

Intelligence Vehicle Counting Menggunakan Metode Combination Value Saturation Pada Video Lalu Lintas

Guruh Adi Purnomo¹, Imam Cholissodin², Fitri Utamingrum³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya

Email: ¹guruh.adi.purnomo@gmail.com, ²imamcs@ub.ac.id, ³f3_ningrum@ub.ac.id

ABSTRAK

Kebutuhan transportasi sudah menjadi hampir kebutuhan setiap kegiatan yang dilakukan oleh manusia, dan hal tersebut sangat mempengaruhi jumlah pertumbuhan kendaraan di Indonesia. Menurut data Korps Lalu Lintas Kepolisian Negara Republik Indonesia mencatat, jumlah kendaraan yang beroperasi meningkat setiap tahunnya, sehingga menimbulkan kemacetan dan perlu adanya solusi untuk mengatasi hal tersebut. Salah satu solusi untuk mengatasi kemacetan dengan cara mengalihkan arus kendaraan ke jalur lainnya, dan untuk mengatasi hal tersebut diperlukan perhitungan kendaraan agar tidak terjadi kemacetan kembali. Dikarenakan pada saat ini perhitungan mobil masih dilakukan manual, maka diperlukan sebuah sistem yang dapat menghitung kendaraan secara otomatis seperti "intelligence vehicle counting menggunakan metode combination value saturation pada video lalu lintas". Berdasarkan pengujian, sistem ini memiliki rata-rata akurasi penghitungan kendaraan sebesar 65.38%.

Kata Kunci : *Combination Value Saturation, Intelligence Vehicle Counting, Pengolahan Citra Digital, Penghitungan Kendaraan*

ABSTRACT

Transportation needs have become almost the needs of every activity undertaken by humans, and it greatly affects the number of vehicle growth in Indonesia according to data Traffic Corps of the State Police of the Republic of Indonesia noted, the number of vehicles that operate increases every year, causing congestion and the need for a solution to overcome it. One solution to overcome the congestion by diverting the flow of vehicles to other lanes, and to overcome this is required to calculate the vehicle so that no congestion occurs again. Because at this time the calculation of the car is still done manually, then required a system that can calculate the vehicle automatically as "intelligence vehicle counting menggunakan combination value method saturation on video traffic". Based on the test, this system has an average vehicle accuracy of 65.38%.

Keywords: *Combination Value Saturation, Intelligence Vehicle Counting, Digital Image Processing, Vehicle Counting*

1. PENDAHULUAN

Dewasa ini, transportasi bukanlah hal yang baru bagi kita, seperti fenomena yang ada, bahwa hampir setiap intensitas kegiatan manusia memerlukan transportasi. Hal ini dapat disimpulkan bahwa, semakin banyak pertumbuhan manusia, semakin banyak pula pertumbuhan alat transportasi yang ada (Adhitama, 2011).

Menurut data Korps Lalu Lintas Kepolisian Negara Republik Indonesia

mencatat, jumlah kendaraan yang masih beroperasi di seluruh Indonesia pada tahun 2013 mencapai 104,211 juta unit, naik 11 % dari tahun sebelumnya (2012) yang cuma 94,299 juta unit. Rata-rata penjualan mobil di Indonesia sekitar 1 juta unit per tahun, Estimasinya, pertumbuhan terjadi sekitar 10% setiap tahun nya (Gatra, 2014).

Salah satu upaya untuk mengurangi kemacetan adalah dengan cara pelaksanaan manajemen lalu lintas yang efektif, yaitu proses pengaturan dan penggunaan sistem jalan raya yang sudah ada tanpa menambah infrastruktur

yang baru (Ekawati, 2014). Hal tersebut biasanya dilakukan dengan cara pengalihan arus kendaraan yang padat ke jalan alternatif lainnya. Dalam upaya pengalihan lalu lintas tersebut perlu diketahui jumlah kendaraan yang melewati ruas-ruas jalan yang menuju titik-titik jalan yang padat dan berpotensi mengakibatkan kemacetan, sehingga muncullah sebuah mekanisme perhitungan manual “counter” sebagai penghitung jumlah kendaraan. Namun, cara seperti ini memiliki kelemahan yaitu perhitungan kendaraan dilakukan oleh manusia (petugas dinas perhubungan) yang pada dasarnya harus memiliki tingkat konsentrasi yang tinggi, selain itu juga akan menguras stamina para petugas yang mengakibatkan ketidakakuratan ketika melakukan proses perhitungan (Hariyanto, 2012).

Dalam upaya meminimalisir pengurasan stamina para petugas dan meningkatkan keakuratan dalam proses perhitungan tersebut dapat memanfaatkan kecanggihan teknologi dan kemajuan ilmu pengetahuan untuk menghitung kendaraan itu dengan memanfaatkan cabang ilmu kecerdasan buatan, pengolahan citra digital dan *computer vision* untuk mengenali, mengobservasi objek, dan menginterpretasikan, serta mendeskripsikannya, sehingga menghasilkan informasi yang berguna seperti mengetahui jumlah kendaraan yang melintas hanya melalui masukan video yang dihasilkan dari rekaman gambar oleh kamera (T.Sutojo, et al., 2011). Sehingga, dalam pengaplikasian sebuah sistem penghitung kendaraan tidak semata hanya menggunakan kamera saja, namun perlu adanya akuisisi citra, pemrosesan, hingga pengambilan keputusan yang diikuti oleh pengidentifikasian citra (Fadlisyah, 2007) dan menerapkan metode-metode untuk pendeteksian objek oleh kamera.

Pada penelitian kali ini menggunakan metode *Combination Value Saturation* (CVS) untuk melakukan pendeteksian objek. Pada dasarnya, metode ini mirip dengan *background subtraction* perbedaannya terletak pada saat

frame extraction metode ini menjadikan gambar pada *background* dan *current frame* menjadi citra *saturation* dan *value* dan nantinya akan digabungkan kembali untuk deteksi objeknya, dan dilakukan beberapa proses *noise and dilation removal* dan *erotion* (Rad, et al., 2010). Setelah dilakukan pendeteksian objek maka dapat dilakukan proses *counting vehicle*.

2. DASAR TEORI

2.1 Kajian Pustaka

Traffic Surveillance by Counting and Classification of Vehicles from video using Image Processing yang Penelitian ini membahas tentang algoritma pada pengamatan video lalu lintas dengan menghitung dan mengklasifikasikan kendaraan yang diperoleh dari inputan video menggunakan *image processing*. Setelah dilakukan pengujian dengan metode tersebut dengan menginputkan video berdurasi 5 detik diperoleh beberapa *error*, pada detik pertama diperoleh 0% *error* dengan 7 *actual count* dan 7 *testing count*, kemudian pada detik kedua diperoleh *error* 20% yang disebabkan pada *actual count* atau perhitungan asli diperoleh 5 kendaraan yang melintas, namun sistem mendeteksi 4 kendaraan saja, kemudian pada detik ke 3 diperoleh *error* 16% yang dikarenakan pada *actual count* diperoleh 6 kendaraan yang melintas dan pada sistem hanya mendeteksi 5 kendaraan saja, sedangkan pada detik ke 4 didapatkan *error* 0% dengan 3 *actual count* dan 3 *testing count* (Susmita A & A.V.Malviya, 2013).

Penelitian selanjutnya adalah *Real Time Vehicle Detection and Counting Method for Unsupervised Traffic Video on Highways*. Pada penelitian kali ini diusulkan dengan menggunakan *background subtraction* dan *image segmentation* yang didasarkan pada *morphological transformation* untuk *tracking* dan *counting vehicle* pada jalan raya. Pengimplementasian sistem ini dilakukan dengan software MATLAB dengan hasil akurasi yang di coba pada 3 video dengan format *grayscale*, pada video 1 terdapat 30 *frame* dan memiliki 6 *actual count* dan 6

detected cars dengan akurasi 100%, kemudian pada video 2 terdapat 120 *frame* dan memiliki 25 *actual count* dan 26 *detected cars* dengan akurasi 96% *error* yang terjadi disini karena terdapat sedikit kemacetan pada video inputan tersebut, sehingga menyebabkan sistem mendeteksi posisi yang ambigu pada saat mendeteksi mobil, dan pada video ke 3 memiliki 30 *frame*, 5 *actual count*, dan 5 *detected cars* (M.Daigavanee & Bajaj, 2010).

Berikutnya *Vehicle Counting Without Background Modelling*. Penelitian ini menyelesaikan beberapa permasalahan yang didapati pada penelitian *vehicle counting* lainnya yang biasanya sering mengalami masalah tentang bayangan, pencahayaan dan goyangan pohon yang dapat mengganggu perhitungan. Penelitian ini diuji pada 3 model video clip, pada video pertama terdapat 85 *actual count* dan 90 *Detected cars*, kemudian pada video ke dua terdapat 88 *actual count* dan 94 *detected cars*, kemudian pada video ke tiga terdapat 76 *actual count* dan 85 *Detected cars* dengan ini menjelaskan bahwa sistem ini dapat berjalan dan menghitung dengan baik pada *rate* dibawah 20fps dan rata rata penghitungan kendaraan mencapai 86 % (Lien, et al., 2013).

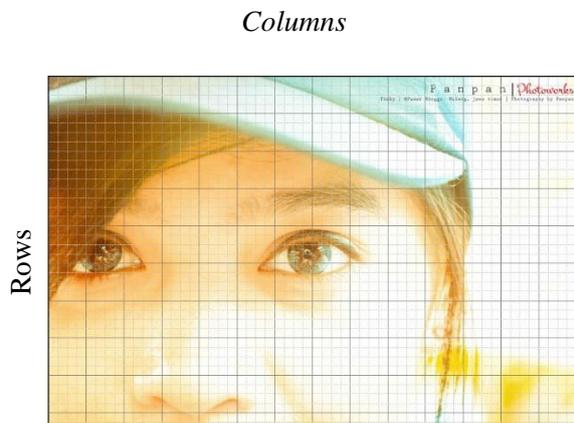
Berikutnya adalah segmentasi kendaraan menggunakan *improve blob analysis* pada video lalu lintas. Pada penelitian tersebut menggabungkan beberapa metode *frame differencing* dan *running average*. penelitian ini menghasilkan akurasi 90% untuk pengujian *threshold* dan 100% untuk ukuran *Structuring element*, sehingga sistem yang dibentuk pada penelitian ini dapat melakukan segmentasi kendaraan dengan baik (Sutrisno, et al., 2015).

Kemudian, penelitian yang terakhir adalah *An Automated Vehicle Counting System Based on Blob Analysis for Traffic Surveillance* Penelitian berikutnya masih sama tentang *vehicle counting* untuk membantu mengatir dan mengontrol ketika terdapat sebuah kemacetan, pada penelitian ini memanfaatkan ROI (*Region of Interest*) yang diterapkan secara virtual pada jalur jalur kendaraan, dalam pengujia nya sistem ini menggunakan Bahasa pemrograman C++ dan melakukan proses pada video 320 x 240 pixel dengan 25 fps pada

computer dual core 2,4 Ghz yang menghasilkan rata rata keakuratan *precision* 98,9%, *Recall* 98,26%, dan 98,57% *Measure* (G.Salvi, n.d.).

2.2 Citra Digital

Kali ini akan di bahas secara mendalam mengenai citra digital, citra digital disini adalah sebuah representasi citra yang memiliki fungsi dua dimensi $f(x,y)$ dimana nilai x dan y disini merupakan koordinat dari citra tersebut, sedangkan nilai f di setiap pasangan koordinat (x,y) adalah nilai dari derajat keabuan (gray level) dari sebuah citra pada titik tersebut (Hermawati, 2013).



Gambar 1 Contoh citra digital

Pada gambar 2.1 diatas merepresentasikan sebuah citra digital yang memiliki ukuran M baris dan N kolom yang dapat di representasikan sebagai matriks berukuran $N \times M$ yang diilustrasikan oleh persamaan (2-1) dibawah ini, dimana N adalah jumlah baris $0 = y = N-1$, dan M adalah jumlah kolom $0 = x = M-1$

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,M-1) \\ f(1,0) & f(1,1) & f(1,M-1) \\ \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & f(N-1,M-1) \end{bmatrix}$$

2.3 Model Warna HSV

Model warna HSV (*Hue, Saturation, Value*) menunjukkan ruang warna dalam komponen dasar pembentuknya yaitu *Hue* dimana *Hue* disini adalah sudut 0 hingga 360 derajat dan biasanya 0 derajat didominasi warna merah,60

derajat didominasi warna kuning, 120 derajat didominasi warna hijau, 180 derajat didominasi warna cyan, 240 derajat didominasi warna biru, dan 300 derajat didominasi oleh warna magenta. Dimana *hue* sendiri menunjukkan jenis warna dimana tempat warna tersebut ditemukan dalam *spectrum* warna, *saturation* menunjukkan seberapa besar dari warna itu sendiri, sebagai contohnya adalah warna kuning tanpa adanya warna putih adalah saturasi penuh, *saturation* sendiri biasanya bernilai 0 sampai 1 dan nilai 0 menunjukkan keabuabuan dan nilai 1 sendiri menunjukkan nilai primer atau murni. Kemudian nilai *Value* menunjukkan kecerahan dalam warna itu sendiri atau seberapa besar cahaya yang ada pada warna tersebut, nilai dari *value* sendiri dapat bernilai 0 hingga 1, untuk mengkonversi warna dari RGB ke HSV dapat dilakukan oleh persamaan hingga persamaan sebagai berikut (Putra, 2010).

$$H = \tan \left[\frac{3(G - B)}{(R - G) + (R - B)} \right]$$

$$S = 1 - \frac{\min(R, G, B)}{V}$$

$$V = \frac{R + G + B}{3}$$

dimana : $H = Hue$ $R = Red$
 $S = Saturation$ $G = Green$
 $V = Value$ $B = Blue$

Namun jika dari rumus diatas nilai S bernilai 0 maka nilai Hue tidak dapat ditentukan maka perlu dilakukan normalisasi seperti pada persamaan hingga persamaan dibawah ini.

$$r = \frac{R}{R+G+B}$$

$$g = \frac{R}{R+G+B}$$

$$b = \frac{R}{R+G+B}$$

dimana : $R = Red$ $r = normalisasi red$
 $G = Green$ $g = normalisasi green$
 $B = Blue$ $b = normalisasi blue$

Dengan menormalisasi nilai $R, G,$ dan B maka persamaan konversi RGB ke HSV dapat dilakukan dengan persamaan hingga persamaan berikut.

$$V = \max(r, g, b)$$

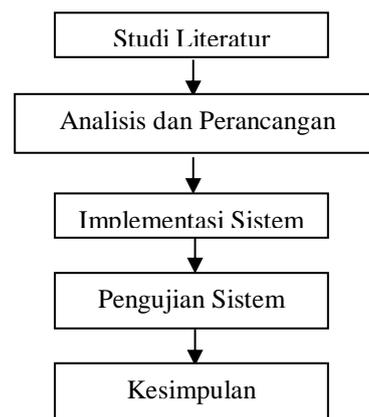
$$S = \begin{cases} 0 & \text{Jika } V = 0 \\ V - \frac{\min(r, g, b)}{V} & \text{Jika } V > 0 \end{cases}$$

$$H = \begin{cases} 0 & \text{Jika } S = 0 \\ \frac{60 \times g - b}{S \times V} & \text{Jika } V = r \\ 60 \times \left[2 + \frac{b - r}{S \times V} \right] & \text{Jika } V = g \\ 60 \times \left[4 + \frac{r - g}{S \times V} \right] & \text{Jika } V = b \end{cases}$$

dimana : $R = Red$ $r = normalisasi red$
 $G = Green$ $g = normalisasi green$
 $B = Blue$ $b = normalisasi blue$
 $H = Hue$
 $S = Saturation$
 $V = Value$

3. METODOLOGI PENELITIAN

Metodologi pada penelitian ini dilakukan dalam beberapa tahap yang ditunjukkan pada Gambar 1.

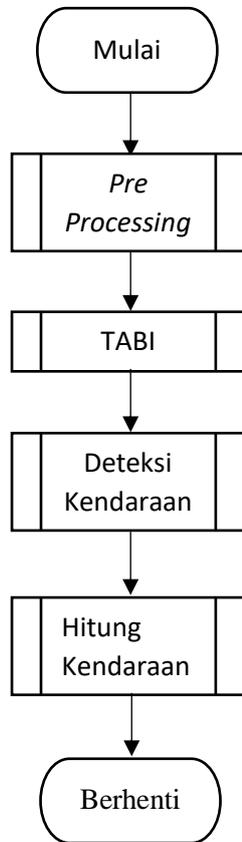


Gambar 2 - Metodologi Penelitian

4. PERANCANGAN

Pada tahap perancangan perangkat lunak ini akan menerapkan metode *Combination Value Saturation* untuk kasus penghitungan kendaraan yang dijalankan pada aplikasi dan digambarkan dengan diagram alir sesuai

dengan analisis yang dilakukan sebelumnya. Dimana diagram alir tentang aplikasi tersebut akan digambarkan dengan diagram alir berikut.



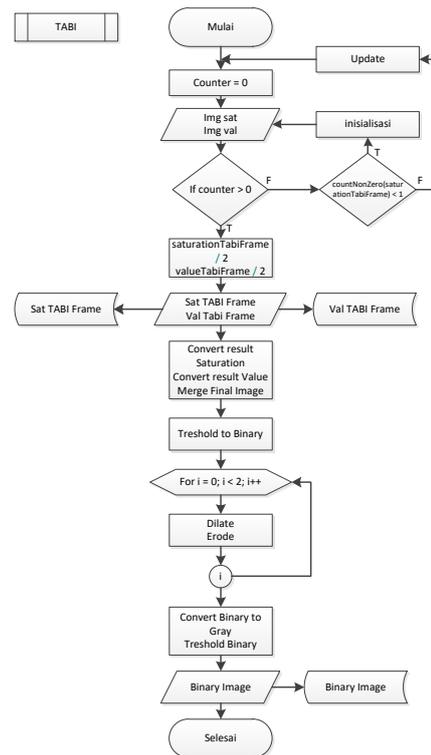
Gambar 3 Perancangan

4.1 Pre-Processing

Pada tahap pre processing didominasi konversi warna yang awalnya memiliki *color space* RGB menjadi *color space* HSV.

4.2 TABI (*Time Average Background Image*)

Setelah melakukan tahap *pre-processing* yang diawali ekstraksi video menjadi sekumpulan gambar dan merubah dimensi warna dari RGB ke HSV, langkah selanjutnya adalah melakukan proses TABI (*Time Average Background Image*), yaitu merata ratakan *frame* untuk merata ratakan nilai dari *frame* yang di deteksi seperti pada gambar berikut



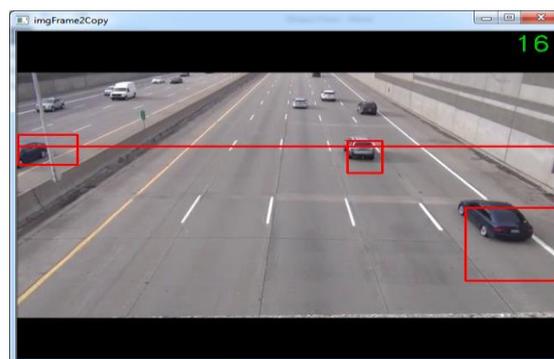
Gambar 4 TABI

4.3 Deteksi Kendaraan

Setelah proses TABI dilakukan, yang selanjutnya dilakukan adalah mendeteksi kendaraan, dimana proses ini akan memproses *Binary frame* yang didapatkan dari proses sebelumnya. Pada *frame* yang diproses kali ini akan menemukan *contour* dari *frame* yang diproses, kemudian akan menggambar dan menampilkan kembali *contour* yang ditemukan. Selanjutnya dari setiap *contour* yang ditemukan, akan di cari *convex hull* dari setiap *contour* yang ditemukan dengan beberapa parameter yang telah di tentukan, kemudian *convex hull* tersebut dimasukan kedalam *container* dan dicari apakah itu terdeteksi sebagai blob, jika blob ditemukan pada *frame* pertama, maka akan langsung dimasukan kedalam *container* blobs, jika bukan *frame* pertama maka akan dilakukan pencocokan blob sesuai dengan parameter yang ada.

4.4 Perhitungan Kendaraan

Setelah melalui proses deteksi kendaraan yang telah dilakukan sebelumnya didapatkan blobs yang sudah memiliki kriteria tertentu, maka blobs yang telah didapatkan tersebut akan di cek, apakah blobs tersebut melewati garis horizontal yang telah ditentukan atau belum, jika blobs tersebut melewati garis horizontal tersebut, garis akan berubah menjadi warna hijau, jika blobs tidak melewati garis yang ditentukan, maka garis akan tetap berwarna merah.



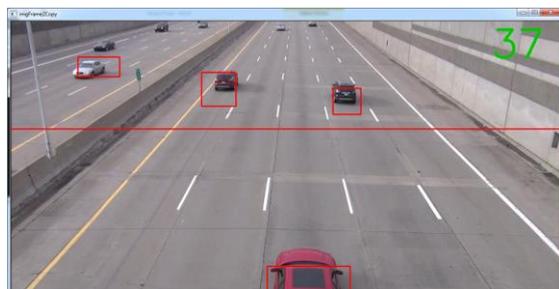
Gambar 6 640 x 480 pada proses 80%

5. PENGUJIAN

Pengujian kendaraan disini akan menguji kemampuan sistem *Intelligence vehicle counting* menggunakan metode *Combination Value Saturation* dalam melakukan penghitungan kendaraan. Dalam pengujian ini, sistem akan di uji pada 2 video berbeda dengan resolusi yang berbeda

5.1 Pengujian video 1024 x 768

Pada proses pengujian video dengan resolusi 1024 x 768 dan durasi selama 33 detik mendapatkan hasil 52 Actual Count dan 52 System count



Gambar 5 1024 x 768 pada proses 80%

5.2 Pengujian Video 640 x 480

Pada proses pengujian video dengan resolusi 640 x 480 dan durasi selama 33 detik mendapatkan hasil 52 Actual Count dan 16 System count

5.3 Perhitungan Akurasi

Pengujian akurasi disini akan menguji akurasi dari sistem *Intelligence vehicle counting* menggunakan metode *Combination Value Saturation* dalam melakukan penghitungan kendaraan dengan cara membandingkan perhitungan yang dilakukan oleh sistem dan perhitungan yang dilakukan secara manual seperti pada persamaan (6-1) dibawah ini.

$$Akurasi (\%) = \frac{\sum \text{Kendaraan yang berhasil di deteksi}}{\sum \text{Kendaraan hitung manual}} \times 100\%$$

$$Akurasi (\%) = \frac{(52+16)/2}{52} \times 100\%$$

$$Akurasi = 65.38 \%$$

Dalam persamaan (6-1) menunjukkan hasil pengujian akurasi dalam sistem *Intelligence vehicle counting* menggunakan metode *Combination Value Saturation* dengan menggunakan 2 video yang sama namun memiliki ukuran yang berbeda, sehingga menghasilkan sebuah perhitungan dari jumlah kendaraan yang berhasil di deteksi di antara 2 video tersebut dengan jumlah kendaraan yang sebenarnya dan dikalikan dengan 100%. Sehingga menghasilkan akurasi perhitungan kendaraan sebesar 68.38%.

6. KESIMPULAN

Berdasarkan hasil penelitian dari implementasi sistem *Intelligence vehicle counting* menggunakan metode *Combination Value Saturation*, maka diambil kesimpulan sebagai berikut :

1. Implementasi dari sistem *Intelligence vehicle counting* menggunakan metode *Combination Value Saturation* ini mampu mendeteksi kendaraan serta dapat mendeteksi kendaraan yang melintas dan menghitung jumlah kendaraan yang melintas pada video lalu lintas.
2. Sistem *Intelligence vehicle counting* menggunakan metode *Combination Value Saturation* ini memiliki akurasi 68.38% dari 2 video yang sama berdurasi 33 detik namun memiliki ukuran aspect ratio yang berbeda .

7. DAFTAR PUSTAKA

- A.MANDELLOS, N., KERAMITSOGLU, I., & KIRANOUDIS, C. T. 2011. A Background Subtraction algorithm for Detecting and Tracking Vehicles. *Expert System with Application*, 38(3), 1619-1631.
- ADHITAMA, M. O. 2011. *Evaluasi Pembangunan Terminal Hamid Rusdi Terhadap Efektivitas Pelayanan Transportasi di Kota Malang*. Malang: Brawijaya University.
- ALHADAR, A. 2011. Analisis Kinerja Jalan Dalam Upaya Mengatasi Kemacetan Lalu Lintas pada Ruas Simpang Bersinyal di Kota Palu. *SMARTek*, 9(4), 327-336.
- ARIS, A. 2013. *Analisis Dampak Sosial Ekonomi Pengguna Jalan Akibat Kemacetan Lalu Lintas (Studi Kasus Area Sekitar Universitas Brawijaya)*. Malang: Brawijaya University.
- EKAWATI, N. N. 2014. *Kajian Dampak Pengembangan Pembangunan Kota Malang Terhadap Kemacetan Lalu Lintas*. Malang: Brawijaya University.
- FADLISYAH. 2007. *Computer Vision dan Pengolahan Citra*. Yogyakarta: Andi.
- G.SALVI. t.thn.. An Automated Vehicle Counting System Based on Blob Analysis for Traffic Surveillance. *Departement of Economic Studies, University of Naples, "Parthenope", Naples, Italy*.
- GATRA, K. 2014, April 15. *Populasi Kendaraan Bermotor di Indonesia Tembus 104,2 Juta Unit*. (Kompas) Dipetik March 18, 2015, dari <http://otomotif.kompas.com/read/2014/04/15/1541211/Populasi.Kendaraan.Bermotor.di.Indonesia.Tembus.104.2.Juta.Unit>
- HARIYANTO, B. 2012. *Counter, Alat Penghitung Kendaraan*. Nagreg, Jawa Barat: Trans TV.
- HERMAWATI, F. A. 2013. *Pengolahan Citra Digital Konsep & Teori* (1 ed.). Yogyakarta: Andi.
- LIEN, C. C., HSIEH, C. T., & TSAI, M. H. 2013. Vehicle Counting Without Background Modelling. *Journal of Marine Science and Technology*, 21(6), 631-638.
- M.DAIGAVANEE, P., & BAJAJ, D. P. 2010. Real Time Vehicle Detection and Counting Method for Unsupervised Traffic Video on Highways. *International Journal of Computer Science and Network Security*, 10(8), 112-116.
- PUTRA, D. 2010. *Pengolahan Citra Digital* (1 ed.). Yogyakarta: Andi.
- RAD, A. G., DEHGhani, A., & KARIM, M. R. 2010. Vehicle speed detection in video image squences using CVS Method. *International Journal of Physical Sciences*, 5(17), 2555-2563.
- SUSMITA A, M., & A.V.MALVIYA. 2013. Traffic Surveillance by counting and classification of vehicles from video using image processing. *International Journal of Advance Research in Computer Science and Management Studies*, 1(6), 2321-7782.
- SUTRISNO., CHOLISSODIN, I., CHRISTANTI, R., DEWI, C., HIDAYAT, N., 2015. Segmentasi

kendaraan menggunakan improve blob analysis (BA) pada video lalu lintas. Jurnal Teknologi Informasidan Ilmu Komputer (JTIK), Volume 2, pp. 67-72.

T.SUTOJO, MULYANTO, E., & SUHARTONO, D. V. 2011. *Kecerdasan Buatan*. Yogyakarta: Andi.

T.SUTOYO, S. M., EDI MULYANTO S.SI, M., SUHARTONO, D., OKY DWI NURHAYATI, M., & M.Kom, W. 2009. *Teori Pengolahan Citra Digital* (1 ed.). Yogyakarta: Andi.