

Deteksi Jumlah Pengunjung Dan Penggunaan Masker Dengan Menggunakan Metode YOLO Dan *Haar Cascade Classifier*

¹Achmad Ubaidillah Ms, ²Achmad Fiqhi Ibadillah. ³Muhammad Mukhlis Febriansyah

^{1,2,3} Teknik Elektro, Universitas Trunojoyo Madura, Kabupaten Bangkalan

¹ ubaidillah.ms@trunojoyo.ac.id, ² fiqhi.achmad@gmail.com, ³ mfebriansyah119@gmail.com

Abstract - Technological developments in the current era of globalization are very rapid. This makes technology a basic need for every human being. In 2022 new regulations will emerge regarding the use of masks when indoors and limiting the number of visitors while indoors. There are restrictions on the number of visitors and also the use of masks in this room as an effort to prevent and control Covid-19 so that the spread of Covid-19 does not occur as in previous years. Therefore, in this study, the authors designed an image processing-based tool to detect the number of visitors and use of masks using the You Only Look Once (YOLO) method and the Haar Cascade Classifier where the results of the two methods will be compared. Using a camera that is used as input for detection which will output from this tool in the form of Led televisions and speakers for information on visitors. After testing, the results obtained from this system using the YOLO method have a success rate of 93.3% for detecting the number of visitors. As for the Haar Cascade Classifier method, the success rate is 85.7% for detecting the number of visitors.

Keywords — *Number of visitors, Use of masks, YOLO, Haar Cascade Classifier.*

Abstrak— Perkembangan teknologi pada era globalisasi saat ini sangat pesat. Hal ini membuat teknologi menjadi kebutuhan dasar setiap manusia. Pada tahun 2022 muncul peraturan baru tentang penggunaan masker saat didalam ruangan dan pembatasan jumlah pengunjung saat didalam ruangan. Adanya pembatasan jumlah pengunjung dan juga penggunaan masker didalam ruangan ini sebagai upaya pencegahan dan pengendalian Covid-19 supaya tidak terjadi penyebaran Covid-19 seperti tahun-tahun sebelumnya. Oleh karena itu pada penelitian ini penulis merancang alat berbasis *image processing* untuk deteksi jumlah pengunjung dan penggunaan masker dengan menggunakan metode *You Only Look Once (YOLO)* dan *Haar Cascade Classifier* yang hasil dari kedua metode tersebut akan dibandingkan hasilnya. Menggunakan kamera yang digunakan sebagai *input* dari pendeteksian yang nantinya *output* dari alat ini berupa *Display Led* dan *speaker* untuk informasi terhadap pengunjung. Setelah dilakukan pengujian didapatkan hasil dari sistem ini dengan menggunakan metode YOLO tingkat keberhasilan 93,3 % untuk deteksi jumlah pengunjung. Sedangkan untuk metode *Haar Cascade Classifier* tingkat keberhasilan 85,7 % untuk deteksi jumlah pengunjung.

Kata Kunci— *Jumlah pengunjung, Penggunaan masker, YOLO, Haar Cascade Classifier,.*

I. Pendahuluan

Perkembangan teknologi pada era globalisasi saat ini sangat pesat dan masyarakat pun telah bergantung pada teknologi. Hal ini membuat teknologi menjadi kebutuhan dasar setiap manusia. Banyak teknologi komputer yang digunakan untuk memudahkan setiap pekerjaan. Pada tahun 2020 muncul peraturan baru tentang pembatasan jumlah pengunjung pada tempat tertentu dan fasilitas umum yang diatur oleh Kementerian Dalam Negeri Republik Indonesia. Adanya pembatasan jumlah pengunjung yang masuk didalam ruangan sebagai upaya pencegahan dan pengendalian *Corona Virus Disease 2019 (COVID-19)* [1].

Penanganan untuk kondisi darurat seperti pandemi membutuhkan bantuan teknologi lainnya untuk mengetahui jumlah pengunjung yang berada didalam ruangan dan juga pengunjung yang tidak memakai masker yang akan memasuki ruangan dengan menggunakan kamera. Hasil video dari kamera digunakan sebagai bahan data uji untuk dihubungkan dengan sistem teknologi yang lebih tinggi untuk mendeteksi jumlah manusia. Tempat dan fasilitas umum merupakan salah satu tempat masyarakat beraktivitas yang akan mendukung keberlangsungan perekonomian, namun berpotensi menjadi tempat penyebaran Covid-19 sehingga diperlukan protokol kesehatan dalam pelaksanaan kegiatan di tempat dan fasilitas umum. Berdasarkan peraturan Intruksi Menteri Dalam Negeri nomor 53 Tahun 2021, yaitu terkait upaya pemerintah dalam mencegah terjadinya penyebaran Covid-19 yaitu dengan membatasi jumlah pengunjung di tempat-tempat tertentu.

Berkembangnya dunia komputasi serta meningkatnya kapasitas dan kecerdasan proses komputer, saat ini muncul ilmu-ilmu komputasi yang memungkinkan komputer dapat mengambil informasi dari suatu citra untuk keperluan pengenalan objek secara otomatis.

Seperti pada penelitian Ilham Nurjabar, Muhamad Nicky pada tahun 2022 menciptakan aplikasi yang dapat mendeteksi penggunaan masker untuk meminimalisir penularan virus Covid-19 yang saat ini menjadi wabah di Indonesia dengan fitur mengeluarkan peringatan yang berupa audio dan memotret jika ada terdeteksi tidak mengenakan masker, sehingga dapat meringankan beban kerja petugas di lapangan. Pada penelitian ini metode yang digunakan ialah *Haar Cascade*. Hasil dari penelitian ini adalah aplikasi dapat mendeteksi masker dari citra yang bersumber dari foto atau video dari *webcam* internal

maupun eksternal dengan baik, dengan total keakuratan tertinggi 88,7% dan terendah 44,9% [2].

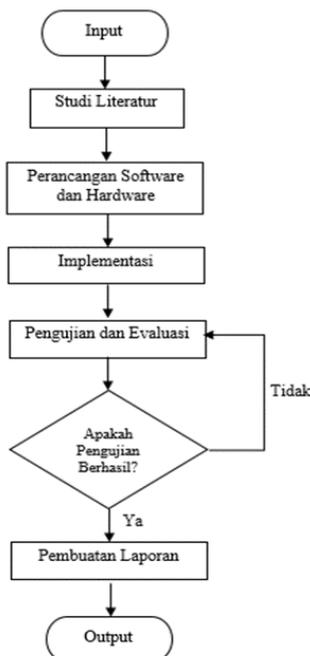
Pada penelitian selanjutnya dilakukan oleh Bathija pada tahun 2019 yang membahas tentang pelacakan objek dengan pendekatan deteksi menggunakan YOLO dan pelacakan menggunakan algoritma SORT. Menggunakan masukan video untuk mengenali kendaraan dan pejalan kaki untuk analisis lalu lintas. Dataset yang digunakan sebanyak 800 gambar dengan 6 kelas. Akurasi dan presisi dapat terdeteksi dengan melatih sistem dengan menggunakan lebih banyak *epoch*. Kinerja SORT untuk pelacakan tergantung pada kinerja *detector* [3].

Sehingga pada penelitian kali ini, penulis menggunakan pembaruan dengan melakukan pendeteksian masker dan perhitungan jumlah pengunjung yang masuk dan keluar ruangan, dengan menggunakan metode *You Only Look Once* (YOLO) dan *Haar Cascade Classifier* yang hasil dari penelitian akan dibandingkan antara metode YOLO dan *Haar Cascade Classifier* berdasarkan presentase keberhasilan pendeteksian.

II. Metode Penelitian

A. Alir Penelitian

Alur penelitian adalah uraian beberapa tahap yang dilakukan pada saat melakukan perancangan dan pengujian. Gambaran umum tahapan tersebut dapat dilihat dalam gambar 1 sebagai berikut. Memulai penelitian dengan mencari literatur yang berhubungan dengan penelitian. Dari hasil literatur dilakukan pembuatan sistem. Setelah melakukan pembuatan, pengujian terhadap simulasi apabila sudah memenuhi tujuan yang dicapai, tahap selanjutnya yang dilakukan adalah implementasi. Apabila telah selesai maka penelitian selesai.



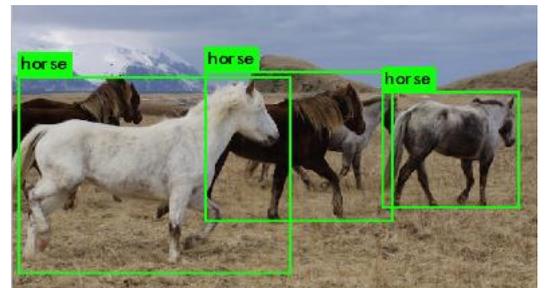
Gambar 1. Alur penelitian

B. Dasar Teori

Berdasarkan definisi dari *United Nations World Tourism Organization* (UNWTO) pengunjung (*visitor*) merupakan bagian dari orang yang melakukan perjalanan berkunjung pada suatu tempat yang terdiri dari banyak orang dengan tujuan yang berbeda-beda [4]. Namun saat adanya pandemi yang disebabkan oleh penyebaran suatu virus yang disebut dengan SARS-COV2 atau sering disebut dengan virus Corona, oleh sebab itu adanya peraturan pembatasan jumlah pengunjung yang masuk didalam ruangan dan penggunaan masker sebagai upaya pencegahan dan pengendalian Covid19. Berikut beberapa materi yang digunakan dalam sistem ini :

1) Metode *You Only Look Once* (YOLO)

You Only Look Once (YOLO) adalah salah satu pendekatan untuk melakukan pendeteksian objek secara *real-time* berbasis *Convolutional Neural Network* (CNN). YOLO menggunakan pendekatan jaringan syaraf tunggal (*Single neural network*) untuk melakukan pendeteksian objek pada sebuah citra [5].



Gambar 2. Deteksi YOLO

Tahapan yang perlu dilakukan dalam mendeteksi objek dengan metode YOLO adalah sebagai berikut :

1. Citra akan dibagi menjadi *grid* dengan ukuran $s \times s$, *bounding box* selanjutnya melakukan prediksi tiap *grid* dan nilai *confidence*. Nilai *Confidence* merupakan nilai keyakinan *bounding box* yang berisi rencana dan akurasi prediksi.

$$\text{Conf}(\text{Class}) = \text{Pr}(\text{Class}) \times \text{IOU} \frac{\text{Truth}}{\text{Pred}} \quad (1)$$

2. Dari persamaan diatas $\text{Pr}(\text{class})$ adalah objek yang memiliki kemungkinan muncul dalam suatu *region* dan IOU merupakan *Intersection Of Union* atau bisa disebut rasio tumpang tindih antara kotak prediksi dengan kota kebenaran (*truth*). Tingkat akurasi dari pendeteksian objek bergantung pada besar tidaknya nilai IOU, jika nilai IOU semakin besar maka tingkat keakuratannya juga bertambah.

$$\text{IOU} = \frac{\text{Truth}}{\text{Pred}} = \frac{\text{Area of overlap}}{\text{Area of union}} \quad (2)$$

3. Kemudian dari tiap *bounding box* akan diperoleh lima *variable* yaitu x , y , w , h dan c . Dimana x dan y adalah nilai koordinat dari titik tengah *bounding box* objek yang terdeteksi dalam citra. Untuk w dan h merupakan nilai ukuran lebar dan tinggi, sedangkan c merupakan nilai *confidence* dari *bounding box*.
4. Tahap selanjutnya tiap *grid* melakukan prediksi nilai probabilitas jika didalam citra tersebut pada objek. Nilai probabilitas dan nilai *confidence* dikalikan untuk memperoleh nilai *confidence* pada tiap *bounding box* di kelasnya dengan akurat.

$$\frac{\Pr(\text{Class}_i | \text{Object}) \times \Pr(\text{Object}) \times \text{IOU}_{\text{Pred}}^{\text{Truth}}}{\Pr(\text{Class}_i) \times \text{IOU}_{\text{Pred}}^{\text{Truth}}} \quad (3)$$

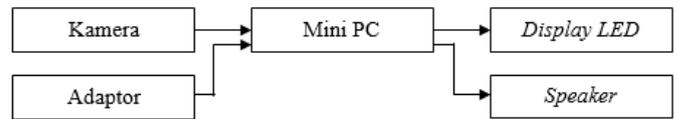
2) Metode Haar Cascade Classifier

Haar cascade dimana algoritma ini merupakan salah satu model *machine learning* yang kerap kali digunakan sebagai pondasi aplikasi *object detection* (terutama *face recognition*) dalam sebuah gambar maupun video. Metode ini sering digunakan untuk mengenali area mana saja terdapat wajah manusia (*face recognition*) pada sebuah gambar menggunakan *OpenCV* [6]. Konsep yang menjadi *Algoritma Haar Cascade Classifier*, yakni :

1. Fitur *rectangular* yang disebut *Haar-like feature* untuk memproses gambar dalam kotak-kotak, di mana dalam satu kotak bisa terdiri dari beberapa *pixel* [7]. Per kotak itu pun lalu diproses dan di cari perbedaan nilai (*threshold*) yang menandakan daerah gelap dan terang.
2. *Integral image* untuk menambahkan nilai dari *pixel* secara bersamaan,. Nilai dari integral pada masing-masing *pixel* adalah penjumlahan dari semua *pixel* di atasnya dan di sebelah kirinya [8]. Dimulai dari kiri atas sampai kanan bawah sampai gambar bisa diintegrasikan sebagai operasi matematika per *pixel*.
3. Metode *machine learning AdaBoost* digunakan untuk menggabungkan banyak *classifier* lemah untuk membuat sebuah *classifier* kuat [9]. dengan menggabungkan beberapa *AdaBoost classifier* sebagai rangkaian filter yang cukup efisien untuk menggolongkan daerah *image*.
4. *Cascade classifier* digunakan untuk filter pada bobot yang diberikan oleh *AdaBoost*. Filter dengan bobot paling besar diletakkan pada proses pertama kali, bertujuan untuk menghapus daerah gambar bukan wajah secepat mungkin [10].

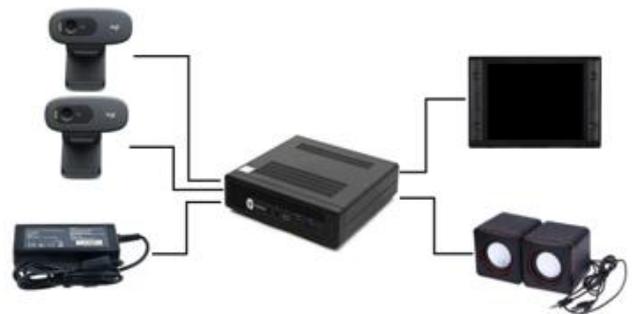
C. Perancangan Sistem

Dalam perancangan sistem ini terdapat beberapa perangkat yang akan di integrasikan. Berikut adalah blok diagram untuk pengujian sistem deteksi jumlah pengunjung dan penggunaan masker.



Gambar 3. Diagram Blok Sistem

Dalam perancangan alat pada sistem ini menggunakan berbagai macam *hardware* yang terintegrasi pada gambar 3. Proses pertama dari sistem ini adalah pengambilan citra manusia dan masker menggunakan kamera sebagai data *input*. Kemudian data akan diproses dengan pengolahan citra menggunakan Mini Pc untuk dilakukan pemrosesan data dengan menggunakan metode YOLO dan *Haar Cascade Classifier*. Setelah itu, data akan diidentifikasi untuk menentukan jumlah pengunjung yang ada pada tempat tersebut dan penggunaan masker pada pengunjung. Display Led pada sistem ini digunakan sebagai *Graphic User Interface* (GUI) untuk menampilkan hasil jumlah pengunjung yang ada didalam ruangan dan *speaker* sebagai notifikasi jika pengunjung telah mencapai batas kapasitas maksimal yang telah ditentukan, dan pengunjung yang tidak menggunakan masker.



Gambar 4. Perancangan Hardware

Pada gambar 4 merupakan perancangan *hardware* yang terdiri dari 2 buah kamera, Display Led, Mini PC, adaptor, dan *speaker*. 2 buah kamera terhubung ke mini PC dengan menggunakan slot usb (*universal serial bus*). Display Led terhubung ke Mini PC dengan menggunakan konektor HDMI (*high definition multimedia interface*). Adaptor terhubung ke mini PC melalui konektor *charger*. *Speaker* terhubung ke mini PC melalui *earphone jack*.

III. Hasil dan Pembahasan

Dalam bab ini akan dijelaskan mengenai hasil dan Analisa dari pengujian sistem deteksi jumlah pengunjung dan penggunaan masker. Dalam pengujian sistem untuk mendeteksi jumlah pengunjung dan penggunaan masker kali ini akan didapatkan sebuah hasil berupa persentase keberhasilan dan kegagalan dalam mendeteksi jumlah pengunjung. Berikut adalah perhitungan untuk menentukan persentase keberhasilan dan kegagalan dari sistem tersebut :

$$\text{Keberhasilan (\%)} = \frac{\text{jumlah data benar}}{\text{total data uji}} \times 100\% \quad (4)$$

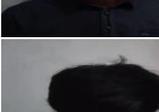
$$\text{Kegagalan (\%)} = \frac{\text{jumlah data salah}}{\text{total data uji}} \times 100\% \quad (5)$$

A. Perancangan Sistem

Pada pengujian deteksi masker ini dilakukan 3 kali pengujian, yang pertama yaitu pengujian deteksi masker berdasarkan posisi wajah, yang kedua deteksi masker berdasarkan jarak, dan yang terakhir pengujian deteksi masker berdasarkan tingkat kecerahan cahaya. Tujuan dari pengujian ini untuk mengetahui keakuratan dari sistem, saat alat mendeteksi masker, dan juga sebagai *output* untuk notifikasi suara dari *speaker* kepada pengunjung ketika pengunjung tidak menggunakan masker. Pada hasil pengujian didapatkan hasil pengujian sebagai berikut :

1) Pengujian Masker Berdasarkan Posisi

Tabel 1. Pengujian berdasarkan posisi

No	Hasil Pengujian	Kondisi	Keterangan
1		Dari depan	Terdeteksi
2		Dari kanan	Terdeteksi
3		Dari kiri	Terdeteksi
4		Hadap ke atas	Tidak Terdeteksi
5		Hadap ke bawah	Tidak Terdeteksi

Pada hasil pengujian deteksi masker dengan menggunakan 5 citra sebagai pengujian, didapatkan hasil tidak semua masker dapat terdeteksi, hanya 3 citra yang dapat terdeteksi maskernya, kondisi yang bisa dideteksi oleh alat yaitu, deteksi dari depan, dari kanan, dan dari kiri. Kemudian 2 citra dengan kondisi menghadap ke atas, dan menghadap ke bawah tidak dapat dideteksi oleh alat.

2) Pengujian Masker Berdasarkan Jarak

Tabel 2. Pengujian berdasarkan jarak

No	Hasil Pengujian	Kondisi	Keterangan
1		Jarak 1 Meter	Terdeteksi
2		Jarak 2 Meter	Terdeteksi
3		Jarak 3 Meter	Terdeteksi
4		Jarak 4 Meter	Terdeteksi

Pada hasil pengujian deteksi masker berdasarkan jarak dengan 4 citra, didapatkan hasil yang maksimal dikarenakan semua masker dapat terdeteksi, citra yang terdeteksi menggunakan masker yakni dengan jarak 1 meter, 2 meter, 3 meter, dan 4 meter.

3) Pengujian Masker Berdasarkan Faktor cahaya

Tabel 3. Pengujian berdasarkan faktor cahaya

No	Hasil Pengujian	Kondisi	Keterangan
1		<i>Lux</i> < 50	Terdeteksi
2		<i>Lux</i> 50-150	Terdeteksi
3		<i>Lux</i> > 150	Terdeteksi

Pada hasil pengujian deteksi masker berdasarkan tingkat kecerahan cahaya dengan satuan *lux* menggunakan 3 citra, didapatkan hasil semua masker dapat terdeteksi dengan *lux* dibawah 50, *lux* antara 50-150, dan *lux* di atas 150.

B. Pengujian Perhitungan Pengunjung

Pada pengujian perhitungan pengunjung ini dilakukan 2 kali pengujian, dengan menggunakan *input* video pengujian yang sama, yang pertama yaitu pengujian perhitungan pengunjung menggunakan metode YOLO dan pengujian kedua menggunakan metode *Haar Cascade Classifier*. Berikut merupakan hasil dari pengujian perhitungan pengunjung :

1) Pengujian Perhitungan Pengunjung YOLO
Tabel 4. Pengujian pengunjung YOLO

No	Masuk	Keluar	Lok	Ket
1			1	✓
2			1	✓
3			1	✓
4			1	✓
5			2	✓
6			2	✓
7			2	✓
8			2	✓
9			2	✗
10			2	✓

Berdasarkan hasil uji dari 15 citra pengunjung diperoleh hasil keberhasilan dan kegagalan seperti dibawah ini :

$$\text{Keberhasilan}(\%) = \frac{14}{15} \times 100\% = 93,3\%$$

$$\text{Kegagalan}(\%) = \frac{1}{15} \times 100\% = 6,7\%$$

Dari pengujian tersebut didapatkan hasil kesalahan data uji dari 15 citra, ada 1 citra yang mengalami kesalahan dikarenakan 1 pengunjung dengan nomor urut pengujian 9 saat memasuki ruangan pengunjung tersebut terhitung 2 kali oleh sistem. Namun untuk sisanya sistem mampu mendeteksi 14 citra tersebut dengan benar dan melakukan perhitungan dengan benar.

2) Perhitungan Pengunjung Haar Cascade Classifier
Tabel 5. Pengujian perhitungan Haar Cascade

No	Masuk	Keluar	Lok	Ket
1			1	✓
2			1	✓
3			1	✓
4			1	✗
5			2	✓
6			2	✓
7			2	✓
8			2	✓
9			2	✓
10			2	✓

Berdasarkan data yang diperoleh dari proses uji, diperoleh nilai tingkat keberhasilan dan kegagalan dari proses data uji seperti dibawah ini :

$$\text{Keberhasilan(\%)} = \frac{13}{15} \times 100\% = 86,6\%$$

$$\text{Kegagalan(\%)} = \frac{2}{15} \times 100\% = 13,4\%$$

Dari pengujian tersebut didapatkan hasil kesalahan data uji dari 15 citra, ada 2 citra yang mengalami kesalahan, untuk kesalahan pertama dikarenakan pengunjung yang memasuki ruangan dan tertutup objek lain. Untuk kesalahan kedua seperti pada pengujian nomor 4 yaitu kesalahan perhitungan keluar. Namun untuk sisanya sistem mampu mendeteksi 13 citra lainnya dengan benar.

C. Analisa Perhitungan

Untuk analisa perhitungan dilakukan 2 kali perhitungan, yaitu perhitungan YOLO dan *Haar Cascade Classifier*. Berikut merupakan hasil dari perhitungan :

1) Perhitungan YOLO

Pada perhitungan YOLO langkah awal yang harus dilakukan adalah melakukan *resize* citra. Data *input* awal dari citra yang ukurannya bervariasi kemudian di ubah menjadi ukuran 416 x 416 dengan 3 *channel*.

Citra yang sudah di *resize* dengan ukuran 416 x 416 kemudian dilakukan pembagian *grid sel* dengan ukuran 7 x 7 seperti gambar 5.



Gambar 5. Pembagian *grid* 7 x 7

Perhitungan *Intersection Over Union* (IoU) Sesuai dengan rumus IoU pada persamaan 2 maka dihitung sebagai berikut: $A \cap G_{truth} = 50 \times 50 = 2500$

$$A \cup TruthP = (51 \times 50) + (51 \times 51) = 2550 + 2601 = 5151$$

$$IoU = \frac{A \cap G_{truth}}{A \cup TruthP} = \frac{2500}{5151} = 0,4853$$

$$Pr(\text{Class}) = Pc_0 \times IoU = 1 \times 0,48 = 0,48$$

Convolution dilakukan pada jaringan YOLO setelah melalui proses *resize* dan pencarian IoU.

Titik RGB (x,y) = (152,100)

Lebar (w) = (51)

Tinggi (h) = (51)

Tabel 6. Hasil Konvolusi

x,y	1	2	50	51
1	0	0	0	0
2	0	1	1	0
....
6	0	1	1	0
7	0	0	0	0

Tabel 7. Matrix Konvolusi

x,y	1	2	44	45
1	755	756	746	649
2	732	727	524	606
....
44	766	761	201	217
45	743	752	178	211

ReLU suatu fungsi yang digunakan untuk mengubah nilai piksel *negative* menjadi 0 dan nilai yang lebih dari 255 di ubah menjadi 255 [11].

Tabel 8. Matrix *ReLU*

x,y	1	2	44	45
1	255	255	255	255
2	255	255	255	255
....
44	255	255	201	217
45	255	255	178	211

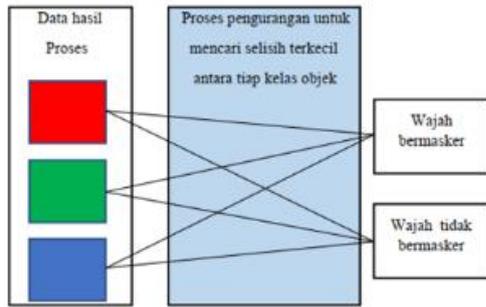
Max Pooling adalah proses penyeleksian nilai matriks yang lebih besar (maksimal) diantara nilai tetangganya [12]. Operasi *max pooling* dengan menggeser matrik 2 x 2 atau disebut *stride*. Hasil operasi *max pooling* yang sudah melalui proses konvolusi dan *ReLU* maka akan menghasilkan matrik baru dengan ukuran 23 x 23 seperti pada tabel 9.

Tabel 9. *Max Pooling*

x,y	1	2	22	23
1	255	255	255	255
2	255	255	255	255
....
22	255	255	232	198
23	743	255	201	204

Setelah semua proses selesai maka *Fully Connected Layer* baru bisa dilakukan, *fully connected layer* merupakan *layer* yang berfungsi sebagai pengklasifikasian kelas objek, dengan pencocokan data citra yang sudah di proses pada *step-step*

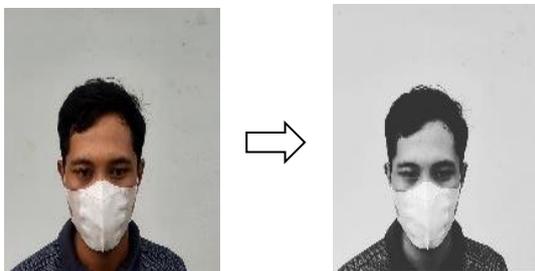
sebelumnya dengan data *training* kelas yang sudah dilabel [13].



Gambar 6. Fully Connected Layer

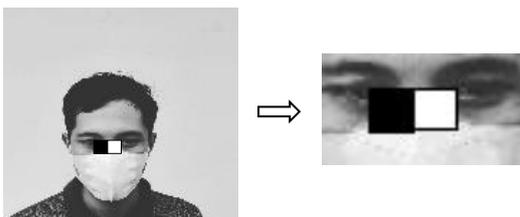
2) Perhitungan *Haar Cascade Classifier*

Pada perhitungan Haar hal pertama yang harus dilakukan dalam metode *haar cascade classifier* adalah dengan mengubah citra rgb menjadi citra *grayscale*, citra *grayscale* adalah citra dengan gradasi warna hitam dan putih dengan nilai paling besar 255 berwarna putih hingga warna hitam dengan nilai intensitas paling kecil (0) [14].



Gambar 7. Rgb ke *grayscale*

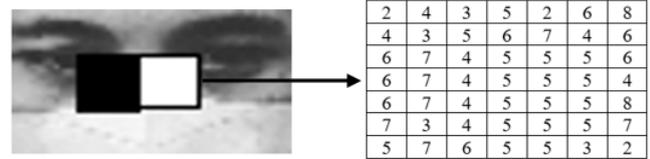
Setelah citra warna diubah menjadi *grayscale*, proses selanjutnya mendapatkan nilai fitur dengan melakukan pemindaian pada citra *grayscale*. Teknik yang dilakukan yaitu memindai bagian citra dengan *haar-like feature* setiap daerah *image* dari mulai ujung kiri atas sampai kanan bawah. Proses ini dilakukan untuk mencari nilai fitur yang menyatakan objek wajah.



Gambar 8. *Haar-like feature*

Proses selanjutnya melakukan *integral image* digunakan untuk proses perhitungan *integral image*

secara cepat dan akurat. Nilai *pixel* yang akan dihitung merupakan nilai dari *pixel* citra yang sudah diproses oleh fitur *haar*.



Gambar 9. Pengambilan *sub-window*

Dari nilai *pixel* yang didapatkan dilakukan perhitungan *integral image*.

	1	2	3	4	5	6	7
1	2	4	3	5	2	6	8
2	4	3	5	6	7	4	6
3	6	7	4	5	5	5	6
4	6	7	4	5	5	5	9
5	6	7	4	5	5	5	8
6	7	3	4	5	5	5	9
7	5	7	6	5	5	3	6

Gambar 10. Nilai asli dari *sub-window 7 x 7*

	1	2	3	4	5	6	7
1	4	9	10	17	23	31	44
2	13	21	32	41	51	65	80
3	26	38	54	68	83	103	124
4	24	52	72	98	122	147	154
5	36	78	106	142	179	217	256
6	41	91	111	315	204	241	288
7	54	200	213	196	241	289	345

Gambar 11. Hasil *integral* dari *sub-window*

	1	2	3	4	5	6	7
1	4	9	10	17	23	31	44
2	13	21	32	41	51	65	80
3	26	108	64	56	72	103	124
4	24	68	83	58	122	147	154
5	36	78	106	142	179	217	256
6	41	91	111	315	204	241	288
7	54	200	213	196	241	289	345

Gambar 12. Hasil pembentukan fitur

Kemudian dilakukan kalkulasi untuk mendapatkan nilai fitur berdasarkan gambar diatas. Prosesnya sebagai berikut :

$$\begin{aligned} \text{Nilai Fitur} &= |(\text{total piksel hitam}) - (\text{total piksel putih})| \\ &= |{(68 + 108) - (83 + 64)} - (56 + 72) - (58 + 122)| \\ &= |29 - 52| \\ &= 23 \end{aligned}$$

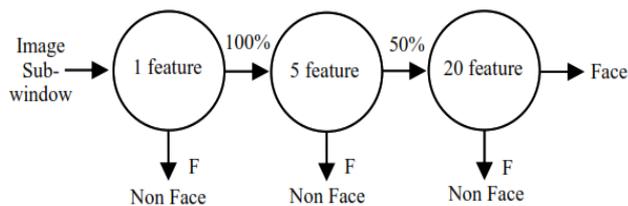
Nilai *haar feature* yang didapatkan diatas yaitu 23 adalah nilai perbedaan antara kotak (fitur putih dan fitur hitam) yang biasa disebut dengan *threshold*. *Threshold* digunakan sebagai parameter klasifikasi objek yang terdeteksi sebagai wajah atau tidak [15].

Adaboost digunakan untuk membentuk *classifier* kuat dengan menentukan beberapa fitur dari banyak fitur, fitur yang ditentukan *weak classifier* yang akan diberikan penambahan bobot dan gabungan dari *weak classifier* lainnya untuk membentuk *strong classifier* yang akan memberikan prediksi pendeteksian lebih baik.

Proses *cascade classifier* digunakan sebagai klasifikasi. *Haar-like feature* mempunyai sifat *learner* dan *classifier* yang lemah, jika ingin mendapatkan hasil yang lebih akurat maka harus dilakukan proses *haar-like feature* secara masal. Semakin banyak proses *haar-like feature* yang dilakukan maka akan semakin akurat hasil yang dicapai. Dibawah ini merupakan alur *cascade classifier*.

Pada klasifikasi pertama tiap subcitra akan diklasifikasi menggunakan satu fitur. Jika hasil nilai fitur dari filter tidak memenuhi kriteria yang diinginkan, maka hasil akan ditolak.

Algoritma kemudian bergerak ke *sub-window* selanjutnya dan menghitung nilai fitur kembali. Jika didapat hasil sesuai dengan *threshold* yang diinginkan, maka dilanjutkan ke tahap filter selanjutnya. Hingga jumlah *sub-window* yang lolos klasifikasi pun akan berkurang hingga mendekati citra yang ada pada sampel.



Gambar 13. Proses Cascade Classifier

Pada gambar 13 di atas, dapat dijelaskan secara keseluruhan proses *cascade classifier*-nya sebagai berikut :

1. Filter pertama (*First classifier*)
1 fitur
Detection rate : 100%, *false positive rate* : 50%.
2. Filter kedua (*second classifier*)
5 fitur
Detection rate : 100%, *false positive rate* : 80%.
3. Filter ketiga dan keempat (*third and fourth classifiers*)
20 fitur

Sedangkan hasil dari filter pada proses *cascade classifier* dapat dilihat pada gambar 14.



Gambar 14. Hasil deteksi

IV. Kesimpulan

Berdasarkan hasil pengujian dan analisa pada sistem deteksi jumlah pengunjung dan penggunaan masker didapatkan sebuah kesimpulan bahwa :

1. Dalam pengujian sistem hasil pendeteksian dengan metode YOLO dan *Haar Cascade Classifier* didapatkan hasil yang cukup baik dikarenakan presentase keberhasilan pendeteksian dari alat masih di atas 80%.
2. Dari hasil pengujian deteksi masker kedua metode mendapatkan hasil yang sama, sehingga peneliti hanya memasukan 1 metode saja untuk deteksi masker.
3. Untuk deteksi jumlah pengunjung menggunakan metode YOLO mendapatkan tingkat keberhasilan 93,3% dan *Haar Cascade Classifier* mendapatkan tingkat keberhasilan 85,7% dengan sama-sama menggunakan 15 data uji.

Dengan adanya kesimpulan seperti diatas, ada beberapa hal yang perlu diperhatikan, oleh karena itu ada beberapa saran yang perlu diperhatikan :

1. Diharapkan untuk penelitian selanjutnya penggunaan *hardware* terutama Mini PC menggunakan spesifikasi yang lebih tinggi, dikarenakan pada penelitian kali ini menggunakan 2 buah kamera secara bersamaan sehingga masih ada *delay/lag* saat proses pengujian.

V. Daftar Pustaka

- [1] B. Putra, G. Pamungkas, B. Nugroho, and F. Anggraeny, "Deteksi dan Menghitung Manusia Menggunakan YOLO-CNN," *J. Inform. dan Sist. Inf.*, vol. 02, no. 1, pp. 67–76, 2021.
- [2] A. R. Wasril, M. S. Ghozali, and M. B. Mustafa, "Pembuatan Pendeteksi Obyek Dengan Metode You Only Look Once (Yolo) Untuk Automated Teller Machine (Atm)," *Maj. Ilm. UNIKOM*, vol. 17, no. 1, pp. 69–76, 2019.
- [3] A. Bathija, "Visual Object Detection and Tracking using YOLO and SORT," *Int. J. Eng. Res. Technol.*, vol. 8, no. 11, pp. 705–708, 2019.

-
- [4] S. E. Priyanto and E. Sugiarto, "Preferensi Pengunjung Terhadap Pelayanan Di Grhatama Pustaka Yogyakarta," *Pringgitan*, vol. 1, no. 02, pp. 87–97, 2020.
- [5] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," 2018.
- [6] A. Thariq and R. Y. Bakti, "Sistem Deteksi Masker dengan Metode Haar Cascade pada Era New Normal COVID-19," *J. Sist. dan Teknol. Inf.*, vol. 9, no. 2, p. 241, 2021.
- [7] Syarif, "Deteksi Kedipan Mata Dengan Haar Cascade Classifier Dan Contour Untuk Password Login," *Techno.com*, vol. 14, no. 4, pp. 242–249, 2017.
- [8] A. R. Syafira and G. Ariyanto, "Sistem Deteksi Wajah dengan Modifikasi Metode Viola Jones," *Emit. J. Tek. Elektro*, vol. 17, no. 1, pp. 26–33, 2017.
- [9] A. Byna and M. Basit, "Penerapan Metode Adaboost Untuk Mengoptimasi Prediksi Penyakit Stroke Dengan Algoritma Naïve Bayes," *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 9, no. 3, pp. 407–411, 2020.
- [10] M. F. Sitorus, R. Fatharani, N. Fadhillah, T. Informatika, F. Teknik, and U. Samudra, "Sistem Deteksi Multi Wajah Menggunakan Metode Haar Cascade Classifier," vol. 01, no. 01, 2020.
- [11] Y. Achmad, R. C. Wihandika, and C. Dewi, "Klasifikasi emosi berdasarkan ciri wajah wenggunakan convolutional neural network," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 11, pp. 10595–10604, 2019.
- [12] A. Hibatullah and I. Maliki, "Penerapan Metode Convolutional Neural Network Pada Pengenalan Pola Citra Sandi Rumput," pp. 1–8, 2019.
- [13] M. R. Alwanda, R. P. K. Ramadhan, and D. Alamsyah, "Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle," *J. Algoritm.*, vol. 1, no. 1, pp. 45–56, 2020.
- [14] N. Nafi'iyah, "Algoritma Kohonen dalam Mengubah Citra Graylevel Menjadi Citra Biner," *J. Ilm. Teknol. Inf. Asia*, vol. 9, no. 2, pp. 49–55, 2015.
- [15] E. Ambar Pambudi and F. Abdul Rosyid, "Penerapan Segmentasi Citra Dengan Metode Threshold Niblack Pada Daun Janda Bolong (*Monstera Adansonii*)," no. November, pp. 76–86, 2021.