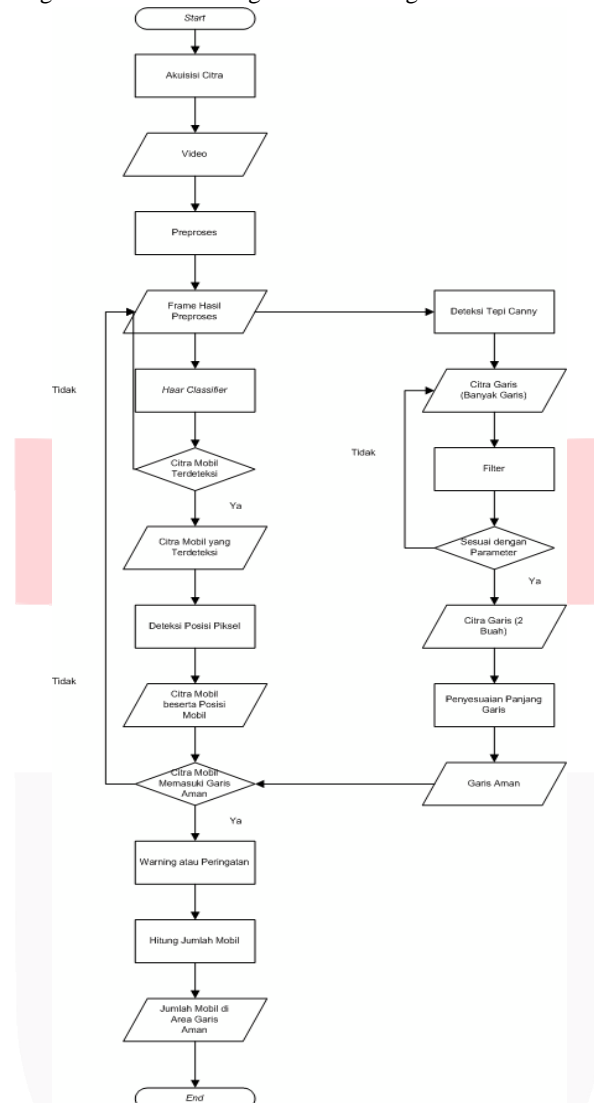
2.4. Deteksi Tepi Canny^[2]

Metode *Canny edge detection* atau deteksi tepi Canny merupakan sebuah metode yang akan mendeteksi semua garis-garis atau *edge* yang membentuk objek gambar dan akan memperjelas kembali pada bagian-bagian tersebut. Tujuan dari deteksi tepi Canny adalah mengenali objek di dalam gambar dan kemudian disederhanakan bentuknya dari bentuk sebelumnya. Kelebihan dari metode deteksi tepi Canny adalah kemampuan untuk mengurangi *noise* sebelum melakukan perhitungan deteksi tepi sehingga tepi-tepi yang dihasilkan lebih banyak.^[1]

3. Perancangan

Sistem pendeteksi mobil dan pergerakannya merupakan bagian dari sebuah sistem besar, yaitu sistem anti tabrakan. Sistem ini mendapatkan input dari *webcam* dan merekam pergerakan mobil, dari video yang telah diambil kemudian video tersebut dianalisis. Dari analisis video, kemudian dapat ditentukan objek mobil dan kendaraan beroda empat atau lebih beserta dengan pergerakannya untuk kemudian akan memberikan *output* pada laptop.

Secara umum rancangan sistem dalam Tugas Akhir ini digambarkan dalam diagram alir berikut:



Gambar 3.1 Diagram alir Aplikasi

3.1. Inisialisasi Kamera

Inisialisasi kamera merupakan proses awal dalam segala hal yang diperlukan untuk menjalankan proses selanjutnya. Pada tahap inisialisasi kamera, untuk memudahkan *user*, GUI telah diinisialisasikan terlebih dahulu dengan berbagai tampilan-tampilan *interface*.

3.2. Akuisisi Citra

Akuisisi citra bertujuan untuk menentukan data yang akan diolah dan memilih metode yang diperlukan. Dalam tugas akhir ini, digunakan *webcam* untuk proses pengambilan data dengan resolusi 640x480 piksel. *Webcam* yang digunakan merupakan *webcam* Logitech HD C525 dengan resolusi 720p. Pada saat proses pengambilan data, *webcam* dipasang diatas *dashoard* mobil, sehingga bagian depan mobil pengemudi tidak terekam oleh kamera.

3.3. Preproses

Pada tahap preproses terdiri dilakukan proses *grayscale* dan *cutting/cropping*.

3.4. Grayscale^[8]

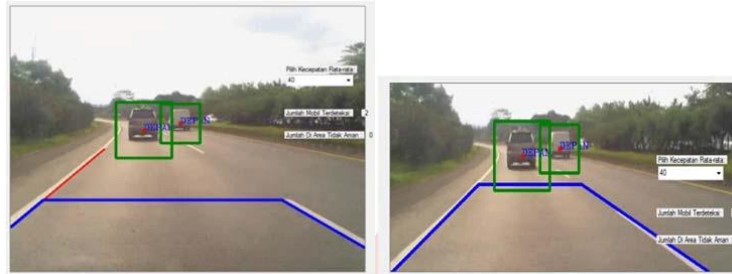
Citra masukan yang telah ditangkap oleh *webcam* kemudian diubah mejadi citra dua warna dengan proses *grayscale*. Proses *grayscale* bertujuan untuk mempermudah komputasi citra sehingga tahap berikutnya

dalam pengolahan citra dapat dilanjutkan. Proses *grayscale* dalam mengkonversi citra RGB menjadi citra dua warna dapat dihitung dengan rumus seperti di bawah ini.^[8]

$$M = (0,2989 * R) + (0,5870 * G) + (0,1140 * B)$$

3.5. Cutting/Cropping

Tujuan dari proses *cutting/cropping* adalah untuk mendapatkan area deteksi yang sesuai dengan kebutuhan proses pengolahan citra. Sehingga jika terdapat area yang tidak memungkinkan untuk terdeteksinya sebuah objek, maka area tersebut dapat dihilangkan.



Gambar 3.2 Citra sebelum proses *cropping/cutting* (kiri) dan setelah proses *cropping/cutting* (kanan)

3.6. Detektor DetectHaarCascade^[5]

Terdapat lima parameter dalam menggunakan *classifier Haar Cascade*, yaitu sebagai berikut^[5].

- a. Haar Classifier
- b. Scale Increase Rate
- c. Minimum Neighbors Threshold
- d. Canny Prunning Flag
- e. Minimum Detection Scale

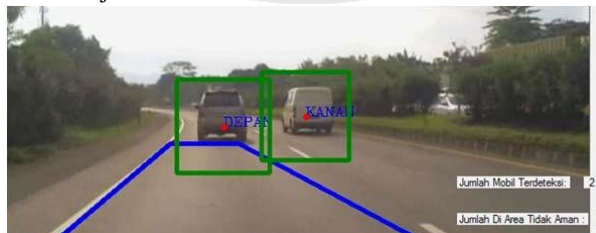
3.7. Haar Classifier

Citra yang telah melewati tahap preproses selanjutnya akan diklasifikasi oleh *cars classifier*. *Classifier* merupakan salah satu parameter (*haar*) dalam *class HaarCascade* yang berfungsi untuk mendeteksi mobil dalam *library EmguCV*. *Classifier* berisi data gambar positif (mobil) dan data gambar negatif (bukan mobil) yang kemudian dijalankan tahap *grayscale* dan juga *scaling*. Data citra kemudian *di training* dan ditentukan nilai ambang atau *threshold* untuk dapat ditentukan apakah citra yang di deteksi termasuk mobil atau bukan mobil.

Dalam Tugas Akhir ini digunakan *cars classifier Haar Cascade* yang telah dibuat oleh Brad Philip dan Paul Updike pada tahun 2001, sehingga tidak pembuatan *classifier* baru tidak dibutuhkan^[6].

3.8. Counting

Setelah sebuah gambar atau *frame* dideteksi oleh *Haar classifier*, langkah selanjutnya adalah menghitung jumlah objek (mobil) yang terdeteksi. Sebelum dilakukan penghitungan, terlebih dahulu dilakukan sebuah proses penanda atau (*marking*) terhadap objek yang telah terdeteksi dengan membuat sebuah kotak yang mengelilingi objek terdeteksi tersebut. Membuat kotak pada deteksi mobil bukanlah tugas dari *classifier/detector*, maka dari itu, harus dilakukan pengkotakan objek secara manual.



Gambar 3.3 Kotak penanda objek yang terdeteksi dan jumlah objek yang terdeteksi

3.9. Deteksi Gerakan

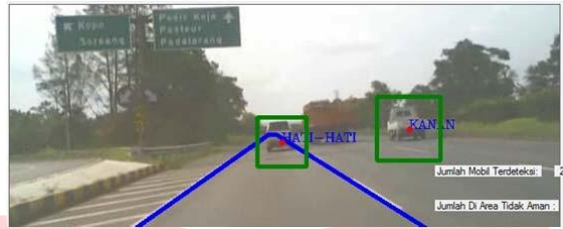
Pada Tugas Akhir ini, untuk mendeteksi gerakan, langkah pertama adalah menentukan titik tengah dari objek yang telah terdeteksi. Untuk mencari titik tengah objek dapat digunakan rumus sebagai berikut.

$C = (\text{panjang kotak} / 2, \text{ lebar kotak} / 2)$

$C =$ titik tengah dari kotak penanda objek terdeteksi

Setelah ditentukan titik tengah dari objek yang terdeteksi maka akan ditemukan posisi piksel dari titik tengah tersebut, misalkan (120, 60). Posisi piksel tersebut nanti akan menjadi acuan untuk mendeteksi gerak dari mobil.

Untuk menentukan pergerakan dari mobil (dalam hal ini gerak ke kiri dan ke kanan), terlebih dahulu di tentukan pembagian area deteksi. Pembeagian area deteksi dibagi menjadi empat area, yaitu depan, kanan, kiri, dan bagian awas atau mobil berada di area aman mobil.

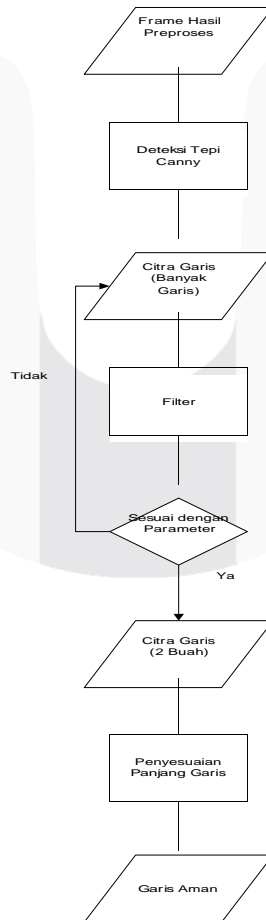


Gambar 3.4 Deteksi gerakan mobil berdasarkan perpindahan piksel

3.10. Menentukan Jarak Aman

Dalam sistem anti tabrakan salah satu aspek yang paling penting adalah menentukan jarak aman kendaraan. Jarak aman berfungsi sebagai penanda atau batas seberapa dekat kendaraan yang kita kemudikan berada di belakang mobil lain.

Berikut merupakan skema atau diagram alir dari proses pembentukan garis area aman menggunakan algoritma deteksi tepi Canny.



Gambar 3.5 Diagram alir pembentukan garis area aman menggunakan deteksi tepi Canny

4. Pengujian

Pengujian aplikasi Sistem Anti Tabrakan dilakukan dengan cara pengujian *non real time* yaitu dengan mengolah video yang sebelumnya sudah di rekam terlebih dahulu. Proses perekaman video sendiri dilakukan di Jalan Tol Buah Batu, dengan rute Buah Batu – Pasir Koja – Buah Batu.

Pengujian pada Tugas Akhir dibagi menjadi dua bagian, yaitu: pengujian terhadap jumlah kendaraan di area aman, dan pengujian terhadap keakuratan garis area aman pada jalan yang cenderung lurus.

Pengujian sistem ini menggunakan tiga buah video yang telah direkam sebelumnya di Jalan Tol Buah Batu, yaitu Video1.wmv, Video2.wmv, dan Video3.wmv. Dengan spesifikasi sebagai berikut.

Tabel 4.1 Spesifikasi video pengujian

Keterangan	Video1.wmv	Video2.wmv	Video3.wmv
Kecepatan mobil pengemudi	40 – 50 KM	50 KM	50 KM
Kecepatan mobil rekan	40 – 50 KM	50 KM	50, 40, 50 KM
Jarak Aman	150 piksel	150 piksel	150 piksel
Durasi Video Pengujian	00:13 – 02:34 (detik ke 13 sampai 2 menit 34 detik)	00:21 – 02:00 (detik ke 21 sampai 2 menit 0 detik)	00:13 – 02:32 (detik ke 13 sampai 2 menit 32 detik)

Keterangan:

- Kecepatan mobil rekan merupakan kecepatan mobil di depan mobil pengemudi yang bekerjasama dalam pengujian.
- Pengujian dilakukan dalam rentang waktu 2 detik dan kemudian diselaraskan antara penglihatan mata dengan penglihatan sistem.

4.1. Pengujian terhadap Jumlah Kendaraan di Area Aman Kendaraan

Pengujian terhadap jumlah kendaraan di seluruh area pandang kamera dimaksudkan untuk menguji keakuratan *classifier* dalam mendeteksi jumlah mobil dan juga untuk menguji keakuratan pergerakan mobil dalam area-area yang telah ditentukan sebelumnya.

Tabel 4.2 Pengujian jumlah kendaraan pada area tidak aman

No.	Nama	Pengujian 1			Pengujian 2			Pengujian 3		
		Total	Kesalahan	Akurasi	Total	Kesalahan	Akurasi	Total	Kesalahan	Akurasi
1.	Video1.wmv	71	10	85,92%	71	14	80,28%	71	15	78,87%
2.	Video2.wmv	50	19	62%	50	7	86%	50	9	82%
3.	Video3.wmv	70	3	95,71%	70	5	86,38%	70	6	91,43%
	Rata-rata Akurasi	81,21%			86,38%			84,1%		

Tabel 4.3 Analisis kesalahan terhadap pengujian jumlah kendaraan pada area tidak aman

No.	Pengujian	Jumlah Kesalahan	Penyebab Kesalahan		
			Faktor Garis Aman	Kesalahan deteksi oleh <i>Classifier</i>	Mobil tidak terdeteksi
1.	Pengujian 1	32 kali	21 kali (65,625%)	6 kali (18,75%)	5 kali (15,625%)
2.	Pengujian 2	26 kali	11 kali (42,3%)	2 kali (7,7%)	13 kali (50%)
3.	Pengujian 3	30 kali	10 kali (33,33%)	3 kali (10%)	17 kali (56,67%)

4.2. Pengujian terhadap Keakuratan Garis Area Aman

Pengujian terhadap keakuratan garis area aman dimaksudkan untuk mendeteksi tingkat akurasi dari sistem untuk mendeteksi garis batas bahu jalan pada jalan yang cenderung lurus.

Tabel 4.4 Pengujian akurasi area aman

No.	Nama	Pengujian 1			Pengujian 2			Pengujian 3		
		Total	Kesalahan	Akurasi	Total	Kesalahan	Akurasi	Total	Kesalahan	Akurasi
1.	Video1.wmv	71	15	78,87%	71	17	76,06%	71	15	78,87%
2.	Video2.wmv	50	17	66%	50	23	54%	50	23	54%
3.	Video3.wmv	70	8	88,57%	70	7	90%	70	5	92,86%
	Rata-rata Akurasi	77,81%			73,35%			75,24%		

Tabel 4.5 Analisis kesalahan terhadap pengujian akurasi area aman

No.	Pengujian	Jumlah Kesalahan	Penyebab Kesalahan			
			Kesalahan pada garis	Kamera tidak stabil	Titik potong	Jalan berbelok
1.	Pengujian 1	40 kali	14 kali (35%)	16 kali (40%)	5 kali (12,5%)	5 kali (12,5%)
2.	Pengujian 2	47 kali	15 kali (31,91%)	16 kali (34,04%)	10 kali (21,28%)	6 kali (12,77%)
3.	Pengujian 3	43 kali	7 kali (16,28%)	19 kali (44,19%)	9 kali (20,93%)	8 kali (18,60%)

4.3. Analisis

- a. $\frac{81,21\% + 86,38\% + 84,1\%}{3} = 83,90\%$
- b. $\frac{77,81\% + 73,35\% + 75,24\%}{3} = 75,47\%$
- c. Analisis penyebab kegagalan pendeteksian kendaraan pada area tidak aman
 - Faktor garis aman (total 42 kali)
 - Kesalahan deteksi oleh *classifier* (total 11 kali)
 - Mobil tidak terdeteksi (total 35 kali)
- d. Analisis penyebab kegagalan pembentukan garis area aman
 - Kesalahan pendeteksian garis (total 36 kali)
 - Kamera (total 51 kali)
 - Titik potong (total 24 kali)
 - Jalan berkelok (total 19 kali)

5. Kesimpulan

Dari hasil pengujian dan analisa yang telah dilakukan pada sistem, dapat diambil beberapa kesimpulan sebagai berikut :

1. Sistem yang dirancang sudah dapat mengenali objek (dalam hal ini mobil) dan mendeteksi pergerakan mobil pada siang hari, tetapi masih memiliki kelemahan dalam membedakan mobil dengan objek lain (misal papan penunjuk arah).

2. Sistem yang dirancang sudah dapat menghitung jumlah mobil yang berada di area aman. Sistem juga telah berhasil membuat garis area aman melalui bantuan garis batas bahu jalan dengan menggunakan deteksi tepi Canny.
3. Tingkat akurasi dari sistem pendeteksi mobil pada area aman kendaraan adalah 81,21%, sedangkan tingkat akurasi keberhasilan sistem dalam membentuk garis area aman adalah 77,81%.

Daftar Pustaka

- [1] Bradski, Gary., Adrian Kaehler. 2008. *Learning OpenCV Computer Vision with the OpenCV Library*. California: O'Reilly Media, Inc.
- [2] Canny, John. 1986. *A Computational Approach to Edge Detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. PAMI-8, No. 6, pp. 679-698.
- [3] Fadlisyah.2007. *Computer Vision dan Pengolahan Citra*. Yogyakarta: Andi.
- [4] Munir, Rinaldi. 2004. *Pengolahan Citra Digital*. Bandung: InformatikaBandung.
- [5] Nasir, Mahvish. 2011. *Emgu CV with C#* [Online]. Tersedia : <http://fewtutorials.bravesites.com/tutorials> [30 Desember 2014].
- [6] Philip, Brad., Paul Updike. 2001. *Vehicle Detection with Haar Cascades* [Online]. Tersedia : <https://www.behance.net/gallery/Vehicle-Detection-Tracking-and-Counting/4057777> [7 Januari 2015]
- [7] Putra, Darma. 2010. *Pengolahan Citra Digital*. Yogyakarta: Andi.
- [8] Santoso, Hadi., Agus Haryoko, *Haar Cascade Classifier dan Algoritma Adaboost untuk Deteksi Banyak Wajah dalam Ruang Kelas*. Pangkalpinang: Universitas Gadjah Mada.
- [9] Seputra, Yulius Eka Agung. 2014. *Buku Pintar Pemrograman C#*. Yogyakarta: MediaKom
- [10] Shi, Shin, *Emgu CV Essentials*. Birmingham: Packt Publishing, 2013.
- [11] Widyawan., Noor Ahmad Setiawan., Teguh Bharata Adji. 2013. *Teori Aplikasi Pengolahan Citra*. Yogyakarta: Andi