

Rancang Bangun Aplikasi Deteksi Objek Untuk Menghitung Jumlah Pengunjung Restoran Berbasis Computer Vision

Ari Kurnia Ramadan, Sularso Budi Laksono

Jurusan Informatika Fakultas Teknik Universitas Persada Indonesia Y.A.I Jakarta

Email : arikrnrd11@gmail.com. sularso@upi-yai.ac.id

ABSTRAK

Mengetahui jumlah pasti orang-orang yang masuk atau keluar di sebuah restoran adalah salah satu cara untuk memperoleh data jumlah pengunjung. Data tersebut bisa digunakan untuk laporan perkembangan minat masyarakat terhadap restoran tersebut. Dimana, sebelumnya untuk mendapatkan data jumlah pengunjung ke suatu tempat biasanya didapat secara manual seperti mengisi buku tamu. Pihak restoran harus bisa dengan mudah mendapatkan informasi mengenai jumlah pengunjung setiap harinya dengan tujuan data-data tersebut bisa digunakan untuk laporan perkembangan minat masyarakat terhadap restoran tersebut. Dari data yang didapatkan, pihak operasional restoran juga bisa memanfaatkan data tersebut untuk evaluasi jika jumlah pengunjung rendah, begitupun sebaliknya. Oleh karena itu, dibutuhkan sebuah sistem untuk memudahkan pendeteksian objek untuk menghitung jumlah pengunjung yang dapat memudahkan pihak restoran mendapatkan data pengunjung. Perancangan aplikasi deteksi objek untuk menghitung jumlah pengunjung menggunakan OpenCV dan metode mobilenet-ssd untuk mendeteksi objek dan Laravel untuk *User Interface* yang berguna untuk menampilkan data pengunjung yang sudah terdeteksi dan terhitung oleh sistem.

Kata Kunci: *Open CV, Mobile Net SSD, Laravel, Python.*

ABSTRACT

Knowing the exact number of people entering or leaving a restaurant is one way to obtain data on the number of diners. This data can be used to report on the development of public interest in the restaurant. Previously, to get data on the number of diners, it was usually obtained manually by filling out a guest book. The restaurant must be able to easily get information about the number of diners each day with the aim that the data can be used for related report. The restaurant's operational division can also use the data for evaluation if the number is low, and vice versa. Therefore, a system is needed to facilitate object detection to count the number of diners which can make it easier for the restaurant to get the number of diners data. The design of an object detection application to count the number of diners uses OpenCV and the mobilenet-ssd method to detect objects and Laravel for the User Interface which is useful for displaying diners data that has been detected and counted by the system.

Keywords: *Open CV, Mobile Net SSD, Laravel, Python.*

1. Pendahuluan

Belakangan ini, banyak isu muncul seperti bagaimana memperkecil sebuah data, menafsirkan dan mengolahnya. Seringkali banyak cara yang ada tidak perlu diarahkan untuk meniru proses manusia secara langsung melainkan hanya perlu untuk ditingkatkan, seperti memprediksi harga crypto currency, menghitung jumlah pengunjung di sebuah restoran atau mengambil informasi dengan cepat. (Barber, 2012)

Dimasa sekarang, siapapun dapat membangun usaha yang menjanjikan terutama dibidang kuliner. Namun untuk mendapatkan

kepercayaan dan keuntungan dalam usaha terutama dibidang kuliner, pengusaha harus mementingkan kualitas dari pelayanan kedai atau restorannya disaat pengunjung sedang ramai berdatangan. Beragam upaya yang mengaitkan dengan teknologi telah banyak dilakukan guna menunjang kualitas pelayanan terhadap customer.

Pendeteksian dan penghitung objek manusia merupakan salah satu sistem yang diperlukan untuk meningkatkan perkembangan teknologi yang dapat dimanfaatkan dalam berbagai bidang kebutuhan. Secara umum cara kerja dari deteksi dan penghitung objek manusia ini mengadaptasi cara pengamatan apa yang ada

pada dunia nyata melalui indra pengelihatian. Dengan kata lain, pendeteksian objek manusia dengan menggunakan indra penglihatan manusia (human vision) yang saat ini dikenal dengan teknologi computer vision. (Solichin & Harjoko, 2013)

Minimnya informasi mengenai data pengunjung dapat berdampak pada kurang optimalnya prediksi penggunaan sumberdaya penunjang kebutuhan operasional. Oleh karena itu, pihak restoran harus bisa dengan mudah mendapatkan informasi mengenai jumlah pengunjung setiap harinya dengan tujuan data-data tersebut bisa digunakan untuk laporan perkembangan minat masyarakat terhadap restoran tersebut. Dari data yang didapatkan, pihak operasional restoran juga bisa memanfaatkan data tersebut untuk evaluasi jika jumlah pengunjung rendah. Begitupun sebaliknya, ketika minat masyarakat tinggi dan ramai pengunjung, pihak restoran yang terkait dengan operasional dan telah mengetahui rata-rata jumlah pengunjung bisa menyiapkan sumberdaya yang dapat mendukung proses operasional dan bisa mengoptimalkan operasional terkait dalam meningkatkan pelayanan yang lebih baik.

Untuk memudahkan pendeteksian objek untuk menghitung jumlah pengunjung restoran, penulis membuat “Aplikasi Deteksi Objek Untuk Menghitung Jumlah Pengunjung Restoran Berbasis Computer Vision”. Dalam penerapannya, untuk mendeteksi objek dan

menghitung jumlah pengunjung, video yang akan diproses diambil menggunakan kamera webcam atau smartphone dan dikirimkan ke dalam computer server yang menghubungkan semua komponen penunjang sistem pendeteksian dan perhitungan jumlah pengunjung.

Video yang sudah diterima computer server, kemudian diproses menggunakan sistem pendeteksian dan perhitungan yang sudah dibuat menggunakan bahasa python, yang menerapkan metode SSD (Single Shot Multibox Detector) untuk melakukan proses pendeteksian objek manusia yang tertangkap oleh kamera. Selanjutnya sistem akan menghitung jumlah objek manusia yang masuk ke dalam restoran. Berdasarkan dari latar belakang diatas maka rumusan masalah dari penulisan ini yaitu:

- a. Bagaimana membangun aplikasi deteksi objek untuk menghitung jumlah pengunjung berbasis *computer vision*?
- b. Bagaimana mengaplikasikan algoritma *single shot multibox detector*?
- c. Bagaimana menghitung jumlah objek manusia yang terdeteksi memasuki dan keluar restoran
- d. Bagaimana aplikasi ini dapat menjadi solusi untuk memudahkan dalam mendapatkan data atau informasi jumlah pengunjung restoran setiap harinya.

canggih modern. Tujuan dari kecerdasan buatan yaitu membuat komputer menjadi lebih cerdas dan membuat sistem menjadi lebih berguna.

Computer Vision

Computer Vision adalah suatu pembelajaran menganalisis suatu gambar atau video untuk memperoleh hasil sebagaimana yang bisa dilakukan manusia. Computer vision adalah salah satu cabang ilmu komputer dengan tujuan untuk memberikan kemampuan melihat suatu objek yang mempelajari sebuah citra berupa gambar atau video. Komputer akan menganalisa input citra yang diberikan kemudian komputer memberikan informasi yang didapatkan dari input yang diberikan.

2. Tinjauan Pustaka

Kecerdasan Buatan

Kecerdasan buatan merupakan salah satu bagian dari ilmu komputer yang membuat agar mesin komputer dapat melakukan seperti yang dilakukan oleh manusia. Pada awal diciptakan, komputer hanya diciptakan sebagai alat hitung saja. Namun seiring dengan perkembangan jaman, maka peran komputer semakin mendominasi kehidupan manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, tetapi diharapkan untuk dapat diberdayakan untuk mengerjakan segala sesuatu yang bisa dikerjakan manusia. (Dr. Hendra Jaya, 2018).

Kecerdasan buatan mengungguli robot fiksi ilmiah, keadalam non-fiksi ilmu komputer

Computer vision bekerja hampir sama dengan penglihatan manusia (human vision), kecuali manusia memiliki permulaan, penglihatan manusia memiliki keuntungan dari konteks masa hidup untuk melatih bagaimana membedakan objek, seberapa jauh mereka, apakah mereka bergerak dan apakah ada sesuatu yang salah dalam sebuah gambar. (Szeliski, 2010).

Open CV

OpenCV adalah library terbuka lintas platform yang menyediakan blok rancangan untuk eksperimen penglihatan pada komputer dan aplikasi. OpenCV menyediakan antarmuka tingkat tinggi untuk menangkap, memproses dan memperlihatkan data gambar. OpenCV banyak digunakan baik di akademis maupun industri.

Sistem

Sistem adalah satu kesatuan yang terdiri dari komponen-komponen atau subsistem yang tertata dengan teratur, saling interkasi, saling ketergantungan satu dengan yang lainnya, dan tidak dapat dipisahkan untuk mewujudkan suatu tujuan. (M.S, 2015).

Website

Website yaitu merupakan sebuah kumpulan halaman yang saling berhubungan yang digunakan untuk menampilkan sebuah informasi teks, gambar diam ataupun gerak, suara dan animasi atau gabungan dari semuanya, yang bersifat statis maupun dinamis yang membentuk suatu bangunan yang saling terkait.

PHP Hypertext Preprocessor

Hypertext Preprocessor atau biasa disebut dengan *PHP* adalah Bahasa pemrograman untuk membuat web server. *PHP* biasanya digunakan untuk membuat *website* yang bersifat dinamis. *Database* yang biasa digunakan dalam Bahasa pemrograman *PHP* adalah *MySQL*. Selain itu *PHP* juga mendukung *system oracle*, *Microsoft Access*, *Interbase*, *D-base*, dan lain-lain. (Sitinjak Daniel Dido Jantce TJ & Suwita, 2020). (Mediana & Nurhidayat, 2018)..

Database

Menurut (Connolly & Berg, 2015, p. 63) menjelaskan bahwa *database* adalah tempat penyimpanan data tunggal atau lebih yang berukuran besar dan dapat digunakan secara bersamaan dengan departemen atau *user* lain. Selain itu, *database* juga tidak hanya menyimpan data operasional suatu organisasi saja, tetapi *database* juga deskripsi dari suatu data.

Single Shot Multibox Detector

SSD dirancang untuk pendeteksian objek secara real-time. Perbedaan utama antara pelatihan detector klasik seperti R-CNN dan pelatihan SSD adalah bahwa detector objek klasik menggunakan proposal wilayah dan metode SSD menggunakan groundtruth yang harus ditetapkan untuk hasil yang ditentukan dalam rangkaian hasil detektor yang jelas.

Dalam mendeteksi objek, SSD menandai area bounding box yang diprediksi kepada koleksi default bounding box melalui berbagai skala dan rasio untuk setiap lokasi feature map. SSD membandingkan objek dengan default bounding box dengan berbagai rasio selama masa training. Setiap default box dengan IoU > 0.5 dikategorikan cocok. Metode SSD juga menggunakan sejumlah layer dalam berbagai skala yang mampu memberikan hasil terbaik terhadap objek yang terdeteksi. Dalam penelitian ini, arsitektur MobileNet digunakan sebagai feature extractor pada metode SSD.

Ketika gambar diteruskan kepada arsitektur MobileNet, SSD menggunakan 6 layer konvolusi ekstra. Tiga dari layer ekstra tersebut dapat menghasilkan enam prediksi untuk setiap sel. Metode SSD secara total mampu menghasilkan 8732 prediksi dengan memanfaatkan 6 layer tersebut. Layer ekstra tersebut juga menghasilkan feature maps dalam berbagai ukuran untuk mendeteksi objek dalam berbagai ukuran sehingga mampu memberikan akurasi yang lebih baik terhadap objek-objek yang memiliki ukuran berbeda dalam suatu gambar. (Liu et al., 2016).

3. Metodologi Penelitian Dan Analisis Perancangan

Metode Pengumpulan Data

a. Studi Pustaka

Pengumpulan data dengan metode studi pustaka dilakukan dengan mencari dan membaca data-data yang mendukung berupa buku-buku dan jurnal yang berkaitan dengan penulisan yang dilakukan.

b. Observasi

Pengumpulan data dengan metode observasi ini adalah melakukan suatu pengamatan secara langsung

Metode perancangan sistem

Metode perancangan sistem yang digunakan pada penulisan kali ini yaitu Metode *Waterfall*.

Berikut ini merupakan tahapan-tahapan pada proses metode waterfall, yaitu:

1. Communication
2. Planning
3. Modeling
4. Construction
5. Deployment

Analisis Kebutuhan

Saat melakukan pembuatan aplikasi atau proyek perangkat lunak, sangat diperlukan adanya analisis kebutuhan sistem untuk sistem yang akan dibuat, agar dalam pengerjaan pembuatan aplikasi dapat selesai dengan tepat waktu. Dengan melakukan analisis kebutuhan sistem dalam pembuatan aplikasi,

Analisis Kebutuhan Fungsional

Analisis fungsional dari sistem deteksi menjelaskan fitur-fitur dari aplikasi deteksi dan penghitung objek manusia :

- a. Sistem deteksi objek untuk menghitung jumlah pengunjung
- b. Menu home
- c. Menu data pengunjung

Analisis kebutuhan non fungsional

Analisis kebutuhan non fungsional ini dilakukan untuk menentukan kebutuhan pada spesifikasi sistem. Kebutuhan pada

spesifikasi ini meliputi analisis perangkat keras (hardware) dan perangkat lunak (software).

Kebutuhan Perangkat Keras

Berikut ini merupakan fungsi Perangkat keras yang digunakan dalam perancangan sebelum di implementasikan yaitu sebagai berikut:

1. Personal Computer
Spesifikasi yang digunakan adalah:
 - a. Processor AMD Radeon R7
 - b. SSD 240 GB
 - c. HDD 1 TB
2. Smartphone

Kebutuhan Perangkat Lunak

Berikut merupakan fungsi dari perangkat lunak yang digunakan dalam perancangan sistem yaitu sebagai berikut:

a. Apache

Apache merupakan sebuah aplikasi web server yang bertugas dalam menciptakan sebuah halaman website berdasarkan kode-kode program yang dibuat oleh programmer. Fungsinya ialah untuk mengakses sistem database dan mendukung sebuah halaman situs yang dihasilkan.

b. MySQL

MySQL merupakan sebuah aplikasi database yang menggunakan bahasa SQL (Structured Query Language). Fungsi dari MySQL sendiri ialah sebagai tempat atau wadah informasi guna mempermudah dalam mengolah data, mengedit data, dan menyimpan sebuah data pada sistem yang berjalan.

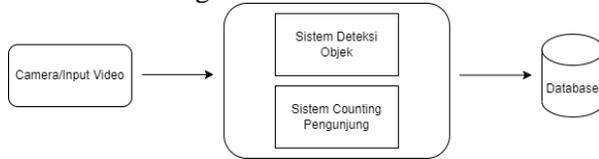
Perancangan Sistem

Dalam perancangan sistem berbasis computer vision ini menggunakan library OpenCV dan algoritma Single Shot Detector (SSD) untuk mendeteksi single objek. Dalam perancangan sistem juga terdapat tahap-tahap rancangan seperti use case diagram.

Perancangan Sistem Deteksi Objek Untuk Menghitung Jumlah Pengunjung

Pada tahap ini ada beberapa tahap untuk melakukan perancangan sistem deteksi objek

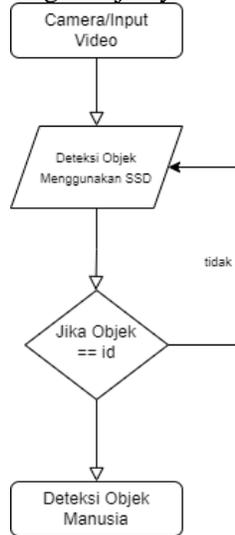
untuk menghitung jumlah pengunjung.
Berikut diagram sistem:



Gambar 1 Diagram Sistem

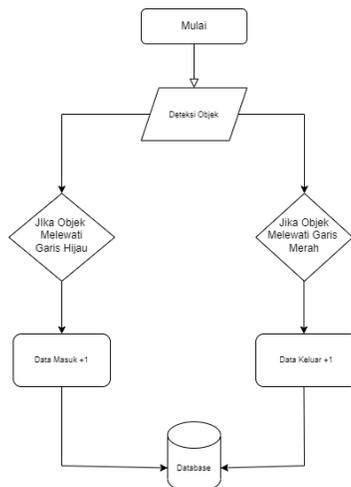
Sistem Deteksi Objek

Sistem deteksi ini menggunakan algoritma Single Shot Detector (SSD) untuk mendeteksi single objek yaitu objek manusia.



Gambar 2 Flowchart Deteksi Objek

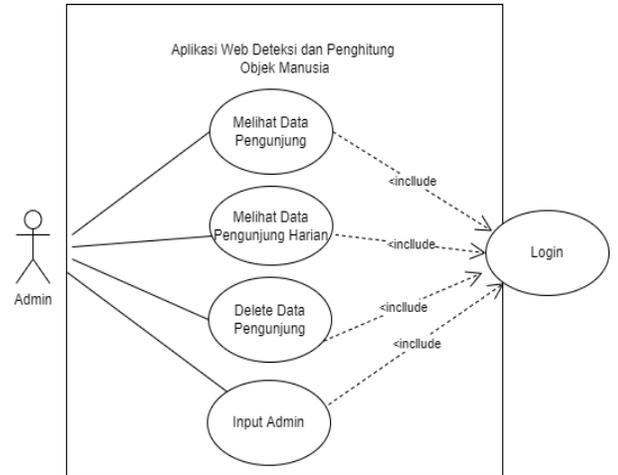
Sistem Penghitung Pengunjung



Gambar 3 Flowchart Penghitung Objek

Flowchart diatas menjelaskan bagaimana sistem penghitung pengunjung bekerja setelah objek dideteksi oleh single shot multibox detector (SSD).

Use Case Diagram

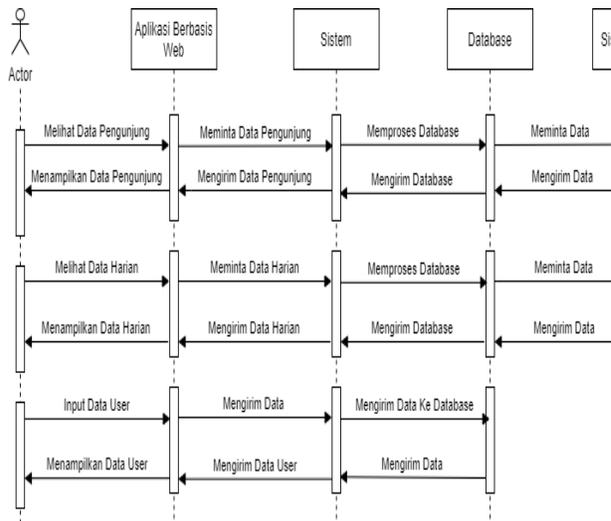


Gambar 4 Use Case Diagram

Pada gambar diatas merupakan rancangan use case diagram antara admin dengan program aplikasi. Pada rancangan tersebut terdapat admin yang dapat membuka aplikasi untuk melihat data pengunjung, data pengunjung harian, dan melakukan action delete data pengunjung. Pada data pengunjung harian, admin dapat melihat data pengunjung pada hari-hari sebelumnya.

Sequence Diagram

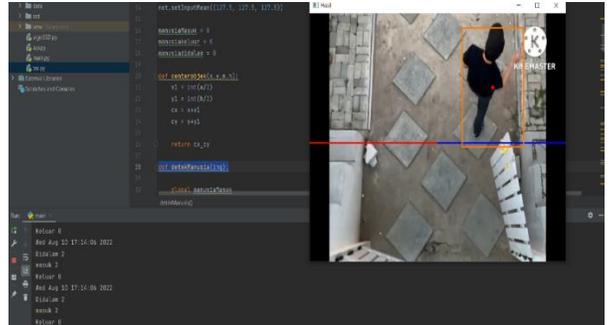
Pada alur squence diagram admin, admin atau pengguna mengakses aplikasi kemudian admin melakukan login untuk bisa mengakses halaman utama aplikasi berbasis web. Pada tampilan home aplikasi akan menampilkan data pengunjung yang sudah dikirim oleh sistem deteksi ke database dan sistem web akan meminta data ke database yang sudah menerima data dari sistem deteksi setelah itu sistem akan mengirim data ke aplikasi berbasis web dan akan menampilkan data ke admin atau pengguna. Pada menu input user, admin dapat menambahkan user yang bisa mengakses aplikasi berbasis web. Setelah menambahkan user, aplikasi akan mengirim data ke database.



Gambar 5 Sequence Diagram

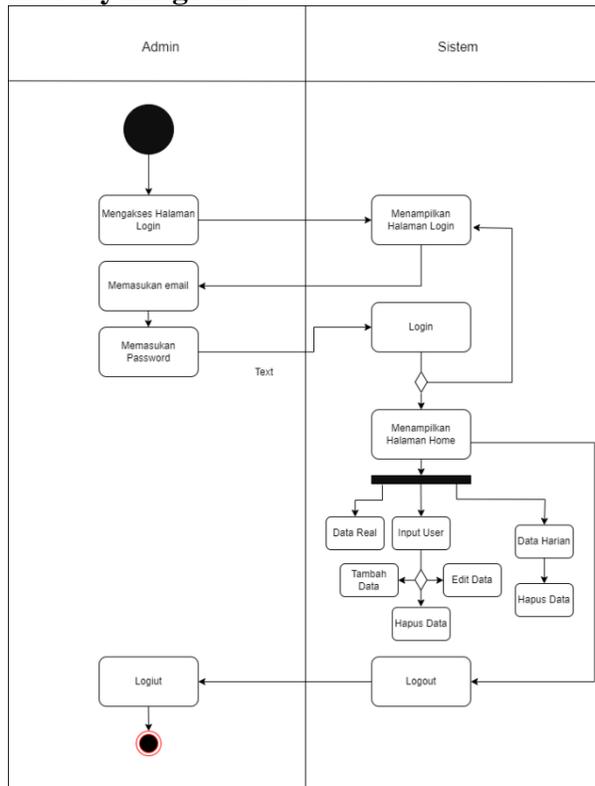
pada tahap implementasi akan dibahas hal-hal yang berhubungan langsung dengan perangkat lunak.

Hasil Implementasi Sistem Deteksi



Gambar 7 Hasil Sistem Deteksi

Activity Diagram



Gambar 6 Activity Diagram

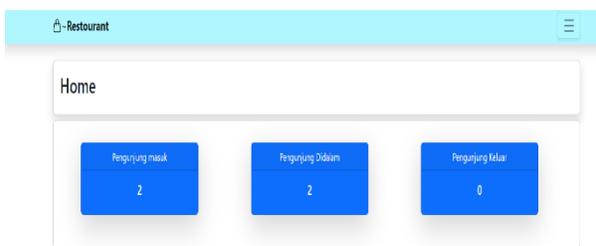
Pada Gambar 7 menampilkan hasil dari sistem deteksi objek dan penghitung manusia yang dimana, hanya objek manusia yang dapat terdeteksi oleh sistem dan menampilkan data pengunjung masuk, keluar dan di dalam.

Hasil Implementasi User Interface

4. Hasil Dan Pembahasan

Implementasi Sistem

Implementasi merupakan suatu tahapan dalam perancangan suatu perangkat lunak. Tahap implementasi dilakukan setelah tahap analisis, desain dan pengembangan sistem.



Gambar 8 Tampilan Halaman Home

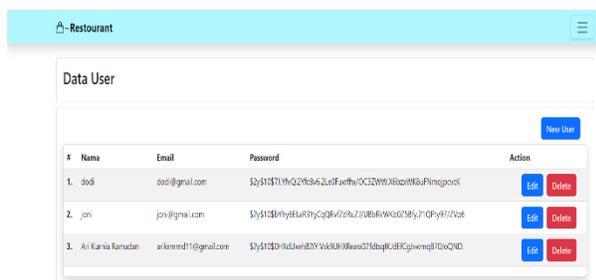


Gambar 10 Tampilan Halaman Data User



Gambar 11 Tampilan Halaman Edit Data User

Gambar 9 Tampilan Halaman Pengunjung Harian



Pengujian

Setelah dilakukannya implementasi, maka tahap selanjutnya adalah tahap pengujian fungsi program untuk aplikasi web maupun sistem deteksi objek dan penghitung manusia. Tahap ini dilakukan untuk memastikan bahwa fungsi yang dirancang dapat berjalan dengan baik dan sesuai dengan tujuan perancangan.

Tabel 1 Pengujian Aplikasi Berbasis Web

No	Test Case Description	Test Step	Expected Result	Result
1	Input email dan password kemudian klik	Akses Aplikasi berbasis web lalu melakukan login	Jika data valid, akan masuk ke halaman home dan halaman home	Pass

<i>No</i>	<i>Test Case Description</i>	<i>Test Step</i>	<i>Expected Result</i>	<i>Result</i>
	<i>button login</i>		menampilkan data pengunjung hari ini, jika tidak valid akan menampilkan Login gagal	
2	<i>Input nama, email, dan password pada tampilan register</i>	<i>Klik Button registrasi sekarang</i>	Berhasil menampilkan tampilan register dan setelah melakukan input data yang diperlukan data akan disimpan kedalam database dan akan kembali ke halaman home	Pass

<i>No</i>	<i>Test Case Description</i>	<i>Test Step</i>	<i>Expected Result</i>	<i>Result</i>
3	<i>Pilih menu data pengunjung harian pada sidebar</i>	<i>Klik menu data pengunjung pada sidebar</i>	Berhasil mengklik dan menampilkan halaman data pengunjung harian pada hari-hari sebelumnya	Pass
4	<i>Pilih menu user</i>	<i>Klik menu user pada sidebar</i>	Berhasil mengklik dan menampilkan data user yang dapat login ke aplikasi	Pass

<i>No</i>	<i>Test Case Description</i>	<i>Test Step</i>	<i>Expected Result</i>	<i>Result</i>
5	Melakukan edit pada data user	Klik button edit pada tampilan data user dan isi form dengan data yang diperlukan. Dan setelah mengisi data yg diperlukan klik button save user dengan harapan data yang diisi telah terupdate di tampilan data user maupun database user	Berhasil mengklik button edit dan menampilkan form untuk mengisi data yang diperlukan. Dan setelah mengisi data yg diperlukan klik button save user dengan harapan data yang diisi telah terupdate di tampilan data user maupun database user	Pass
6	Melakukan tambahkan	Klik button new user pada	Berhasil mengklik button new user dan	Pass

<i>No</i>	<i>Test Case Description</i>	<i>Test Step</i>	<i>Expected Result</i>	<i>Result</i>
	user pada fitur new user	halaman data user dan mengisi data yang diperlukan pada form yang disediakan pada tampilan new user	masuk ke halaman new user. Halaman new user akan menampilkan form untuk mengisi data yang diperlukan dengan harapan setelah form terisi akan menyimpan data ke database user	

No	Test Case Description	Test Step	Expected Result	Result
7	Melakukan aksi delete user pada	Klik button delete pada halaman data user	Berhasil klik button delete dengan harapan jika diklik button delete data user akan terhapus dari tampilan data user maupun database user	Pass

Pada tabel 1 diatas merupakan hasil pengujian aplikasi berbasis web secara manual.

Tabel 2 Hasil Pengujian Sistem Deteksi

No	Test Case Description	Expected Result	Pengamatan	Result
1	Deteksi Objek	Berhasil mendeteksi objek hanya manusia	Bisa dilihat pada gambar 4.14 sistem hanya mendeteksi manusia	Pass
2	Menghitung objek masuk yang terdeteksi oleh sistem	Berhasilan +1 untuk objek yang terdeteksi mengenai atau melewati garis biru	Sistem berhasil menghitung objek yang terdeteksi i ketika mengenai atau melewati garis biru	Pass
3	Menghitung objek masuk yang terdeteksi oleh sistem	Berhasilan +1 untuk objek yang terdeteksi mengenai atau	Sistem berhasil menghitung objek yang terdeteksi i ketika mengenai atau	Pass

<i>No</i>	<i>Test Case Description</i>	<i>Expected Result</i>	<i>Pengamatan</i>	<i>Result</i>
		<i>melewati garis biru</i>	melewati garis biru	
4	<i>Mengirim data pengunjung hari ini dan data pengunjung harian</i>	<i>Sistem dapat mengirim data pengunjung ke user interface atau database</i>	Sistem berhasil mengirim data ke database untuk ditampilkan oleh user interface	<i>Pas</i>

Berdasarkan pada tabel 2 Sistem deteksi dan penghitung objek manusia berhasil berjalan dengan cukup baik.

Kesimpulan

Berdasarkan rumusan masalah dan hasil pengujian yang telah dilakukan dalam penelitian tugas akhir yang berjudul rancang bangun aplikasi deteksi objek untuk menghitung jumlah pengunjung restoran mengambil kesimpulan sebagai berikut :

1. Perancangan sistem deteksi objek dan penghitung manusia menggunakan algoritma Single Shot Multibox Detector dengan mengimplementasikan kode-kode untuk mendeteksi objek berhasil berjalan dengan cukup baik. Hasil tersebut bisa dilihat pada gambar 7
2. Penerapan algoritma Single Shot Multibox Detektor untuk mendeteksi objek manusia

bisa dilihat pada sub bab Implementasi Kode Deteksi Objek Manusia.

3. Proses menghitung jumlah objek manusia yang terdeteksi memasuki atau keluar ruangan dapat dilihat pada sub bab Implementasi Kode Penghitung Objek Manusia dan hasil dan pengujian dapat dilihat pada Gambar 7
4. Aplikasi Deteksi Dan Penghitung Objek Manusia berhasil menampilkan data pengunjung hari ini dan data pengunjung harian bisa dilihat pada Gambar 8 dan 9

Daftar Pustaka

- Barber, D. (2012). Bayesian Reasoning and Machine Learning. *Bayesian Reasoning and Machine Learning*, 462–478.
<https://doi.org/10.1017/cbo9780511804779.026>
- Delisle, M. (n.d.). *Effective MySQL Management*.
- Dr. Hendra Jaya, S. P. . M. T. . dkk. (2018). *Buku Referensi - Kecerdasan Buatan*.
- Howse, J. (2013). OpenCV Computer Vision with Python. In *Cs_Python_in*. www.it-ebooks.info
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9905 LNCS, 21–37.
https://doi.org/10.1007/978-3-319-46448-0_2
- M.S, D. K. (2015). Konsep Dasar Sistem Informasi. *Konsep Dasar Sistem Informasi*, 1–36.
- McCool, S. (2012). Laravel Starter. In *Packtlib*.
<https://compeng.abu.edu.ng/dept/tutorial/lara>

- vel_starter.pdf
- Mediana, D., & Nurhidayat, A. I. (2018). Rancang Bangun Aplikasi Helpdesk (A-Desk) Berbasis Web Menggunakan Framework Laravel (Studi Kasus di PDAM Surya Sembada Kota Surabaya). *Jurnal Manajemen Informatika*, 8(2), 75–81. <http://ejournal.ukrida.ac.id/ojs/index.php/TIK/article/view/1495/1617>
- Mursyidin, A. (2018). Prototype Sistem Penghitung Objek Manusia Berbasis Computer Vision Menggunakan Algoritma. In *Prodi Informatika dan Sistem Informasi* (Vol. 7).
- Novendri. (2019). Pengertian Web. *Lentera Dumai*, 10(2), 46–57.
- Presmann, R. (2014). Software Quality Engineering: A Practitioner’s Approach. In *Software Quality Engineering: A Practitioner’s Approach* (Vol. 9781118592). <https://doi.org/10.1002/9781118830208>
- Satzinger, J. W., Jackson, R. B., & Burd, S. D. (2015). *Systems analysis and design in a changing world*. Cengage learning. C, 826–833. https://doi.org/10.1057/978-1-349-96042-2_5150
- Shahu Gaikwad, S., & Adkar, P. (2019). A Review Paper on Bootstrap Framework. *IRE Journals*, 2(10), 349–351. https://www.think247.com/vertical?s_pt=sou
- Sianturi, J., Rahmat, R. F., & Nababan, E. B. (2018). Sistem Pendeteksian Manusia untuk Keamanan Ruangan Human Detection System For Room Security Using Viola - Jones. *Journal of Informatics and Telecommunication Engineering*, 1(2), 61–72.
- Solichin, A., & Harjoko, A. (2013). Deteksi Pejalan Kaki pada Video dengan Metode Fastest Pedestrian Detector in The West (FPDW). *Ticom*, 2(1), 202–205.