

Sistem Penghitung Jumlah Pengunjung Restoran Menggunakan Kamera Berbasis *Single Shot Detector* (Ssd)

Yosia Pradeska Admaja¹⁾ Heri Pratikno²⁾ Weny Indah Kusumawati³⁾

¹ Program Studi/Jurusan Teknik Komputer, Universitas Dinamika

² Jl. Raya Kedung Baruk 98 Surabaya, 60298

Email: 17410200022@dinamika.ac.id, heri@dinamika.ac.id, weny@dinamika.ac.id

Abstrak: Wabah pandemi virus *Covid-19* pada awal tahun 2020 menuntut seluruh lapisan masyarakat dan pemerintah untuk menerapkan pola hidup *new normal* demi mencegah penyebaran virus yang semakin meluas. Langkah pencegahan yang dilakukan pemerintah adalah melalui sosialisasi pola hidup *new normal*, yaitu: mewajibkan masyarakat selalu mencuci tangan, wajib menggunakan masker, mengurangi mobilitas, menjaga jarak aman kira-kira 1 meter, serta membatasi jumlah pengunjung pada pusat perbelanjaan dan pusat keramaian, seperti restoran. Berdasarkan level PPKM-nya dilakukan pembatasan jumlah pengunjung dan kapasitas maksimal orang dalam suatu ruangan. Dalam upaya pencegahan virus *Covid-19* tersebut, peneliti membuat sebuah sistem penghitung jumlah pengunjung yang masuk, keluar dan yang masih berada dalam suatu ruangan pada restoran. Apabila jumlah pengunjung melebihi kapasitas ruangan yang telah ditentukan maka *buzzer* berbunyi sebagai tanda bahwa kapasitas ruangan *overload*. Metode yang digunakan pada aplikasi ini adalah *Single Shot Detector* (SSD) dengan integrasi *dataset* dari *MobileNet-SSD*, sehingga mampu mendeteksi *object* manusia secara akurat. Berdasarkan hasil pengujian dari rekaman video perorangan yang telah dilakukan, tingkat akurasi deteksi untuk *object* berjalan adalah 100%, *object* berjalan cepat sebesar 90%, dan tingkat akurasi *object* berlari adalah 50%. Sedangkan pada pengujian secara *realtime* di pintu masuk restoran, didapatkan hasil akurasi 86% untuk penghitungan jumlah pengunjung masuk, 66% untuk akurasi penghitungan jumlah pengunjung keluar, dan akurasi sebesar 79% untuk jumlah pengunjung yang masih berada di dalam ruangan restoran.

Kata Kunci: Python, *Single Shot Detector* (SSD), *MobileNet*, Covid-19, protokol kesehatan, restoran.

Abstract: The outbreak of the Covid-19 virus pandemic in early 2020 required all levels of society and the government to implement a new normal lifestyle in order to prevent the spread of the virus from spreading. Prevention measures taken by the government are through socializing the new normal lifestyle, namely: requiring people to always wash their hands, must wear masks, reduce mobility, maintain a safe distance of about 1 meter, and limit the number of visitors to shopping centers and crowd centers, such as restaurants. Based on the PPKM level, there are restrictions on the number of visitors and the maximum capacity of people in a room. In an effort to prevent the Covid-19 virus, researchers created a system to calculate the number of visitors who entered, left and who were still in a room at the restaurant. If the number of visitors exceeds the specified room capacity, the buzzer sounds as a sign that the room capacity is overloaded. The method used in this application is *Single Shot Detector* (SSD) with dataset integration from *MobileNet-SSD*, so that it is able to detect human objects accurately. Based on the test results from individual video recordings that have been carried out, the detection accuracy rate for walking objects is 100%, objects run fast by 90%, and the accuracy rate for running objects is 50%. Meanwhile, in realtime testing at the restaurant entrance, the results obtained accuracy of 86% for calculating the number of visitors entering, 66% for the accuracy of calculating the number of visitors leaving, and an accuracy of 79% for the number of visitors who are still in the restaurant room.

Keywords: Python, *Single Shot Detector* (SSD), *MobileNet*, Covid-19, health protocols, restaurants.

PENDAHULUAN

Pada tanggal 12 Maret 2020, WHO menyebarkan pengumuman mengenai Covid-19 sebagai pandemik. Sampai dengan tanggal 29 Maret 2020, data kematian tercatat sebanyak 33.106 dari 634.835 kasus di seluruh dunia. Sedangkan di Indonesia sendiri terdapat 1.528 kasus positif Covid-19 dan yang meninggal ada 136 kasus [1]. Covid-19 merupakan penyakit baru yang disebabkan oleh Virus Corona, perlu mendapat perhatian karena penyebarannya relatif cepat, dan memiliki angka kematian yang tidak dapat diabaikan [2].

Covid-19 merupakan penyakit yang diakibatkan virus SARS-CoV-2. Pasien Covid-19 dengan komorbid memiliki tingkat resiko lebih tinggi [3]. Faktor resiko kematian paling tinggi pada Covid-19 adalah Laki-laki, berusia lanjut, diabetes, dan hipertensi [4]. Pada penelitian lain yang membahas tentang penyakit bawaan dipaparkan bahwa ada beberapa penyakit bawaan yang berhubungan erat dengan Covid-19 yaitu: penyakit jantung, hipertensi dan *diabetes mellitus*, hipertensi dan *diabetic ketoacidosis* [5].

Sejak virus mulai menyerang Indonesia, pemerintah telah menetapkan kebijakan protokol kesehatan yang direkomendasikan oleh WHO, yaitu menyediakan berbagai fasilitas medis bagi pasien yang terinfeksi virus corona, dan menerapkan *social distancing*, bahkan sampai menerapkan Pembatasan Sosial Berskala Besar (PSBB) di wilayah-wilayah Indonesia, terutama daerah rawan terinfeksi virus corona [6]. Arahan PSBB tersebut mengacu pada Undang-Undang Nomor 6 Tahun 2018 tentang Keekarantinaan Kesehatan, serta didukung oleh Peraturan Pemerintah tentang Pembatasan Sosial Berskala Besar (PP Nomor 21 Tahun 2020) dan Keputusan Presiden Nomor 11 Tahun 2020 tentang Kedaruratan Kesehatan [7].

Pemerintah menerapkan kebijakan pada masa dan menghimbau kepada seluruh masyarakat yang untuk patuh dan mengikuti aturan protokol kesehatan [8]. Ada 7 aturan yang harus dipatuhi pada masa *new normal*, yaitu: mencuci tangan, dilarang menyentuh area wajah, menutup mulut pada saat batuk dan bersin, menggunakan masker, menjaga jarak, melakukan isolasi mandiri bila terinfeksi, dan menjaga kesehatan, berjemur, makan makanan bergizi, dan berolahraga [9].

Beberapa negara termasuk Indonesia, saat ini sedang merencanakan tatanan *new normal* di saat pandemi Covid-19. Restoran adalah salah satu fasilitas umum yang harus dipersiapkan, karena mempunyai potensi menyebarkan penularan Covid-19 [10]. Pemilik usaha makanan atau restoran harus memperoleh kepercayaan konsumen dengan memberikan kualitas produk yang sehat, higienis, dan aman. Selain itu juga menjalankan protokol kesehatan, meningkatkan penjualan *online*, *take away*, dan *delivery order*, serta menciptakan ragam produk makanan sehat yang berupa *frozen food* [11].

Berdasarkan adanya pembatasan jumlah pengunjung yang ditetapkan oleh pemerintah Indonesia, maka pada penelitian ini membuat sebuah sistem yang mampu menghitung jumlah pengunjung

pada suatu restoran, baik pengunjung yang memasuki restoran, pengunjung yang sedang di dalam restoran, maupun pengunjung yang sudah keluar dari restoran. Penelitian ini menggunakan metode *Single Shot Detector* (SSD) yang terintegrasi dengan *dataset* dari *MobleNet-SSD*. Integrasi tersebut mampu untuk mendeteksi *object* manusia dari data *input* yang berasal dari hasil rekaman kamera, sehingga dapat dihitung jumlah pengunjung yang masuk, keluar, dan masih berada di dalam ruangan. Hasil penghitungan ditampilkan pada layar monitor PC atau TV yang berfungsi sebagai *output*.

Single Shot Detector (SSD) adalah algoritma yang dapat mendeteksi objek dalam sebuah gambar atau video dan memiliki akurasi yang lebih tinggi serta memproses gambar yang ditangkap oleh kamera lebih cepat [12]. (Sik-Ho Tsang, 2018). Metode SSD memiliki tingkat akurasi yang tinggi dibandingkan dengan metode sekelasnya seperti YOLO dan RCNN. Metode ini mampu memberikan akurasi lebih akurat dari YOLO dan kecepatan pemrosesan lebih cepat dari RCNN. Kelemahan dari metode ini adalah kemampuannya masih dirasa kurang dalam mendeteksi *object* dengan ukuran kecil [13].

Terdapat beberapa penelitian lain yang menggunakan permasalahan penghitungan *object*. Salah satunya adalah “Perancangan Program Pendeteksi Dan Pengklasifikasi Jenis Kendaraan Dengan Metode *Convolutional Neural Network* (CNN) *Deep Learning*”, yang membahas mengenai proses deteksi, klasifikasi, dan menghitung kendaraan menggunakan metode *Convolutional Neural Network* (CNN) *Deep Learning* dengan algoritma *You Only Look Once* (YOLO). Hasilnya masih memiliki kekurangan yaitu akurasi belum maksimal ketika mendeteksi *object* di jalan yang padat, dan belum ada fitur untuk *reporting* hasil proses identifikasi *object* kedalam bentuk file PDF atau *spreadsheet* [14].

Penelitian yang lain adalah “*Implementation Of Tensorflow In The CCTV-Based People Counter Application at PT Matahari Department Store, Tbk*”. Penelitian ini membahas tentang digunakan untuk menghitung keluar masuknya orang-orang di toko PT. Matahari Department Store menggunakan algoritma *TensorFlow*. Namun penelitian tersebut memiliki beberapa kekurangan yaitu penempatan kamera yang belum tepat, serta akurasi pengenalan *object* masih belum akurat, penggunaan kamera yang memiliki resolusi kecil, dan belum ada fitur untuk menuliskan hasil deteksi ke dalam *file* dengan format lain (PDF atau *spreadsheet*) [15].

Untuk aplikasi dalam kehidupan yang nyata cukup sulit untuk menentukan atau membandingkan model detektor obyek mana yang terbaik antara YOLO, SSD dan Faster-RCNN. Ada beberapa faktor yang mempengaruhi kinerja dari ketiga model detektor obyek tersebut diatas, diantaranya: banyaknya fitur, resolusi citra, jumlah dataset dan data augmentation.

Pada penelitian [16] telah melakukan komparasi 6 model detektor obyek, diantaranya: YOLO, SSD dan

Faster-RCNN dengan kesimpulan sebagai berikut: YOLO mempunyai waktu komputasi yang tercepat dari ketiga model tersebut tetapi akurasi lebih rendah daripada SSD [17]. Faster-RCNN mempunyai akurasi tinggi tetapi mempunyai kecepatan komputasi terendah, sedangkan SSD yang digunakan pada penelitian ini mempunyai kelebihan, yaitu: keseimbangan yang baik antara kecepatan komputasi dan akurasi, disamping itu SSD paling memungkinkan dapat berjalan pada sistem komputasi dengan kebutuhan spesifikasi perangkat keras yang paling minimalis daripada YOLO [18,19] dan Faster-RCNN.

Kontribusi utama dalam penelitian ini adalah memberikan solusi untuk monitoring jumlah pengunjung serta pemantauan kapasitas ruangan di suatu restoran, sehingga tidak melanggar aturan yang telah ditetapkan oleh pemerintah daerah terkait dengan pembatasan jumlah pengunjung yang diperbolehkan masuk. Selain itu penelitian menggunakan metode SSD mampu mendeteksi pengunjung restoran dengan tingkat akurasi yang tinggi.

METODE

Penelitian ini mempunyai tujuan untuk membuat sebuah sistem yang dapat menghitung jumlah pengunjung masuk, keluar, dan jumlah pengunjung yang masih berada di dalam suatu ruangan restoran.

1. Perancangan Perangkat Keras (Hardware)



Gambar 1. Model perancangan

1. *Input*: data yang digunakan untuk pengujian aplikasi berupa hasil rekaman video dari kamera *web* serta proses deteksi obyek secara *realtime* menggunakan *wireless* IP Camera.
2. *Computer Vision*: pemrosesan video *input* menggunakan bahasa pemrograman *python* dengan penerapan metode *Single Shot Detector (SSD)* serta penggabungan dataset hasil proses *training* dari *pre-trained network MobileNet-SSD* untuk seleksi dan penghitungan *object* manusia.
3. *Output*: Hasil pemrosesan akan ditampilkan pada layar monitor atau TV yang digunakan dengan bentuk tampilan *output* berupa waktu (jam, menit, detik, tanggal, bulan, tahun) serta jumlah hasil penghitungan jumlah pengunjung yang masuk, keluar, dan di dalam area restoran.

2. Perancangan Perangkat Lunak (Software)

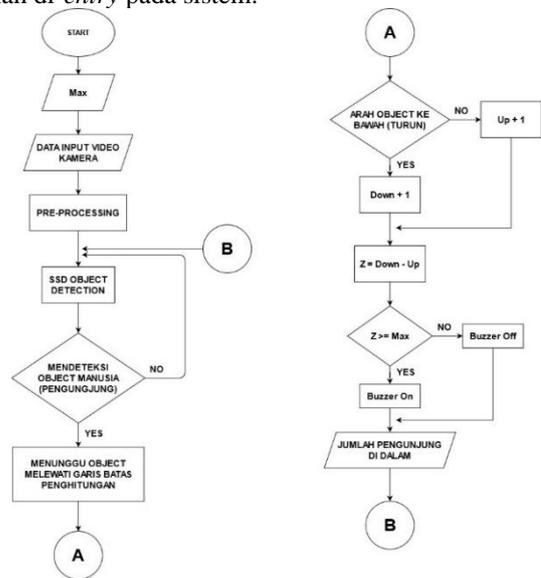
Sistem diawali dengan memasukkan jumlah kapasitas maksimal dari suatu ruangan, yang kemudian ditampung pada variabel *Max*. Kemudian sistem berlanjut ke proses pembacaan data *input* hasil rekaman

peneliti dan mengubah video menjadi *greyscale* serta memperkecil resolusi videonya dengan tujuan agar lebih mudah dalam proses analisis dan deteksi *object* manusia.

Setelah proses sebelumnya selesai, dilanjutkan ke proses *tracking object* dengan tujuan untuk mengetahui arah pergerakan *object* manusia, sehingga dapat ditentukan apakah *object* tersebut sedang berjalan masuk atau keluar. Penentuan kriteria *object* yang masuk adalah ketika *object* tersebut berjalan turun dan melewati garis batas penghitungan, untuk kriteria *object* yang keluar adalah ketika *object* tersebut berjalan naik dan melewati garis batas penghitungan.

Data jumlah *object* yang masuk tersimpan pada variabel *Down*, dan data jumlah *object* yang keluar akan tersimpan pada variabel *Up*. Dari data jumlah tersebut, dapat dihitung jumlah pengunjung yang masih berada di dalam ruangan dengan cara mengurangi jumlah variabel *Down* dengan jumlah variabel *Up*.

Sistem ini juga dilengkapi dengan fitur notifikasi ketika jumlah pengunjung yang berada di dalam sudah mencapai batas maksimal sesuai yang telah ditentukan sebelumnya pada variabel *Max*. Notifikasi memunculkan pesan *text* pada layar seperti pada gambar 3 yang menandakan bahwa kapasitas sudah mencapai batas maksimal, dan alarm dari *buzzer* berbunyi pada saat kapasitas ruangan melebihi batas maksimal nilai yang telah di-*entry* pada sistem.



Gambar 2. Algoritma sistem

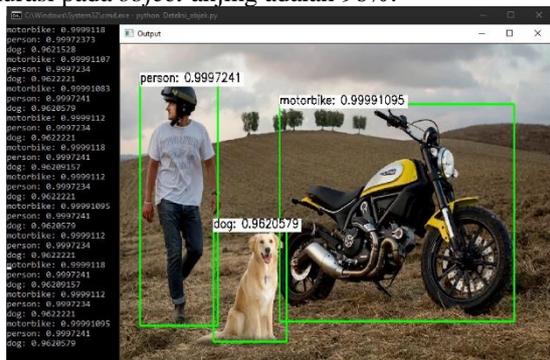


Gambar 3. Tampilan notifikasi

HASIL DAN PEMBAHASAN

1. Pengujian Deteksi Object

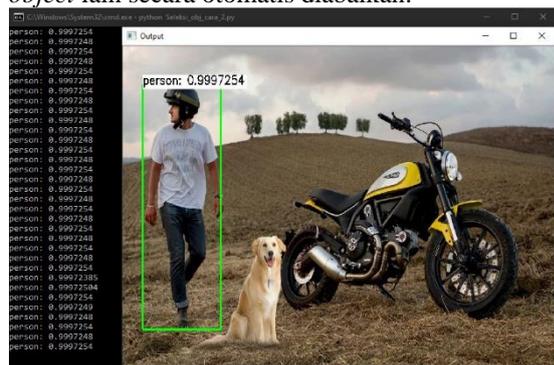
Tujuan dari pengujian ini adalah mengetahui kemampuan sistem dalam mendeteksi seluruh *object* yang ada pada suatu video maupun *object* secara *realtime*. Dengan menggunakan algoritma *Single Shot Detector* (SSD) yang terintegrasi dengan *dataset* dari *MobileNetSSD* dapat disimpulkan bahwa nilai akurasi deteksi *object* ini mampu membedakan jenis *object* dengan akurat sesuai dengan Gambar 4 nilai akurasi deteksi *object* manusia adalah 100%, nilai akurasi deteksi *object* sepeda motor adalah 100%, dan nilai akurasi pada *object* anjing adalah 96%.



Gambar 4. Deteksi multi-object

2. Seleksi Object Manusia

Seleksi *object* manusia bertujuan untuk mempermudah proses penghitungan jumlah pengunjung, karena yang dideteksi dan dihitung adalah *object* manusia, sehingga nantinya tidak ada interferensi dari *object* lain seperti hewan atau kendaraan yang melintas. Pada Gambar 5 menandakan bahwa proses seleksi *object* manusia sudah berhasil dengan akurasi 100%, sehingga *object* lain secara otomatis diabaikan.



Gambar 5. Deteksi object manusia

3. Pengujian Akurasi Person Detection

Tujuan dari proses ini adalah untuk memastikan nilai akurasi terhadap *object* manusia yang berada dalam video dengan kondisi *object* sedang berjalan, berjalan cepat, dan berlari. Kecepatan normal orang berjalan kaki apabila tidak ada halangan sekitar 4,8 Km per jam atau 79,25 m per menit, akan meningkat sedikit untuk laki-laki serta sebaliknya bagi perempuan. Jika jalannya menanjak atau adanya halangan, misalnya ada

kerumunan orang, tanda lalu lintas dan sebagainya maka dapat memperlambat sekitar 25%.

Kecepatan lari rata-rata seorang wanita adalah 10,21 per mil atau 16,428 Km per jam, sedangkan kecepatan lari rata-rata untuk pria adalah 9,03 per mil atau sekitar 14,529 Km per jam. Sedangkan definisi berjalan cepat adalah kondisi kecepatan rata-rata setengah berlari [20].

Tabel 1. Hasil pengujian akurasi *person detection*

Data Ke-	Kondisi Object	Tinggi Badan	Berat Badan	Hasil	
				Terdeteksi	Tidak
1	Jalan	175 Cm	90 Kg	✓	
	Jalan cepat Berlari			✓	✓
2	Jalan	178 Cm	88 Kg	✓	
	Jalan cepat Berlari			✓	✓
3	Jalan	174 Cm	66 Kg	✓	
	Jalan cepat Berlari			✓	✓
4	Jalan	173 Cm	60 Kg	✓	
	Jalan cepat Berlari			✓	✓
5	Jalan	155 Cm	88 Kg	✓	
	Jalan cepat Berlari			✓	✓
6	Jalan	150 Cm	62 Kg	✓	
	Jalan cepat Berlari			✓	✓
7	Jalan	170 Cm	87 Kg	✓	
	Jalan cepat Berlari			✓	✓
8	Jalan	176 Cm	81 Kg	✓	
	Jalan cepat Berlari			✓	✓
9	Jalan	173 Cm	69 Kg	✓	
	Jalan cepat Berlari			✓	✓
10	Jalan	172 Cm	64 Kg	✓	
	Jalan cepat Berlari			✓	✓
11	Jalan	157 Cm	85 Kg	✓	
	Jalan cepat Berlari			✓	✓

Dari hasil pengujian tersebut, dapat ditentukan nilai akurasi untuk setiap kondisi yaitu: berjalan, jalan cepat, dan berlari dengan menggunakan rumus (1).

$$\text{Akurasi} = \left(\frac{\text{terdeteksi}}{\text{banyak data percobaan}} * 100\% \right) \quad (1)$$

$$\begin{aligned} \text{Akurasi Object Berjalan} &= \left(\frac{10}{10} * 100\% \right) \\ &= 100\% \end{aligned}$$

$$\begin{aligned} \text{Akurasi Object Berjalan Cepat} &= \left(\frac{9}{10} * 100\% \right) \\ &= 90\% \end{aligned}$$

$$\begin{aligned} \text{Akurasi Object Berlari} &= \left(\frac{5}{10} * 100\% \right) \\ &= 50\% \end{aligned}$$

Berdasarkan hasil dari Tabel 1 serta hasil perhitungan akurasi yang dilakukan menggunakan rumus (1) dapat dijelaskan sebagai berikut: untuk mendeteksi *object* orang dengan kondisi berjalan maka SSD dapat mendeteksi pergerakannya dengan tepat 100%, sedangkan untuk proses deteksi pergerakan orang dengan kondisi berjalan cepat atau bisa dikatakan setengah berlari maka kemampuan deteksi dari metode SSD akurasi 90%. Adapun pada saat *object* berlari akurasi deteksi dari SSD turun menjadi 50%, hal ini berarti proses deteksi *object* melalui *bounding box* pada beberapa *frame* telah terjadi kehilangan *center point* sebagai ID untuk proses seleksi dan penghitungan sehingga pergerakan dari *object* tidak terdeteksi maka sudah barang tentu hal ini akan berdampak pada hasil akurasi.

4. Pengujian Akurasi Penghitungan Jumlah Pengunjung

Tujuan dari pengujian ini adalah untuk memastikan penghitungan dapat berfungsi dengan baik. Langkah awal adalah menghitung manual jumlah pengunjung yang masuk, keluar dan berada di dalam ruangan, kemudian membandingkan dengan hasil penghitungan dari aplikasi.

Berdasarkan hasil percobaan yang telah dilakukan secara empiris penghitungan jumlah pengunjung yang masuk maupun keluar pintu restoran yang hampir secara bersamaan masih tetap dapat dihitung dengan tepat karena pada saat obyek masuk dalam area pantauan kamera langsung secara otomatis diberi nomer ID pada titik tengah *bounding box* (*anchor point*), terkecuali para pengunjung tersebut saling berhimpitan kearah samping atau depan-belakang maka kondisi tersebut akan dihitung sebagai satu *object* saja.

Percobaan dilakukan dengan menghitung selisih penghitungan antara aplikasi dengan penghitungan manual, setiap percobaan dilakukan dengan durasi 1 menit dengan total akumulasi setiap menitnya dari 1 menit sampai 10 menit.

Tabel 2. Pengujian akurasi penghitungan

Percobaan Ke-	Hasil Penghitungan						Selisih Penghitungan		
	Aplikasi			Manual			Penghitungan		
	Masuk	Keluar	Di dalam	Masuk	Keluar	Di dalam	Masuk	Keluar	Di dalam
1	7	1	6	8	1	7	1	0	1
2	13	1	12	13	2	11	0	1	1
3	21	1	20	23	1	22	2	0	2
4	29	1	28	31	1	30	2	0	2
5	36	3	33	38	3	35	2	0	2
6	46	5	41	51	4	47	5	1	4
7	51	8	43	58	4	54	7	4	11
8	58	9	49	66	4	62	8	5	13
9	62	9	53	71	5	66	9	4	13
10	67	9	58	78	5	73	11	4	15

Dari hasil pengujian yang dilakukan, masih ditemukan adanya selisih, selanjutnya adalah menghitung nilai akurasi penghitungan aplikasi menggunakan rumus (2).

$$Akurasi = 100\% - \left(\frac{selisih}{hitungmanual} * 100\% \right) \quad (2)$$

$$\begin{aligned} Object \text{ Masuk} &= 100\% - \left(\frac{selisih}{hitungmanual} * 100\% \right) \\ &= 100\% - \left(\frac{11}{78} * 100\% \right) \\ &= 100\% - 14\% \\ &= 86\% \end{aligned}$$

$$\begin{aligned} Object \text{ Keluar} &= 100\% - \left(\frac{selisih}{hitungmanual} * 100\% \right) \\ &= 100\% - \left(\frac{4}{9} * 100\% \right) \\ &= 100\% - 44\% \\ &= 66\% \end{aligned}$$

$$\begin{aligned} Object \text{ di Dalam} &= 100\% - \left(\frac{selisih}{hitungmanual} * 100\% \right) \\ &= 100\% - \left(\frac{15}{73} * 100\% \right) \\ &= 100\% - 21\% \\ &= 79\% \end{aligned}$$

5. Pengujian Notifikasi Kapasitas Maksimal

Hasil percobaan pada Tabel 3 dapat diketahui tingkat akurasi notifikasi ketika pengunjung mencapai batas maksimal adalah 100%. Nilai akurasi tersebut dihitung dengan rumus (3).

Tabel 3. Pengujian melebihi kapasitas ruangan yang telah ditentukan

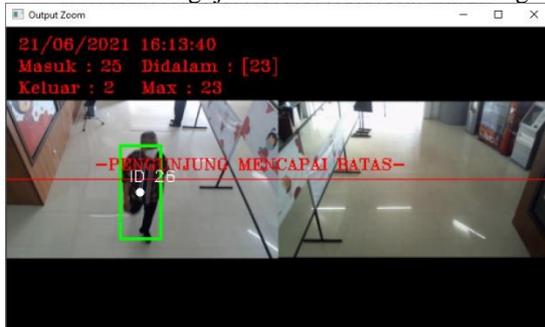
Percobaan Ke-	Jumlah Pengunjung		Notifikasi	
	Ter-deteksi	Batas Max	Text Pesan	Suara Buzzer
1	1 Orang	1 Orang	Tampil	Aktif
2	2 Orang	2 Orang	Tampil	Aktif
3	3 Orang	3 Orang	Tampil	Aktif
4	4 Orang	4 Orang	Tampil	Aktif
5	5 Orang	5 Orang	Tampil	Aktif
6	6 Orang	6 Orang	Tampil	Aktif
7	7 Orang	7 Orang	Tampil	Aktif
8	8 Orang	8 Orang	Tampil	Aktif
9	9 Orang	9 Orang	Tampil	Aktif
10	10 Orang	10 Orang	Tampil	Aktif
11	11 Orang	11 Orang	Tampil	Aktif
12	12 Orang	12 Orang	Tampil	Aktif
13	13 Orang	13 Orang	Tampil	Aktif
14	14 Orang	14 Orang	Tampil	Aktif
15	15 Orang	15 Orang	Tampil	Aktif
16	16 Orang	16 Orang	Tampil	Aktif
17	17 Orang	17 Orang	Tampil	Aktif
18	18 Orang	18 Orang	Tampil	Aktif
19	19 Orang	19 Orang	Tampil	Aktif
20	20 Orang	20 Orang	Tampil	Aktif
21	21 Orang	21 Orang	Tampil	Aktif
22	22 Orang	22 Orang	Tampil	Aktif
23	23 Orang	23 Orang	Tampil	Aktif
24	24 Orang	24 Orang	Tampil	Aktif
25	25 Orang	25 Orang	Tampil	Aktif
26	26 Orang	26 Orang	Tampil	Aktif
27	27 Orang	27 Orang	Tampil	Aktif
28	28 Orang	28 Orang	Tampil	Aktif
29	29 Orang	29 Orang	Tampil	Aktif

Perco- baan Ke-	Jumlah Pengunjung		Notifikasi	
	Ter- deteksi	Batas Max	Text Pesan	Suara Buzzer
30	30 Orang	30 Orang	Tampil	Aktif

$$\begin{aligned}
 \text{Akurasi} &= \left(\frac{\text{berhasil}}{\text{total percobaan}} * 100\% \right) \quad (3) \\
 &= \left(\frac{30}{30} * 100\% \right) \\
 &= 100\%
 \end{aligned}$$



Gambar 6. Pengujian batas maksimal 15 orang



Gambar 7. Pengujian batas maksimal 23 orang



Gambar 8. Pengujian batas maksimal 30 orang

Gambar 6, gambar 7, dan gambar 8 merupakan hasil pengujian notifikasi ketika pengunjung mencapai batas maksimal.

6. Pengujian Data Reporting

Tujuan dari pengujian ini adalah untuk memastikan data yang tampil pada layar monitor pada waktu tertentu bernilai sama dengan hasil yang tersimpan pada file data reporting.

Tabel 4. Pengujian ketepatan data reporting

Percobaan Ke-	Penghitungan Aplikasi			Data Report		
	Masuk	Keluar	Di dalam	Masuk	Keluar	Di dalam
1	8	1	7	8	1	7
2	13	1	12	13	1	12
3	22	2	20	22	2	20
4	29	3	26	29	3	26
5	38	4	34	38	4	34
6	49	8	41	49	8	41
7	56	8	48	56	8	48
8	61	10	51	61	10	51
9	66	10	56	66	10	56
10	66	15	51	66	15	51
11	66	21	45	66	21	45
12	67	26	41	67	26	41
13	67	33	34	67	33	34
14	67	42	25	67	42	25
15	67	52	15	67	52	15
16	68	57	11	68	57	11
17	68	65	3	68	65	3
18	68	68	0	68	68	0

Contoh hasil pengujian data reporting tersaji pada Gambar 9 dan Gambar 10.



Gambar 9. Hasil penghitungan pada aplikasi

Sebagai contoh pembahasan dari hasil Tabel 4 terkait dengan pengujian ketepatan data reporting dapat dilihat pada percobaan pertama melalui perbandingan antara hasil penghitungan aplikasi dan hasil data yang telah tersimpan pada file .csv sebagai data reporting. Hasil perbandingannya antara jumlah pengunjung yang masuk, keluar dan yang masih berada dalam ruangan mempunyai kesamaan data 100% sebagaimana tampak pada Gambar 10.

Waktu	Masuk	Keluar	Di Dalam
8:18:36 PM		1	1
		2	
		3	
		4	
		5	
		6	
		7	
		8	

Gambar 10. Hasil penghitungan pada *file data reporting*

KESIMPULAN DAN SARAN

Berdasarkan hasil dari pengujian yang dilakukan pada penelitian ini didapatkan beberapa poin kesimpulan sebagai berikut:

1. Hasil aplikasi dalam mendeteksi *object* orang atau pengunjung satu per satu yang dikondisikan berjalan normal akurasi 100%, berjalan cepat akurasi deteksinya 90% sedangkan pada kondisi berlari akurasi hanya sebesar 50% obyek orang yang terdeteksi.
2. Akurasi aplikasi penghitung jumlah pengunjung ketika diterapkan dengan kondisi nyata dengan komputasi banyak orang pada pintu masuk restoran mendapatkan nilai akurasi sebesar 86% dalam menghitung pengunjung masuk, akurasi 66% dalam menghitung pengunjung keluar, dan 79% dalam menghitung jumlah pengunjung yang masih berada di dalam ruangan. Perbedaan persentase akurasi yang terjadi antara kesimpulan 1 dan kesimpulan 2 ini dikarenakan, pertama perbedaan komputasi jumlah banyaknya orang yang dideteksi, yang kedua pada kesimpulan pertama menggunakan kamera kabel dengan panjangnya 1,5 meter yang langsung terkoneksi ke komputer sedangkan kamera yang digunakan pada kesimpulan 2 menggunakan IP *Camera wireless* dengan jarak sekitar 5 meter. Dikarenakan menggunakan kamera IP *wireless* itulah transmisi data dalam bentuk video terdapat *delay* yang mana besar kecilnya *delay* dipengaruhi oleh kecepatan koneksi dan besarnya *bandwidth* jaringan *wifi* setempat.
3. Notifikasi berupa bunyi *buzzer* sudah berjalan sesuai dengan alur aplikasi ketika mengalami jumlah pengunjung yang berada di dalam ruangan melebihi batas maksimal yang sudah ditentukan.
4. Perbandingan antara nilai data hasil penghitungan pada aplikasi sudah sama dengan nilai data yang tersimpan pada *file data reporting (.csv)*.

Untuk penelitian selanjutnya terdapat beberapa saran yang dapat diimplementasikan untuk pengembangan aplikasi ini:

1. Meningkatkan efisiensi *reporting* dengan menjadikan data pada *file .csv* menjadi lebih sedikit yang mencatat data setiap menit atau jam, tidak mencatat setiap *frame* seperti saat ini.

2. Meningkatkan tampilan GUI (*Graphical User Interface*) agar lebih interaktif dan mudah dipahami oleh pengguna.
3. Posisi pemasangan kamera akan lebih maksimal ketika berada di ketinggian 3 meter sampai 4 meter dari lantai dengan sudut kamera 45° menghadap ke bawah, sehingga POV (*Point Of View*) kamera menjadi lebih luas.
4. Mengintegrasikan sistem penghitungan jumlah pengunjung dengan sistem pendeteksi suhu dan sistem pendeteksi jarak antar pengunjung (*physical distancing*) agar menjadi suatu sistem yang kompleks dan lebih efektif.

Ucapan Terimakasih

Tim mengucapkan terima kasih kepada Universitas Dinamika atas dukungan terhadap atas terbitnya Jurnal (JoTI). Selain itu tim mengucapkan terima kasih dan penghargaan diberikan kepada editor yang telah membahas makalah ini.

DAFTAR PUSTAKA

- [1] Susilo, Adityo., 2020, Coronavirus Disease 2019: Tinjauan Literatur Terkini, *Jurnal Penyakit Dalam Indonesia*, vol 7, hal 45-67. DOI: 10.7454/jpdi.v7i1.415.
- [2] Handayani, Diah., Rendra Hadi, Dwi., Isbaniah, Fathiyah., Burhan, Erlina., Agustin, Heidy., 2020, Penyakit Virus Corona 2019, *Jurnal Respirologi Indonesia*, vol 40, hal 119-129.
- [3] Alkautsar, Ahmad., 2021, Hubungan Penyakit Komorbid Dengan Tingkat Keparahan Pasien Covid-19, *Jurnal Medika Utama*, vol 3, hal 1488-1494.
- [4] Muhammad Ali Satria, Raden., Varia Tutupoho, Resty., Chalidyanto, Djazuly., 2020, Analisis Faktor Risiko Kematian Dengan Penyakit Komorbid COVID-19, *Jurnal Keperawatan Silampari*, vol 4, hal 48-55. DOI: 10.31539/jks.v4i1.1587.
- [5] Parwanto, Edy., 2021, Virus Corona (SARS-Cov-2) Penyebab COVID-19 Kini Telah Bermutasi, *Jurnal Biomedika dan Kesehatan*, vol 3, hal 47-49. DOI: 10.18051/JBiomedKes.2021.v4.47-49.
- [6] Hanifa, Nurul., Wajuba Perdini Fisabilillah, Ladi., 2021, Peran Dan Kebijakan Pemerintah Indonesia Di Masa Pandemi COVID-19, *Welfare: Jurnal Ilmu Ekonomi*, vol 2, hal 9-19.
- [7] Wiryawan, I Wayan., 2020, Kebijakan Pemerintah Dalam Penanganan Pandemi Virus Corona Disease 2019 (COVID-19) Di Indonesia, *Prosiding Webinar Nasional Universitas Mahasaraswati Denpasar*, hal 179-188.
- [8] Firmansyah, Yudi., Kardina, Fani., 2020, Pengaruh New Normal Ditengah Pandemi Covid-19 Terhadap Pengelolaan Sekolah Dan Peserta Didik, *Jurnal Buana Ilmu*, vol 4, hal 99-112. DOI: 10.36805/bi.v4i2.1107.
- [9] Muhyiddin, 2020, Covid-19, New Normal dan Perencanaan Pembangunan di Indonesia, *The*

- Indonesian Journal of Development Planning, vol 4, hal 240-252.
- [10] Suci Amelia, Decha., Suwarni, Linda., Selviana., Mawardi., 2020, Kesiapan Rumah Makan di Era New Normal, *Jurnal Ilmu Kesehatan Masyarakat*, vol 4, hal 216-221. DOI: 10.33221/jikm.v9i04.769.
- [11] Ezizwita., Sukma, Tri., 2021, Dampak Pandemi Covid-19 Terhadap Bisnis Kuliner dan Strategi Beradaptasi di Era New Normal, *Jurnal Ekonomi dan Bisnis Dharma Andalas*, vol 23, hal 51-63.
- [12] Sik-Ho Tsang., 2018, <https://towardsdatascience.com>. Retrieved 02 20, 2021, from <https://towardsdatascience.com/review-ssd-single-shot-detector-object-detection-851a94607d11>.
- [13] Chen Z., K. R., 2019, Real Time Object Detection, Tracking, and Distance and Motion Estimation based on Deep Learning: Application to Smart Mobility, *Proceedings of the 2019 Eighth International Conference on Emerging Security Technologies (EST)*, hal 1-6.
- [14] R. G. Fajri, I. S., 2021, Perancangan Program Pendeteksi Dan Pengklasifikasi Jenis Kendaraan Dengan Metode Convolutional Neural Network (CNN) Deep Learning, *Ransient: Jurnal Ilmiah Teknik Elektro*, vol 1, hal 97-106. DOI: 10.14710/transient.1.1.%25p.
- [15] Kusno Prasetya, A. A., 2020, Implementation of Tensorflow in the CCTV-Based People Counter Application at PT Matahari Department Store, Tbk, *IAIC International Conferences*, vol 3, hal 38-44.
- [16] S. A. Sanchez., *et all.*, 2019, A Review: Comparison of Performance Metrics of Pretrained Models for Object Detection Using the TensorFlow Framework. IOP Conf. Series: Materials Science and Engineering 844 (2020) 012024, DOI: 10.1088/1757-899X/844/1/012024.
- [17] Sukusvieri, A., 2020, *Sistem keamanan menggunakan kamera berbasis wajah dengan metode Single Shot Detector (SSD)*. Tugas Akhir. Fakultas Teknologi Informatika. Universitas Dinamika. Surabaya.
- [18] Rifai, A., 2020, *Analisis perhitungan bibit ikan gurame menggunakan webcam dengan metode YOLO (You Only Look Once)*. Tugas Akhir. Fakultas Teknologi Informatika. Universitas Dinamika. Surabaya.
- [19] Hidayatulloh, M., S., 2021, *Sistem pengenalan wajah menggunakan metode YOLO (You Only Look Once)*. Tugas Akhir. Fakultas Teknologi Informatika. Universitas Dinamika. Surabaya.
- [20] Zhang K., *et all.*, 2005, Assessment of human locomotion by using an insole measurement system and artificial neural networks. ELSEVIER: Journal of Biomechanics 38 (2005) 2276-2287. USA.