

# FACO (FIRST AID COPTER) AUTONOMOUS AEROROBOT FOR BRING MEDICINE OR FOOD TO VICTIMS CATASTROPHE WITH PEOPLE DETECTION CAMERA AND STREAMING OVER IP

Agil Setiawan<sup>\*)</sup>, Yusfi Florianto<sup>\*)</sup>, M. Fadhil Abdullah<sup>\*)</sup>  
<sup>\*)</sup>Telkom University  
 e-mail: agil.setiawan03@gmail.com

## Abstract

The problem from slow searching and handling for victims catastrophe still can't be resolved until now, there are many cause from this problems like access for go to the place of catastrophe is sealed or there are many congestion if using landline, because this problems there are many victims of catastrophe cannot be saved. One Of Solutions from this problem is using airline to save victims, but the cost if using plane or helicopter is too big. So the solution is using aerorobot like UAS (Unmanned Aircraft System). UAS (Unmanned Aircraft System) is an aircraft without a human pilot on board and usually using for military purposes. UAS will be prepare for search and handle the victims catastrophe we call it FACO (FIRST AID COPTER). FACO has payload so it can bring medicine or food. FACO using flight controller with stability system and GPS (Global Positioning System) so it can fly autonomous in its destination point, with give input coordinat in its GUI (Graphical User Interface). FACO has camera using OPENCV Library and integrated with Raspberry Pi so it can detect object which identify as a people, with this feature Victims Catastrophe will be easy to find. The camera in FACO can be access using IP (Internet Protocol) so all of people in access point of FACO can watch the camera using their gadget with streaming method. With This Solution many victims catastrophe will be faster to be found, so that victims will be handled quickly and with this way the number of deaths from catastrophe could be minimized.

**Key Words :** *Aerorobot, GUI (Graphical User Interface), OPENCV, Raspberry Pi, IP (Internet Protocol)*

## Abstrak

Masalah dari lambatnya pencarian dan penanganan bagi korban bencana alam masih belum terselesaikan sampai saat ini ,hal ini diakibatkan dari beberapa penyebab seperti akses untuk menuju ke tempat bencana tertutup ataupun karena banyaknya kemacetan jika menggunakan jalur darat. Karena masalah ini begitu banyak korban bencana alam yang tak dapat diselamatkan. Salah satu solusi dari masalah ini adalah dengan menggunakan jalur udara , tapi biaya yang akan dikeluarkan jika menggunakan pesawat ataupun helikopter sangat besar. Jadi solusinya adalah menggunakan Aerorobot seperti UAS (*Unmanned Aircraft System*). UAS (*Unmanned Aircraft System*) adalah sebuah pesawat tanpa awak yang banyak digunakan untuk keperluan militer. UAS yang akan dipersiapkan untuk pencarian dan penangan korban bencana alam kami menyebutnya FACO (*FIRST AID COPTER*). FACO memiliki *payload* yang berfungsi sebagai tempat untuk membawa obat-obatan atau makanan. FACO juga menggunakan kontrol penerbangan dengan sistem stabilitas yang baik serta menggunakan *GPS (Global Positioning System)*. Dengan sistem tersebut maka FACO dapat terbang secara otomatis ke tempat yang diinginkan, dengan cara menginput koordinat pada grafis antar mukanya. FACO memiliki kamera yang menggunakan perpustakaan OpenCV yang terintegrasi pada Raspberry Pi, sehingga dapat mendeteksi objek yang diidentifikasi sebagai orang. Dengan fitur ini maka Korban Bencana alam akan lebih mudah untuk ditemukan. Kamera pada FACO dapat diakses dengan menggunakan IP (*Internet Protocol*) sehingga semua orang yang berada dalam jangkauan akses point dari FACO, dapat melihat hasil dari kamera menggunakan metode *streaming*. Dengan solusi ini maka korban bencana alam akan lebih cepat untuk ditemukan sehingga bisa dengan cepat ditangani, dengan begini maka jumlah kematian akibat bencana alam bisa diminimalisir.

**Kata Kunci :** *Robot Udara, Grafis antar muka, OpenCV, Raspberry Pi, IP (Internet Protocol)*

## 1. Latar Belakang

Penerapan Teknologi kedirgantaraan yang paling banyak dikembangkan dalam beberapa tahun terakhir ini adalah sistem pesawat udara nir awak, atau biasa dikenal dengan *Unmanned Aircraft System* (UAS). Pengembangan UAS ini dulu dikenal dengan *Unmanned Aerial Vehicle* (UAV). Perkembangan IPTEK yang begitu pesat juga menuntut agar UAS dapat bergerak secara otonom. Dengan adanya sistem otomatisasi ini, UAS dapat menentukan sendiri rute gerakannya.

Namun Sangat Disayangkan Pemanfaatan dari UAS biasanya hanya digunakan sebagai keperluan militer ataupun untuk keperluan pribadi misalnya fotografi melalui udara. Padahal jika kita teliti lebih dalam lagi begitu banyak hal yang bisa kita lakukan untuk memanfaatkan UAS ini, salah satunya adalah menggunakannya sebagai alat bantu untuk mencari korban pada bencana alam. Apalagi melihat bahwa tim pencarian korban bencana alam sering kali mengalami masalah apabila menghadapi medan yang berat dan susah untuk dijangkau sehingga membuat pencarian korban bencana alam jadi terhambat. Akibatnya korban bencana alam tidak dapat ditangani dengan cepat sehingga berdampak pada meningkatnya jumlah kematian.

Oleh karena itu maka kami berinisiatif untuk membuat sebuah UAS yang diberi nama dengan “FACO (*FIRST AID COPTER*)”. FACO akan dilengkapi dengan sistem navigasi yang berfungsi untuk melakukan kontrol terhadap sikap dan perpindahannya dari satu tempat ke tempat yang lain. FACO juga akan dilengkapi dengan kamera yang berfitur *system people detection* yang diintegrasikan pada Raspberry Pi. Selain itu juga kamera pada FACO bisa diakses dengan metode streaming melalui *IP (internet protocol)* sehingga semua orang yang berada pada jangkauan akses point FACO bisa melihat hasil dari kamera pada FACO. FACO juga akan dilengkapi dengan *payload* yang berguna sebagai tempat untuk membawa makanan ataupun obat-obatan. Semoga dengan adanya FACO maka jumlah kematian akibat bencana alam dapat diminimalisir.

## 2. Perumusan Masalah

Rumusan masalah yang akan dibahas pada makalah ini yaitu bagaimana cara membuat sebuah sistem navigasi dan sistem kontrol yang baik pada FACO, bagaimana sistem kerja dari kamera yang terintegrasi dengan Raspberry PI sehingga dapat melakukan sistem deteksi orang yang dapat diakses dengan metode *streaming* melalui IP, serta bagaimana cara kerja dari pembawa muatan FACO.

## 3. Tujuan

Tujuan dari perancangan makalah ini yaitu untuk membuat *aerorobot* FACO yang memiliki sistem kontrol yang baik dan dapat melakukan gerak *otonom* mengikuti titik yang telah ditentukan, kemudian membuat *aerorobot* FACO yang memiliki kemampuan pada kameranya yang dapat melakukan sistem deteksi orang yang dapat diakses dengan metode *streaming* melalui IP, serta membuat *aerorobot* FACO yang dapat membawa muatan berupa obat-obatan ataupun makanan.

## 4. Metodologi

### 4.1 Pembuatan Mekanik dan Sistem Terbang dari FACO

Dalam tahap ini membutuhkan rangka yang kokoh, bukan cuman itu saja dalam pembuatan mekanik dari FACO sendiri harus efektif dan efisien, yang dimaksud dengan efektif dan efisien ialah memperhatikan dari segi berat FACO karena merupakan benda terbang yang menggunakan sumber tenaga berupa *LiPo Battery (Lithium Polimer)* sehingga hal ini berdampak pada lama waktu terbang dari FACO itu sendiri. Dalam pembuatan FACO kami memilih model *quadcopter* sebagai rangkanya dikarenakan dari segi efektif dan efisien merupakan yang paling efektif diantara yang lainnya. Dalam pembuatan mekanik kami menggunakan software design Autocad kemudian diimplementasikan ke Corel

Draw agar bisa dicetak pada PCB. Adapun PCB yang kami gunakan disini terbuat dari bahan FR 4. Kemudian setelah mendesain kami membuat sistem dari FACO. FACO menggunakan *Flight Controller* (Kontrol Penerbangan) yang dilengkapi dengan system *GPS (Global Positioning System)*. Dalam hal ini *Flight Controller* yang digunakan yaitu *Arduflyer* yang dilengkapi dengan GPS Ublox CN-06. Nantinya FACO akan terbang secara otomatis menuju titik yang telah ditentukan pada GUI (Grafis Antar Muka). Setelah mencapai titik koordinat maka *GPS System* akan melakukan *holding Position* dan *Altitude* sehingga FACO tidak terbawa oleh angin, kemudian kamera akan mulai melakukan pemantauan secara *realtime* pada *ground station*, *System People detection* dan *Streaming Over IP* akan berfungsi ketika kamera telah melakukan pemantauan. Setelah kamera mendeteksi objek yang diidentifikasi sebagai manusia maka *Quadcopter* akan dikontrol menuju ke titik tersebut, kemudian FACO akan menjatuhkan *payload* yang berisi obat-obatan atau makanan dan GPS akan mengirimkan koordinat dari FACO sehingga memudahkan untuk menemukan objek yang diidentifikasi sebagai manusia

Tabel 4.1 Spesifikasi alat yang digunakan

NAMA ALAT	SPESIFIKASI
<b>SISTEM SENSOR</b>	
ArduFlyer	Sebagai Flight Controller dari FACO
GPS U Blox CN-06	Sebagai GPS dari FACO
433Mhz Radio Telemetry Kit	Sebagai Telemetry FACO dan Ground Station
Logitech HD Pro Webcam C920	Sebagai Kamera Dari FACO
Raspberry PI	Sebagai Alat untuk melakukan Image processing dan Streaming Over IP
EDIMAX 712 hpn	Sebagai Akses Point Raspberry PI
<b>AIR FRAME FACO</b>	
Aluminium	Kerangka Lengan dari FACO
FR 4	Tempat Flight Controller dari FACO
<b>SISTEM ELEKTRONIK FACO</b>	
SunnySky X2212-13 Brushless Motor KV980	Sebagai Motor Penggerak FACO
TURNIGY Plush 30amp Speed Controller	Sebagai Pengatur Kecepatan motor
Futaba 8JA 8-Channel 2.4GHz S-FHSS Transmitter	Sebagai Remote Control dari FACO
RMRC 5100mAh 3S 35C Lipo Pack	Sebagai Battery dari FACO

#### 4.2 Pembuatan System People detection dan Streaming Over IP

Pada System People detection FACO menggunakan Histogram of Oriented Gradients (HOG). *Histogram of Oriented Gradients* (HOG) adalah salah satu jenis algoritma dalam *Computer Vision* untuk mendeteksi suatu objek. Algoritma ini termasuk dalam *feature detection*, bersama dengan algoritma

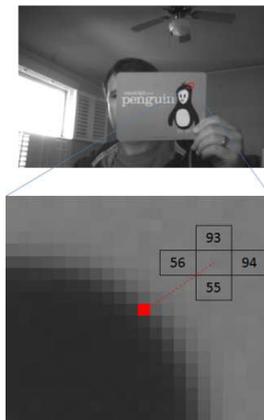
deteksi objek lainnya seperti SURF, SIFT, dan GLOH. HOG lebih dikenal untuk mendeteksi orang, karena pengembangan awalnya dikhususkan pada deteksi pejalan kaki oleh Navneet Dalal dan Bill Trigs pada tahun 2005[1]. Prinsip kerja HOG ialah menghitung nilai dan arah (vektor) gradien dalam daerah tertentu, sehingga akan tampak karakteristik distribusi gradien pada suatu gambar. Karakteristik distribusi gradien inilah yang akan menggambarkan bagaimana bentuk suatu objek pada gambar, lalu akan ditentukan seberapa dekat kemiripan bentuk dari objek pada gambar tersebut terhadap objek yang ingin dideteksi.



Gambar 4-1. Diagram blok metode HOG

### 4.3 Histogram Vektor Gradien

Vektor gradien dapat diukur pada setiap pixelnya. Cara pengukurannya ialah dengan menghitung selisih nilai pixel setelah dan sebelumnya dalam arah sumbu-x dan arah sumbu-y dalam kartesian.



Gambar 4-2. Ilustrasi nilai suatu pixel pada gambar *grayscale*.

(<http://chrisjmccormick.files.wordpress.com/2013/05/brightness50.png?w=470>)

Sehingga akan didapatkan:

$$\begin{bmatrix} 38 \\ 38 \end{bmatrix}$$

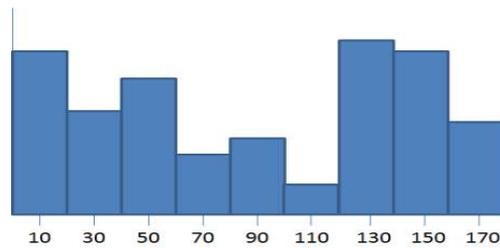
Selain itu, dapat diukur pula besar dan arah (dalam derajat) dari suatu vektor gradien dengan menggunakan formula[3]:

$$\text{Magnitude} = \sqrt{(38)^2 + (38)^2} = 53.74$$

$$\begin{aligned} \text{Angle} &= \arctan\left(\frac{38}{38}\right) = 0.785 \text{ rads} \\ &= 45 \text{ degrees} \end{aligned}$$

Implementasi dari HOG dapat diperoleh dengan membagi gambar kedalam area kecil yang masih saling terhubung atau biasa disebut *cell*[3]. Pada sebuah *cell*, kita menghitung vektor gradien dari masing-

masing pixel. Setiap vektor gradien memiliki kontribusinya pada suatu histogram dengan memberi nilai *magnitude* dari vektor gradien tersebut.

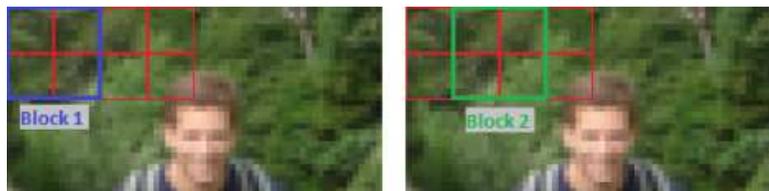


Gambar 4-3. Histogram suatu gambar dengan parameter besar vektor gradien (sumbu-y) dan besar derajat vektor gradien (sumbu-x) yang menggambarkan distribusi vektor gradien dari suatu gambar.

(<http://chrisjmccormick.files.wordpress.com/2013/05/histogram.png?w=470&h=332>)

#### 4.4 Normalisasi Blok

Normalisasi blok digunakan untuk menormalisasikan seluruh histogram yang ada pada suatu blok. Ini lebih baik daripada harus menormalisasi setiap histogram di seluruh *cell*. Cara penormalisasiannya sama dengan normalisasi histogram vektor gradien, hanya saja ukuran normalisasi blok lebih luas dan efektif[3]. Sebagai ilustrasi dari blok, lihat gambar dibawah:

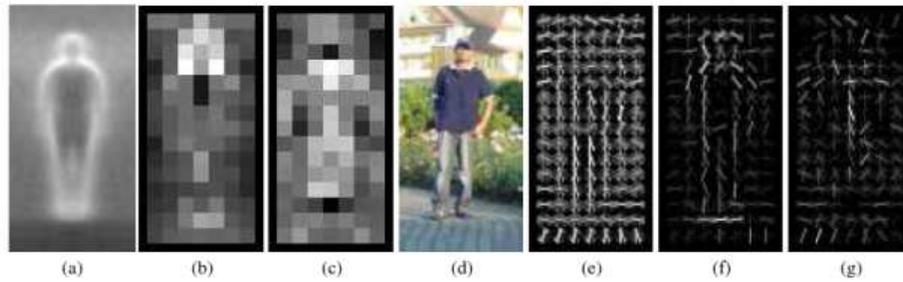


Gambar 4-4. Ilustrasi pembagian blok pada suatu gambar.

(<http://chrisjmccormick.files.wordpress.com/2013/05/blocks.png?w=470&h=113>)

#### 4.5 Pengklasifikasian SVM

Ini adalah langkah terakhir dari mendeteksi atau mengenal suatu objek menggunakan *Histogram of Oriented Gradien* yang harus dilakukan oleh sistem pendeteksi. SVM ialah *Support Vector Machine* yang digunakan untuk menemukan garis pemisah (*hyperplane*) dari *class +1* yang berisi *feature vector positif* dan *class -1* yang berisi *feature vector negatif*. Ada dua tahapan pada learning SVM, yaitu training SVM untuk menghasilkan model, dan *classify* SVM untuk menguji model. Sebagai bahan untuk pengujian, maka dibutuhkan himpunan data dari objek yang akan dideteksi, dalam hal ini orang (pejalan kaki). Pada pengujian yang dilakukan oleh Dalal dan Triggs, himpunan data diambil dari basis data pejalan kaki milik MIT. Metode pengujiannya ialah dengan menggunakan gambar yang ukurannya sama, yakni 128x64 [1,2]. Kecuali untuk proses deteksi, yakni bebas.



Gambar 4-5.

- (a) *average gradient image over training examples*
- (b) *each "pixel" shows max positive SVM weight in the block centered on that pixel*
- (c) *same as (b) for negative SVM weights*
- (d) *test image*
- (e) *its R-HOG descriptor*
- (f) *R-HOG descriptor weighted by positive SVM weights*
- (g) *R-HOG descriptor weighted by negative SVM weights*

#### 4.6 Streaming Over IP

Dalam *Streaming Over IP Software* yang digunakan yaitu *MJPEG-Streamer* yang sudah terintegrasi pada Raspberry PI. Dalam hal ini Raspberry PI akan dijadikan akses *point* sehingga bisa diakses dengan orang yang berada dalam jangkauan akses *point* tersebut. Kanal transmisi yang digunakan yaitu 802.11n[4]. Kemudian Raspberry PI akan terintegrasi dengan *library* OPENCV yang dilengkapi dengan sistem HOG sehingga bisa melakukan pendeteksian. Modul yang digunakan sebagai akses poin dari Raspberry PI yaitu EDIMAX 712 hpn dengan *data rate* hingga 300 mbps.

### 5. Hasil dan Pembahasan

#### 5.1 Pengukuran Jarak telemetri FACO

Pada pengukuran yang dilakukan pada Jarak maksimal  $\pm 300$  meter dengan ketinggian  $\pm 20$  meter telemetri masih berjalan dengan baik.

Tabel 5.1. Pengukuran jarak telemetri FACO

JARAK	KETERANGAN
$\pm 100$ meter	OK
$\pm 200$ meter	OK
$\pm 300$ meter	OK

#### 5.2 Pengukuran Jarak Video Streaming dan Akses Point

Pada pengukuran jarak video streaming dan akses *point* yang dilakukan pada jarak  $\pm 300$  meter sudah tidak didapatkan sinyal dari akses *point* yang ditransmisikan oleh Raspberry PI.

Tabel 5.2. Pengukuran Jarak Video Streaming dan Akses Point

JARAK	KETERANGAN
± 100 meter	Gambar bagus dan sinyal akses point full
± 200 meter	Gambar bagus dan sinyal akses point 1 bar
± 300 meter	Gambar putus dan tidak ada sinyal akses point

### 5.3 Pengukuran Ketinggian dan delay dari System People Detection

Pada pengukuran seberapa tinggi FACO dapat mendeteksi objek yang diidentifikasi sebagai manusia, kondisi terbaik ada pada ketinggian ± 10 meter dengan *delay* yang hanya 3 – 5 detik dan ketika melewati ketinggian diatas 40 meter *delay*-nya pun semakin besar yaitu > 30 detik serta FACO sangat susah untuk mendeteksi objek yang diidentifikasi sebagai manusia.

Tabel 5.3. Pengukuran Ketinggian dan delay dari System People Detection

KETINGGIAN	DELAY	KETERANGAN
± 10 meter	3 Detik – 5 Detik	Bagus
± 20 meter	5 Detik – 15 Detik	Bagus
± 30 meter	15 – 30 Detik	Jelek
± 40 meter	> 30 detik	Sangat Jelek

### 5.4 Hasil Pengujian Streaming Over IP Address

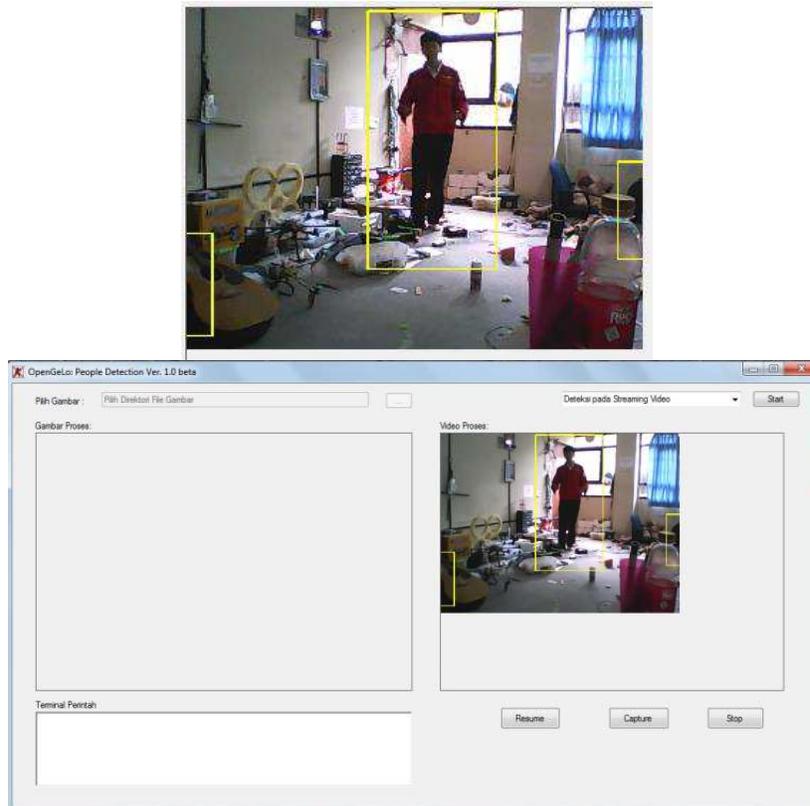
Uji streaming video dengan komunikasi via IP di Laptop Dell dan Tablet Samsung dari Raspberry Pi B sebagai server utamanya. Untuk melakukan uji ini dibutuhkan *Wifi Adapter* dan *IP address server*, selanjutnya hanya butuh pengkoneksian kepada alamat IP dari Raspberry Pi. Hasil yang ditampilkan sangat baik sesuai data atau gambar yang diambil kamera. Namun, terdapat *delay* pada pengiriman video melalui IP.



Gambar 5-1. Streaming over IP di Tablet dan Laptop dari R Pi.

### 5.5 Hasil Pengujian Deteksi Orang atau Pejalan Kaki dengan Komunikasi IP Pada GUI

Hasil deteksi orang menggunakan raspberry pi sangatlah lambat. Delay penerimaan gambar dari kamera saja dapat memakan waktu sekitar 3 detik. Serta masih terdapat kesalahan deteksi yang terjadi dalam menggunakan algoritma HOG pada OpenCV.



Gambar 5-2. dan 5-3. Hasil deteksi orang dengan algoritma HOG menggunakan OpenCV Diterima di GUI C# via IP address.

## 6. Kesimpulan

Berdasarkan pada percobaan diatas, didapatkan kesimpulan sebagai berikut yaitu Jarak *Telemetry* FACO bergantung pada *transceiver telemetry* yang digunakan sehingga jika diinginkan jarak yang jauh perlu ditambahkan *booster* dan antena yang dapat menambah jarak dari proses *telemetry*. Kemudian pada proses pendeteksian objek yang diidentifikasi sebagai manusia dengan menggunakan metode HOG dengan *library* OPENCV memiliki *delay* yang cukup besar sehingga mengganggu proses pendeteksian. Selain itu juga masih terdapat kesalahan pada deteksi orang menggunakan algoritma *Histogram of Oriented Descriptor* (HOG), Sehingga diperlukan pengklasifikasian orang atau pejalan kaki yang lebih detail. Pada metode *Streaming Over IP* dengan menggunakan Raspberry PI memiliki kelemahan pada jarak akses point yang dapat dijangkau yang bergantung pada jenis *router* akses *point* yang digunakan, selain itu juga pada metode *steaming Over IP* memiliki *delay* yang cukup besar dikarenakan paket yang dikirim pada kanal transmisi 802.11n itu dikirim secara utuh dan langsung.

## 7. Saran

Berikut adalah saran yang diberikan pada percobaan di atas yaitu, sebaiknya menggunakan *Booster* serta antena yang dapat memaksimalkan jarak pencarian korban bencana alam. Kemudian sebaiknya mencari algoritma pendeteksian yang lebih handal dibandingkan metode HOG, seperti *particle blur*. Sebaiknya dalam metode *streaming* menggunakan metode UDP dikarenakan pada metode ini paket dikirim secara terus menerus sehingga dapat mengurangi *delay*. Kemudian kamera yang digunakan dalam

pendeteksian harus sebgus mungkin dengan pixel yang besar. Untuk mempercepat pemrosesan data deteksi menggunakan algoritma HOG, sebaiknya digunakan GPU (*Graphical Processing Unit*) sehingga, pemrosesan data menjadi 8x lipat lebih cepat dari sebelumnya.

## 8. Daftar Rujukan

- Navneet Dalal, and Bill Triggs. 2005. *Histograms of Oriented Gradients for Human Detection*. INRIA Rhone-Alps, 655 avenue de l'Europe, Mountbonnot 38334. France.
- C. Permata, I. Ketut E.P., dan Muhtadin, 2010. *Deteksi Mobil Menggunakan Histogram of Oriented Gradient*. Institut Teknologi Surabaya, Surabaya,
- JM, Chris. Mei, 2013. *Histogram Tutorial*. Diperoleh 4 Maret 2014 dari (<http://chrisjmccormick.files.wordpress.com/2013/05/histogram.png?w=470&h=332>)
- Ramadhoni, Moehammad. 2014. *ANALYSIS 802.11n TRANSMISSION CHANNEL AS COMMUNICATION MEDIA BETWEEN UAV- GROUND STATION* : Telkom University.
- Forum Raspberry Pi. Mei, 2013. *Streaming using Raspberry Pi*. Diperoleh 7 Maret 2014 dari (<http://www.raspberrypi.org/phpBB3/viewtopic.php?f=43&t=55371>)
- Forum Linux Circle. 2 Juni, 2013. *Faster video streaming on raspberry media server with mjpg streamer*. Diperoleh 7 Maret 2014 dari (<http://www.linuxcircle.com/2013/02/06/faster-video-streaming-on-raspberry-media-server-with-mjpg-streamer>).