

PERANCANGAN DAN IMPLEMENTASI SISTEM PENDETEKSI MANUSIA DALAM PENCARIAN KORBAN BENCANA ALAM MENGGUNAKAN DRONE BERBASIS MIKRO KOMPUTER

DESIGN AND IMPLEMENTATION OF HUMAN DETECTION SYSTEM IN SEARCH OF NATURAL DISASTER VICTIMS USING DRONE BASED ON MICRO COMPUTER

Ahmad Devi Gunawan^[1], Rumani M^[2], Casi Setianingsih^[3]

Prodi S1 Sistem Komputer

Fakultas Teknik Elektro, Universitas Telkom

Jl. Telekomunikasi Terusan Buah Batu 40257, Indonesia

^[1]adevgunawan@student.telkomuniversity.ac.id, ^[2]rumani@telkomuniversity.ac.id,

^[3]setiacasie@telkomuniversity.ac.id

Abstrak

Pengolahan citra menjadi sangat penting dalam perkembangan teknologi, pengolahan citra banyak diaplikasikan pada berbagai kasus dalam kehidupan manusia yang dapat membantu mengefisienkan pekerjaan manusia. Penelitian ini mengimplementasikan pengolahan citra untuk mendeteksi korban bencana alam yang akan memudahkan dalam proses pengevuasian korban bencana alam, proses pendeteksian ini menggunakan cabang ilmu komputer yaitu *Computer Vision* pada bidang *Object Detetction*. Objek yang di deteksi adalah pola keseluruhan tubuh manusia dengan pencahayaan yang terang yang memiliki perbedaan warna dengan objek disekitarnya, dalam pelaksanaannya penulis menggunakan metode *Histogram of Oriented Gradient* (HOG) sebagai metode yang akan digunakan untuk mendeteksi objek dan metode *Support Vector Machine* (SVM) Selain hasil deteksi sistem juga dapat mengirim data *GPS* ke server yang diasumsikan letak keberadaan korban berada. Parameter yang akan diuji yaitu keakurasian dalam mendeteksi, kecepatan dalam mendeteksi dan jarak deteksi objek yang dapat ditempuh oleh sistem.

Kata kunci : *Object Detection, HOG, SVM, GPS*.

Abstract

Image processing is becoming very important in the development of technology, image processing was applied to many cases in people's lives that can help simplify the work of man. This study implements image processing to detect victims of natural disasters that will make it easier in the process pengevuasian the victims of natural disasters, this detection process using the branch of computer science that is *Computer Vision* on the field of *Detetction Object*. The object is the overall pattern detection in the human body with bright lighting that has a difference in color with surrounding objects, in practice the author uses the method of *Histogram of Oriented Gradient* (HOG) as a method to be used to detect the object and the method of *Support Vector Machine* (SVM) in addition to the results of a detection system can also send the *GPS* data to a server that is assumed to be the location of the whereabouts of the victims are. The parameters to be tested i.e. accuracy in detecting, speed and distance detection in detecting objects that can be reached by the system.

1. Pendahuluan

Bencana alam adalah suatu kejadian yang bisa terjadi kapan saja tanpa terduga. Bencana dapat mengakibatkan kerugian yang besar baik harta, benda bahkan dapat merenggut nyawa. Oleh karena itu diperlukan evakuasi bencana yang cepat dan tepat agar korban bencana alam dapat segera di evakuasi dan diselamatkan, akan tetapi dalam proses evakuasi banyak terjadi kendala, seperti sulitnya akses menuju tempat korban bencana alam, kondisi cuaca yang tidak memungkinkan dan sulitnya penyebaran tim SAR dalam pencarian korban bencana yang diakibatkan tidak ada informasi letak korban bencana dan tingkat kerusakan yang diakibatkan bencana alam.

Dari permasalahan diatas maka pada penelitian ini dibuat sistem untuk mendeteksi korban bencana alam dengan menggunakan pengolahan citra yaitu deteksi objek menggunakan *metode Histogram of Oriented gradient*. Dalam penelitian yang telah dibuat sistem dapat mendeteksi objek berupa manusia dan sistem dapat lokasi objek berada sehingga dalam pencarian korban bencana tim SAR dapat dengan mudah mengevakuasi korban bencana alam.

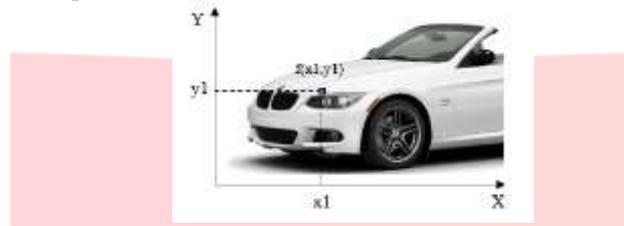
2. Dasar Teori

2.1 Computer vision

Computer Vision merupakan ilmu yang mempelajari, merekonstruksi, menginterpretasikan dan memahami sebuah gambar. *Computer Vision* merupakan kemampuan sebuah komputer yang di desain agar mampu melihat sebuah obyek sehingga mampu menampilkan obyek tersebut dalam bentuk digital dengan tujuan dapat mengetahui informasi informasi dari bentuk digital tersebut(gambar)^[1].

2.2 Image Processing

Pengolahan citra merupakan pemrosesan gambar berdimensi dua melalui komputer^[2]. Pengolahan citra merupakan teknik pengolahan gambar atau citra, citra yang dimaksud yaitu gambar diam (foto) atau gambar bergerak (video) yang bertujuan untuk memperbaiki, mengolah dan merubah sebuah citra. Citra atau gambar merupakan kumpulan dari beberapa pixel, baik citra diam ataupun citra bergerak. Representasi dari beberapa pixel ini dapat didefinisikan dengan fungsi $f(x,y)$ dimana x dan y adalah koordinat spasial dari pixel dengan x adalah garis horizontal dan y adalah garis vertikal. Nilai $f(x,y)$ adalah nilai intensitas citra pada koordinat tersebut, biasanya intensitas berupa informasi citra baik warna atau informasi lain.



Gambar 2.1 Contoh Pengambilan Pixel Citra

Konsep dasar pemrosesan suatu objek pada gambar menggunakan pengolahan citra diambil dari kemampuan indera penglihatan manusia yang selanjutnya dihubungkan dengan kemampuan otak manusia yang diimplementasikan pada sebuah komputer. Contoh operasi pengolahan citra yaitu perubahan kontras citra, penghilangan derau (*noise*) dengan operasi filtering, pengolahan data histogram suatu citra, dan sebagainya

2.3 Histogram of Oriented Gradient

Histograms of Oriented Gradients (HOG) yang merupakan metode dalam *image processing* yang digunakan untuk mendeteksi objek. Metode ini menggunakan teknik menghitung nilai gradien dalam daerah tertentu dalam sebuah citra atau gambar. Tiap gambar memiliki karakteristik yang berbeda yang ditunjukkan oleh distribusi gradien, karakteristik yang ditunjukkan oleh distribusi gradien. Karakteristik ini diperoleh dengan membagi image kedalam daerah kecil yang disebut cell. Tiap cell disusun sebuah histogram dari sebuah gradient. Kombinasi dari histogram ini dijadikan sebagai deskriptor yang mewakili sebuah objek^[5].

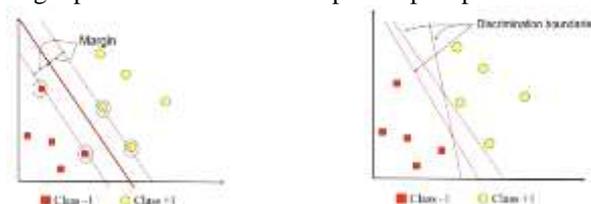


Gambar 2.2 Visualisasi HOG^[5]

2.4 Support Vektor Machine (SVM)

Support Vektor Machine (SVM) adalah metode untuk menemukan fungsi pemisah yang dapat memisahkan 2 obyek data yang berbeda^[3]. Pada penelitian tugas akhir ini SVM digunakan sebagai pemisah obyek manusia dengan obyek yang bukan manusia dan pencocokan obyek yang akan di deteksi.

Konsep Support Vector Machine (SVM) dapat dijelaskan secara sederhana sebagai usaha mencari hyperplane 2 terbaik yang berfungsi sebagai pemisah dua buah class pada input space.



Gambar 2.5 Proses SVM mencari hyperplane dari 2 class objek

2.5 GPS

GPS(Global Positioning System) adalah sistem untuk menentukan letak di permukaan bumi dengan bantuan penyelarasan sinyal satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang mikro ke Bumi. Sinyal ini diterima oleh alat penerima (receiver) dan digunakan untuk menentukan letak, kecepatan, arah, dan waktu.



gambar 2.3 contoh letak longitude latitude^[7]

Gambar diatas merupakan contoh dalam mengetahui lokasi suatu tempat dengan mengetahui letak longitude latitude, gambar diatas adalah letak longitude latitude kampus Universitas Telkom yang dicari di google maps.

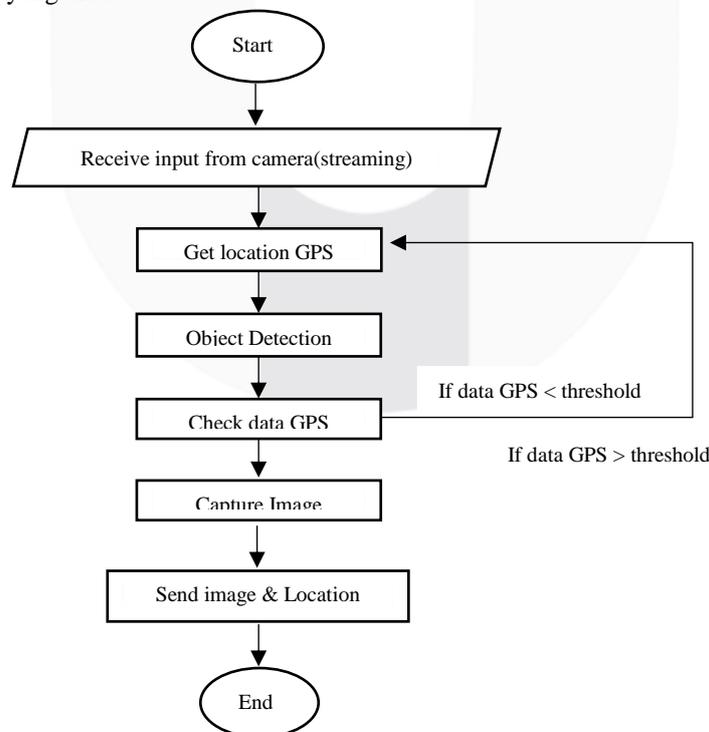
3. Perancangan

3.1 Deskripsi

Dalam penelitian tugas akhir ini, dibangun sebuah aplikasi pengolahan citra untuk mendeteksi manusia (korban bencana alam) yang terintegrasikan dan diimplementasi pada perangkat raspberry pi. Aplikasi ini di dibangun agar dalam pengevakuasian korban bencana alam korban dapat ditangani dengan cepat, korban yang dimaksud adalah manusia baik yang masih hidup atau yang sudah meninggal. Sistem deteksi manusia ini dibangun menggunakan metode *Histogram of Oriented Gradient* dan *support vector machine* hal ini dilakukan agar dapat membantu tim sar dalam mencari korban bencana alam dengan cara mendeteksi keberadaan korban menggunakan kamera dan hasil deteksi akan dikirimkan kepada tim sar untuk ditindak lanjuti lebih jauh, selain gambar yang dikirim letak koordinat lokasi pada korbanpun dikirim dengan mengambil data gps, akan tetapi data yang dikirim bukan letak koordinat asli korban melainkan letak koordinat sistem berada yang akan mempermudah tim sar dalam mengetahui lokasi korban berada, karena letak korban tidak akan jauh dari letak koordinat sistem berada.

3.2 Perancangan Sistem

Berikut alur sistem yang dibuat



Gambar 3.1 Flowchart umum perancangan

Dari *flowchart* umum diatas didapat urutan proses sebagai berikut :

1. Sistem melakukan *scanning* dengan melakukan *streaming* dengan tujuan untuk mendeteksi manusia.

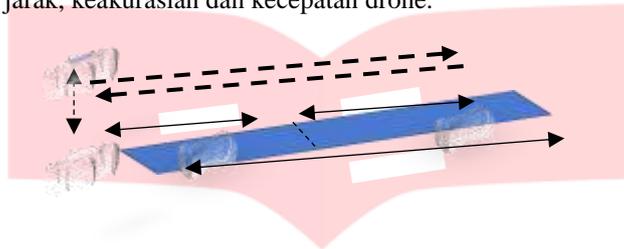
2. Melakukan proses pendeteksian manusia dengan cara memberi ciri pada objek manusia.
3. Setelah mendeteksi sistem akan membaca nilai latitude longitude dari data GPS, ketika data gps masih di *range* yang sama sistem tidak akan melakukan *capture image*.
4. Proses terakhir yaitu mengirim gambar dan lokasi gambar ke server.

4. Analisis

Dalam pengujian tugas akhir ada beberapa parameter pengujian yaitu jarak deteksi, kecepatan sistem bergerak (drone) dan keakurasian deteksi. Dalam melakukan deteksi sistem menyesuaikan letak koordinat latitude longitude, pada pendeteksian pertama sistem akan mengambil data latitude longitude dari gps sedangkan pada pendeteksian berikutnya sistem akan melakukan penyesuaian terhadap data gps yang diambil, jika masih di region yang sama sistem tidak akan melakukan *capture image* walaupun sistem mendeteksi objek. Toleransi jarak yang diberikan yaitu $\pm 7m$ dengan *threshold* latitude 0.000048 dan *threshold* longitude 0.00015. setelah data gps lebih atau kurang dari *threshold* maka sistem akan *capture image* .

4.1 Skenario

Skenario yang dilakukan yaitu sistem ditempel pada drone dan dibawa ke udara untuk melakukan pendeteksian, kemudian di bawah terdapat objek berupa orang yang akan di deteksi. Dalam skenario ini parameter yang diuji yaitu jarak, keakurasian dan kecepatan drone.



Keterangan :



Sistem yang dibawa oleh drone akan bergerak ke udara dengan 1 putaran garis horizontal untu mendeteksi objek. Ketinggian drone di set secara auto sehingga setelah 1 putara ketinggian drone akan berubah.

4.2 Hasil dan Analisa

4.2.1 Analisis Threshold GPS

Pengujian ini dilakukan untuk membuat jarak toleransi sistem dapat mendeteksi objek, karena jika tidak dibatasi oleh data gps makan sistem dapat mengcapture hasil deteksi objek beberapa kali dalam posisi sistem yang sama sehingga akan terjadi penumpukan data, oleh karena itu dibatasi oleh jarak posisi sistem berada. Ketika posisi sistem masih berada di jarak toleransi maka sistem tidak akan melakukan *capture image* walaupun sistem mendeteksi objek, tetapi pada saat sistem keluar dari jarak toleransi maka sistem akan mengcapture image apabila mendeteksi adanya objek. Berikut data hasil pengujian jarak toleransi latitude longitude yang telah dilakukan:

Tabel 4.1 Pengujian *threshold latitude*

| No | Latitude 1 | Latitude 2 | Selisih |
|-----------|--------------|--------------|---------------|
| 1 | -6,970965787 | -6,970908952 | -0,0000568350 |
| 2 | -6,970964507 | -6,970912351 | -0,0000521560 |
| 3 | -6,970961103 | -6,970910949 | -0,0000501540 |
| 4 | -6,970960169 | -6,970906919 | -0,0000532500 |
| 5 | -6,970958143 | -6,970904319 | -0,0000538240 |
| 6 | -6,970950793 | -6,970900929 | -0,0000498640 |
| 7 | -6,970952944 | -6,970906383 | -0,0000465610 |
| 8 | -6,970953159 | -6,970911146 | -0,0000420130 |
| 9 | -6,970951276 | -6,970904646 | -0,0000466300 |
| 10 | -6,970951693 | -6,970902645 | -0,0000490480 |
| Rata-Rata | | | -0,0000500335 |

Tabel 4.2 Pengujian *threshold longitude*

| No | Longitude1 | Longitude2 | Selisih |
|----|---------------|---------------|--------------|
| 1 | 107,636088321 | 107,635966555 | 0,0001217660 |
| 2 | 107,636004324 | 107,635966404 | 0,0000379200 |
| 3 | 107,635999913 | 107,635967759 | 0,0000321540 |
| 4 | 107,636022666 | 107,635967821 | 0,0000548450 |

| | | | |
|-----------|---------------|---------------|--------------|
| 5 | 107,636019132 | 107,635969683 | 0,0000494490 |
| 6 | 107,636080263 | 107,635963949 | 0,0001163140 |
| 7 | 107,636084096 | 107,635945982 | 0,0001381140 |
| 8 | 107,636019132 | 107,635939367 | 0,0000797650 |
| 9 | 107,636070880 | 107,635934892 | 0,0001359880 |
| 101 | 107,636055523 | 107,635919799 | 0,0001357240 |
| Rata-Rata | | | 0,0000902039 |

Letak latitude dan longitude 1 adalah posisi awal sistem berada kemudian posisi berubah dengan jarak 7m ke arah depan dan didapat nilai latitude longitude 2. Dari 10 data yang diambil diatas didapat nilai perubahan latitude sebesar -0,0000500335 dan longitude 0,000902039 dengan nilai jarak 7m. Sehingga pada saat melakukan pendeteksian sistem akan membaca nilai dari toleransi latitude longitude, ketika masih berada pada nilai toleransi tersebut maka sistem tidak akan melakukan capture image dan pada saat keluar dari nilai toleransi tersebut maka sistem akan mengcapture image sesuai perintah awal. Nilai latitude-longitude yang berubah rubah diakibatkan oleh spesifikasi modul gpsnya yang lemah dalam mengambil sinyal dari satelit sehingga perubahan nilainya tidak stabil.

4.2.2 Analisis Kecepatan, Jarak dan Akurasi Deteksi

Skenario yang dilakukan yaitu dengan menggunakan drone dengan langkah-langkah pengujian sebagai berikut :

1. Mengukur jarak deteksi dengan kecepatan drone 50cm/s, drone akan naik ke udara untuk melakukan pendeteksian objek dengan jarak yang diuji yaitu 5m-15m.
2. Masing-masing jarak diuji sebanyak 5 kali untuk mengetahui hasil keakurasian dengan jarak terbaik.
3. Jarak terbaik yang diambil dilakukan pengujian ulang dengan kecepatan drone yang berbeda yaitu 50cm/s, 80cm/s, 100cm/s, 120cm/s, 140cm/s, 160cm/s, 180cm/s, 200cm/s, 220cm/s, 240cm/s, 250cm/s dan 260cm/s.

Berikut data hasil pengujian yang telah dilakukan :

Tabel 4.3 Pengujian jarak deteksi objek dengan kecepatan 50cm/s

| Kecepatan Drone | Jarak ketinggian drone | Jumlah objek | Jumlah objek terdeteksi/5 kali percobaan |
|-----------------|------------------------|--------------|--|
| 50 cm/s | 5m | 5 | 1-2-2-2-2 |
| 50 cm/s | 6m | 5 | 2-3-1-3-2 |
| 50 cm/s | 7m | 5 | 3-2-2-1-2 |
| 50 cm/s | 8m | 5 | 3-2-2-3-2 |
| 50 cm/s | 9m | 5 | 4-3-4-5-4 |
| 50 cm/s | 10m | 5 | 2-3-2-2-2 |
| 50 cm/s | 11m | 5 | 3-2-2-2-2 |
| 50 cm/s | 12m | 5 | 4-2-2-5-3 |
| 50 cm/s | 13m | 5 | 2-2-2-1-2 |
| 50 cm/s | 14m | 5 | 2-2-2-2-1 |
| 50 cm/s | 15m | 5 | 2-1-1-2-2 |
| 50 cm/s | 16m | 5 | 3-1-1-1-1 |
| 50 cm/s | 17m | 5 | Tidak terdeteksi |

Percobaan yang diambil yaitu menggunakan kecepatan drone 50cm/s karena pada kondisi sebenarnya kecepatan 50cm/s sudah pelan sehingga dapat menjadikan sistem akurat dalam mendeteksi korban bencana alam.

Dari data diatas dapat dihitung jarak terakurat dalam mendeteksi objek yaitu dengan menggunakan perhitungan sebagai berikut :

A = Total objek yang ada

B = Total Objek yang terdeteksi

Maka dapat dihitung dengan rumus : $(B/A) \times 100\%$

Keterangan :

Dalam 1x percobaan terdapat 5 objek, percobaan dilakukan sebanyak 5 kali maka $A=5 \times 5=25$. Sedangkan nilai B adalah total objek yang terdeteksi dari 5x percobaan yang telah dilakukan. Dengan menggunakan rumus diatas diperoleh nilai keakurasian sebagai berikut :

Tabel 4.4 Perhitungan keakurasian dengan kecepatan 50cm/s dan jarak 9m

| Jarak Deteksi | Perhitungan keakurasian $(B/A) \times 100\%$ |
|---------------|--|
| 5m | $(9/25) \times 100\% = 36\%$ |
| 6m | $(11/25) \times 100\% = 44\%$ |
| 7m | $(10/25) \times 100\% = 40\%$ |

| Jarak Deteksi | Perhitungan keakurasian (B/A) x100% |
|---------------|--|
| 8m | $(12/25) \times 100\% = 48\%$ |
| 9m | $(20/25) \times 100\% = 80\%$ |
| 10m | $(11/25) \times 100\% = 44\%$ |
| 11m | $(11/25) \times 100\% = 44\%$ |
| 12m | $(16/25) \times 100\% = 64\%$ |
| 13m | $(9/25) \times 100\% = 36\%$ |
| 14m | $(9/25) \times 100\% = 36\%$ |
| 15m | $(8/25) \times 100\% = 32\%$ |
| 16m | $(7/25) \times 100\% = 28\%$ |

Dari hasil pengujian diatas dapat dilihat bahwa terbaik yaitu berada pada ketinggian 9m dengan tingkat akurasi 80%, hal tersebut diakibatkan oleh factor cuaca karena pada saat pengujian terjadi kondisi dimana drone terbawa oleh angin sehingga posisi drone naik turun dan pada ketinggian 9m posisi drone lebih stabil dari ketinggian yang lain , oleh karena itu pada ketinggian tersebut dilakukan pengujian kecepatan drone agar diketahui kecepatan paling baik untuk melakukan deteksi objek. Berikut tabel pengujian kecepatan drone dengan kecepatan 50cm/s, 80cm/s, 100cm/s, 120cm/s, 140cm/s, 160cm/s, 180cm/s, 200cm/s, 220cm/s, 240cm/s, 250cm/s dan 260cm/s.

Tabel 4.5 pengujian Kecepatan pada jarak 9m

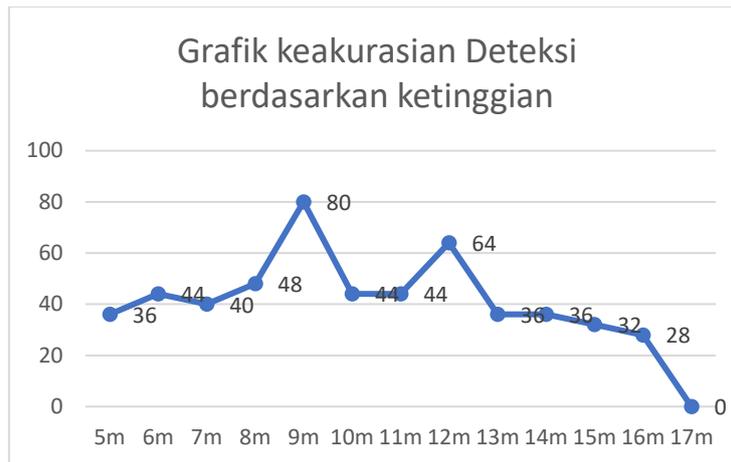
| Kecepatan Drone | Jarak ketinggian drone | Jumlah objek | Jumlah objek terdeteksi/5 kali percobaan |
|-----------------|------------------------|--------------|--|
| 50 cm/s | 9m | 5 | 4-3-4-5-4 |
| 80 cm/s | 9m | 5 | 2-2-2-2-1 |
| 100 cm/s | 9m | 5 | 2-1-2-1-1 |
| 120 cm/s | 9m | 5 | 1-3-1-2-3 |
| 140 cm/s | 9m | 5 | 2-1-3-0-0 |
| 160 cm/s | 9m | 5 | 1-1-3-2-0 |
| 180 cm/s | 9m | 5 | 1-1-2-3-0 |
| 200 cm/s | 9m | 5 | 2-1-0-0-0 |
| 220 cm/s | 9m | 5 | 2-3-3-2-0 |
| 240 cm/s | 9m | 5 | 2-2-2-2-1 |
| 250 cm/s | 9m | 5 | Tidak terdeteksi |
| 260cm/s | 9m | 5 | Tidak terdeteksi |

Tabel 4.6 Perhitungan Keakurasian Deteksi Objek Berdasarkan Kecepatan pada jarak 9m

| Kecepatan Drone | Perhitungan keakurasian (B/A) x100% |
|-----------------|--|
| 50 cm/s | $(20/25) \times 100\% = 80\%$ |
| 80 cm/s | $(9/25) \times 100\% = 36\%$ |
| 100 cm/s | $(7/25) \times 100\% = 28\%$ |
| 120 cm/s | $(10/25) \times 100\% = 40\%$ |
| 140 cm/s | $(6/25) \times 100\% = 24\%$ |
| 160 cm/s | $(7/25) \times 100\% = 28\%$ |
| 180 cm/s | $(7/25) \times 100\% = 28\%$ |
| 200 cm/s | $(3/25) \times 100\% = 12\%$ |
| 220 cm/s | $(10/25) \times 100\% = 40\%$ |
| 240 cm/s | $(9/25) \times 100\% = 36\%$ |
| 250 cm/s | Tidak terdeteksi |
| 260 cm/s | Tidak terdeteksi |

Secara logika seharusnya semakin besar kecepatan yang diberikan maka tingkat akurasi semakin menurun, tetapi pada percobaan yang telah dilakukan tingkat akurasi naik turun yang diakibatkan oleh kestabilan drone saat terbang di udara naik turun hal tersebut dipengaruhi oleh besar kecilnya angin yang terjadi pada saat pengujian.

Dari hasil pengujian yang telah dilakukan dapat ditarik kesimpulan bahwa sistem dapat mendeteksi sejauh 16m dari udara dengan kecepatan terbaik untuk mendeteksi objek yaitu pada kecepatan 50cm/s karena pada kecepatan ini dron bergerak secara perlahan dan stabil. Jarak stabil sistem dapat mendeteksi yaitu diketinggian 9m. Berikut grafik hasil pengujian yang telah dilakukan.



Dari grafik diatas dapat dilihat bahwa jarak terbaik sistem dapat mendeteksi objek yaitu pada ketinggian 9m dengan tingkat keakurasian 80% dan pada kecepatan drone 50cm/s.



Dari grafik diatas dapat dilihat bahwa kecepatan terbaik dengan nilai akurasi tertinggi yaitu pada kecepatan 50cm/s dengan tingkat akurasi 80%.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Setelah dilakukan perancangan, realisasi dan pengujian deteksi objek menggunakan metode histogram of oriented gradient pada tugas akhir ini, dapat diambil beberapa kesimpulan diantaranya :

1. Secara kinerja sistem dapat mendeteksi objek dan mendeteksi letak koordinat latitude longitude serta dapat mengirimkannya ke server.
2. Terdapat beberapa faktor yang dapat menghambat proses pendeteksian yaitu cahaya, kecepatan dore dan posisi objek.
3. Hasil pengujian dengan nilai terbaik yaitu pada kecepatan drone 50cm/s pada jarak 9m dengna tingkat keakurasian 80%.
4. Terjadi perubahan tingkat keakurasian dalam pengujian, hal itu disebabkan oleh faktor cuaca, karena pada saat pengujian tingkat angin di pagi, siang dan sore berbeda.
5. Batas maksimal sistem dapat mendeteksi objek yaitu pada ketinggian 16m.
6. Metode *Histogram of Oriented Gradient* merupakan salah satu metode yang cocok dalam mendeteksi objek berupa manusia, karena pada metode ini terdapat vitur SVM yang di dalamnya telah menyimpan data sheet berupa gradient dari pola manusia, sehingga training yang dilakukan sudah include pada metode HOG dan SVM.

5.2 Saran

Untuk mendapatkan kinerja yang lebih baik lagi pada penelitian selanjutnya ada beberapa pertimbangan yaitu :

1. Pada realisasi nyata letak dan posisi korban bencana alam tidak teratur bisa saja hanya perpotongan tubuhnya saja yang terlihat. Diharapkan pada penelitian selanjutnya objek yang akan di deteksi di perbanyak disesuaikan dengan potongan potongan pola tubuh manusia.
2. Pada proses training dilakukan beberapa posisi karena pada sistem yang dibuat posisi objek harus tegak lurus terhadap posisi kamera.

DAFTAR PUSTAKA

- [1]. Ari Purno Wahyu Wibowo. " Implementasi Teknik Computer Vision Dengan Metode Colored Markers Trajectory Secara Real Time ". Jurnal Teknik Informatika Vol. 8 No. 1, Januari 2016
- [2]. Endina Putri Purwandani. "Peningkatan Kualitas Pembelajaran Pengolahan Citra Digital Pada Program Studi Teknik Informatika Menggunakan Model Project Based Learning ". Teknik Informatika Universitas Bengkulu. Jurnal Rekursif, Vol. 2 No. 1 Maret 2014, ISSN 2303-0755
- [3]. Farizi Rachman, Santi Wulan Purnami.(2012). *Perbandingan Klasifikasi Tingkat Keganasan Breast Cancer Dengan Menggunakan Regresi Logistik Ordinal dan Suort Vector Machine (SVM)*. Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam ITS, Vol. 1, (Sept 2012) ISSN:2301-928X.
- [4]. Gary Bradski, Adrian Kaechler. 2008. "Learning OpenCV". United States of America. O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.
- [5]. Haofu Liao, BSEE. *Human Detection Based on Histograms of Oriented Gradients and SVM*. EECE-7313. April 23, 2013
- [6]. Kadir Abdul, Adhi Susanto. 2013. *Teori dan Aplikasi Pengolahan Citra*. Yogyakarta:ANDI.
- [7]. Letak koordinat Telkom University. Diakses pada 21 Mei 2017 dari (<https://www.google.co.id/maps/search/gps+longitude+latitude/@-6.9677577,107.6255859,15.25z>).
- [8]. Nindhi."Image Processing and Object Detection". Dept. Of Computer Application, NIT Kurukshetra, Haryana, India. *International of Applied Research* 2015:1(9):396-399. ISSN:2394-5869.
- [9]. Permata Cahyo, I Ketut Eddy Purnama, Muhtadin.(2013). *Deteksi Mobil Menggunakan Histogram of Oriented Gradient*. Teknik Elektro ITS.
- [10]. Spesifikasi Raspberry pi 3 model B. [Diakses Pada 15 Mei 2017 dari https://www.raspberrypi.org/products/raspberry-pi-3-model-b/](https://www.raspberrypi.org/products/raspberry-pi-3-model-b/)