

Klasifikasi Golongan Kendaraan Berdasarkan Fitur Histogram of Oriented Gradients (HOG) Menggunakan metode K-Nearest Neighbors (K-NN) Berbasis Raspberry PI 3

Lilo Nofrizal Akbar¹, Fitri Utamingrum²

Program Studi Teknik Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹nofrizal.lilo@gmail.com, ²f3_ningrum@ub.ac.id

Abstrak

Antrean yang terjadi pada saat melakukan pembayaran jalan tol masih menjadi permasalahan di Indonesia, permasalahan tersebut terjadi salah satunya karena tidak adanya sistem yang secara otomatis melakukan klasifikasi terhadap kendaraan yang lewat pada jalan tol, sehingga kendaraan besar seperti truk yang dibedakan menjadi 5 golongan harus secara manual dibedakan oleh petugas pintu tol. Salah satu upaya untuk mengatasi permasalahan tersebut, pada penelitian ini dirancang sebuah sistem untuk bisa secara otomatis mengklasifikasi kendaraan yang melewati jalan tol menjadi 5 golongan sesuai golongan yang berlaku pada jalan tol di Indonesia, sehingga proses klasifikasi yang sebelumnya dilakukan secara manual dapat dilakukan secara otomatis, ini tentunya membawa manfaat pada waktu transaksi pembayaran tiap kendaraan yang semakin singkat, yang pada akhirnya dapat mengurangi potensi kemacetan yang terjadi pada pintu tol. Sistem ini bekerja dengan berdasar pada *image processing*, proses klasifikasi diawali dengan pengambilan *video* kendaraan menggunakan *webcam*, selanjutnya *video* kendaraan diproses pada Raspberry Pi 3 untuk dilakukan ekstraksi fitur citra dari kendaraan menggunakan metode *Histogram of Oriented Gradients (HOG)*, kemudian fitur yang didapatkan dilakukan proses klasifikasi menggunakan metode *k-Nearest Neighbors (k-NN)* untuk menentukan kelas golongan kendaraan, hasil dari klasifikasi selanjutnya ditampilkan pada layar LCD. Dari pengujian yang dilakukan terhadap sistem dengan menggunakan 25 data uji, dengan tiap golongan sebanyak 5 data uji, didapatkan hasil akurasi sistem pada golongan 1 sebesar 80%, golongan 2 sebesar 80%, golongan 3 sebesar 60%, golongan 4 sebesar 60%, dan golongan 5 sebesar 60%.

Kata kunci: *klasifikasi kendaraan, jalan tol, histogram of oriented gradients, k-nearest neighbors, background subtraction, raspberry pi*

Abstract

The queues that occur when making toll road payments are still a problem in Indonesia, one of which is the problem because there is no system that automatically classifies vehicles passing on the toll road, so that large vehicles such as trucks that are divided into 5 classes must be manually differentiated by toll gate officers. One effort to overcome these problems, in this study a system was designed to be able to automatically classify vehicles passing the toll road into 5 classes according to the class applicable to the Indonesian toll road, so the classification process that was previously done manually can be done automatically, this of course bring benefits to the payment transaction time for each vehicle that is getting shorter, which in turn can reduce the potential for congestion that occurs at the toll gate. This system works based on image processing, the classification process begins with vehicle video capture using a webcam, then the vehicle video is processed on Raspberry Pi 3 to extract image features from the vehicle using the Histogram of Oriented Gradients (HOG) method, then the features obtained are processed The classification uses the k-Nearest Neighbors (k-NN) method to determine the class of vehicle, then results of the classification are displayed on the LCD screen. From the tests conducted on the system using 25 test data, with each class as many as 5 test data, the results obtained for system accuracy in class 1 by 80%, class 2 by 80%, class 3 by 60%, class 4 by 60%, and class 5 by 60%.

Keywords: *vehicle classification, tol road, histogram of oriented gradients, k-nearest neighbors, background subtraction, raspberry pi*

1. PENDAHULUAN

Antrean yang terjadi pada saat melakukan pembayaran jalan tol masih menjadi permasalahan tersendiri bagi Indonesia, permasalahan tersebut terjadi salah satunya karena tidak adanya sistem yang secara otomatis melakukan klasifikasi terhadap kendaraan yang lewat pada jalan tol, sehingga kendaraan besar seperti truk yang dibedakan menjadi 5 golongan harus secara manual dibedakan oleh petugas pintu tol untuk membedakan golongannya, ini tentunya semakin menambah potensi kemacetan. Bahkan tragedi Brexit beberapa tahun yang lalu, yang diiringi dengan adanya korban jiwa dari pengguna jalan yang disebabkan kelelahan saat menunggu antrean yang sangat panjang di saat libur lebaran. Belasan korban meninggal karena kecelakaan hingga kelelahan akibat macet yang tak kunjung terurai (Kumpran, 2016).

Dari permasalahan yang sudah dipaparkan tadi akan dibuat alat untuk mendeteksi kendaraan yang lewat pada gerbang tol untuk dilakukan klasifikasi sesuai golongan kendaraan tersebut secara otomatis berdasar pada *image processing* dengan menggunakan metode *Histogram of Oriented Gradient* (HOG) untuk mendapatkan fitur citra dari kendaraan. *Histogram of Oriented Gradient* (HOG) adalah *feature descriptor* yang digunakan untuk mengenali tampilan lokal dari objek. Teknik pada HOG melakukan perhitungan orientasi gradien pada bagian gambar yang akan dicari fitur HOG-nya. Digunakan metode HOG karena memiliki keuntungan fleksibilitas jika dibanding metode lain, karena pada metode ini mengacu pada *feature based* yang melihat tampilan lokal serta melihat objek yang terdapat pada gambar yang berdasar pada intensitas dan distribusi gradien (Kachouane, et al., 2012).

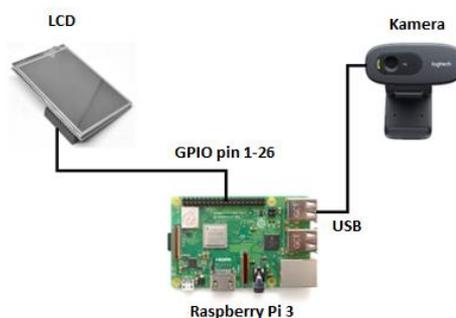
Setelah mendapat nilai fitur HOG citra dari kendaraan, selanjutnya dilakukan proses untuk mengklasifikasi objek yang terdapat pada citra menggunakan algoritma *k-Nearest Neighbors* (k-NN) untuk menentukan golongan kendaraan. Digunakan metode k-NN karena merupakan teknik yang sederhana, efisien, dan efektif digunakan untuk mengenali pola, maupun pengolahan objek yang lainnya, dan mampu untuk melakukan *training* data dengan skala besar (Bathia, 2010). K-NN melakukan klasifikasi terhadap data baru dengan cara membandingkan data baru dengan data latih yang sudah diberi label, dengan menghitung

kedekatan nilai tiap fitur yang dimiliki antara data uji dengan data pelatihan (Kusrini dan Lutfhi, 2009). Kemudian dicari sebanyak *k* data yang memiliki jarak terdekat, selanjutnya kelas yang paling dominan pada *k data* terdekat digunakan untuk prediksi kelas data uji. Selanjutnya sistem akan menampilkan hasil klasifikasi berupa jenis golongan kendaraan pada layar LCD.

2. PERANCANGAN DAN IMPLEMENTASI

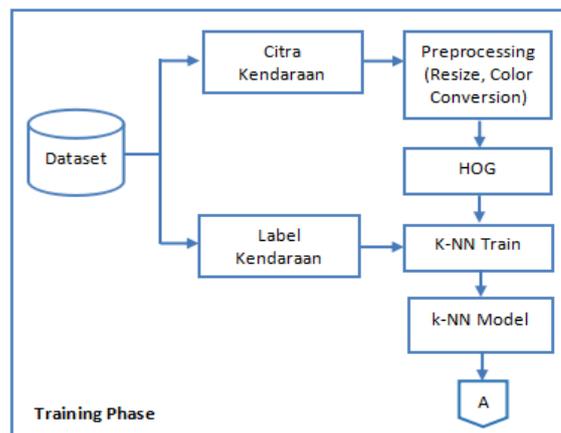
2.1. Perancangan Perangkat Keras

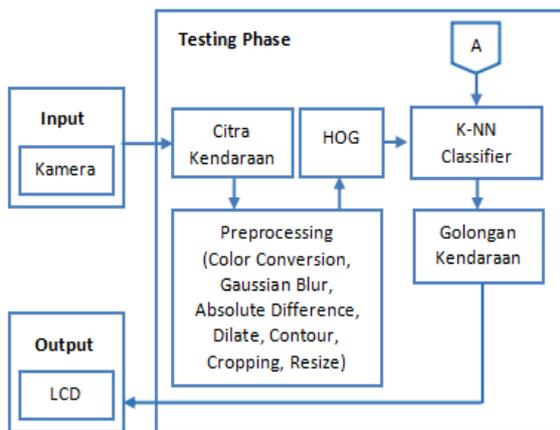
Sistem ini menggunakan tiga perangkat keras berupa kamera Logitech C207 yang digunakan sebagai penangkap citra berupa video, Raspberry Pi 3 digunakan sebagai pemroses masukan berupa citra video yang kemudian dilakukan proses klasifikasi untuk menentukan golongan dari kendaraan, dan layar LCD sebagai penampil keluaran sistem berupa video tangkapan kamera secara *real time* beserta hasil klasifikasi dari golongan kendaraan. Perancangan perangkat keras dapat dilihat pada Gambar 1.



Gambar 1. Perancangan rangkaian perangkat keras

2.1 Perancangan Perangkat Lunak





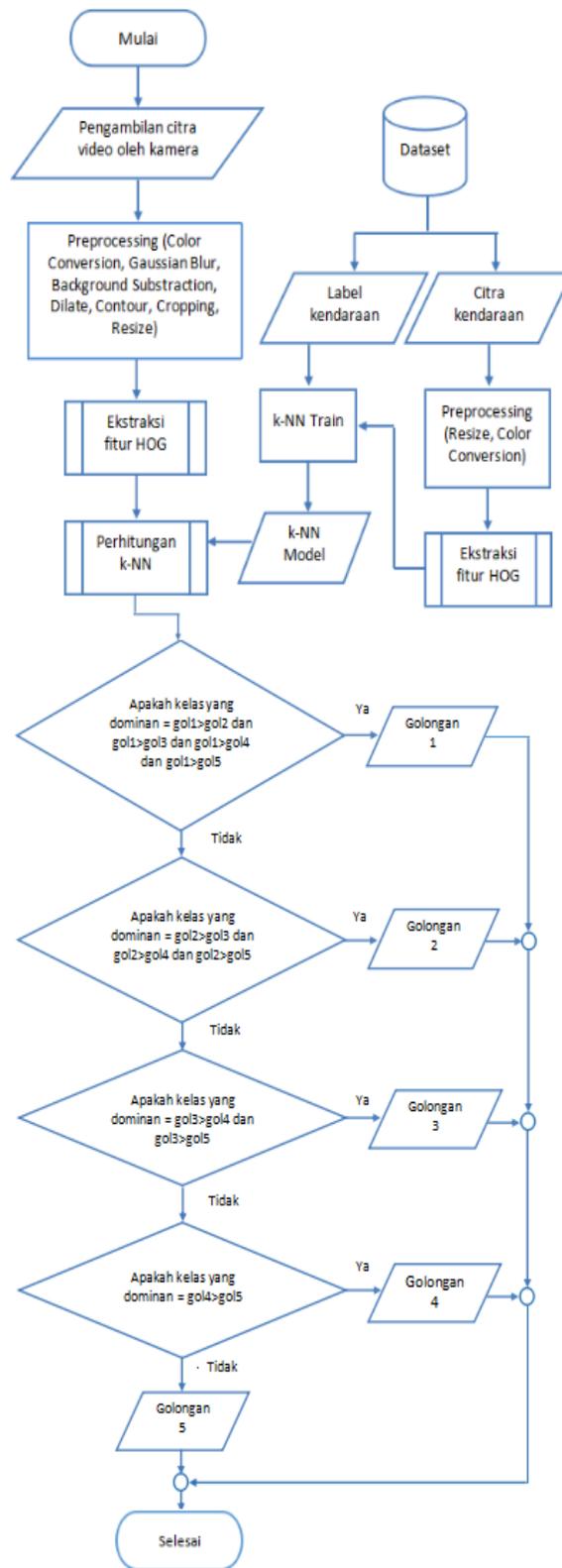
Gambar 2. Diagram blok perangkat lunak

Diagram blok perangkat lunak seperti terlihat pada Gambar 2. Terdapat 4 bagian proses pada program yaitu, *Input* dimana program mengambil masukan berupa citra video kendaraan, *Training phase* dimana program melakukan pembuatan model k-NN berdasarkan dataset yang sudah ditentukan, *Testing phase* dimana dilakukan pemrosesan pada citra video dari masukan kamera untuk dicari nilai fitur HOG untuk kemudian dilakukan proses klasifikasi k-NN untuk menentukan golongan kendaraan berdasarkan model k-NN yang sudah dibuat, dan *Output* dimana klasifikasi yang dihasilkan dari *testing phase* akan ditampilkan pada layar LCD.

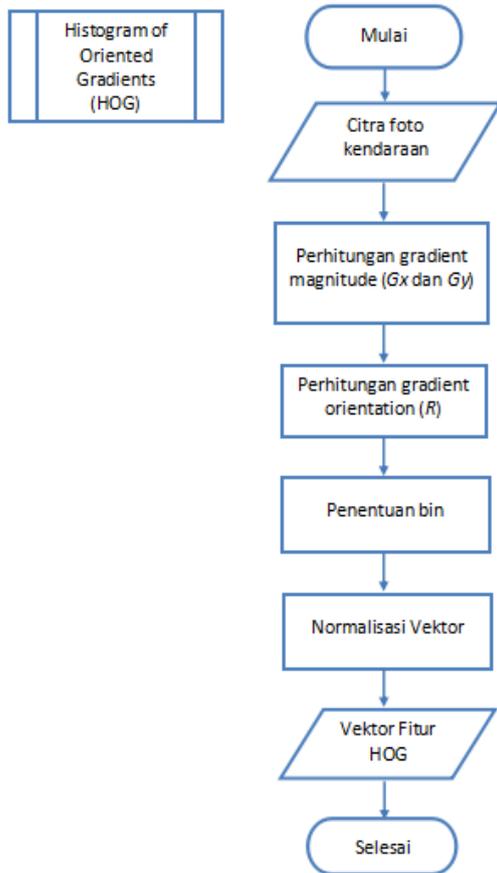
Untuk alur pemrosesan pada perangkat lunak dapat dilihat seperti pada Gambar 3. proses berjalannya sistem berawal pada *training phase*, dimana dilakukan pengambilan citra kendaraan dari *dataset* untuk kemudian dilakukan *preprocessing* seperti *resize*, konversi warna sebelum kemudian dilakukan proses ekstraksi fitur HOG. Selanjutnya nilai fitur HOG dari citra kendaraan dilakukan *training* k-NN *model* dengan pemberian label diambil dari *dataset*.

Tahap selanjutnya masuk pada *testing phase*, dimana dilakukan pengambilan citra video kendaraan sebagai data uji dengan menggunakan kamera, setelah itu dilakukan *preprocessing* citra seperti, konversi warna, *Gaussian blurring*, *background subtraction*, dilasi, pencarian kontur, *cropping*, *resize* sebelum kemudian dilakukan proses ekstraksi fitur HOG. Selanjutnya nilai fitur HOG dari citra kendaraan dilakukan klasifikasi k-NN dengan mencari k tetangga terdekat antara data uji dengan data latih menggunakan perhitungan jarak Euclidean, selanjutnya data yang muncul

dalam k tetangga terdekat menentukan probabilitas prediksi untuk tiap kelas golongan kendaraan. kemudian hasil akhir sistem berupa klasifikasi kelas golongan kendaraan ditampilkan pada layar LCD.



Gambar 3. Diagram alir perangkat lunak



Gambar 4. Diagram alir proses *Histogram of Oriented Gradients*

Pada Gambar 4 merupakan diagram alir proses ekstraksi fitur *Histogram of Oriented Gradients (HOG)*. Pada proses tersebut dibagi menjadi beberapa tahap yaitu:

1. Preprocessing citra

pada tahap ini terdapat proses konversi warna dari RGB menjadi *grayscale*, *cropping* dan *resize*.

2. Perhitungan nilai gradient pada citra gambar.

Menghitung nilai gradient sumbu *x* dan sumbu *y* pada tiap *pixel* citra dengan perkalian kovolusi menggunakan kernel operator sobel seperti pada persamaan (1). Dari penelitian yang sudah dilakukan sebelumnya menunjukkan metode sobel memiliki hasil paling bagus dalam melakukan deteksi tepi citra dengan pengurangan *noise* yang signifikan (Adistya, Muslim, 2016).

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (1)$$

Selanjutnya menghitung nilai *gradient magnitude* dan *gradient orientation* setiap *pixel* pada *cell* dengan menggunakan Persamaan (2) dan (3) pada kedua sumbu *horizontal* maupun *vertical*.

$$G = \sqrt{G_x^2 + G_y^2} \quad (2)$$

Keterangan:

G = Gradient Magnitude

G_x = Matriks operator Sobel sisi *horizontal*

G_y = Matriks operator Sobel sisi *vertical*

$$R0 = \arctan \frac{G_y}{G_x} \quad (3)$$

Keterangan :

R0 = Gradient Orientation

G_x = Matriks operator Sobel tepi *horizontal*

G_y = Matriks operator Sobel tepi *vertical*

3. Proses penentuan bin

Proses penentuan *bin* dilakukan untuk membentuk *histogram* dari fitur HOG, *histogram* terdiri dari 9 *bin* dengan rentang nilai 0° - 160° sehingga tiap *bin* selisih 20°. Proses penentuan bin dilakukan dengan menggunakan persamaan (4).

$$Vote = \frac{\Delta R_i}{H} \times G_i \text{ dan } \frac{1 - \Delta R_i}{H} \times G_i \quad (4)$$

Keterangan:

H = Rentang nilai sudut.

G_i = Nilai Gradient.

ΔR_i = Batas atas gradien - Nilai orientasi gradien

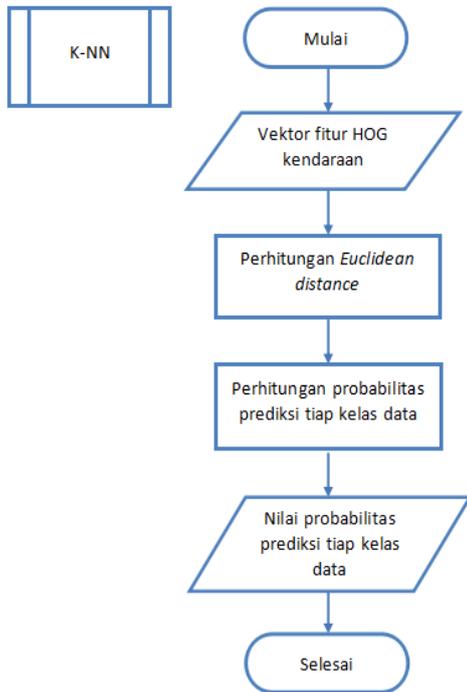
4. Proses normalisasi

Proses normalisasi berguna agar gambar tidak terpengaruh oleh variasi pencahayaan, pada tahap ini dilakukakan penggabungan *histogram* untuk seluruh sel dalam satu blok. Hasil penggabungan tersebut akan dinormalisasi dengan menggunakan Persamaan (5).

$$\|x\|_2 := \sqrt{x_1^2 + \dots + x_n^2} \quad (5)$$

Keterangan:

x = fitur *histogram* vektor



Gambar 5. Diagram alir proses k-NN

Pada Gambar 5. Ditunjukkan diagram alir dari proses *k-nearest neighbors*. Pada proses *k-nearest neighbors* vektor fitur HOG dilakukan perhitungan *Euclidean distance* menggunakan Persamaan (6) untuk mengetahui jarak terdekat antara data uji dengan data latih.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (6)$$

Keterangan:

d = jarak *Euclidean*.

x = nilai sumbu x .

y = nilai sumbu y .

Selanjutnya dari jarak *Euclidean* antara data uji dengan data latih diambil sebanyak k data yang memiliki jarak *Euclidean* terdekat dengan data uji. Kemudian kelas dari data yang paling dominan dijadikan sebagai prediksi kelas untuk data uji.

2.2 Implementasi Sistem

Dalam implementasinya sistem ini menggunakan tiga komponen perangkat keras, yaitu kamera Logitech C270, Raspberry Pi 3, dan layar LCD 3,5 inch. Pemasangan kamera pada port USB Raspberry Pi 3 seperti pada Gambar 6, untuk layar LCD dipasang pada *interface* GPIO pin 1 - 26 Raspberry Pi 3 seperti pada Gambar 7, dan untuk sumber daya yang

dibutuhkan Raspberry Pi 3 yaitu tegangan dan arus sebesar 5V 2A yang masuk pada *port micro* USB seperti pada Gambar 8.

Pada implementasi sistem terdapat kebutuhan yang harus dipenuhi yaitu, posisi kamera saat sistem berjalan harus dalam kondisi tetap tidak berubah posisi, lingkungan sekitar pada objek/kendaraan yang akan dilakukan klasifikasi harus memiliki pencahayaan yang cukup dan stabil/tidak berubah-ubah, dan badan kendaraan yang akan diklasifikasi harus dapat tertangkap kamera secara keseluruhan/berjarak sekitar 15 meter dari kamera.



Gambar 6. Pemasangan kamera pada *port* USB Raspberry Pi 3



Gambar 7. Pemasangan LCD pada GPIO *pin* 1 - 26 Raspberry Pi 3



Gambar 8. Sumber daya Raspberry Pi 3 masuk melalui *port micro* USB

3. PENGUJIAN DAN ANALISIS

3.1 Pengujian Nilai *k* Paling Optimal

Tabel 1. Hasil pegujian nilai *k*

Nilai <i>k</i>	Hasil akurasi (%)
1	92
3	80
5	84
7	92
9	88
11	76
13	76
15	84

Pengujian dilakukan dengan data uji sebanyak 25 data, dengan menggunakan data latih berjumlah 200 data, kemudian dilakukan perhitungan akurasi tiap nilai $k = 1, k = 3, k = 5, k = 7,$ dan $k = 9, k = 11, k = 13, k = 15$. Kemudian nilai akurasi yang didapatkan tiap nilai *k* dicari yang memiliki akurasi paling tinggi. Hasil pengujian sistem dalam melakukan klasifikasi dengan nilai *k* berbeda dapat dilihat pada Tabel 1.

Dari hasil yang didapat dari pengujian, nilai $k = 1$ dan $k = 7$ memiliki akurasi paling tinggi memiliki rata-rata akurasi yang paling baik, namun dipilih nilai $k = 1$ karena komputasi yang dilakukan untuk menghitung 1 tetangga terdekat lebih ringan daripada menghitung 7 tetangga terdekat dalam melakukan klasifikasi golongan kendaraan.

3.2 Pengujian Sistem Dalam Melakukan Klasifikasi Golongan Kendaraan

Pengujian dilakukan dengan memposisikan kamera pada sisi jalan tol dengan jarak sekitar 15 meter dari kendaraan, kemudian program dijalankan untuk melakukan klasifikasi dengan mengambil masukan berupa video kendaraan yang lewat menggunakan kamera *webcam*. Hasil dari klasifikasi golongan kendaraan selanjutnya akan ditampilkan pada layar LCD. Hasil pengujian sistem yang didapatkan seperti pada Tabel 2. Dari hasil pengujian, sistem sudah dapat mengeluarkan hasil berupa klasifikasi golongan kendaraan, menunjukkan sistem sudah berjalan sesuai yang diharapkan.

Tabel 2. Hasil pegujian sistem dalam mengklasifikasi golongan kendaraan

No.	Nilai fitur HOG	Nilai kontribusi tetangga terdekat tiap kelas	Hasil klasifikasi
1	0.006312, 0.002621, 0.002966, 0.006695, 0.013420, 0.006645, 0.005200, 0.002665, 0.001094, ... 0.010210	Golongan 1 = 1 Golongan 2 = 0 Golongan 3 = 0 Golongan 4 = 0 Golongan 5 = 0	Golongan 1
2	0.001732, 0.005307, 0.006161, 0.008654, 0.004284, 0.000375, 0.002909, 0.006608, 0.001531, ... 0.000862	Golongan 1 = 0 Golongan 2 = 1 Golongan 3 = 0 Golongan 4 = 0 Golongan 5 = 0	Golongan 2
3	0.012219, 0.006619, 0.123790, 0.010647, 0.016836, 0.005058, 0.001763, 0.098049, 0.054057, ... 0.032019	Golongan 1 = 0 Golongan 2 = 0 Golongan 3 = 1 Golongan 4 = 0 Golongan 5 = 0	Golongan 3
4	0.041877, 0.333808, 0.033485, 0.026387, 0.048365, 0.058765, 0.004349, 0.000738, 0.002261, ... 0.002624	Golongan 1 = 0 Golongan 2 = 0 Golongan 3 = 0 Golongan 4 = 1 Golongan 5 = 0	Golongan 4

Tabel 2. Hasil pegujian sistem dalam mengklasifikasi golongan kendaraan (lanjutan)

No.	Nilai fitur HOG	Nilai kontribusi tetangga terdekat tiap kelas	Hasil klasifikasi
5	0.003686, 0.001825, 0.000160, 0.001239, 0.002815, 0.025561, 0.006251, 0.007659, 0.007604, ... 0.047790	Golongan 1 = 0 Golongan 2 = 0 Golongan 3 = 0 Golongan 4 = 0 Golongan 5 = 1	Golongan 5

3.1 Pengujian Akurasi Klasifikasi Sistem

Tabel 3. Tingkat akurasi sistem

No	Kondisi Aktual Golongan Kendaraan	Hasil Klasifikasi Sistem	Kesesuaian	Akurasi
1	Golongan 1	Golongan 1	Sesuai	80%
2	Golongan 1	Golongan 1	Sesuai	
3	Golongan 1	Golongan 1	Sesuai	
4	Golongan 1	Golongan 1	Sesuai	
5	Golongan 1	Golongan 3	Tidak Sesuai	
6	Golongan 2	Golongan 2	Sesuai	80%
7	Golongan 2	Golongan 2	Sesuai	
8	Golongan 2	Golongan 3	Tidak Sesuai	
9	Golongan 2	Golongan 2	Sesuai	
10	Golongan 2	Golongan 2	Sesuai	

11	Golongan 3	Golongan 3	Sesuai	60%
12	Golongan 3	Golongan 3	Sesuai	
13	Golongan 3	Golongan 2	Tidak Sesuai	
14	Golongan 3	Golongan 3	Sesuai	
15	Golongan 3	Golongan 2	Tidak Sesuai	
16	Golongan 4	Golongan 4	Sesuai	60%
17	Golongan 4	Golongan 4	Sesuai	
18	Golongan 4	Golongan 4	Sesuai	
19	Golongan 4	Golongan 3	Tidak Sesuai	
20	Golongan 4	Golongan 2	Tidak Sesuai	
21	Golongan 5	Golongan 5	Sesuai	60%
22	Golongan 5	Golongan 3	Tidak Sesuai	
23	Golongan 5	Golongan 5	Sesuai	
24	Golongan 5	Golongan 5	Sesuai	
25	Golongan 5	Golongan 3	Tidak Sesuai	
Akurasi Rata-rata Seluruh Klasifikasi Golongan Kendaraan				68%

Pengujian untuk mendapatkan akurasi sistem dilakukan dengan menguji tiap golongan kendaraan sebanyak 5 data uji dengan total 25 data uji untuk 5 golongan kendaraan. Kemudian dicari persentase akurasi tiap golongan dan keseluruhan sistem dengan perhitungan menggunakan Persamaan (7).

$$Akurasi = \frac{Total\ data\ uji - Total\ data\ uji\ tidak\ sesuai}{Total\ data\ uji} * 100\% \quad (7)$$

Pada Tabel 3 ditunjukkan tingkat akurasi pada tiap golongan, dan tingkat akurasi secara

keseluruhan sistem dalam melakukan klasifikasi kendaraan.

4. KESIMPULAN

Berdasarkan hasil yang didapatkan dari beberapa pengujian didapatkan kesimpulan sebagai berikut:

1. Nilai $k = 1$ merupakan nilai paling optimal dalam melakukan klasifikasi golongan kendaraan.
2. Sistem sudah dapat mengeluarkan hasil berupa klasifikasi golongan kendaraan pada layar LCD namun dengan *frame rate* yang masih tergolong rendah, ini dikarenakan proses klasifikasi dilakukan pada tiap *frame* video sehingga menyebabkan proses komputasi yang berat.
3. Tingkat akurasi sistem yang didapatkan dalam melakukan klasifikasi tiap golongan kendaraan yaitu, 80% untuk golongan 1, 80% untuk golongan 2, 60% untuk golongan 3, 60% untuk golongan 4, 60% untuk golongan 5. Sehingga secara keseluruhan sistem memperoleh akurasi sebesar 68% dari perhitungan rata-rata penjumlahan akurasi tiap golongan kendaraan yang didapatkan dari 25 data pengujian, dengan tiap golongan kendaraan sebanyak 5 data pengujian.

5. DAFTAR PUSTAKA

- Adisty, R., Muslim, M. A., 2016. Deteksi dan Klasifikasi Kendaraan menggunakan Algoritma Backpropagation dan Sobel. *Journal of Mechanical Engineering and Mechatronics* [online] Tersedia di: <<http://e-journal.president.ac.id/presunivojs/index.php/JMEM/article/view/94/67>> [Diakses 19 Januari 2019]
- Bhatia, M., Vandana., 2010. Survey of nearest neighbor techniques. *International Journal of Computer Science and Information Security* 8, 1947-5500.
- Kumpan, 2016. Mengenang Mereka yang 'Gugur' di Jalur Brexit. *kumpanNEWS* [online] Tersedia di: <<https://kumpan.com/@kumpannews/mengenang-mereka-yang-gugur-di-jalur-brexit>> [Diakses 19 Januari 2019]
- Kachouane, M., Sahki, S., Lakrouf, M., Quadah, N., 2012. HOG Based fast Human Detection. In: Université Saad Dahlab

de Blida, Centre de Développement des Technologies Avancées, 2012 24th International Conference on Microelectronics (ICM). Algiers, Algeria, 16-20 December 2012. IEEE

Kusrini dan Lutfhi, E.T., 2009. Algoritma Data Mining, Andi Publishing, Yogyakarta.