

BAB IV

HASIL DAN PEMBAHASAN

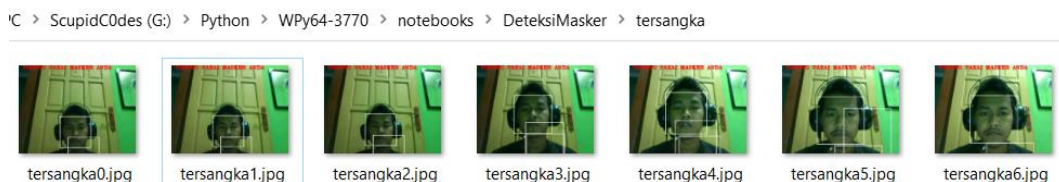
4.1 Pengujian Hardware

Pengujian komponen *internal* dari laptop yaitu webcam untuk pendeteksian secara *realtime* dapat berjalan dengan baik. Aplikasi dapat mendeteksi objek yaitu wajah dan mulut, dengan aturan jika mulut terdeteksi maka akan memunculkan tulisan “Tolong Pakai Masker Anda” dan membuat kotak putih seperti pada gambar 4.1.



Gambar 4.1 Webcam

Gambar 4.1 merupakan hasil capture dari webcam pada kondisi malam hari dengan pencahayaan yang kurang baik, namun aplikasi masih dapat mendeteksi wajah dengan cukup baik walaupun objek menggunakan *headset*. Aplikasi juga dapat memotret objek yang tidak menggunakan masker yang kemudian disimpan ke dalam folder yang telah diatur seperti pada gambar 4.2



Gambar 4.2 Fitur *capture* objek tidak memakai masker

Pengujian selanjutnya yaitu komponen *internal* perangkat keras yaitu kartu suara yang digunakan untuk memainkan fitur peringatan berupa audio apabila objek

mulut terdeteksi. Pengujian dilakukan dengan sistem *realtime* dan didapatkan hasil aplikasi dapat menjalankan fitur peringatan dengan baik.

Pengujian Webcam Eksternal dilakukan dengan scenario yaitu menggunakan perangkat telepon pintar untuk dijadikan kamera eksternal. Di table 4.1 merupakan hasil pengujian kamera eksternal.

Tabel 4.1 Pengujian kamera eksternal

No	Merk	Resolusi	Gambar	Hasil
1	Mi Max 3	64 MP		✓
2	Realme XT	64MP		✓
3	Samsung J7 Pro	13MP		✓
4	Xiaomi Redmi Note 3 Pro	16MP		✓

Keterangan:

✓ = terdeteksi

Berdasarkan table pengujian 4.1, dapat disimpulkan aplikasi tetap dapat berjalan dan mendeteksi masker dengan baik walaupun menggunakan kamera eksternal.

4.2 Pengujian Aplikasi

Pengujian dilakukan dengan total data yang diujikan yaitu 723 citra pengujian otomatis dan 35 citra pengujian manual. Beberapa skenario pengujian manual dilakukan untuk mengetahui apakah aplikasi dapat bekerja mengenali objek masker dengan baik, pengujian dilakukan dengan menggunakan beberapa ketentuan dan skenario, seperti:

- Citra akan diubah ukurannya menjadi 240x300
- Deteksi wajah menggunakan faktor skala 1.1 dan kemiripan tetangga 4.
- Deteksi masker menggunakan faktor skala 1.2 dan kemiripan tetangga 4.
- Deteksi mulut menggunakan faktor skala 1.4 dan kemiripan tetangga 5.
- Jarak objek dari kamera 50cm atau 100cm.
- Posisi objek ideal yaitu tegak lurus.

Sedangkan untuk beberapa scenario tersebut diantaranya:

4.2.1 Citra Masker Berdasarkan Jarak dan Sudut

Pengujian pertama dilakukan dengan dipengaruhi oleh 2 kondisi yaitu jarak dan sudut. Pertama dengan jarak objek ke kamera webcam yaitu 50cm dan kedua dengan jarak objek dengan kamera 100cm. Pengujian dilakukan dengan 5 variasi posisi sudut objek saat pengambilan gambar sebagai berikut;

1. Objek tegak lurus dengan kamera.
2. Objek bergeser 45° miring ke kanan
3. Objek bergeser 45° miring ke kiri
4. Objek bergeser 15° ke atas
5. Objek bergeser 15° ke atas

Tabel 4.2 Pengujian Aplikasi Pada Citra Berdasarkan Jarak dan Sudut

No	Jarak	Posisi/Sudut	Gambar	Hasil
1	20cm	Tegak Lurus		x
2	20cm	Miring Kanan 45°		x
3	20cm	Miring Kiri 45°		x
4	20cm	Dongak 15°		x

5	20cm	Nunduk 15°		✗
1	50cm	Tegak Lurus		✓
2	100cm	Tegak Lurus		✓
3	50cm	Miring Kanan 45°		✓
4	100cm	Miring Kanan 45°		✓

5	50cm	Miring Kiri 45°		✓
1	100cm	Miring Kiri 45°		✓
2	50cm	Dongak 15°		✓
3	100cm	Dongak 15°		✓
4	50cm	Nunduk 15°		✓

5	100cm	Nunduk 15°		✓
1	≤ 180 cm	Tegak Lurus		✓
2	≤ 180 cm	Miring Kanan 45°		✓
3	≤ 180 cm	Miring Kiri 45°		✓
4	≤ 180 cm	Dongak 15°		✓

5	≤ 180 cm	Nunduk 15°		✓
---	---------------	-------------------	--	---

Keterangan:

✗ = tidak terdeteksi

✓ = terdeteksi

Tabel 4.2 menunjukkan bahwa aplikasi dapat mendeteksi masker dengan berbagai sudut pengambilan gambar dan jarak. Pendeteksian masker ditandai dengan bingkai berwarna kuning untuk membedakan yang terdeteksi pada aplikasi apakah pendeteksian masker dengan wajah dan mulut.

Tabel 4.3 Hasil Pengujian Aplikasi Pada Citra Berdasarkan Jarak dan Sudut

No	Jarak	Sudut	Hasil
1	50, 100, & ≤ 180 cm	Tegak Lurus	Terdeteksi
2	50, 100, & ≤ 180 cm	Miring Kanan 45°	Terdeteksi
3	50, 100, & ≤ 180 cm	Miring Kiri 45°	Terdeteksi
4	50, 100, & ≤ 180 cm	Nunduk 15°	Terdeteksi
5	50, 100, & ≤ 180 cm	Dongak 15°	Terdeteksi
6	20cm & > 180 cm	Tegak Lurus	Tidak Terdeteksi
7	20cm & > 180 cm	Miring Kanan 45°	Tidak Terdeteksi
8	20cm & > 180 cm	Miring Kiri 45°	Tidak Terdeteksi
9	20cm & > 180 cm	Nunduk 15°	Tidak Terdeteksi
10	20cm & > 180 cm	Dongak 15°	Tidak Terdeteksi

Berdasarkan tabel 4.2 dan tabel 4.3 maka dapat disimpulkan aplikasi deteksi masker ini dapat mendeteksi masker dengan baik berdasarkan jarak dan juga posisi objek. Dapat disimpulkan pula jarak minimum untuk aplikasi deteksi masker ini yaitu 20cm dan maksimum yaitu 180cm, lebih dari 180cm aplikasi tidak bisa mendeteksi masker lagi.



Gambar 4.3 Hasil Tangkapan Layar Deteksi Masker

Pengambilan gambar otomatis menggunakan fitur tangkapan layar yang ada pada aplikasi deteksi masker dengan sistematis apabila aplikasi mendeteksi adanya masker, maka akan diambil tangkapan layar, yang kemudian di simpan pada folder yang sudah ditentukan seperti pada gambar 4.3.

4.2.2 Citra Berdasarkan Intensitas Cahaya

Pengujian ini dilakukan berdasarkan kondisi intensitas cahaya, di mana terdapat 3 kondisi saat pengujian dilakukan, yaitu:

- Terang yaitu kondisi pada luar ruangan yang mendapatkan cahaya berlebih baik dari sumber matahari langsung ataupun lampu.
- Normal yaitu kondisi di dalam ruangan dengan pencahayaan yang tidak terlalu terang dan gelap.
- Gelap yaitu kondisi di mana objek tidak mendapatkan cahaya dengan baik.

Tabel 4.4 Pengujian Berdasarkan Intensitas Cahaya

No	Warna Masker	Kondisi	Lux Meter	Hasil
1	Putih	Terang	13000 lx	
2	Putih	Normal	400 lx	
3	Putih	Gelap	5 lx	
4	Biru	Terang	10000 lx	

5	Biru	Normal	400 lx	
6	Biru	Gelap	5 lx	
7	Hitam	Terang	10000 lx	
8	Hitam	Normal	30 lx	
9	Hitam	Gelap	5 lx	

Tabel 4.4 menunjukkan bahwa aplikasi deteksi masker ini dapat mendeteksi masker dengan warna berbeda di intensitas cahaya berbeda pula. Namun pada masker berwarna hitam, aplikasi kesulitan mendeteksi penggunaan masker hitam pada intensitas cahaya normal & gelap yang menyebabkan keakuratan aplikasi berkurang.

Tabel 4.5 Hasil Pengujian Berdasarkan Intensitas Cahaya

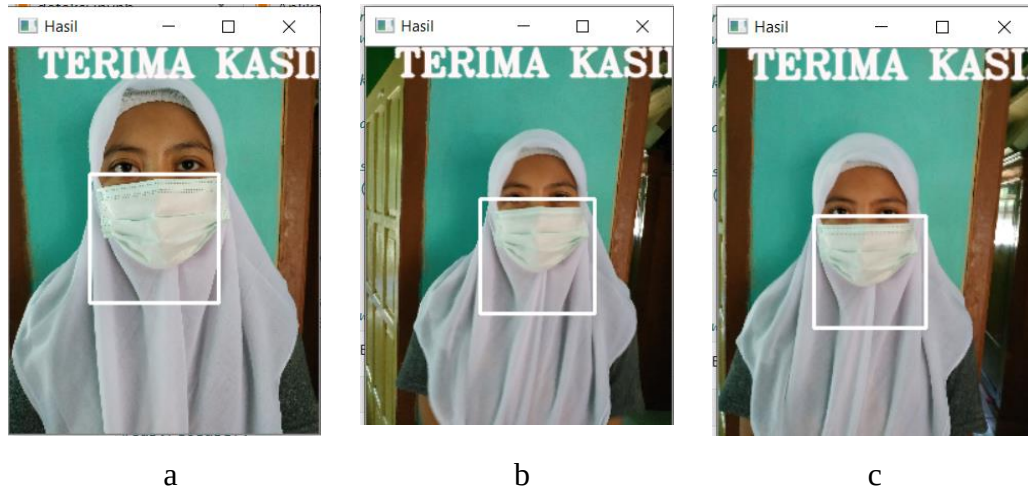
No	Warna Masker	Kondisi	Lux	Hasil	Waktu Deteksi
1	Biru	Terang	13000	Terdeteksi	< 1 detik
2	Biru	Normal	400	Terdeteksi	< 1 detik
3	Biru	Gelap	5	Terdeteksi	< 5 detik
4	Putih	Terang	10000	Terdeteksi	< 1 detik
5	Putih	Normal	400	Terdeteksi	< 1 detik
6	Putih	Gelap	5	Terdeteksi	< 3 detik
7	Hitam	Terang	10000	Terdeteksi	< 1 detik
8	Hitam	Normal	30	Terdeteksi	20 detik
9	Hitam	Gelap	5	Tidak Akurat	53 detik

Berdasarkan tabel 4.4 dan tabel 4.5 maka dapat disimpulkan aplikasi deteksi masker ini dapat mendeteksi masker dengan baik jika masker yang digunakan berwarna terang. Intensitas cahaya juga berpengaruh pada proses pendeteksian, karena semakin baik intensitas cahaya semakin baik kinerja aplikasi dalam mengenali masker dan tidak memerlukan waktu yang lama.

Pengujian aplikasi ini menggunakan bantuan aplikasi yang bernama Lux Meter untuk mengetahui nilai pencahayaan dalam suatu ruangan. Penguji mendapatkan hasil bahwa aplikasi ini sangat akurat untuk mendeteksi masker yang cenderung berwarna cerah dengan daya pancar cahaya yang direkomendasikan yaitu minimal 30 lx dan maksimal 13000lx, kurang dari nilai minimal ini atau lebih dari nilai maksimal ini aplikasi masih bisa mendeteksi namun membutuhkan waktu lama dan keakuratannya berkurang.

4.2.3 Citra Berdasarkan Warna Masker

Selanjutnya dilakukan pengujian dengan menggunakan citra wanita berhijab dengan menggunakan masker putih, tujuan dilakukannya pengujian dengan scenario seperti ini adalah untuk mengetahui apakah aplikasi dapat mengenali objek walaupun objek dominan berwarna putih dan didapatkan hasil aplikasi dapat mengenali objek dengan baik terlihat pada gambar 4.4.



Gambar 4.4 Pengujian dengan citra hijab putih masker putih

Dapat dilihat pada gambar 4.4 aplikasi dapat mengenali objek masker yang digunakan oleh wanita berhijab putih dan bermasker putih dengan berbagai kondisi, pada citra (a) dengan pengambilan gambar dari jarak 30cm, citra (b) dengan pengambilan jarak 50cm dengan posisi sedikit miring, citra (c) pengambilan gambar dari jarak 50cm.

Pada gambar 4.5 merupakan hasil dari pengujian selanjutnya yang dilakukan dengan menggunakan citra wanita berhijab dengan menggunakan masker hitam, tujuan dilakukannya pengujian dengan skenario seperti ini adalah untuk mengetahui apakah aplikasi dapat mengenali objek walaupun objek dominan berwarna hitam dan didapatkan hasil aplikasi dapat mengenali objek dengan baik terlihat.



Gambar 4.5 Pengujian dengan citra hijab & masker hitam

Tabel 4.6 Pengujian Aplikasi Pada Citra Wanita Berhijab

No	Warna Hijab	Warna Masker	Hasil
1	Putih	Putih	Terdeteksi
2	Hitam	Hitam	Terdeteksi

Berdasarkan tabel 4.6 maka dapat disimpulkan aplikasi deteksi masker ini dapat mendeteksi masker dengan baik berdasarkan skenario wanita berhijab dan memakai masker berwarna hitam atau putih.

4.2.4 Citra Pria Berkepala Botak

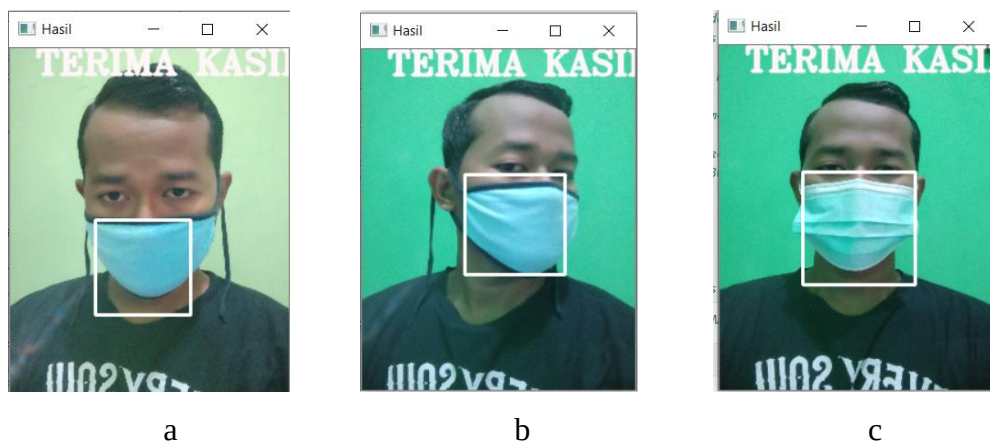
Pada gambar 4.6 merupakan hasil dari pengujian selanjutnya yang dilakukan dengan menggunakan citra pria dengan kepala botak, tujuan dilakukannya pengujian dengan skenario seperti ini adalah untuk mengetahui apakah aplikasi dapat mengenali objek dan didapatkan hasil aplikasi dapat mengenali objek dengan baik terlihat.



Gambar 4.6 Citra pria botak

4.2.5 Citra Pria Jenis Masker

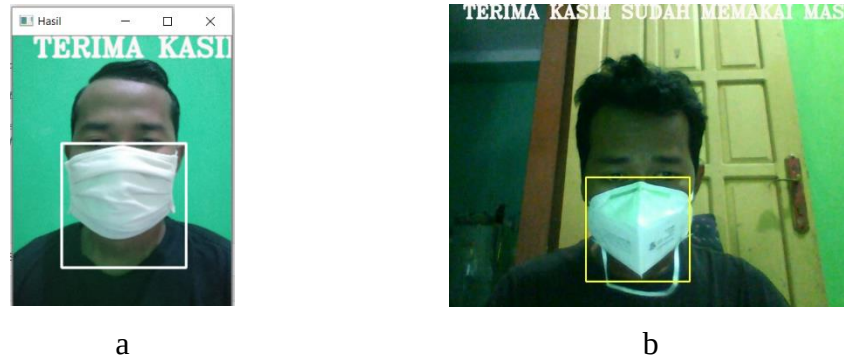
Pengujian ini dilakukan pada pria dengan menggunakan masker berwarna biru dan dapat dilihat pada gambar 4.7 merupakan hasil dari pengujian, tujuan dilakukannya pengujian dengan skenario seperti ini adalah untuk mengetahui apakah aplikasi dapat mengenali objek walaupun objek dominan berwarna biru dan didapatkan hasil aplikasi dapat mengenali objek dengan baik terlihat.



Gambar 4.7 Citra pria masker biru

Pada gambar 4.7 aplikasi dapat mengenali objek masker yang digunakan oleh pria berbaju hitam dan bermasker biru dengan berbagai kondisi, pada citra (a) jenis masker kain dengan pengambilan gambar objek lurus atau menghadap depan, citra (b) dengan pengambilan gambar dengan posisi sedikit miring dan menggunakan masker berjenis kain, citra (c) pengambilan gambar menggunakan masker dengan warna biru muda dan jenis masker bedah.

Lalu pengujian ini dilakukan menggunakan citra pria berbaju hitam dengan menggunakan masker putih, tujuan dilakukannya pengujian dengan scenario seperti ini adalah untuk mengetahui apakah aplikasi dapat mengenali objek dengan jenis bahans masekr berbeda walaupun objek dominan berwarna putih dan didapatkan hasil aplikasi dapat mengenali objek dengan baik terlihat pada gambar 4.8.



Gambar 4.8 Citra pria masker putih

Pada gambar 4.8 dapat dilihat bahwa aplikas deteksi masker ini dapat mendeteksi masker dengan bahan dan jenis yang berbeda, pada gambar 4.8 (a) masker yang digunakan yaitu masker berbahan dan jenis kain, sedangkan pada gambar 4.8 (b) masker yang digunakan yaitu jenis n95. Hasil pengujian dapat dilihat pada tabel 4.4

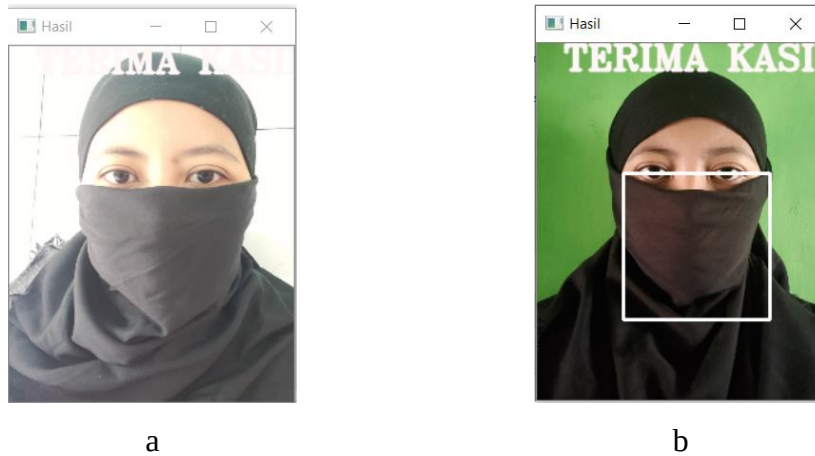
Tabel 4.7 Pengujian Aplikasi Pada Citra Pria Berdasarkan Warna Masker

No	Warna Masker	Jenis Masker	Hasil
1	Putih Cream	Kain	Terdeteksi
2	Putih	N95	Terdeteksi
3	Biru	Kain	Terdeteksi
4	Biru Muda	Bedah	Terdeteksi

Berdasarkan hasil pengujian yang dilakukan, didapatkan hasil seperti table 4.7 yang dapat disimpulkan bahwa aplikasi deteksi masker ini dapat mendeteksi masker berdasarkan warna dan jenis masker dengan baik, walaupun berbeda jenis dan warna masker, aplikasi masih dapat mendeteksi dengan baik.

4.2.6 Citra Wanita Menggunakan Cadar

Skenario terakhir dilakukan menggunakan citra wanita berhijab dengan menggunakan cadar, tujuan dilakukannya pengujian dengan scenario seperti ini adalah untuk mengetahui apakah aplikasi dapat mengenali objek walaupun objek menggunakan cadar dan didapatkan hasil aplikasi dapat mengenali objek dengan baik terlihat pada gambar 4.8.

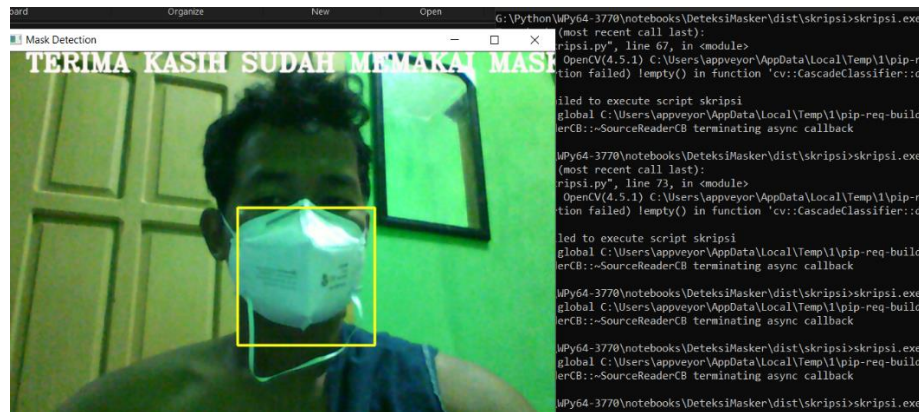


Gambar 4.9 Citra wanita berhijab dan cadar hitam

Citra (a) tidak terdeteksi sebagai masker karena terdapat gangguan pada citra yaitu pencahayaan yang terlalu terang dari belakang, sedangkan pada citra (b) dapat dideteksi sebagai masker karena tidak ada gangguan. Pencahayaan sangat berpengaruh pada proses pendeteksian objek ini seperti pada penjelasan tabel 4.4.

4.2.7 Pengujian *File Executable*

Untuk mempermudah penggunaan, pembuatan file *executable* diperlukan, agar pengguna dapat langsung menggunakannya tanpa perlu menjalankan sebuah IDE/Server terlebih dahulu. Pembuatan *file exe* ini langsung dibuat dari *script* Python menggunakan Pyinstaller. Hasilnya aplikasinya dapat berjalan dengan baik seperti saat pengujian menggunakan *script* python langsung.



Gambar 4.10 Pengujian File Exe

Pada gambar 4.10 dapat dilihat pengujian *file exe* dapat berjalan dengan baik dan menjalankan fitur deteksi masker. Penggunaan *file exe* ini bisa langsung dijalankan dengan 2 kali klik pada file exe atau bisa melalui cmd. Sedangkan untuk pengujian pada perangkat berbeda dengan merk dan spesifikasi berbeda, hasilnya dapat dilihat pada table 4.8:

Tabel 4.8 Pengujian Aplikasi

No	Merk	Versi Windows	Processor	RAM	VGA	Resolusi Layar	Konsumsi RAM ketika aplikasi dijalankan	Hasil Uji
1	Asus	10	I7	4GB	GTX 9000	1366x766	109MB	Berjalan Lancar
2	Asus	8.1	Amd	4GB	AMD	1366x768	110MB	Berjalan Lancar
3	Dell	7	Intel	4GB	IntelHD	1366x768	107MB	Berjalan Lancar
4	Dell	10	I5 8 th gen	8GB	Nvidia GTX 1050	1920x1080	117MB	Berjalan Lancar
5	MSI	10	I7 4 th	16GB	GTX 960M	1920x1080	111MB	Berjalan Lancar

Berdasarkan hasil pengujian pada table 4.8 dapat disimpulkan aplikasi dapat berjalan dengan lancar walaupun pada laptop dengan spesifikasi tergolong rendah. Dari ke-6 skenario di atas yang telah diuji sebelumnya, maka dapat ditarik kesimpulan dari ke-6 skenario tersebut bahwa aplikasi dapat mengenali objek masker dengan sangat baik, beberapa faktor yang dapat mempengaruhi hasil pengujian diantaranya kualitas citra tersebut, intensitas cahaya, dan jenis masker yang digunakan. Dari total 45 citra hasil pengujian, didapatkan hasil yang ditampilkan pada table 4.9:

Tabel 4.9 Hasil Pengujian Citra Secara Manual

No	Jumlah	Jenis Kelamin	Warna	Jenis	Jumlah Terdeteksi
1	10	Pria	Biru	Kain	10
2	10	Pria	Biru Muda	Bedah	8
3	10	Pria	Putih	N95	10
4	5	Pria	Putih	Kain	5
6	5	Wanita	Putih	Kain	3
7	3	Wanita	Hitam	Kain	2
8	2	Wanita	Hitam	Cadar	1
9	3	Pria	Hitam	Kain	2

Dari tabel 4.9 dapat dilihat bahwa total data uji sebanyak 48 citra, citra masker yang dapat dideteksi sebanyak 41 citra (positif). Dilakukan perhitungan dengan menggunakan persamaan 4.1 dan didapatkan nilai keakuratannya sebesar:

$$akurat = \frac{Positif}{Total\ Citra} \times 100 \quad (4.1)$$

$$akurat = \frac{41}{48} \times 100 = 83\%$$

Sedangkan untuk pengolahan secara otomatis 1 folder dataset, berisi 723 citra positif. Didapatkan hasil yang berbeda-beda tergantung dengan faktor skala. Algoritma yang digunakan ialah terdapat variabel positif dengan inisialisasi nilai awal 0, lalu program akan melakukan pengolahan citra secara otomatis secara berulang-ulang sebanyak citra yang ada. Apabila masker terdeteksi, maka nilai dari variabel positif ditambah +1. Di akhir program, variabel positif akan ditampilkan

untuk mengetahui berapa banyak citra positif. Hasilnya dapat dilihat pada tabel 4.6 ini.

Tabel 4.10 Hasil Pengujian Otomatis Dengan Berbagai Faktor Skala

Skala	Terdeteksi	Keakuratan	Waktu
1.1	517	71.5 %	183 s
1.2	642	88,7 %	172 s
1.3	325	44,9 %	170 s

Berdasarkan pengujian didapatkan hasil seperti pada table 4.10 dan didapatkan kesimpulan bahwa aplikasi deteksi masker ini memiliki tingkat deteksi yang tinggi pada factor skala 1.2 dengan tingkat keakuratan sebesar 88,7% dengan waktu penyelesaian 172 detik.

4.3 Pengujian Fungsionalitas Aplikasi

Pengujian fungsionalitas aplikasi dilakukan dengan melakukan uji pada fitur yang ada di aplikasi sesuai dengan kebutuhan dan rancangan aplikasi seperti pada tabel 4.11. Pengujian dilakukan dengan menggunakan peramban Google Chrome v87 (64bit), Mozilla Firefox v75 (64 bit), Microsoft Edge v 44.18362.449.0 (64bit), dan Opera v73 (64 bit). Pengujian fitur aplikasi dilakukan untuk mengecek apakah aplikasi dapat berjalan sesuai dengan yang deprogram dan diinginkan. Fitur pada aplikasi deteksi masker ini yaitu:

- Deteksi Masker
- Deteksi Wajah
- Deteksi Mulut
- Memotret objek yang tidak memakai masker
- Memainkan peringatan berupa audio
- Membuat bingkai persegi pada objek
- Memunculkan peringatan berupa teks citra.

Hasil dari pengujian fitur aplikasi ini dapat dilihat pada tabel 4.12 ini

Tabel 4.11 Pengujian Fitur Aplikasi

No	Fitur	Chrome	Firefox	Opera	Edge
1	Deteksi Masker	✓	✓	✓	✓
2	Deteksi Wajah	✓	✓	✓	✓
3	Deteksi Mulut	✓	✓	✓	✓
4	Memotret objek tidak memakai masker	✓	✓	✓	✓
5	Memainkan peringatan audio	✓	✓	✓	✓
6	Membuat bingkai persegi pada objek	✓	✓	✓	✓
7	Memunculkan peringatan berupa teks pada citra	✓	✓	✓	✓

Keterangan:

✓ = Bekerja

x = Tidak bekerja

Dilihat dari tabel 4.11 di atas menunjukkan bahwa keseluruhan fungsional aplikasi dapat bekerja dengan sangat baik di browser berbeda.

4.4 Pengujian User

Pada tahap selanjutnya dilakukan pengujian *user* untuk mengetahui apakah aplikasi ini dapat bekerja dengan baik dengan fitur yang memadai. Pengujian dilakukan dengan pengguna yang berbagai macam jenis/merk perangkat laptop/komputer. Hasil pengujian ini terdapat pada table 4.12:

Tabel 4.12 Pengujian User

No	Pertanyaan	Penilaian		
		Iya	Mungkin	Tidak
1	Apakah aplikasi deteksi masker yang menerapkan metode haar cascade ini mudah digunakan?	80%	20%	0%
2	Apakah menurut anda aplikasi deteksi masker yang menerapkan metode haar cascade ini dapat membantu memeprmudah kinerja petugas di lapangan?	70%	30%	0%
3	Apakah hasil dari aplikasi deteksi masker yang menerapkan metode haar cascade ini akurat?	60%	40%	0%
4	Apakah fitur dari aplikasi deteksi masker yang menerapkan metode haar cascade ini sudah cukup?	80%	20%	0%