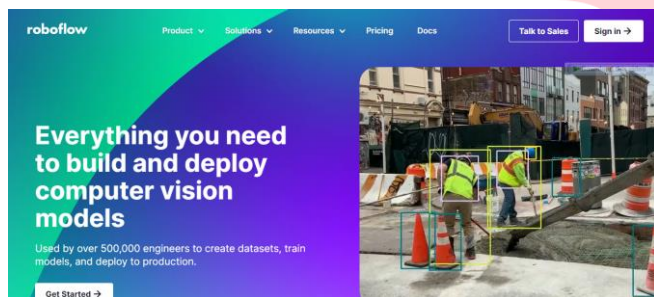


YOLO (*You only look once*) merupakan pendekatan *object detection* secara *real-time* yang menggunakan konvusalional seperti yang ditunjukkan pada gambar 2.1 di atas. YOLOv8 adalah model terbaru dari *ultralytics*, salah satu fitur utamanya adalah SOTA atau *State Of The Art* [4]. Dari uraian tersebut, YOLO adalah pendekatan deteksi objek secara *real-time* yang menggunakan jaringan saraf konvusalional untuk memproses seluruh gambar dalam satu kali operasi. YOLOv8n yang memiliki fitur SOTA, artinya model ini memiliki performa yang baik dalam tugas deteksi objek untuk mengenali dan mengklasifikasi objek.

C. Dataset Roboflow

Dalam deteksi objek, data yang digunakan untuk melatih suatu model algoritma berupa kumpulan gambar. Tanpa adanya data maka algoritma tidak dapat dilatih serta tidak dapat mengenali objek. Gambar tersebut dapat berjumlah puluhan, ratusan, hingga ribuan. Data yang cukup terkenal dan sering digunakan oleh peneliti untuk melakukan pengujian terhadap suatu model adalah PASCAL VOC, dan COCO [5].



Gambar 2. 2 Roboflow

Pengambilan dataset berupa gambar-gambar untuk penelitian sistem deteksi objek pada website ini bisa dengan memanfaatkan API untuk terhubung dengan mudah antara dua aplikasi atau mendownload semua file dalam format .zip.

D. Telegram

Merupakan *platform* perpesanan instan berbasis cloud yang memungkinkan pengguna untuk mengirimkan pesan teks, foto, video, dokumen, dan file lainnya. Salah satu fitur utama di aplikasi ini adalah bot, telegram memiliki API yang memungkinkan pengembang membuat bot otomatis untuk berbagai fungsi, seperti memberikan informasi, mengirim notifikasi, dan bahkan dapat mengontrol perangkat sistem IoT (*Internet of Things*) [6].

E. Confusion matrix

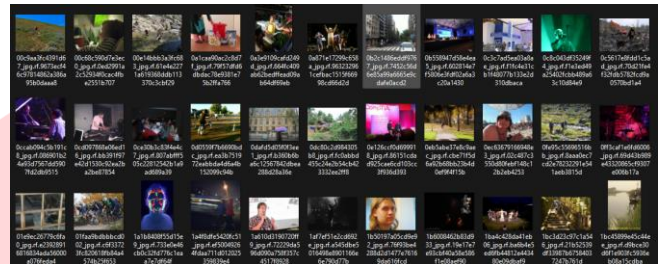
Confusion matrix adalah alat evaluasi yang digunakan untuk mengukur kinerja model. Dalam matriks ini, analisis *confusion matrix* biasanya terdiri dari empat komponen utama: *True Positive* (TP) adalah total dari data positif dan terklasifikasi oleh sistem dengan benar, *True Negative* (TN) adalah total dari data positif dan terklasifikasi oleh sistem dengan salah, *False Positive* (FP) adalah total dari data negatif dan terklasifikasi dengan benar, *False Negative* (FN) adalah total data negatif yang terklasifikasi oleh sistem dengan salah [7]. Dengan melakukan analisis ini, kita dapat mengetahui berbagai nilai metrik dari akurasi, presisi, *recall*,

dan *f1-score*. Ini membantu untuk kita memahami kelebihan dan kekurangan model dalam melakukan deteksi objek.

III. METODE

A. Pengumpulan dataset

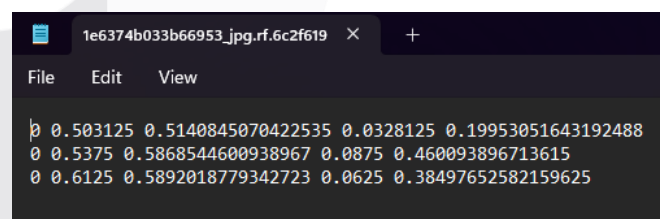
Proses pengumpulan dataset gambar manusia dilakukan pada website roboflow. Dataset yang sudah dikumpulkan kemudian diunduh dan disimpan dengan format .tar pada sebuah folder yang menghasilkan kumpulan gambar manusia sebanyak 1.534 yang terdiri dari berbagai kondisi. Berikut adalah hasil pengumpulan dataset citra manusia seperti yang ditunjukkan pada gambar 3.1



Gambar 3. 1 Pengumpulan dataset

B. Pengolahan dataset

Tahap ini dilakukan untuk eksplorasi terhadap dataset yang sudah didapatkan, proses ini mengenai gambaran dari informasi dengan mengeksplorasi dan mendeskripsikan struktur dataset yang sudah dilakukan pengumpulan data dan proses anotasi citra. Hasil dari akuisisi data mendapatkan dataset sebanyak 1.534 merupakan kumpulan dataset *singleclass* yang sudah dianotasi yang terdiri dari 1 pada website roboflow. Proses anotasi pada setiap gambar dilakukan pelabelan dengan ukuran 320×320 pixel, anotasi citra YOLO akan tersimpan menggunakan format file .txt dengan nama yang sama untuk setiap file gambar dalam direktori yang sama. Setiap file .txt berisi anotasi untuk file gambar terkait, yaitu kelas objek, koordinat objek, tinggi, dan lebar. Format anotasi citra YOLO pada file .txt seperti ditunjukkan pada gambar 3.2



Gambar 3. 2 Anotasi dataset

Hasil dari tahap eksplorasi struktur dataset yang sudah dianotasi akan terbagi menjadi *data testing*, *validation*, dan *training* yang tersimpan pada sebuah folder dengan folder label seperti yang ditunjukkan pada gambar 3.3

Name	Date modified	Type	Size
test	5/17/2024 6:06 PM	File folder	
train	5/17/2024 6:06 PM	File folder	
valid	5/17/2024 6:06 PM	File folder	
data	5/11/2024 12:25 PM	Yaml Source File	1 KB
README dataset	5/9/2024 12:04 PM	Text Document	1 KB
README roboflow	5/9/2024 12:04 PM	Text Document	1 KB

Gambar 3. 3 File dataset

Kemudian pada tahap selanjutnya yaitu konfigurasi program dilakukan untuk melatih model yang dibuat menggunakan *Visual Studio Code* menggunakan bahasa pemrograman python. Algoritma deteksi objek yang digunakan adalah YOLOv8. Sebelum dilakukan *training* pada model, dilakukan beberapa tahapan seperti melakukan *clone Repo* pada GitHub Ultralytics, menginstal beberapa *requirements* yang diperlukan, kemudian proses *training* model menggunakan algoritma YOLOv8 yang dilakukan sebanyak 100 iterasi atau *epoch*, menentukan ukuran *image size* sebesar 320×320 pixel, dan data dari dataset yang sudah terkumpul.

```
Pip install ultralytics

Yolo detect train model=yolov8n.pt
data=human/data.yaml epoch=100
imgsz=320
```

Kode diatas adalah konfigurasi untuk melakukan proses *training* pada dataset.

C. Pengujian dataset

Setelah melakukan proses pengumpulan, pengolahan dataset pada model, dilakukan tahap pengujian atau *testing*. Berikut merupakan proses *testing* pada demo video dan gambar, dimana keduanya akan disimpan pada direktori yang didapat dari hasil *testing object detection* yang mana di dalamnya berisi hasil rekaman yang berdurasi 22 detik di rumah Pak Burhan di daerah Bandung. Proses *testing object detection* untuk mendeteksi manusia dengan menjalankan

```
yolo detect predict
model=yolov8n320.onnx source=test.mp4'
```

Dimana *yolov8n320.onnx* adalah model hasil *export* dan *training* yang digunakan untuk *testing object detection*. Hasil proses *testing* pada rekaman seperti yang ditunjukkan pada gambar 3.4 dan 3.5



Gambar 3. 4 Pengujian pada siang hari



Gambar 3. 5 Pengujian pada malam hari

D. Pengujian dengan aplikasi telegram

Seperti yang telah dijelaskan sebelumnya tentang sistem notifikasi yang berperan sebagai fitur peringatan kepada pengguna ketika ada manusia yang terdeteksi keberadaannya. Sistem notifikasi ini melibatkan penggunaan server flask dengan aplikasi Telegram.

Jika ada manusia yang terdeteksi keberadaannya maka bot telegram akan langsung mengirim notifikasi berupa gambar seperti yang ditunjukkan pada gambar 3.6 berikut.



Gambar 3. 6 Notifikasi Telegram

IV. HASIL DAN PEMBAHASAN

Pengujian yang dilakukan pada pengembangan sistem deteksi objek bertujuan untuk memastikan bahwa setiap komponen yang terpasang pada sistem bekerja sesuai dengan spesifikasi yang telah ditentukan dan sistem dapat berfungsi sesuai dengan yang seharusnya. Algoritma akan diuji dan

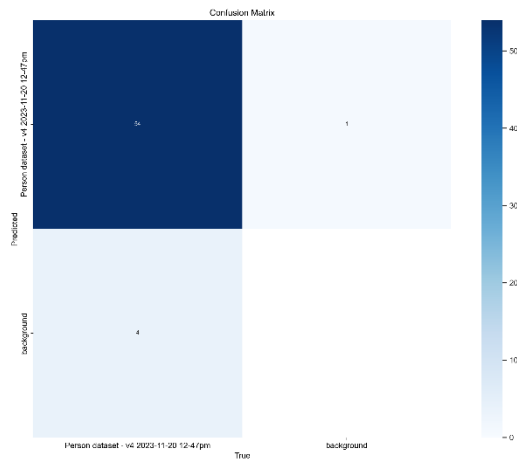
diukur untuk mengetahui nilai akurasi, presisi, recall, dan f1-score menggunakan persamaan (1), (2), (3), (4).

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$precision = \frac{TP}{TP + FP} \quad (2)$$

$$recall = \frac{TP}{TP + FN} \quad (3)$$

$$f1 - score = 2 \times \frac{precision \times recall}{precision + recall} \quad (4)$$



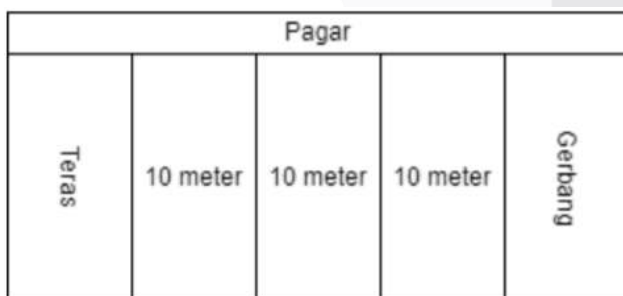
Gambar 3. 7 Nilai confusion matrix model

Hasil dari pengukuran nilai akurasi, presisi, recall, dan f1-score pada model seperti yang ditunjukkan pada tabel 4.1 berikut.

Tabel 4. 1 Tabel nilai evaluasi

Akurasi	Presisi	Recall	F1-score
91.5%	98.1%	100%	93.1%

Kemudian dilakukan pengujian di lapangan. Pengujian akan dilakukan dengan cara mencari posisi yang efektif agar sistem dapat menjangkau keseluruhan halaman.



Gambar 4. 1 Denah halaman rumah

Dari gambar 4.1 di atas, hasil pengujian akan disajikan pada tabel 4.2 berikut.

Tabel 4. 2 Tabel pengujian di lapangan

Jenis pengujian	Hasil
10 meter dari teras rumah	Sistem mampu mendeteksi >10 m ke halaman tengah namun tidak sampai ke gerbang rumah

10 meter di tengah halaman	Sistem mampu mendeteksi >10 m ke teras rumah dan ke gerbang rumah
10 meter dari gerbang rumah	Sistem mampu mendeteksi >10 m ke tengah halaman namun tidak sampai ke teras rumah

Setelah itu dilakukan pengujian di kondisi yang berbeda yaitu siang hari dan malam hari. Pengujian ini dilakukan untuk mengetahui performa yang diberikan dari model terhadap tugas deteksi objek. Berikut adalah tabel 4.3 tentang pengujian fungsionalitas umum pada sistem.

Tabel 4. 3 Tabel fungsionalitas umum

Lingkungan	Inference (ms)	Akurasi (%)	Kestabilan deteksi (1-5)
Indoor terang	35	89	5
Outdoor terang	42	90	5
Indoor redup	38	75	5
Outdoor redup	45	70	3.5

V. KESIMPULAN

Pengembangan sistem deteksi objek manusia menunjukkan tingkat keberhasilan yang tinggi. Pengujian pengukuran nilai akurasi, presisi, recall, dan f1-score dari model yang sudah dilatih menghasilkan nilai akurasi 91.5%, presisi 98.1%, recall 100%, dan f1-score 93.1% kemudian untuk hasil pengujian di lapangan menunjukkan posisi yang paling efektif adalah di tengah halaman karena dapat mendeteksi ke arah teras dan gerbang sejauh >10 m, dan untuk menguji fungsionalitas di berbagai kondisi yaitu siang hari dan malam hari, hasilnya adalah dalam keadaan outdoor redup akurasi menurun menjadi 70% dan kestabilan sistem bisa mencapai 3.5 ini menunjukkan sistem membutuhkan waktu untuk mendeteksi karena pencahayaan yang rendah.

REFERENSI

- [1] R. Samsinar, G. Gatot Aditya, D. Almanda, F. Amrulloh, dan A. Ilmar Ramdhan, "Sistem Pendeteksi Kurir Menggunakan Smart Closed Circuit Television (CCTV) Berbasis Internet Of Things (IoT) dengan Media Komunikasi Bot Telegram (Studi Kasus : Rumah Indekost)," vol. 6, no. 1.
- [2] C. Janiesch, P. Zschech, dan K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, vol. 31, no. 3, 2021, doi: 10.1007/s12525-021-00475-2.
- [3] R. Roscher, B. Bohn, M. F. Duarte, dan J. Garcke, "Explainable Machine Learning for Scientific Insights and Discoveries," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.2976199.
- [4] J. Redmon, S. Divvala, R. Girshick, dan A. Farhadi, "You Only Look Once: Unified, Real-Time Object

- Detection,” Jun 2015, [Daring]. Tersedia pada: <http://arxiv.org/abs/1506.02640>
- [5] R. I. Tiyyar dan D. H. Fudholi, “Kajian Pengaruh Dataset dan Bias Dataset terhadap Performa Akurasi Deteksi Objek,” *PETIR*, vol. 14, no. 2, 2021, doi: 10.33322/petir.v14i2.1350.
- [6] H. A. Kusuma, S. B. Wijaya, dan D. Nussyirwan, “SISTEM KEAMANAN RUMAH BERBASIS ESP32-CAM DAN TELEGRAM SEBAGAI NOTIFIKASI,” *Infotronik : Jurnal Teknologi Informasi dan Elektronika*, vol. 8, no. 1, 2023, doi: 10.32897/infotronik.2023.8.1.2291.
- [7] T. Fawcett, “An introduction to ROC analysis,” *Pattern Recognit Lett*, vol. 27, no. 8, 2006, doi: 10.1016/j.patrec.2005.10.010.

