

Sistem Deteksi Pengendara Sepeda Motor Tanpa Helm Menggunakan Drone

1st Zalva Ihilani Pasha
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

zalvapsh@student.telkomuniversity.ac.id

2nd Casi Setianingsih
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

setiacasie@telkomuniversity.ac.id

3rd Anggunmekha Luhur Prasasti
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

anggunmekha@telkomuniversity.ac.id

Abstrak— Dalam menghadapi masalah pelanggaran lalu lintas yang meningkat di Indonesia, metode konvensional seperti CCTV memiliki keterbatasan dalam cakupan dan mobilitas. Studi ini mengusulkan penggunaan drone yang terintegrasi dengan teknologi penginderaan visual dan kecerdasan buatan sebagai alat deteksi pelanggaran lalu lintas. Kami menggunakan YOLOv7 sebagai model objek deteksi. Untuk memproses data dan komunikasi dengan drone, kami memanfaatkan backend yang dikembangkan dengan Flask, sementara frontend aplikasi dihadirkan menggunakan React JS. Drone ini mampu mencakup area lebih luas dan mendeteksi pelanggaran seperti pengendara yang tidak menggunakan helm. Data diperoleh dari drone diolah secara real-time untuk tindakan penegakan hukum. Hasil pengujian menunjukkan model ini memiliki precision 0.742, recall 0.798, F1-score 0.768, mAP@[0.5] 0.81, dan mAP@[0.5:0.95] 0.595, parameter tersebut dihasilkan oleh model data dengan Rasio Data 88:12, Learning Rate 0.0001, Batch Size 8, dan Epochs 100. Hasil dari pengujian sistem dengan tingkat akurasi 90.51% untuk deteksi tidak memakai helm dengan scenario penangkapan video drone yang terbaik. skenario terbaik pengujian sistem, drone diterbangkan pada ketinggian 3-5m, dengan sudut kamera 45°-60°, pencahayaan di atas 7000 lux, dan kecepatan drone di bawah 1 km/j atau dalam posisi diam dengan pandangan optimal.

Kata kunci— Deteksi pelanggaran lalu lintas, Kecerdasan buatan, Drone, Real-time

I. PENDAHULUAN

Teknologi, dalam dekadenya yang cepat berkembang, telah merevolusi banyak aspek kehidupan, termasuk bidang penindakan hukum. Di Indonesia, pihak kepolisian telah memanfaatkan kamera CCTV yang terpasang di persimpangan untuk memantau dan menindak pelanggaran lalu lintas. Namun, ada keterbatasan pada sistem ini. Mobilitas terbatas CCTV berarti bahwa banyak pelanggar dapat mengelak dari pantauan, terutama di titik-titik buta atau dengan trik-trik tertentu seperti menutupi plat nomor. Sebagai respons terhadap kelemahan ini, kepolisian memulai uji coba dengan drone atau UAV. Berbeda dengan CCTV, drone menawarkan mobilitas yang jauh lebih tinggi, memberikan pandangan yang lebih luas dan mampu mendeteksi pelanggar yang mencoba bersembunyi [1].

Latar belakang proyek ini muncul dari kesadaran akan dampak serius dari pelanggaran lalu lintas. Data menunjukkan bahwa pelanggaran lalu lintas berkontribusi pada sebagian besar kematian akibat kecelakaan jalan. Namun, kemajuan dalam penuntasan kasus-kasus ini seringkali terhambat oleh kurangnya bukti substansial. Dengan ini, tujuan utama adalah untuk meningkatkan

pengawasan dan memastikan bahwa pelanggaran dapat didokumentasikan dengan lebih efektif. UAV dengan kemampuannya yang unik diharapkan dapat mengatasi hambatan yang ada saat ini, terutama dalam mendeteksi pengendara sepeda motor yang tidak mengenakan helm.

Namun, menerapkan drone dalam penegakan hukum bukanlah tanpa tantangannya. Meskipun drone memiliki potensi besar, ada tantangan teknis yang perlu diatasi, khususnya dalam pelacakan target. Teknologi pelacakan visual, khususnya yang berbasis deep learning, menawarkan hasil yang menjanjikan tetapi memerlukan komputasi yang berat. Ini menimbulkan pertanyaan tentang efektivitasnya dalam situasi nyata dan ketersediaan sumber daya untuk mendukungnya. Meskipun demikian, dengan pendekatan yang tepat dan penelitian lebih lanjut, drone dapat menjadi alat yang sangat berharga dalam meningkatkan keselamatan di jalan raya [2].

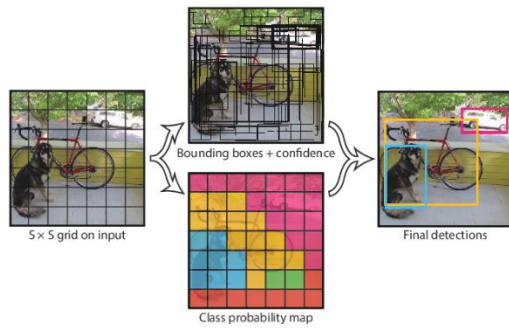
II. KAJIAN TEORI

A. Undang-Undang Lalu Lintas

Dalam rangka meningkatkan keselamatan lalu lintas, khususnya bagi para pengendara dan penumpang sepeda motor pentingnya menerapkan undang-undang lalu lintas. Berikut undang-undang yang berkaitan dengan pelanggaran tidak memakai helm :

1. Pasal 106 ayat (8) UU No. 22/2009
“Setiap orang yang mengemudikan Sepeda Motor dan Penumpang Sepeda Motor wajib mengenakan helm yang memenuhi standar nasional Indonesia”.
2. Pasal 291 ayat (1) UU No. 22/2009
“Setiap orang yang mengemudikan Sepeda Motor tidak mengenakan helm standar nasional Indonesia sebagaimana dimaksud dalam Pasal 106 ayat (8) dipidana dengan pidana kurungan paling lama 1 (satu) bulan atau denda paling banyak Rp250.000,00 (dua ratus lima puluh ribu rupiah)”.
3. Pasal 291 ayat (2) UU No. 22/2009
“Setiap orang yang mengemudikan Sepeda Motor yang membiarkan penumpangnya tidak mengenakan helm sebagaimana dimaksud dalam Pasal 106 ayat (8) dipidana dengan pidana kurungan paling lama 1 (satu) bulan atau denda paling banyak Rp250.000,00 (dua ratus lima puluh ribu rupiah)”.

B. You Only Look Once



Gambar 1 Ilustrasi YOLO

YOLO (You Only Look Once), diilustrasikan oleh Redmon et al. (2016), adalah metode deteksi objek berdasarkan Convolutional Neural Network (CNN) yang unik. Dengan satu jaringan konvolusional, YOLO dapat memprediksi banyak bounding boxes dan probabilitas kelasnya secara bersamaan. Keunggulannya terletak pada kecepatan; versi dasarnya bisa menghasilkan 45 frame per detik, dan versi yang lebih cepat mencapai lebih dari 150 frame per detik. Dengan kapasitas ini, YOLO memproses video real-time dengan latensi kurang dari 25 milidetik dan memiliki presisi rata-rata yang tinggi dibandingkan sistem real-time lainnya [3].

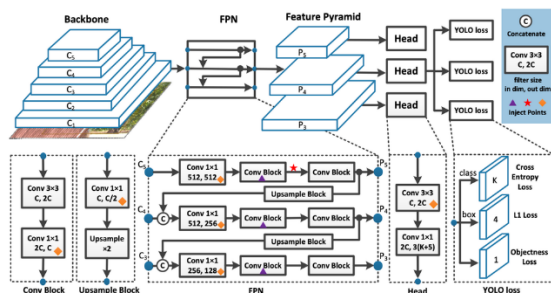
$$\text{conf}(\text{class}) = \text{Pr}(\text{class}) \times \text{IOU}_{\text{Pred}}^{\text{Truth}}$$

Gambar 2 Rumus Confidence

$$\text{IOU}_{\text{Pred}}^{\text{Truth}} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

Gambar 3 Rumus Intersection of Union

Algoritma ini membagi gambar menjadi grid berdimensi $s \times s$ untuk mendeteksi objek. Setiap grid memprediksi bounding box dan menghitung confidence score, yang mengukur keakuratan deteksi objek dalam box tersebut. Score ini penting untuk deteksi yang tepat. Alih-alih fokus hanya pada area tertentu, YOLO mempertimbangkan seluruh objek dalam gambar. Hal ini memungkinkan YOLO mendeteksi objek secara real-time dengan presisi tinggi dan responsif terhadap perubahan skala dan orientasi objek. setiap bounding box memiliki lima variabel: x , y , w , h , dan c . Variabel x dan y menunjukkan koordinat pusat bounding box, sedangkan w dan h menentukan lebarnya dan tingginya. Variabel c menunjukkan kepercayaan sistem bahwa box tersebut mengandung objek target. IOU dan struktur bounding box ini meningkatkan efisiensi deteksi objek oleh YOLO [4].



Gambar 4 Arsitektur YOLO

Dalam arsitektur YOLOv7, terdapat tiga komponen: Backbone, Neck, dan Head. Backbone menggunakan model ELAN dan E-ELAN untuk mengumpulkan fitur gambar pada tingkat granularitas yang berbeda dan biasanya dilatih terlebih dahulu. Neck, dengan model CSPSP+ (ELAN, E-ELAN) PAN, mengekstrak feature maps dari berbagai tahap di Backbone. Sedangkan Head memakai model YOLOR untuk membuat prediksi bounding box dan klasifikasi [5].

C. Parameter Kinerja YOLO

• Precision

Bagian dari semua deteksi positif yang sebenarnya benar menunjukkan kualitas dari prediksi, menggambarkan seberapa sering prediksi positif kita tepat adanya [6]. *Precision* memiliki rumus seperti berikut:

$$\text{Precision} = \frac{TP}{TP + FP} \quad [6]$$

Gambar 5 Rumus Precision

Penjelasan :

TP = True Positive

FP = Fake Positive

• Recall

Bagian dari deteksi positif yang sebenarnya benar berdasarkan semua objek asli menunjukkan sejauh mana cakupan deteksi, yaitu, berapa banyak objek nyata yang berhasil dideteksi [7]. *Recall* memiliki rumus seperti berikut:

$$\text{Recall} = \frac{TP}{TP + FN} \quad [7]$$

Gambar 6 Rumus Recall

Penjelasan :

TP = True Positive

FN = Fake Negative

• F1-Score

Nilai harmonik dari presisi dan recall. Ini menciptakan suatu keseimbangan di antara *precision* dan *recall*, menjadi ukuran yang efektif terutama ketika distribusi kelas tidak merata [8]. *F1-score* memiliki rumus seperti berikut :

$$F1 - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad [8]$$

Gambar 7 Rumus F1-Score

• Mean Average Precision

Metrik evaluasi untuk model deteksi objek seperti YOLO. Ini dihitung sebagai rata-rata dari AP (*Average Precision*) untuk tiap kelas, dengan AP merepresentasikan area di bawah grafik *precision-recall*. Ini menilai efektivitas model pada semua kategori [9]. mAP memiliki rumus seperti berikut:

$$mAP = \sum_{i=1}^N \frac{AP_i}{N} \times 100\% \quad [9]$$

Gambar 8 Rumus mAP

Penjelasan :

Σ (AP) adalah jumlah total dari Average Precision untuk semua kelas.

N adalah jumlah kelas.

D. Library *Frontend* dan *Backend*

a. *Frontend*

- React

React JS adalah library JavaScript untuk membangun antarmuka aplikasi web dan mobile dengan pendekatan berbasis komponen yang dapat digunakan kembali dan penggunaan Virtual DOM untuk pembaruan tampilan yang efisien [10].

- Tailwind CSS

Tailwind CSS adalah framework CSS berorientasi utility untuk desain antarmuka cepat. Menggunakan class-class utility, memungkinkan desain kustom langsung dari HTML dengan pendekatan utility-first, desain responsif, dan mode gelap [11].

- React Router DOM

React Router DOM adalah library untuk React yang memudahkan pembuatan rute dan navigasi antar halaman dalam aplikasi web. Menyediakan dua jenis router: HashRouter untuk situs web statis tanpa server untuk data dinamis, dan BrowserRouter untuk situs dengan server yang merender data dinamis [12].

b. *Backend*

- argparse

Argparse adalah library Python yang memudahkan pengelolaan argumen di command-line interface, dengan fitur untuk mendefinisikan argumen mandatory, opsional, tipe data, dan nilai default [13].

- io

Library io di Python memfasilitasi operasi input/output, dengan kelas untuk membaca dan menulis data dari berbagai sumber, termasuk file dan aliran data seperti BytesIO, StringIO, BufferedReader, dan BufferedWriter. Esensial untuk operasi I/O di Python [14].

- PIL

Pillow adalah versi ditingkatkan dari PIL, library Python untuk manipulasi gambar, yang kompatibel dengan Python 3. Menawarkan alat untuk berbagai operasi gambar dan mendukung format seperti BMP, PNG, JPEG, dan TIFF [15].

- datetime

Library datetime di Python digunakan untuk mengelola dan memanipulasi tanggal dan waktu, dengan kelas seperti date, time, dan timedelta. Memudahkan operasi seperti menghitung selisih waktu dan memformat tanggal/waktu [16].

- torch

PyTorch adalah library Python untuk pengembangan jaringan saraf dalam machine learning. Dioptimalkan untuk GPU dan CPU, menawarkan grafik komputasi dinamis, perhitungan tensor, diferensiasi otomatis, dan dukungan deep learning [17].

- cv2

cv2, bagian dari OpenCV di Python, digunakan untuk pengolahan gambar dan video. Memungkinkan perubahan ukuran, warna, deteksi wajah, dan mendukung format seperti BMP, PNG, dan JPEG. Esensial untuk computer vision dan machine learning [18].

- numpy

NumPy adalah library Python untuk operasi array dan aljabar linier, mendukung komputasi numerik, ilmiah, pengolahan data, machine learning, dan computer vision dengan kecepatan tinggi [19].

- flask

Flask adalah framework Python untuk pembuatan aplikasi web, ideal untuk proyek kecil hingga kompleks, memudahkan pembangunan dengan cepat [20].

- os

Library os di Python memfasilitasi operasi terkait sistem operasi seperti akses file, direktori, dan pengaturan variabel lingkungan. Esensial untuk interaksi dengan sistem operasi [21].

- re

Library re di Python digunakan untuk operasi ekspresi reguler, memungkinkan pencocokan dan manipulasi string berdasarkan pola. Esensial untuk tugas-tugas manipulasi string [22].

- json

Library json di Python memfasilitasi interaksi dengan format JSON, mengkonversi data Python ke JSON dengan json.dumps() dan mengubah data JSON ke Python dengan json.loads(). Esensial untuk pertukaran data aplikasi web [23].

- math

Library math di Python menyediakan fungsi untuk operasi matematika seperti akar kuadrat dan logaritma. Esensial untuk pengolahan data, machine learning, dan computer vision [24].

- flask_cors

Library Flask-Cors di Python memungkinkan pengaturan Cross-Origin Resource Sharing (CORS) pada aplikasi web Flask. Dengan ini, pengguna dapat mengatur akses sumber daya antar domain dan mengatur header CORS. Esensial untuk komunikasi antar-domain dalam aplikasi Flask [25].

E. Koneksi Drone ke Aplikasi

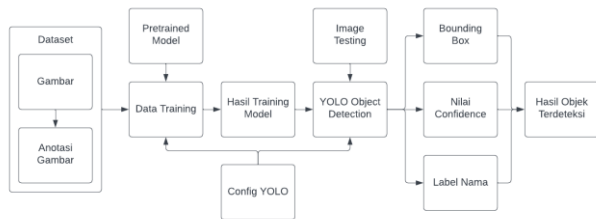


Gambar 9 Ilustrasi Koneksi

MonaServer adalah server multimedia yang mendukung berbagai protokol termasuk RTMP. OBS adalah perangkat lunak sumber terbuka untuk merekam dan streaming video dengan dukungan plugin seperti OBS Virtual Camera, yang memungkinkan outputnya digunakan sebagai webcam virtual. Rencana penggunaannya melibatkan MonaServer sebagai pusat koneksi RTMP untuk menerima stream dari drone melalui OBS dan menyiarkannya ke aplikasi web. OBS akan menampilkan dan mengolah video dari drone sebelum dikirim ke *backend*.

III. RANCANGAN SISTEM

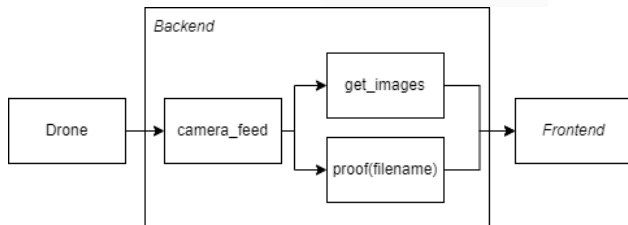
A. Model Training



Gambar 10 Diagram Block Model Training

Pertama, proses yang dilakukan anotasi gambar dengan memungkinkan pengguna menandai objek menggunakan bounding box dan memberi label pada masing-masing. Hasil anotasi disimpan dalam file .txt yang berisi koordinat dan label objek untuk setiap bounding box. File ini kemudian digunakan untuk melatih model YOLO. Aplikasi Roboflow juga mendukung augmentasi data, yang meningkatkan variasi data dan kinerja model. Dataset yang telah dianotasi diproses melalui tahap training menggunakan *pre-trained model* dan konfigurasi dari YOLO. Hasilnya adalah file model dengan format .pt yang mewakili model kustom baru. Saat menggunakan model ini dengan YOLO, objek yang sesuai dengan dataset dianotasi akan terdeteksi, menampilkan bounding box, label, dan nilai confidence. Untuk melatih model menggunakan *clone repository* dari YOLOv7, langkah-langkah meliputi instalasi modul, menggunakan *pre-trained model* YOLO, melatih dengan data foto anotasi kustom dari *roboflow*, dan inferensi menggunakan *weight* kustom.

B. Desain Sistem



Gambar 11 Diagram Block Sistem

Aplikasi ini menggunakan Flask yang mengintegrasikan berbagai library dan fungsionalitas untuk memproses gambar khususnya yang berkaitan dengan pelanggaran tidak menggunakan helm dan akses folder *backend* untuk ditampilkan ke *frontend*.

1. Inisialisasi Modul dan Variabel

Aplikasi ini dimulai dengan mengimpor modul yang diperlukan dari berbagai library Python, seperti Flask, Torch (PyTorch), OpenCV, dan lainnya. Aplikasi ini juga menginisialisasi beberapa variabel, seperti *classNames* yang berfungsi untuk menamai jenis objek yang terdeteksi.

2. Flask App

Aplikasi Flask dibuat dan diberi nama *webapp*. CORS juga diterapkan pada aplikasi ini untuk menangani isu-isu terkait *Cross-Origin Resource Sharing*.

3. Rute

Ada beberapa rute yang didefinisikan di backend aplikasi ini :

- `@app.route("/proof/<path:filename>")`
Memungkinkan akses ke gambar tertentu yang disimpan di direktori *proof*.
- `@app.route("/get_images")`
Mengembalikan daftar gambar yang ada dalam direktori *proof* dalam format JSON.
- `@app.route("/camera_feed")`
Mengirimkan gambar dalam format stream dari webcam. video tersebut kemudian diproses menggunakan model YOLO yang digunakan pada aplikasi untuk deteksi objek dan hasilnya ditampilkan kembali.

4. Menjalankan *backend*

Bagian `if __name__ == "__main__":` menjalankan *backend*. Ini pertama-tama memuat model dari PyTorch yang akan digunakan untuk deteksi objek. Kemudian, memulai pemantauan direktori dan akhirnya menjalankan aplikasi Flask pada port yang ditentukan.

5. Koneksi *Frontend*

Metode yang digunakan React js untuk berinteraksi dengan *backend* melalui rute atau *endpoint* untuk mendapatkan atau mengirim data. Setelah mengirimkan permintaan ke backend, frontend menunggu respons dan kemudian memproses data yang diterima. Namun, saat frontend dan backend berjalan pada domain yang berbeda, masalah CORS (Cross-Origin Resource Sharing) bisa muncul, dengan itu aplikasi ini menggunakan library *flask_cors*.

IV. HASIL DAN PEMBAHASAN

A. Pengujian *Training Dataset*

Dalam pengujian pertama ini, fokus pengujian diberikan kepada metode deteksi objek dengan menggunakan teknologi YOLO (You Only Look Once). Metode ini terkenal dengan kecepatan dan efisiensinya dalam mendeteksi objek pada citra digital. Hasil dari pengujian ini menunjukkan bahwa YOLO memberikan tingkat kepercayaan yang sangat tinggi dalam mendeteksi objek yang telah dilatih dalam model tersebut. Hal ini mengindikasikan bahwa metode YOLO efektif dan dapat diandalkan dalam mengidentifikasi dan mengklasifikasi objek dengan akurasi yang memuaskan, sehingga menegaskan keunggulan metodenya dalam aplikasi deteksi objek.

Tabel 1 Model Training

Parameter	Nilai	Precision	Recall	F1-Score	mAP@[0.5]	mAP@[0.5:0.95]
Rasio Data	58% : 42%	0.613	0.616	0.614	0.577	0.427
	68% : 32%	0.565	0.694	0.622	0.624	0.458
	78% : 22%	0.651	0.598	0.623	0.66	0.484

Parameter	Nilai	Precision	Recall	F1-Score	mAP@[0.5]	mAP@[0.5:0.95]
	88% : 12%	0.784	0.746	0.764	0.8	0.603
	98% : 2%	0.657	0.765	0.697	0.799	0.625
Learning Rate	0.001	0.64	0.88	0.741	0.782	0.593
	0.0001	0.742	0.798	0.768	0.81	0.592
	0.00001	0.784	0.746	0.764	0.8	0.603
	0.000001	0.743	0.732	0.738	0.784	0.595
Batch Size	2	0.724	0.736	0.729	0.767	0.564
	4	0.751	0.775	0.762	0.8	0.611
	8	0.742	0.798	0.768	0.81	0.592
Epochs	100	0.742	0.798	0.768	0.81	0.592
	200	0.648	0.75	0.695	0.766	0.588
	300	0.791	0.654	0.716	0.772	0.594

Dapat dilihat dari tabel, hasil pengujian terbaik yaitu menggunakan rasio data 88%:12%, *learning rate* 0.0001, *batch-size* 8, dan *epochs* 100 dengan hasil rata-rata seluruh parameter sebesar 0.742.

B. Pengujian Sistem

Dalam pengujian ini, ada lima skenario berbeda akan diterapkan untuk menguji performa drone terhadap sistem. Pertama, drone akan diuji pada ketinggian 3 meter dengan tiga variasi sudut pandang atau angle yang berbeda. Kedua, ketinggian drone ditingkatkan menjadi 5 meter dengan tiga variasi angle yang sama. Selanjutnya, pada skenario ketiga, ketinggian dinaikkan kembali menjadi 7 meter dengan tiga variasi angle. Pada skenario keempat, evaluasi akan dilakukan berdasarkan tiga kecepatan berbeda saat drone bergerak. Terakhir, pada skenario kelima, drone akan diuji pada tiga kondisi penerangan yang berbeda, yang diukur berdasarkan hasil pembacaan lux meter dari tiga waktu yang berbeda. Pengujian ini bertujuan untuk mendapatkan gambaran kinerja drone pada berbagai kondisi dan situasi yang mungkin dihadapi dalam operasional sehari-hari.

Berikut ru

Tabel 2 Pengujian Ketinggian dan Angle kamera

Ketinggian	Nilai Akurasi setiap Angle Kamera		
	45	60	90
3	100%	100%	33.4%
5	83.4%	83.4%	33.4%
7	66.7%	33.4%	16.7%

Tabel 2 merupakan penjabaran hasil nilai akurasi dari pengujian sistem berbagai ketinggian dan *angle* kamera.

Tabel 3 Pengujian Kecepatan Drone

Kecepatan Drone (km/h)	Nilai Akurasi
0.7-1.1	83.4%

Kecepatan Drone (km/h)	Nilai Akurasi
1.4-1.8	66.7%
3.2-4	16.7%

Tabel 3 merupakan penjabaran hasil nilai akurasi dari pengujian sistem berbagai kecepatan drone.

Tabel 4 Pengujian Pencahayaayan

Pencahayaayan	Nilai Akurasi
7536 lux	83.4%
17644 lux	100%
2251 lux	33.4%

Tabel 4 merupakan penjabaran hasil nilai akurasi dari pengujian sistem berbagai kondisi pencahayaayan pada lokasi.

Hasil pengujian terbaik sistem menunjukkan bahwa kisaran ketinggian drone optimal berada antara 3 hingga 5 meter. Disarankan untuk menggunakan angle kamera antara 45° hingga 60° agar mendapatkan pandangan yang optimal. Untuk memastikan kualitas gambar, pencahayaayan minimal sebaiknya lebih dari 7000 lux. Selain itu, pergerakan drone sebaiknya tidak melebihi kecepatan 1 kilometer per jam atau dapat dipilih opsi posisi diam yang memberikan pandangan yang baik dari sudut pandang drone. Kombinasi dari faktor-faktor ini membantu memastikan akurasi dan kualitas pendeteksian serta pelacakan objek oleh sistem.

C. Pengujian Notifikasi

Pengujian dilakukan untuk memastikan frontend dapat menerima notifikasi pelanggaran yang diproses oleh backend dari folder 'proof'. Pengujian melibatkan pemantauan folder hasil pemrosesan berdasarkan jumlah file yang diproses dan dikirim ke frontend. Lima foto pelanggaran, dinamai berdasarkan jenis pelanggaran, dan waktu.

Tabel 5 Pengujian Notifikasi

Skenario	Akurasi		
	Percobaan 1	Percobaan 2	Percobaan 3
Video 1	100%	100%	100%
Video 2	100%	100%	100%
Video 3	100%	100%	100%

Pengujian fitur notifikasi pada Tabel 5 frontend pada aplikasi website telah dilakukan dengan metode komprehensif, mencakup semua skenario dan kasus uji. Hasil pengujian menunjukkan fitur berfungsi dengan baik, tanpa kesalahan, dan memiliki akurasi 100%. Hal ini menunjukkan kualitas dan keandalan fitur, memastikan pengguna menerima notifikasi sesuai ekspektasi.

V. KESIMPULAN

Berdasarkan serangkaian uji coba yang telah dilaksanakan, kita bisa menyatakan bahwa sistem bekerja dengan efektif. Melaksanakan pengujian di lapangan secara real time menunjukkan variasi hasil sesuai dengan parameter yang diharapkan. Hingga saat ini, feedback dari pengujian langsung sangat positif dan layak untuk dianalisa lebih mendalam. Pelatihan teroptimal diperoleh dengan proporsi data 88% berbanding 12%, tingkat pembelajaran 0.0001, ukuran batch 8, dan 100 epochs dengan skor rata-rata parameter sebesar 0.742. Skenario deteksi ideal melibatkan ketinggian drone antara 3m hingga 5m, sudut kamera antara 45° hingga 60°, intensitas cahaya di atas 7000 lux, dan kecepatan drone tidak melebihi 1 km/j atau dengan posisi tetap yang memberikan perspektif terbaik. Keseluruhan uji coba sistem menunjukkan tingkat keakuratan yang impresif.

REFERENSI

- [1] Ridwan Arifin, (Oktober 2022). "Polisi kini bisa tilang dari udara pakai drone pelanggaran apa yang diincar". <https://www.oto.detik.com/>.
- [2] Shanthi.K.G, Sesha Vidhya S, Vishakha K, Subiksha S, Srija.K.K, Srinee Mamtha.R. Algorithms for Face Recognition Drones, June 2021.
- [3] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. <http://arxiv.org/abs/1506.02640>
- [4] Sarosa, M., & Muna, N. (2021). IMPLEMENTASI ALGORITMA YOU ONLY LOOK ONCE (YOLO) UNTUK DETEKSI KORBAN BENCANA ALAM. 8(4). <https://doi.org/10.25126/jtiik.202184407>
- [5] Jacob Solawetz. (2022, July 17). What is YOLOv7? A Complete Guide. <https://blog.roboflow.com/yolov7-breakdown/>
- [6] Salma Ghoneim. (2019, April 2). Accuracy, Recall, Precision, F-Score & Specificity, which to optimize on? Towards Data Science. <https://towardsdatascience.com/accuracy-recall-precision-f-score-specificity-which-to-optimize-on-867d3f11124>
- [7] Gowtham S R. (2022, June 30). Confusion Matrix to No Confusion Matrix in Just 5mins. Towards AI. <https://pub.towardsai.net/confusion-matrix-179b9c758b55>
- [8] Hari Krishnan N B. (2019, December 11). Confusion Matrix, Accuracy, Precision, Recall, F1 Score. Analytics Vidhya. <https://medium.com/analytics-vidhya/confusion-matrix-accuracy-precision-recall-f1-score-ade299cf63cd>
- [9] Ahmed Fawzy Gad. (2020). Evaluating Object Detection Models Using Mean Average Precision (mAP). Paperspace. <https://blog.paperspace.com/mean-average-precision/>
- [10] Chinmayee Deshpande. (2023, February 7). The Best Guide to Know What Is React. <https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs>
- [11] Tailwind CSS. (n.d.). Rapidly build modern websites without ever leaving your HTML. Retrieved August 6, 2023, from <https://tailwindcss.com/>
- [12] React Router. (n.d.). Feature Overview v6.13.0 | React Router. Retrieved August 6, 2023, from <https://reactrouter.com/en/main>
- [13] argparse. (n.d.). Parser for command-line options, arguments and sub-commands. Python. Retrieved August 6, 2023, from <https://docs.python.org/3/library/argparse.html>
- [14] io. (n.d.). Core tools for working with streams. Python. Retrieved August 6, 2023, from <https://docs.python.org/3/library/io.html>
- [15] Fredrik Lundh. (n.d.). PIL is the Python Imaging Library. Python. Retrieved August 6, 2023, from <https://pillow.readthedocs.io/en/stable/>
- [16] datetime. (n.d.). Basic date and time types. Python. Retrieved August 6, 2023, from <https://docs.python.org/3/library/datetime.html>
- [17] torch. (2023, May 8). Tensors and Dynamic neural networks in Python with strong GPU acceleration. Python. <https://pytorch.org/>
- [18] opencv-python. (2023, June 30). Wrapper package for OpenCV python bindings. Python. <https://pytorch.org/project/opencv-python/>
- [19] numpy. (2023, July 31). Fundamental package for array computing in Python. Python. <https://pytorch.org/project/numpy/>
- [20] flask. (n.d.). Flask's documentation. Python. Retrieved August 6, 2023, from <https://flask.palletsprojects.com/en/2.3.x/>
- [21] os. (n.d.). Miscellaneous operating system interfaces. Python. Retrieved August 6, 2023, from <https://docs.python.org/3/library/os.html>
- [22] re. (n.d.). Regular expression operations. Python. Retrieved August 6, 2023, from <https://docs.python.org/3/library/re.html?highlight=re#module-re>

[23]json. (n.d.). JSON encoder and decoder. Python. Retrieved August 6, 2023, from <https://docs.python.org/3/library/json.html?highlight=json#module-json>

[24]math. (n.d.). Mathematical functions. Python. Retrieved August 6, 2023, from

<https://docs.python.org/3/library/math.html?highlight=math#module-math>

[25]flask_cors. (n.d.). making cross-origin AJAX possible. Python. Retrieved August 6, 2023, from <https://flask-cors.readthedocs.io/en/latest/>

