

# PERANCANGAN DAN IMPLEMENTASI HISTOGRAMS OF ORIENTED GRADIENTS DAN SUPPORT VECTOR MACHINES (HOG+SVM) UNTUK DETEKSI OBYEK PEJALAN KAKI PADA APLIKASI MOBILE BERBASIS ANDROID

## DESIGN AND IMPLEMENTATION OF HISTOGRAMS OF ORIENTED GRADIENTS AND SUPPORT VECTOR MACHINES (HOG+SVM) FOR PEDESTRIAN DETECTION ON ANDROID BASED MOBILE APPLICATION

<sup>1</sup>Reza Ilmi  
1104110045

<sup>2</sup>Astri Novianty, S.T, M.T  
10800597-1

<sup>3</sup>Umar Ali Ahmad, S.T, M.T  
11850763-1

<sup>123</sup>Prodi S1 Teknik Komputer, Fakultas Teknik Elektro, Universitas Telkom  
Jl. Telekomunikasi, Dayeuh Kolot, Bandung 40257, Indonesia

<sup>1</sup>[rezailmi@live.com](mailto:rezailmi@live.com), <sup>2</sup>[astrinov@telkomuniversity.ac.id](mailto:astrinov@telkomuniversity.ac.id), <sup>3</sup>[umar@telkomuniversity.ac.id](mailto:umar@telkomuniversity.ac.id)

### ABSTRAK

Setiap tahun, 1,24 juta kematian di seluruh dunia terjadi di jalan raya. 92% kecelakaan lalu lintas terjadi di negara-negara berkembang, termasuk Indonesia. 22% korban dari kecelakaan di jalan raya adalah pejalan kaki, karena itu pejalan kaki dikenal sebagai pengguna jalan paling rentan. Kecelakaan pejalan kaki sering kali disebabkan oleh kelalaian pengemudi kendaraan di jalan raya. Untuk meminimalisir kecelakaan terhadap pejalan kaki, dibutuhkan suatu *driver assistance system* di kendaraan yang praktis diaplikasikan serta handal.

Oleh karena itu diimplementasikan pengolahan citra dengan HOG+SVM sebagai metode untuk mendeteksi obyek pejalan kaki. HOG+SVM dibangun pada aplikasi *mobile* berbasis Android yang diaplikasikan untuk kendaraan dengan menggunakan *smartphone car holder* sebagai alat bantu menempatkan *device* di kendaraan. Aplikasi dapat mendeteksi obyek pejalan kaki lalu memberi *warning* kepada pengemudi jika obyek terlalu dekat dengan kendaraan atau saat sebelum kemungkinan terjadinya kecelakaan.

Hasil dari penelitian tugas akhir ini adalah didapat akurasi untuk kasus-kasus obyek pejalan kaki di jalan raya perkotaan di Indonesia rata-rata 81,75%. Sistem deteksi mampu mencapai performansi kecepatan deteksi hingga 27 fps, artinya sistem sudah dapat dikategorikan sebagai sistem deteksi yang *real-time*.

**Kata Kunci :** *computer vision, pedestrian detection, HOG+SVM, pengolahan citra, aplikasi mobile*

### ABSTRACT

Each year, about 1,24 million people die as a result of road traffic accidents around the world. 92% of road traffic accidents occur in developed countries, including Indonesia. 22% of the victims are pedestrians, because of that pedestrians known as most vulnerable road users. The majority of pedestrian accidents are caused by driver's negligence on the road. To minimize accidents on pedestrians, we need to develop a driver assistance system for vehicles that is simple to use and reliable.

Therefore, we implement the HOG+SVM, a pattern recognition method, to detect pedestrian objects. The HOG+SVM method will be implemented for Android mobile application and placed in the vehicle with smartphone car holder as a tool to put the device on the car dashboard. The apps can detect pedestrian objects then give a warning to the driver if the pedestrian objects are too close with the vehicle or before the possibilities of accident occur.

The results of this research were obtained average accuracy for cases of pedestrians in urban roads in Indonesia about 71.38%. As for the performance gained speed detection is up to 27 fps. So that, the system can be categorized as a real-time detection system.

**Keywords :** *computer vision, pedestrian detection, HOG+SVM, image processing, mobile application*

## 1. PENDAHULUAN

### 1.1 Latar Belakang

Setiap tahun, 1,24 juta kematian di seluruh dunia terjadi di jalan raya. 92% kecelakaan lalu lintas terjadi di negara-negara berkembang, termasuk Indonesia. 22% korban dari kecelakaan di jalan raya adalah pejalan kaki, karena itu pejalan kaki dikenal sebagai pengguna jalan paling rentan. Kecelakaan pejalan kaki sering kali disebabkan oleh kelalaian pengemudi kendaraan di jalan raya<sup>1</sup>. Untuk itu dibutuhkan solusi untuk mengatasi masalah ini demi meningkatkan keamanan dan kenyamanan tiap pengguna jalan, baik pejalan kaki maupun pengemudi kendaraan bermotor.

*Pedestrian Detection* atau deteksi obyek pejalan kaki adalah kunci penting dalam pengembangan bidang ilmu *Computer Vision*, karena berkaitan langsung dengan pengembangan aplikasi sistem keamanan kendaraan, sistem kamera pemantau hingga robotika, dan mendapat perhatian lebih dalam beberapa tahun terakhir [2]. Dalam sepuluh tahun pengembangan deteksi obyek pejalan kaki, terdapat hingga empat puluh lebih metode deteksi. Metode-metode ini memiliki empat pola dasar utama, yaitu *Viola & Jones*, *Histograms of Oriented Gradients* dengan pendekatan *Support Vector Machines* (HOG+SVM), *Deformable Part Detectors* (DPM), dan *Convolutional Neural Networks* (ConvNets) [6]. HOG+SVM dikenal sebagai metode yang paling umum digunakan untuk deteksi obyek pejalan kaki dan telah didukung dengan baik untuk implementasi di *platform mobile*. Dasar dari metode ini adalah penampilan serta bentuk obyek dapat diketahui melalui hasil komputasi gradien dari citra yang dideteksi [4].

Deteksi obyek pejalan kaki dapat diimplementasikan dalam kehidupan untuk membantu mengatasi masalah-masalah yang selama ini menyebabkan kecelakaan di jalan raya, terutama untuk meningkatkan kewaspadaan pengemudi kendaraan bermotor terhadap pejalan kaki sebagai pengguna jalan paling rentan. Kurangnya kewaspadaan pengemudi seperti mengantuk dan kehilangan fokus saat mengemudi adalah penyebab terjadinya kecelakaan terhadap pejalan kaki. Untuk meminimalisir kecelakaan terhadap pejalan kaki, dibutuhkan suatu *driver assistance system* disetiap kendaraan bermotor. Solusi yang dikembangkan harus dapat digunakan dengan mudah, memiliki akurasi dan waktu deteksi yang baik. Pada penelitian tugas akhir ini, metode HOG+SVM diimplementasikan di aplikasi *mobile* berbasis Android karena saat ini telah banyak *smartphone* dengan Android yang didukung dengan *graphics processing unit* (GPU) yang baik sehingga cocok untuk implementasi metode HOG+SVM untuk kebutuhan komputasi. Selain itu, *driver assistance system* dengan menggunakan perangkat *smartphone* memiliki keunggulan karena praktis untuk diintegrasikan dengan setiap kendaraan.

### 1.2 Tujuan

Penelitian tugas akhir ini dilakukan dengan tujuan membuat *Driver Assistance System* (DAS) dengan mengimplementasikan metode HOG+SVM untuk deteksi obyek pejalan kaki di aplikasi *mobile* berbasis Android, serta memahami performansi metode HOG+SVM untuk deteksi obyek pejalan kaki pada aplikasi *Driver Assistance System* yang dikembangkan.

### 1.3 Rumusan Masalah

Dalam pengerjaannya, penelitian tugas akhir ini mengacu pada rumusan masalah sebagai berikut:

1. Bagaimana mengembangkan *Driver Assistance System* yang praktis serta handal pada perangkat *mobile* dengan implementasi metode HOG+SVM untuk deteksi obyek pejalan kaki?
2. Bagaimana performansi metode HOG+SVM dalam implementasi di perangkat *mobile* dan efektifitas metode dalam deteksi obyek pejalan kaki untuk pengembangan *Driver Assistance System*?

### 1.4 Batasan Masalah

Pada penelitian tugas akhir ini ditentukan batasan-batasan masalah sebagai berikut:

1. *Driver Assistance System* yang akan dibangun difokuskan untuk membantu pengemudi menghindari *obstacle* atau halangan berupa obyek pejalan kaki.
2. Aplikasi berjalan pada sistem operasi *mobile* berbasis Android dengan perangkat yang memiliki kamera *built-in*.
3. Deteksi obyek pejalan kaki dilakukan saat cuaca cerah dan tidak pada malam hari.
4. Obyek pejalan kaki didefinisikan sebagai:
  - a. Manusia.
  - b. Posisi diatas jalan/dataran lainnya.
  - c. Pejalan kaki berada dalam pose tegak dari ujung kepala hingga kaki. Tidak melakukan pose selain berdiri, berjalan dan berlari.

<sup>1</sup> Berdasarkan data WHO *Global Status Report on Road Safety* di [www.who.int/roadsafety/en/](http://www.who.int/roadsafety/en/)

5. Sistem deteksi untuk *learning* menggunakan SVM menggunakan 2000 data positif (gambar manusia) dan 2000 data negatif (gambar bukan manusia) dari INRIA *Person Dataset*. Setiap data gambar berukuran 64 x 128 piksel.
6. Obyek pejalan kaki minimal mempunyai tinggi 128 piksel dari ukuran resolusi *viewfinder* kamera.

## 2. Dasar Teori

### 2.1 Computer Vision

*Computer Vision* merupakan bidang ilmu yang mempelajari bagaimana komputer dapat merekonstruksi, menginterpretasikan, dan memahami suatu tampilan 3 dimensi dalam tampilan 2 dimensi berdasarkan sifat dari struktur tampilan tersebut. *Computer Vision* juga mempelajari bagaimana komputer dapat mengenali obyek yang diamati. Selain itu *Computer Vision* mencakup metode-metode dalam perolehan, pemrosesan, analisis, pemahaman citra, dan data-data lain dari dunia nyata secara umum untuk menghasilkan informasi numerik atau simbolik.

*Computer Vision* telah berkembang dalam berbagai bidang dengan tujuan yang berbeda-beda. Beberapa contoh dari pengaplikasian *Computer Vision* antara lain proses pengontrolan di bidang industri, pemodelan medis dan topografis, navigasi dalam transportasi, perangkat dalam interaksi manusia-komputer, dan pendeteksian kejadian seperti dalam kamera pengawas.

Informasi numerik dan simbolik dalam *Computer Vision* disebut visualisasi data. Bentuk visualisasi data yaitu Pengolahan Citra (*Image Processing*) dan Pengenalan Pola (*Pattern Recognition*).

Pengolahan Citra (*image processing*) merupakan bidang dalam *Computer Vision* yang berhubungan dengan proses transformasi citra atau gambar. Proses ini bertujuan untuk mendapatkan kualitas citra yang lebih baik. Proses Penangkapan Citra (*image acquisition*), Proses Pengolahan Citra (*image processing*), Analisa data citra (*image analysis*), Proses pemahaman data citra (*image understanding*).

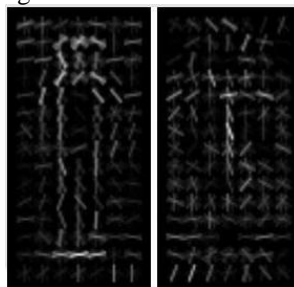
### 2.2 Histogram of Oriented Gradient

Dalam *Computer Vision* khususnya untuk kasus deteksi obyek, seringkali perlu untuk melihat nilai dari masing-masing piksel dari sebuah gambar, baik itu nilai intensitas cahaya dalam piksel, nilai komposisi warna tertentu dalam piksel maupun informasi lain yang terkandung dalam piksel. Nilai dari tiap-tiap piksel akan dibagi ke dalam kelompok *Channel* tertentu dan direpresentasikan dengan diagram batang yang biasa disebut dengan Histogram.

*Histogram of Oriented Gradient* (HOG) adalah metode yang digunakan untuk deteksi obyek, histogram berisi *channel-channel* arah/orientasi *gradient* dari piksel-piksel pada gambar, dimana penampilan serta bentuk obyek dapat diketahui melalui hasil komputasi *gradient* dari citra. Tahap awal dari metode ini adalah dengan melakukan normalisasi *gamma* pada citra. Kemudian menghitung *gradient* dari setiap piksel citra, pembagian citra ke dalam *cell*, membentuk histogram dari masing-masing *cell*, membentuk blok dari setiap histogram dan terakhir melakukan normalisasi dari setiap blok.

Pada metode ini, fitur HOG dapat diperoleh dari membagi gambar ke dalam sel-sel berukuran  $n \times n$ , lalu dikelompokkan ke dalam blok-blok berukuran  $2n \times 2n$  yang saling beririsan satu sama lain. Dari tiap sel masing-masing blok, di hitung *magnitude* dan orientasi *gradient-nya*. Nilai orientasi ini dihitung dengan menghitung konvolusi dengan matriks  $\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$  dan  $\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}^T$  terlebih dahulu.

Kemudian nilai orientasi tiap pikselnya dikuantisasi kedalam 9 kanal, yaitu  $10^\circ, 30^\circ, 50^\circ, 70^\circ, 90^\circ, 110^\circ, 130^\circ, 150^\circ$ , dan  $170^\circ$  menggunakan histogram. Kontribusi piksel terhadap tiap kanal bergantung pada nilai *gradient magnitude-nya*. Nilai-nilai dari seluruh kanal dari tiap sel dimasukkan kedalam vektor. Vektor inilah yang menggambarkan fitur HOG dari suatu gambar.



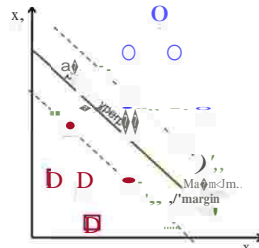
Gambar 2-1 Contoh fitur HOG yang dihasilkan [4]

### 2.3 Support Vector Machine (SVM)

Bahasan tentang *machine learning* merupakan bahasan yang menjadi bagian dari *computer vision*, yaitu dengan tidak memandang konten citra hanya dari sisi proyeksi perspektif atau cahaya saja, melainkan juga dengan melihatnya sebagai bahasan menyangkut pengenalan pola. Dalam kasus penelitian tugas akhir ini, pola yang harus dikenali adalah tubuh manusia sebagai obyek pejalan kaki melalui fitur HOG-nya menggunakan algoritma *Support Vector Machine* (SVM) dalam melakukan proses *training*.

SVM merupakan metode *learning* melalui *training* yang bertujuan untuk menentukan garis pemisah antara 2 kelas agar dapat menentukan termasuk ke dalam kelas manakah sebuah input. SVM melakukan ini dengan memetakan data *training* positif dan negatif ke dalam *high-dimensional feature space* berdasarkan fitur-fitur yang terkandung di dalam data tersebut [3].

Setelah seluruh data *training* dipetakan, kemudian dihitung garis pemisah (*hyperplane*) diantara dua kelompok data tersebut yang memiliki margin paling besar ke vektor data latih positif dan negatif yang terdekat. Penentuan margin vektor-vektor data latih inilah yang disebut dengan *support vector*.



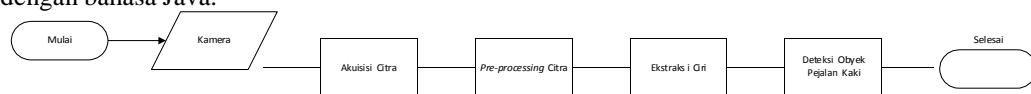
Gambar 2-2 Hyperplane SVM sebagai pemisah kelas positif dan negatif<sup>2</sup>

### 3. Perancangan dan Implementasi

Dalam penelitian tugas akhir ini, akan dibangun *Driver Assistance System* (DAS) yang terintegrasi dan diimplementasikan pada perangkat bergerak berbasis sistem operasi Android. Pengembangan sistem ini didasarkan kebutuhan atas aplikasi DAS yang ringkas dan mudah diaplikasikan ke berbagai jenis kendaraan yang umum digunakan. Pada kasus implementasinya di jalanan perkotaan, aplikasi DAS yang dibangun diberi pembatasan fokus pengembangannya untuk membantu pengemudi menghindari *obstacle* atau halangan berupa obyek pejalan kaki. Obyek pejalan kaki dipilih karena jumlah korban kecelakaan di jalan raya sebahagian besar adalah pejalan kaki menurut data penelitian WHO.

Sistem deteksi obyek pejalan kaki pada aplikasi DAS yang dibangun menggunakan metode *Histogram of Oriented Gradient* dan *Support Vector Machine* pada aplikasi mobile berbasis android. Aplikasi *Driver Assistance Systems* ini akan membantu pengemudi kendaraan agar selalu waspada terhadap pejalan kaki, dengan memberikan tanda peringatan dini (*warning alarm*) jika terdapat obyek pejalan kaki yang berada di depan kendaraan dengan jarak tertentu dan berada pada ruang tangkap *viewfinder* kamera perangkat.

Agar Aplikasi *Driver Assistance Systems* yang dikembangkan diatas *platform* Android dapat efektif dan tepat guna ketika implementasinya, sistem harus dapat melakukan *processing* yang cepat dan *real-time*. Berdasarkan [1], untuk meningkatkan akurasi dan waktu deteksi disarankan menggunakan pendekatan *Native C/C++ Development*, yaitu pengembangan aplikasi Android yang proses logikanya dilakukan di lapisan *Native*, ditulis dengan bahasa C/C++. Kemudian ditampilkan hasil prosesnya ke lapisan *Application* yang ditulis dengan bahasa Java.

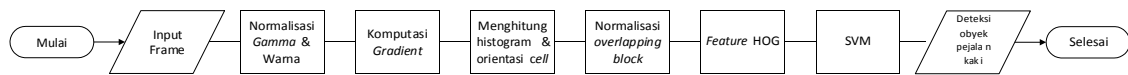


Gambar 3-1 Flowchart sistem

#### 3.1 Proses Deteksi HOG+SVM

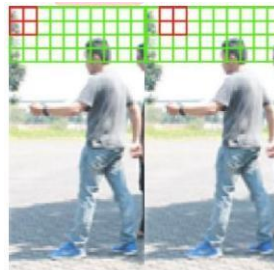
Proses deteksi obyek pejalan kaki yang diimplementasikan di aplikasi *mobile* berbasis Android ini menggunakan metode *Histogram of Oriented Gradients*. Pada metode ini juga dilakukan proses *learning* dengan *Support Vector Machine* (SVM). Alur proses deteksi digambarkan pada gambar 3-2 Proses deteksi obyek dengan metode HOG+SVM.

<sup>2</sup> Sumber gambar dari <http://docs.opencv.org>



**Gambar 3-2** Proses deteksi obyek dengan metode HOG+SVM

Tahap pertama metode HOG+SVM adalah melakukan normalisasi Gamma terhadap masukan citra digital. Pada tahap normalisasi ini dihitung akar kuadrat dari setiap *channel* warna. Hasil dari tahap normalisasi *channel* warna adalah garis tepi antara obyek-obyek dicitra atau *frame* yang sedang diproses. Semakin jelas garis tepi atau *edge* yang tampak, maka orientasi arah vektor *histogram*-nya akan lebih mudah didapat. Di tahap selanjutnya, *gradient* dari citra digital dihitung untuk kemudian citra dibagi ke dalam *cell-cell* yang berukuran 8x8 piksel, digambarkan dalam gambar 3.3-4 dengan kotak berwarna merah. Masing-masing *cell* tersebut dibentuk histogram-histogram yang nantinya akan digabungkan ke kelompok-kelompok yang lebih besar berukuran 16x16 piksel. Kelompok-kelompok gabungan histogram ini disebut dengan *block*. Setelah itu, dilakukan normalisasi terhadap histogram di setiap *block*. Hasil normalisasi kemudian digabungkan untuk mendapatkan data fitur HOG. *Block* ini saling beririsan dengan *block* satu dengan yang lainnya seperti digambarkan di gambar 3-3, *block* pertama digambar kiri, dan *block* kedua digambar kanan.



**Gambar 3-3** Contoh *Cell* 8x8 (Kotak Hijau) dan *Block* (Kotak Merah)

Setiap *cell* dikuantisasi nilai *gradient vector*-nya kedalam 9 kanal, untuk setiap *gradient vector* di tiap kanal histogram tersebut nilainya mempengaruhi nilai orientasi vector pada fitur HOG yang akan dibentuk. Semakin besar kuantitas vektor histogram pada kanal tersebut maka akan semakin jelas arah orientasi vektornya.

Fitur-fitur HOG dari tiap citra ditandai dengan nilai +1 untuk citra positif dan -1 untuk citra negatif obyek target. Untuk mempercepat proses deteksi obyek pejalan kaki, dilakukan proses *learning* dengan *linear SVM*, dimana konsep utamanya adalah mencari *hyperplane* yang optimal untuk memisahkan kedua kelas [3]. Lebih jauh tentang SVM akan dijelaskan di subbab berikutnya. Diakhir proses, akan diklasifikasikan obyek-obyek pejalan kaki dan yang bukan obyek pejalan kaki.

Secara garis besar, metode HOG adalah tentang bagaimana mencari garis tepi atau *edge* antar obyek di citra yang sedang diproses dari *gradients* yang dibentuk dari gradasi antar piksel, untuk kemudian diklasifikasikan ke obyek target atau bukan dengan melihat kecocokan dengan fitur-fitur HOG dari data latih. Orientasi arah dari garis-garis tepi tersebutlah alasan mengapa metode ini disebut dengan *Oriented Gradients*. Sedangkan *Histogram* berkaitan dengan perhitungan seberapa banyak piksel yang memiliki garis tepi dengan orientasi arah yang spesifik.

## 4. Pengujian dan Analisis

### 4.1 Waktu Deteksi

Untuk mengetahui sejauh mana aplikasi dapat mendeteksi obyek pejalan kaki dengan optimal, maka akan dilakukan pengujian waktu deteksi. Analisis waktu sistem deteksi dilakukan dengan beberapa skenario yaitu dengan mengubah resolusi frame untuk mengetahui nilai waktu deteksi dalam satuan *Frame per second* (fps), dan *detection time* (ms). Perubahan ukuran resolusi layar mengacu pada ukuran resolusi layar sebenarnya *smartphone* Xiaomi Redmi 1s yaitu 1280 x 720 piksel, kemudian diubah  $\frac{1}{2}$  dari ukuran resolusi layar sebenarnya yaitu 640 x 480 piksel, dan  $\frac{1}{4}$  dari ukuran resolusi layar yang sebenarnya yaitu 320 x 240 piksel. Dari pengujian waktu deteksi yang telah dilakukan, didapatkan hasil pada tabel 4-1 hasil pengujian waktu deteksi.

**Tabel 4-1** Hasil pengujian waktu deteksi

Resolusi Frame	Kecepatan Deteksi (fps)	Detection Time (ms)
320 x 240	22 - 27	512,892 - 376,235
640 x 480	7 - 10	1412,02 - 1147,69
1280 x 720	1-3	8914.03 - 8564.29

Waktu deteksi yang direpresentasikan dalam *frame per second* menggambarkan seberapa *real-time* yang dihasilkan oleh *preview* sistem dengan kondisi sebenarnya. Semakin besar nilai fps yang dihasilkan oleh sistem maka akan semakin sesuai dengan kondisi sebenarnya. Pada skenario pengujian ini didapatkan hasil bahwa resolusi *frame* terbaik adalah dengan menggunakan resolusi 320 x 240 piksel. Pada resolusi ini, sistem dapat memproses setiap *frame*-nya dengan waktu deteksi mulai dari 376,235 ms hingga 512,892 ms. Kecepatan deteksi tersebut membuktikan bahwa sistem telah dapat menentukan klasifikasi deteksi obyek pejalan kaki dalam waktu yang cepat dan cukup sesuai dengan kondisi sebenarnya. Pada metode deteksi HOG+SVM, sebelum klasifikasi obyek pejalan kaki dimulai, tiap *frame* yang akan diproses akan dipetakan menjadi matriks. Semakin kecil ukuran resolusinya, maka ukuran matriks yang akan diproses juga akan semakin kecil. Dari hasil pengujian ini, parameter resolusi paling optimal untuk metode HOG+SVM adalah dengan resolusi 320 x 240 piksel dan akan digunakan untuk skenario pengujian berikutnya.

#### 4.2 Akurasi Deteksi

Pengujian akurasi deteksi obyek pejalan kaki dilakukan untuk mengetahui sejauh mana aplikasi dapat mendeteksi obyek pejalan kaki secara akurat. Untuk mengetahui nilai akurasi deteksi (%), digunakan parameter uji yaitu False Positive (background atau obyek lain yang terdeteksi sebagai orang) dan False Negative (obyek orang yang tidak terdeteksi). Serta kasus bagaimana orang melewati suatu area, yaitu dengan renggang satu sama lain, berhimpitan, bergerombol, baju mirip background, berpapasan, berlari dan membawa obyek besar. Kasus-kasus tersebut diambil berdasarkan hasil pengamatan dari video-video yang digunakan dalam penelitian sebelumnya [13] [14] [15].

Dari pengujian akurasi deteksi yang telah dilakukan, didapatkan hasil pada tabel 4-2 Hasil Pengujian Akurasi Deteksi. Resolusi paling optimal yang memiliki kecepatan waktu deteksi paling baik untuk metode HOG+SVM pada pengujian waktu deteksi sebelumnya, diperoleh sistem dengan akurasi antara 41,11% hingga 100% tergantung oleh kasus-kasus obyek pejalan kaki pada skenario pengujian. Perolehan akurasi terendah terjadi saat sistem melakukan deteksi obyek pejalan kaki dimana obyek-obyek berada pada jarak yang sangat berdekatan dan bergerombol. Hal ini disebabkan karena sistem kesulitan menghitung derajat orientasi gradien dari obyek sehingga fitur HOG-nya menjadi lebih mirip dengan data latih citra obyek negatif. Akurasi tertinggi diperoleh saat sistem melakukan deteksi obyek pejalan kaki yang jarak antar obyeknya saling berjauhan. Sistem dapat dengan baik menghitung jumlah pejalan kaki dengan kasus berhimpitan hingga akurasi 100%. Hal ini disebabkan karena metode HOG telah mengaplikasikan sistem *grouping* yaitu sistem pengelompokan apabila obyek target yang dideteksi berada berdekatan. Pengelompokan yang dilakukan adalah pengelompokan kotak deteksi atau *bounding box*. Dari hasil pengujian akurasi secara keseluruhan didapat rata-rata akurasi sistem adalah 81,75%. Mengingat implementasi metode dilakukan di perangkat *mobile* yang kemampuan komputasinya sangat terbatas, perolehan akurasi ini sudah cukup untuk dapat mendeteksi minimal 1 pejalan kaki di setiap kasus pejalan kaki yang menyeberang jalanan di perkotaan.

**Tabel 4-2** Hasil pengujian akurasi deteksi

Kasus Pejalan kaki	Jumlah Orang	Jumlah Kotak yang muncul	T	Fp	Fn	Akurasi	Akurasi per kasus
Berhimpitan	2	2	2	0	0	100%	100%
	3	3	3	0	0	100%	
	5	2	5	0	0	100%	
Tidak Berhimpitan	2	2	2	0	0	100%	93,33%
	3	3	3	0	0	100%	
	5	4	4	0	1	80%	
Begerombol	2	1	1	0	1	50%	41,11%
	3	1	1	0	2	33,33%	
	5	1	2	0	3	40%	

Berpapasan	2	2	2	0	0	100%	93,33%
	3	3	3	0	0	100%	
	5	2	4	0	1	80%	
Baju Mirip Background	2	2	2	0	0	100%	75,56%
	3	2	2	0	1	66,67%	
	5	3	3	0	2	60%	
Membawa obyek besar	2	2	2	0	0	100%	75,56%
	3	2	2	0	1	66,67%	
	5	3	3	0	2	60%	
Berlari	2	2	2	0	0	100%	93,33%
	3	3	3	0	0	100%	
	5	4	4	0	1	80%	
Akurasi rata-rata							81,75%

#### 4.3 Sistem DAS

Pengujian sistem deteksi obyek pejalan kaki dilakukan untuk mengetahui apakah sistem deteksi sudah dapat berjalan dengan optimal dalam kondisi tertentu. Pengujian sistem akan dilakukan dengan mengubah skenario kecepatan kendaraan berdasarkan kondisi jalan raya yang sebenarnya yaitu sesuai dengan Peraturan Pemerintah Nomor 79 Tahun 2013, Pasal 23, ayat 4 tentang Jaringan Lalu Lintas dan Angkutan Jalan, Undang-Undang No.22 Tahun 2009, Pasal 287, Ayat 5 Tentang batas kecepatan kendaraan.

**Tabel 4-3** Batas kecepatan Kendaraan

Kecepatan Kendaraan (km/jam)	Kondisi Jalanan
10	Jalanan Padat
25	Jalanan Pemukiman
50	Jalanan Dalam Kota

Dari hasil pengujian sistem yang telah dilakukan, didapatkan hasil pada tabel 4-4 hasil pengujian sistem DAS. Dari hasil pengujian pada skenario sistem *Driver Assistance System* dalam implementasinya dalam kasus-kasus jalanan perkotaan di Indonesia yang dilakukan pada kondisi sebenarnya didapatkan hasil bahwa pada hampir seluruh kasus pejalan kaki, sistem dapat dengan tepat memberikan *warning alarm* atau nada peringatan kepada pengemudi agar lebih berhati-hati karena sistem mendeteksi ada pejalan kaki yang sedang melintas di ruang pandang kamera sistem. Hanya saja, terdapat pada 1 kasus dengan kecepatan mobil 10 km/h, sistem gagal mendeteksi ada pejalan kaki yang melintas. Hal ini dapat disebabkan oleh kondisi pejalan kaki yang warna bajunya menyerupai latar belakang sekitarnya. Sesuai dengan hasil pengujian akurasi sistem di skenario sebelumnya, pada kasus pejalan kaki yang menyerupai latar belakang sekitarnya, akurasi sistem tidak terlalu baik. Namun demikian, sistem dapat dengan mudah mengenali pejalan kaki dengan pose tegak dan berjalan yang melintasi kendaraan yang sedang bergerak.

**Tabel 4-4** Hasil pengujian sistem DAS

Kecepatan(km/h)	Kasus	Jumlah Orang Sebenarnya	Jumlah Terdeteksi	Detection Time(ms)
10	1	1	1	398.379ms
	2	1	1	424.505ms
	3	1	1	418.463ms
	4	1	1	528.563ms
	5	2	0	648.483ms
25	1	2	1	534.243ms
	2	3	2	473.297ms
50	1	1	1	392.953ms

## 5. Kesimpulan

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan, dapat disimpulkan beberapa poin sebagai berikut:

1. Metode HOG+SVM dapat diimplementasikan pada perangkat aplikasi mobile berbasis Android untuk sistem deteksi obyek pejalan kaki dengan akurasi yang baik dan waktu deteksi cukup baik. Untuk mendapatkan waktu deteksi yang lebih baik dibutuhkan perangkat dengan *processor* dengan kemampuan komputasi yang lebih tinggi dari *Quad-core processor*.
2. Nilai performansi dengan parameter waktu deteksi dan akurasi pada perangkat *mobile* untuk sistem deteksi obyek pejalan kaki menggunakan metode HOG+SVM bergantung pada nilai resolusi *frame* kamera, semakin besar nilai resolusi *frame* kamera maka nilai waktu deteksi semakin berkurang, tetapi nilai akurasi deteksi meningkat. Sebaliknya, semakin kecil nilai resolusi *frame* kamera maka nilai waktu deteksi akan semakin meningkat. Hal ini disebabkan karena lebih banyak jumlah piksel yang harus diproses oleh sistem.
3. Metode HOG+SVM memiliki nilai akurasi yang cukup baik untuk kasus implementasi di perangkat *mobile* dengan *Quad-core Processor* sebagai *Driver Assistance System*, dengan nilai akurasi hingga 100% dan nilai akurasi rata-rata di semua kasus pejalan kaki berkisar 81,75%.
4. Berdasarkan nilai akurasi yang didapatkan dari pengujian dengan beberapa kasus, metode HOG+SVM tidak cocok digunakan pada kasus baju obyek mirip background, bergerombol, dan membawa obyek besar karena orientasi vektor-vektor yang terbentuk memiliki kecenderungan mirip dengan data latih citra negatif.
5. Sistem dapat menghasilkan kecepatan waktu deteksi yang *real-time*. Dapat dikategorikan kecepatan waktu deteksi yang *real-time* karena berdasarkan data *Average Frame Rate Used for Recording*<sup>3</sup> yang mempelajari tentang rata-rata kecepatan kamera yang digunakan untuk kamera keamanan atau pengawasan dan sebagainya termasuk kamera untuk pengawasan di jalan raya memiliki kecepatan 15 fps. Sistem deteksi DAS dengan metode HOG+SVM menghasilkan kecepatan kamera saat deteksi hingga 27 fps.
6. Dalam kasus pengembangan *Driver Assistance System* yang paling penting adalah bagaimana sistem dapat mendeteksi obyek pejalan kaki tepat sesuai kondisi *real-time* sebenarnya, walaupun akurasi jumlah obyek pejalan kaki yang dideteksi tidak sesuai dengan kondisi sebenarnya, namun kondisi minimal sistem dianggap berhasil jika telah memberi *warning alarm* ketika terdapat obyek pejalan kaki di *viewfinder* kamera.

## DAFTAR PUSTAKA

- [1] Lee, S., & Wook Jeon, J. (2010). Evaluating Performance of Android Platform Using Native C for Embedded Systems. *International Conference on Control, Automation and Systems*.
- [2] Benenson, R., Omran, M., Hosang, J., & Schiele, B. (2014). Ten Years of Pedestrian Detection, What Have We Learned? *European Conference on Computer Vision*.
- [3] Burges, C. J. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Kluwer Academic Publishers*.
- [4] Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. *Computer Vision and Pattern Recognition*.
- [5] Dinh, T., & dkk. (2011). *Online Multiple Object Tracking by*. Ho Chi Minh City: University of Science.
- [6] Dollar, P., Wojek, C., Schiele, B., & Perona, P. (2009). Pedestrian detection: A benchmark. *Computer Vision and Pattern Recognition*.
- [7] Guo, L., & dkk. (2010). *A New Method Combining HOG and Kalman Filter for Video-based Human Detection and Tracking*. Ningbo: Ningbo University.
- [8] Kadir, A., & Susanto, A. (2013). *Teori dan Aplikasi Pengolahan Citra*. Yogyakarta: ANDI.
- [9] Lv, P., & dkk. (2010). *A People Counting System based on Head-shoulder Detection and Tracking in Surveillance Video*. Wuhan: Wuhan University of Technology.

<sup>3</sup> Dari <http://ipvm.com/updates/1100> oleh J. Honovicg