BAB IV
PENGUJIAN DAN ANALISA

1.1 Pengujian Perangkat Lunak

Penulis melakukan pengujian pada perangkat lunak sistem dengan mencari persentase akurasi Pelacakan objek pada alat ini.

Pengujian akurasi pelacakan objek dilakukan dengan cara menghimpun per-*frame* gambar dan melihat seberapa banyak objek yang berhasil di deteksi dan membandingkan dengan jumlah objek terdeteksi pada setiap *frame* citra. Terdapat empat konndisi dalam observasi *frame* ini diantaranya:

a. *True Positive*, kondisi *frame* dimana ada objek manusia dan sistem berhasil mendeteksinya.
b. *True Negative*, kondisi *frame* dimana tidak ada objek manusia dan sistem tidak menampilkan bounding box sebagai tanda berhasil mendeteksi objek.
c. *False Positive*, kondisi *frame* dimana ada objek manusia dalam frame namun sistem tidak berhasil mendeteksinya.
d. *False Negative*, kondisi *frame* dimana tidak ada objek manusia dalam frame tetapi sistem menampilkan bounding box sebagai tanda berhasil mendeteksi objek.

Parameter yang digunakan untuk pengujian kesalahan yaitu pada dua kondisi yaitu pada saat kondisi *false positive* dan *false negative*. Untuk kondisi tersebut dapat dilihat pada contoh yang ada pada gambar 4.1.1.

Gambar 1.1.1 Kondisi False Positive dan False Negative pada Frame

Pada gambar 4.1.1 adalah suatu contoh kondisi dimana *frame* tidak berhasil mendeteksi objek yang tertera pada *frame* citra sementara itu sistem mendeteksi dimana pada latar belakang citra tidak ada objek manusia dengan adanya satu *bounding box* yang tampil pada *frame*.

1.1.1 Pengujian Akurasi Deteksi Satu Objek Manusia

Pengujian dilakukan dengan mengambil sampel video berdurasi 31 detik dengan resolusi 320 x 240 p dan jumlah sebanyak 604 *frame*.

Gambar 1.1.2 Frame Pengujian Deteksi Satu Objek Manusia

- Keberhasilan deteksi objek dimana ada objek di dalam *frame* dan sistem berhasil mendeteksinya.

$$\frac{jumlah\ objek\ True\ positive}{jumlah\ objek\ dari\ semua\ frame} x100\% = persentase\ keberhasilan$$

$$\frac{469}{482}\ x\ 100\% = 97{,}30\%$$

$$\frac{jumlah\ objek\ False\ Positive}{jumlah\ objek\ dari\ semua\ frame} x100\% = persentase\ kesalahan$$

$$\frac{13}{482}\ x\ 100\% = 2{,}70\%$$

- Keberhasilan sistem dimana tidak ada objek manusia dalam *frame* dan sistem tidak mendeteksi apapun.

$$\frac{true\ negative}{julmah\ frame\ tanpa\ objek} x100\% = persentase\ keberhasilan$$

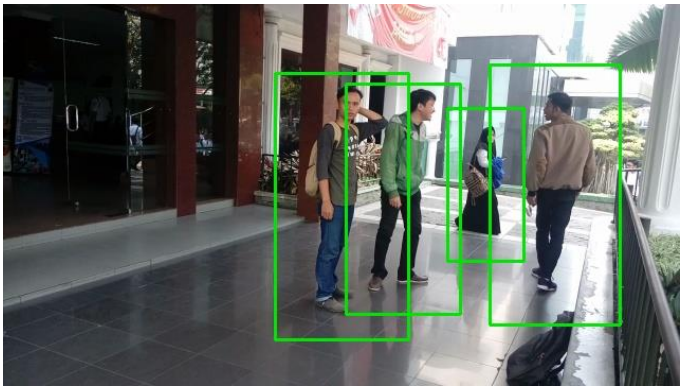
$$\frac{3}{123}\ x\ 100\% = 2{,}44\%$$

$$\frac{false\ negative}{julmah\ frame\ tanpa\ objek} x100\% = persentase\ kesalahan$$

$$\frac{120}{123}\ x\ 100\% = 97{,}56\%$$

1.1.2 Pengujian Akurasi Deteksi Multi Objek Manusia

Pengujian dilakukan dengan mengambil sampel dari video berdurasi 22 detik. Output data dari hasil pemrosesan algoritma HOG sebesar ±8 fps. Dengan jumlah 218 *frame* dengan resolusi 320 x 240.



Gambar 1.1.3 Frame Pengujian Deteksi Multi Objek Manusia

- Tingkat keberhasilan jika ada objek yang ada di *frame* dan sistem mendeteksinya.

$$\frac{jumlah\ objek\ True\ positive}{jumlah\ objek\ dari\ semua\ frame} x100\% = persentase\ keberhasilan$$

$$\frac{273}{605}\ x\ 100\% = 45{,}12\%$$

$$\frac{False\ Positive}{jumlah\ objek\ dari\ semua\ frame} x100\% = Persentase\ kesalahan$$

$$\frac{332}{605}\ x\ 100\% = 54{,}88\%$$

- Tingkat keberhasilan pada *frame* yang tidak terdapat objek manusia dan sistem tidak mendeteksi apapun.

$$\frac{true\ negative}{julmah\ frame\ tanpa\ objek} x100\% = persentase\ keberhasilan$$

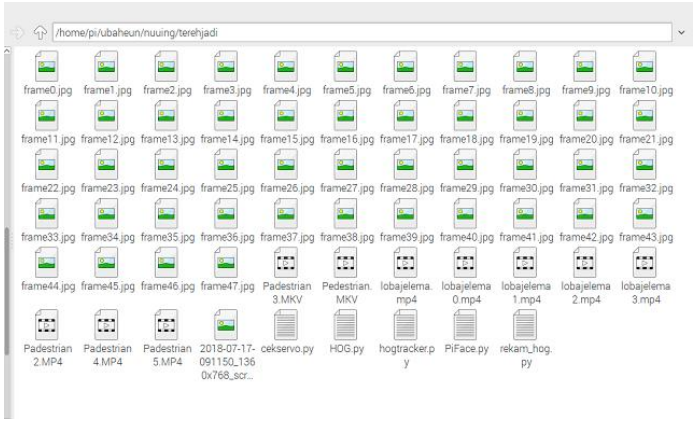
$$\frac{9}{12}\ x\ 100\% = 75\%$$

$$\frac{false\ negative}{julmah\ frame\ tanpa\ objek} x100\% = persentase\ kesalahan$$

$$\frac{2}{11}\ x\ 100\% = 25\%$$

1.1.3 Pengujian Rekam Frame

Pengujian untuk Perekaman data berupa *frame-frame* citra pada hasil pengolahan algoritma HOG di Raspberry Pi dilakukan saat tampil *bounding box* sebagai tanda berhasil mendeteksi objek manusia dan *frame* tersebut akan disimpan pada direktori penyimpanan program.



Gambar 1.1.4 Tampilan Hasil Penyimpanan pada Direktori Penyimpanan

Berikut adalah hasil pengujian penyimpanan hasil deteksi objek dimana objek yang terdeteksi akan disimpan berupa *frame* citra pada direktori penyimpanan Raspberry Pi yang ditunjukan pada tabel 4.1.1.

Tabel 1.1.1 Hasil Pengujian Penyimpanan *Frame* Hasil Deteksi

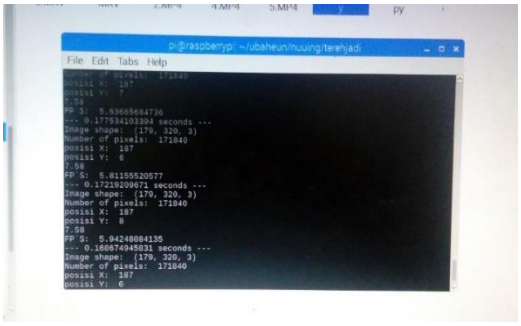
Frame	Bounding box pada monitor		simpan frame	keterangan
	ada	tidak		
1	✓		Frame0.png	frame disimpan
2	✓		Frame1.png	frame disimpan
3	✓		Frame2.png	frame disimpan
4	✓		Frame3.png	frame disimpan
5	✓		Frame4.png	frame disimpan
6	✓		Frame5.png	frame disimpan
7	✓		Frame6.png	frame disimpan
8	✓		Frame7.png	frame disimpan

1.2 Pengujian Perangkat Keras

Pengujian perangkat keras dilakukan dengan melihat performa algoritma HOG yang dijalankan di Raspberry Pi dan membandingkan arah gerak manusia yang ditampilkan pada monitor dengan pegerakan servo.

1.2.1 Pengujian performa algoritma HOG pada Raspberry Pi

Pengujian dilakukan dengan menjalankan program pada terminal di Raspberry Pi. Data yang diamati meliputi fps dan waktu eksekusi untuk tiap frame yang diolah dengan algoritma HOG pada Raspberry Pi.



Gambar 1.2.1 Tampilan Informasi *Framerate* dan Waktu Proses pada Terminal

1.2.1.1 Pengujian rata-rata fps

Pengujian performa dilakukan saat menjalankan program pada terminal pada Raspberry, pengujian dilakukan dengan cara mengamati data berupa rata-rata *frame* atau *framerate* dalam satu detik dengan satuan fps (*frame per secon*) dan waktu eksekusi setiap satu *frame* yang dieksekusi, data tersebut ditampilkan pada terminal ketika program sedang berjalan. Berikut adalah hasil pengujian yang ada pada tabel 4.2.1 dan 4.2.2.

- Pengujian untuk rata-rata *frame* dalam satu detik dilakukan dengan mencari framerate dengan nilai yang terbesar dan terkecil. waktu rata-rata fps algoritma HOG yang dijalankan di Raspberry Pi pada tabel 4.2.1.

Tabel 1.2.1 Rata-rata *frame* pada Raspberry Pi

Framerate terkecil	5,8 fps
Framerate terbesar	12 fps
Rata-rata framerate	8,9 fps

- Pengujian Waktu Rata-rata Eksekusi algoritma HOG. Pengujian dilakukan dengan menghitung waktu proses tiap satu *frame* dengan waktu proses terlama dan tercepat pada Raspberry Pi data tersebut ditunjukan pada tabel 4.2-2

Tabel 1.2.2 Rata-rata Waktu Proses Tiap Satu *Frame* pada Raspberry Pi

Waktu proses tercepat	0,08 detik
Waktu proses terlambat	0,16 detik
Rata-rata waktu proses	0,12 detik

1.2.2 Pengujian Servo

Pengujian servo dilakukan dengan cara membandingkan koordinat dari titik tengah objek dengan koordinat dari titik tengah kamera setelah mengikuti pergerakan objek.

Tabel 1.2.3 Perbandingan TItik Tengah Objek pada *Frame* dan Kamera pada sumbu X

Titik tengah seharusnya (pixel) sb X	**Titik tengah servo (pixel)**
84	82
82	95
95	105
106	102
72	102
83	75
84	93
62	73
69	69
72	74

$$\frac{titik\ tengah\ kamera - titik\ tengah\ di\ frame}{titik\ tengahdi\ frame}\ x\ 100\%$$

$$\frac{870-809}{809}\ x\ 100\% = 7{,}54 \text{ pixel}$$

Tabel 1.2.4 Perbandingan Titik Tengah Objek pada *Frame* dan Kamera pada Sumbu Y

Titik tengah seharusnya (pixel) sb y	**Titik tengah servo**
95	78
88	90
72	77
80	83
82	85
83	81
88	79
78	86

85	89
82	90

$$\frac{titik\ tengah\ kamera - titik\ tengah\ di\ frame}{titik\ tengahdi\ frame}\ x\ 100\%$$

$$\frac{838}{833}\ x\ 100\% = 0{,}6 \text{ pixel}$$

1.3 Analisa

Berdasarkan data yang didapatkan dari hasil pengujian akurasi pelacakan objek baik dengan satu objek ataupun lebih dari satu objek, uji performa dari Raspberry dengan mengitung rata-rata fps dan rata-rata waktu proses, serta membandingkan pergerakan servo dengan pergerakan koordinat pada output dan terakhir pengujian seluruh perangkat yang dibangun, didapat analisa dengan hasil sebagai berikut:

1. Pengujian deteksi lebih dari satu objek manusia dalam *frame* mendapatkan tingkat keberhasilan sebesar 45,12% dan pengujian deteksi satu objek manusia mendapatkan tingkat keberhasilan sebesar 97,30%. Untuk tingkat kesalahan pendeteksian dimana tidak ada objek manusia dalam frame, tingkat kesalahan untuk lebih dari satu objek manusia sebesar 25% dan untuk satu objek manusia sebesar 97,56%
2. Pada uji performa algoritma HOG yang dijalankan di Raspberry Pi didapat rata-rata 8,9 fps dan rata-rata waktu proses 0,12 detik dengan resolusi output yaitu 320 x 240 pixel.
3. Untuk penyimpanan *frame* pada pengujian penyimpanan data hasil pendeteksian oleh sistem berhasil dilakukan namun penyimpanan hanya berupa frame citra bukan berupa video dari 8 *frame* yang terdeteksi oleh sistem kedelapan *frame* tersebut disimpan di dalam direktori penyimpanan.
4. Tingkat error di koordinat y lebih kecil dari koordinat x yaitu 0,6 pixel dibanding 7,54 pixel.