



## **SISTEM KEAMANAN RUANGAN DENGAN *HUMAN DETECTION* MENGUNAKAN SENSOR KAMERA BERBASIS *DEEP LEARNING***

**Muhammad Alizal Surya<sup>1)</sup>, Misfa Susanto<sup>2)</sup>, Arinto Setyawan<sup>3)</sup>, Helmy Fitriawan<sup>4)</sup>,  
Mardiana<sup>5)</sup>**

<sup>1,2,3,4,5</sup>Teknik Elektro, Universitas Lampung

<sup>1,2,3,4,5</sup>Jalan Prof. Dr. Ir. Sumantri Brojonegoro No.1, Bandar Lampung

Email: <sup>1</sup>Muhammadalizal11@gmail.com, <sup>2</sup>Misfa@eng.unila.ac.id, <sup>3</sup>fx.arinto@eng.unila.ac.id,

<sup>4</sup>helmy.fitriawan@eng.unila.ac.id, <sup>5</sup>mardiana@eng.unila.ac.id

### **Abstract**

*This research focuses on the application of TensorFlow, an open-source platform for machine learning, in the context of object detection and visual analysis. The aim of this study is to develop and train object detection models using deep learning techniques available in TensorFlow to identify objects in various visual contexts. The research methodology involves gathering representative datasets, implementing suitable model architectures such as Convolutional Neural Networks (CNNs), and a training process involving techniques like transfer learning or fine-tuning. TensorFlow is used as the primary framework in the development and evaluation of models, leveraging the strength of available algorithms and the computational capabilities of GPUs. Experimental results indicate that the developed model achieves significant accuracy levels in object detection tasks, demonstrating good performance in distinguishing and identifying objects within diverse datasets. These findings suggest that TensorFlow plays a vital role in facilitating the development of deep learning models capable of effectively handling complex visual tasks. This research provides valuable insights into the application of TensorFlow in the context of object detection and visual analysis, highlighting its potential and strengths in addressing challenges in the visual computing domain.*

**Keyword:** TensorFlow, Deep Learning, Object Detection, Security Systems.

### **Abstrak**

Penelitian ini memusatkan perhatian pada penerapan TensorFlow, sebuah platform open-source untuk machine learning, dalam konteks deteksi objek dan analisis visual. Tujuan dari penelitian ini adalah untuk mengembangkan dan melatih model deteksi objek menggunakan teknik-teknik deep learning yang terdapat dalam TensorFlow guna mengidentifikasi objek-objek dalam berbagai konteks visual. Metodologi penelitian ini melibatkan pengumpulan dataset yang representatif, penerapan arsitektur model yang sesuai seperti Convolutional Neural Networks (CNNs), serta proses training yang melibatkan teknik-teknik seperti transfer learning atau fine-tuning. TensorFlow digunakan sebagai framework utama dalam proses pengembangan dan evaluasi model, memanfaatkan kekuatan algoritma-algoritma yang telah tersedia dan kemampuan komputasi yang terdapat pada GPU. Hasil eksperimen menunjukkan bahwa model yang dikembangkan mampu mencapai tingkat akurasi yang signifikan dalam tugas deteksi objek, memberikan nilai performa yang baik dalam membedakan dan mengidentifikasi objek-objek dalam dataset yang beragam. Hasil ini mengindikasikan bahwa TensorFlow memiliki peran penting dalam memfasilitasi pengembangan model deep learning yang mampu menangani tugas-tugas visual kompleks secara efektif. Penelitian ini memberikan wawasan penting terkait aplikasi TensorFlow dalam konteks deteksi objek dan analisis visual, menyoroti potensi dan kekuatan yang dimiliki oleh platform ini dalam menghadapi tantangan-tantangan dalam domain visual computing.

**Kata Kunci:** Tensorflow, Deep learning, Deteksi Objek, Sistem Keamanan.

## **1. PENDAHULUAN**

Menurut Kamus Besar Bahasa Indonesia rumah adalah bangunan tempat tinggal yang pada umumnya seperti gedung. Tidak hanya dimanfaatkan sebagai tempat tinggal, rumah juga menjadi tempat penyimpanan harta benda dan barang berharga. Meski memiliki nilai yang berharga dan pentingnya peranan rumah, namun rumah sering kali ditinggalkan pemiliknya tanpa adanya pengawasan. Di Indonesia terdapat budaya mudik yang merupakan kegiatan tahunan yang dilakukan oleh hampir kebanyakan masyarakat dengan tujuan untuk berkumpul bersama sanak saudara di kampung halaman. Hal ini membuat rumah harus ditinggalkan selama beberapa waktu tanpa adanya pengawasan. Rumah



kosong yang ditinggalkan oleh penghuninya tanpa pengawasan akan rentan terhadap tindak kriminal seperti pembobolan dan pencurian.

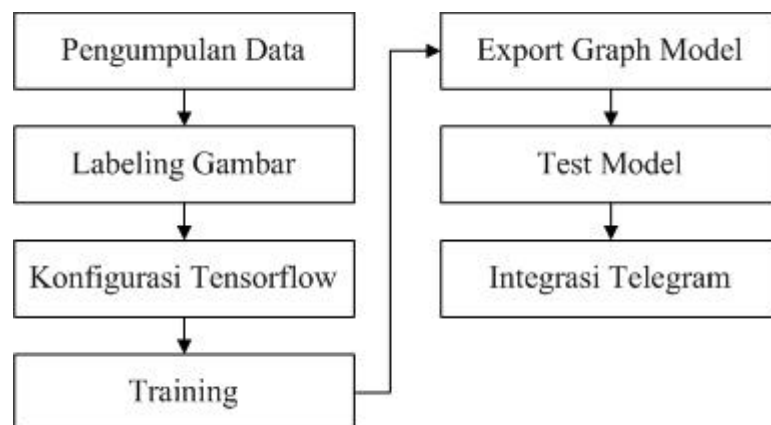
Data registrasi Polri mencatat bahwa tindak kriminalitas di Indonesia pada tahun 2019 sebanyak 269.324 kejadian. Sedangkan pada kasus kejahatan terhadap hak milik tanpa menggunakan kekerasan atau pencurian berada pada angka 80.450 kejadian pada tahun 2019 [1]. Hal ini dapat mengakibatkan kecemasan bagi pemilik rumah ketika akan meninggalkan rumah tanpa adanya pengawasan. Seiring dengan perkembangan jaman, beberapa cara telah dilakukan untuk membuat rumah aman terhadap masuknya pencuri di antaranya adalah menggunakan kunci gembok tambahan. Tetapi hal ini kurang efektif karena pencuri dapat masuk rumah melalui jendela maupun atap rumah. Hal lainnya yang dapat dilakukan adalah menggunakan kamera CCTV (Closed Circuit Television). Kamera CCTV cukup baik untuk mengurangi tingkat pencurian tetapi kamera CCTV ini hanya mampu merekam kejadian tetapi belum dapat membuat notifikasi jika ada pencuri yang masuk ke dalam rumah.

Dengan beberapa kelemahan sistem keamanan yang ada, penulis membuat inisiatif untuk memanfaatkan kamera sebagai sensor deteksi manusia dengan notifikasi, dengan menggunakan sistem ini, kita akan mendapatkan notifikasi dalam bentuk push notification telegram jika ada orang yang terdeteksi di dalam rumah kita. Dengan begini, kita dapat memberitahukan kepada pihak terkait seperti satpam ataupun warga sekitar bahwa adanya orang yang masuk ke dalam rumah kita dengan bukti dari gambar yang di tangkap oleh kamera.

Teknologi yang digunakan adalah dengan menggunakan Artificial Inteligent (AI) untuk mendeteksi manusia, dengan AI Deep Learning ini mempermudah kita untuk mendeteksi manusia dan non-manusia. Penelitian sebelumnya mengenai deteksi manusia dengan menggunakan Deep Learning ini telah dilakukan oleh Sireesha et al dengan judul Human Detection in Aerial Images using Deep Learning Techniques telah berhasil membuat sebuah model deteksi manusia pada gambar yang dihasilkan oleh drone dengan hasil deteksi yang memiliki akurasi tinggi pada objek citra udara [2]. Namun dalam penelitian ini cara yang di gunakan untuk mendeteksi manusia melalui sensor kamera yang terpasang di depan rumah yang langsung terhubung dengan Telegram sebagai media notifikasi.

## 2. METODE PENELITIAN

Langkah-langkah yang dilakukan pada penelitian ini adalah seperti yang terlihat pada Gambar 1.1 berikut :



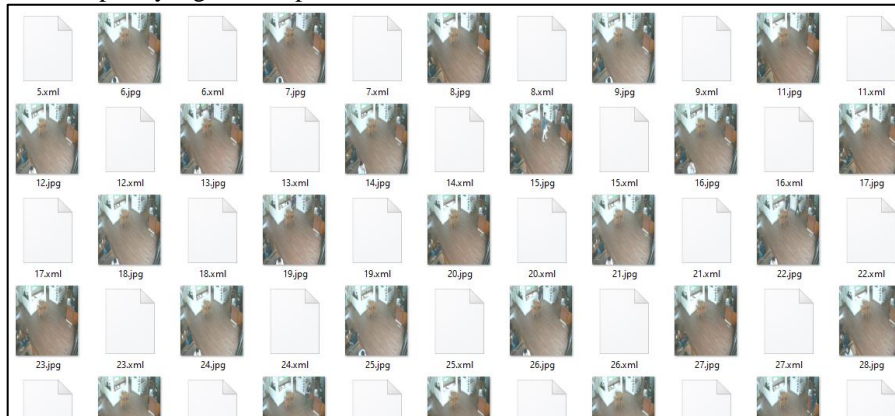
**Gambar 1.** Langkah Penelitian

### 2.1 Pengumpulan Data

Pengumpulan data dalam konteks deep learning merupakan langkah krusial untuk membangun model yang akurat dan handal. Salah satu metode yang digunakan adalah mengambil data awal berupa rekaman video dari sistem CCTV. Proses ini melibatkan ekstraksi frame-frame dari video yang kemudian diolah menjadi gambar-gambar individu. Setiap gambar ini kemudian diberi label atau anotasi sesuai dengan informasi yang ingin diperoleh dari model. Proses labeling ini penting untuk melatih model dalam mengenali pola, objek, atau fitur yang spesifik dalam setiap gambar. Dengan demikian, pengumpulan data tidak hanya sebatas mengonversi video menjadi gambar, tetapi juga melibatkan tahapan penting dalam memberi makna pada data yang dihasilkan agar model deep learning dapat belajar dengan lebih efektif.

Melalui pengumpulan data yang terstruktur dan diberi label dengan cermat, model deep learning dapat memperoleh informasi yang kaya dan representatif. Proses inilah yang memungkinkan model untuk memahami karakteristik dari setiap gambar dan membangun pola yang diperlukan untuk mengenali objek atau fenomena tertentu. Dengan memiliki dataset yang terdiversifikasi dan terlabel dengan baik, model deep learning dapat dilatih secara optimal untuk menghasilkan prediksi yang akurat dan responsif terhadap berbagai situasi yang mungkin terjadi. Dengan demikian, pengumpulan data bukan hanya sekadar proses mengumpulkan informasi, tetapi fondasi utama dalam mengembangkan model deep learning yang dapat memberikan hasil yang berkualitas tinggi. Total dataset yang digunakan berupa data gambar dengan pembagian 312 data train dan 35 data test.

Dataset yang digunakan seperti yang terlihat pada Gambar 2 berikut.



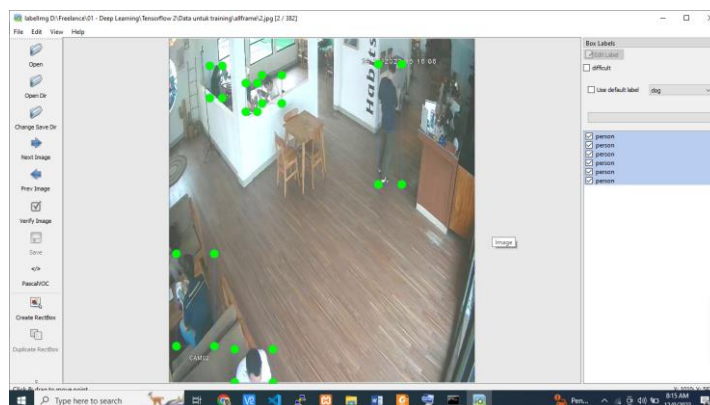
**Gambar 2.** Dataset

## 2.2 Labeling Gambar

Dalam ranah penelitian kecerdasan buatan, terdapat pendekatan yang semakin populer dalam mengatasi masalah pengenalan citra, yaitu penggunaan jaringan neural konvolusional (CNNs). Jaringan ini terinspirasi oleh struktur visual korteks manusia dan telah menunjukkan keunggulan dalam memproses informasi citra. Misalnya, pada analisis citra medis, CNNs mampu mengidentifikasi pola yang kompleks dalam data gambar seperti CT scans atau MRI. Penggunaan CNNs dalam pengenalan citra juga muncul dalam berbagai domain seperti pengenalan wajah, deteksi objek, dan pengolahan citra geospasial, menawarkan potensi untuk pemrosesan yang lebih efisien dan akurat dalam skenario-skenario yang beragam.

Selain itu, penggunaan alat bantu seperti LabelImg atau alat pelabelan lainnya memiliki peran yang signifikan dalam meningkatkan kualitas dataset yang digunakan untuk melatih model CNNs. Proses pelabelan gambar ini memungkinkan penandaan objek atau fitur penting dalam gambar yang menjadi fondasi utama dalam melatih model. Dengan anotasi yang teliti dan terstruktur, dataset yang dihasilkan mampu memberikan informasi yang diperlukan bagi model untuk mempelajari representasi yang lebih baik dari entitas visual yang ada. Kombinasi antara pendekatan jaringan neural konvolusional dengan dataset yang terlabel dengan baik menjadi inti dalam membangun model yang andal dalam pengenalan citra.

Proses pelabelan gambar menggunakan LabelImg seperti yang terlihat pada Gambar 3. berikut.

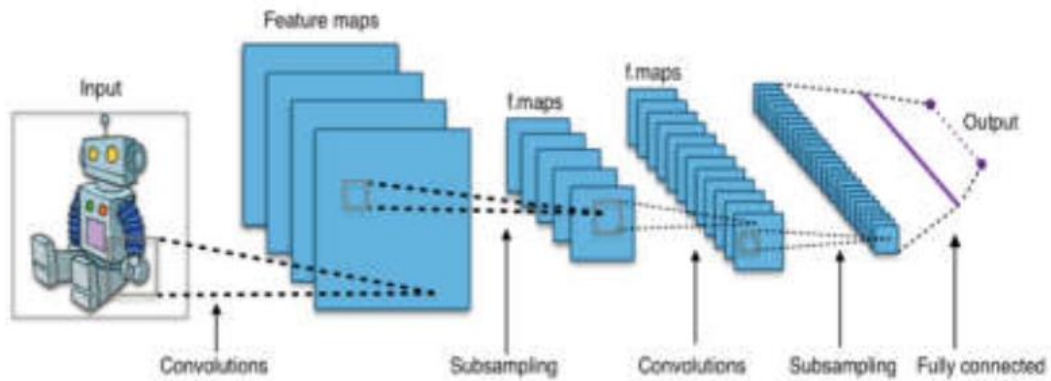


**Gambar 3.** Proses labelling gambar



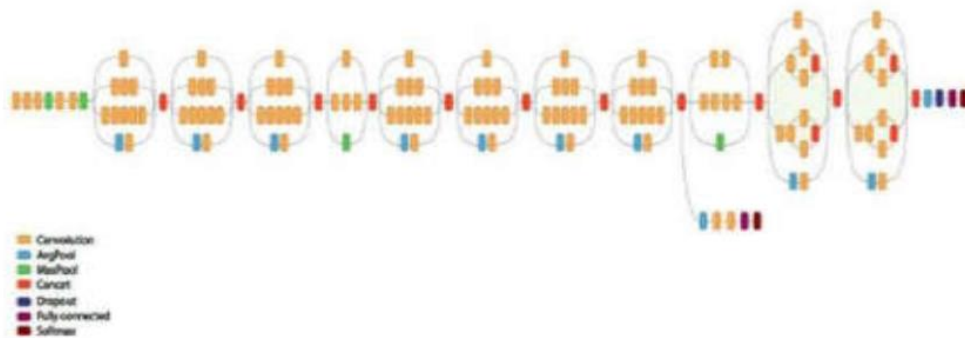
2.3 Konfigurasi Tensorflow

Model pelatihan yang diterapkan dalam studi ini memanfaatkan struktur Convolutional Neural Network (CNN) Inception-V3. Metode optimasi yang digunakan mengadopsi algoritma Adam, sementara loss function yang diimplementasikan adalah binary cross entropy. Secara mendasar, Inception-V3 merupakan turunan dari arsitektur Convolutional Neural Network (CNN). Fungsi dan prinsip kerjanya diuraikan dalam ilustrasi pada Gambar 4 di bawah ini.



**Gambar 4.** Convolutional Neural Network (CNN)

Langkah awal terdiri dari pemrosesan gambar input melalui proses konvolusi. Hasil dari tahapan ini menjadi masukan bagi langkah berikutnya, dan seterusnya hingga menghasilkan output akhir. Inception-V3 menggunakan bobot dari dataset Imagenet dan rancangannya tersaji dalam Gambar 5 di bawah ini.



**Gambar 5.** Arsitektur Inception-V3

Pada struktur inception-V3 yang ditampilkan dalam Gambar 4, beberapa tahapan dilakukan, termasuk proses konvolusi, Average Pooling, MaxPooling, dropout, fully connected, dan softmax. Dalam penelitian ini, prinsipnya mengadopsi langkah-langkah serupa, namun perbedaannya terletak pada pemilihan fungsi aktivasi yang berbeda, yaitu relu dan sigmoid. Hal ini disebabkan oleh fokus klasifikasi dalam penelitian ini yang terdiri dari dua kelas, yang dikenal sebagai binary classification.

Adapun spesifikasi perangkat keras yang diterapkan dalam studi ini melibatkan Intel-Core I3 @ 2.20GHz, 8 Gb DDR4 SDRAM, serta NVIDIA GPU. Sedangkan perangkat lunak yang digunakan meliputi sistem operasi Windows 10, dengan penggunaan Python versi 3.6 dan TensorFlow versi 1.12 dalam penelitian ini.

2.4 Training

Langkah utama dalam jaringan neural adalah melatih model, di mana seluruh dataset diberikan latihan untuk mengenali dan mempelajari pola-pola pada rambu jalur evakuasi dan rambu alat pemadam api. Semua langkah yang terdapat pada CNN akan aktif saat proses pelatihan model. Peneliti memanfaatkan model pra-pelatihan `ssd_mobilenet_v1_coco` yang memiliki kecepatan sekitar 30 gambar per milidetik. Tujuan akhir dari proses pelatihan adalah mendapatkan model yang mampu mendeteksi objek dengan tingkat akurasi yang tinggi. Proses latihan diawali dengan menjalankan perintah yang diilustrasikan pada Gambar 5.20, yakni perintah untuk melatih model.



Model hasil latihan akan disimpan dalam direktori latihan sesuai dengan sintaks `train_dir=training`. Konfigurasi pipeline yang digunakan ada dalam direktori latihan dengan nama file `rambu_v1.config`. Setelah perintah itu dijalankan, proses pelatihan dimulai dari langkah 0. Untuk memantau proses pelatihan, peneliti dapat menggunakan modul yang telah tersedia pada TensorFlow, yakni Tensorboard. Perintah untuk menjalankan Tensorboard melalui command prompt diperlihatkan dalam Gambar 5.21.

Setelah menjalankan perintah tersebut, Tensorboard dapat diakses dengan membuka alamat `localhost:6006` pada browser. Tensorboard merujuk pada file checkpoint yang dihasilkan selama proses pelatihan dalam direktori latihan. Pada Tensorboard, peneliti dapat memonitor berbagai grafik ketika proses pelatihan berlangsung, seperti grafik `total_loss` dan grafik `global_step`. Gambar 5.22 menampilkan tampilan awal dari Tensorboard.

### 2.5 Export Graph Model

Proses eksportasi grafik (graph) dalam konteks pelatihan TensorFlow menjadi tahap penting dalam pengembangan model. Pada titik ini, setelah pelatihan model selesai, langkah berikutnya adalah mengekspor arsitektur dan konfigurasi model ke dalam sebuah file yang dapat digunakan untuk inferensi di lingkungan yang berbeda atau pada perangkat yang berbeda. Tahap ini dikenal sebagai ekspor grafik, di mana model yang sudah dilatih diubah menjadi representasi yang lebih ringkas dan portabel. Proses ini melibatkan pengambilan informasi penting dari model yang telah dilatih, termasuk struktur arsitektur, bobot (weights), dan konfigurasi lainnya, untuk kemudian disimpan dalam sebuah file yang dapat diakses untuk keperluan pengujian atau implementasi pada platform lain.

Setiap detail dalam struktur model yang diekspor haruslah sesuai dengan kebutuhan aplikasi atau lingkungan implementasi selanjutnya. Hal ini mencakup penyesuaian format file output dan penamaan yang jelas agar kompatibel dengan lingkungan yang akan menerapkan model tersebut. Proses eksportasi grafik ini juga mempertimbangkan performa model secara keseluruhan, mencari keseimbangan antara ukuran file yang dihasilkan dan kinerja yang optimal saat model dijalankan untuk tujuan inferensi. Dengan demikian, tahapan eksportasi grafik dalam proses pelatihan TensorFlow bukan hanya sekadar membuat model dapat digunakan secara independen, tetapi juga mempertimbangkan aspek keterjangkauan dan performa saat model diterapkan di berbagai lingkungan yang berbeda.

### 2.6 Test Model

Langkah penting setelah melatih model dalam lingkungan TensorFlow adalah menguji performa model menggunakan data uji yang diambil dari video percobaan. Proses pengujian ini melibatkan penggunaan sejumlah gambar dari video percobaan yang belum pernah dilihat oleh model sebelumnya. Data uji ini dimanfaatkan untuk mengukur seberapa baik model dapat menggeneralisasi pengetahuan yang diperoleh selama pelatihan terhadap situasi yang belum pernah ditemuinya sebelumnya. Dengan memasukkan gambar-gambar baru dari video percobaan, kita dapat mengevaluasi tingkat akurasi, keandalan, dan responsivitas model terhadap kondisi yang belum dikenal.

Proses pengujian ini bertujuan untuk mengidentifikasi kinerja model di luar lingkup data yang digunakan selama pelatihan. Dengan mendapatkan pemahaman yang lebih baik tentang bagaimana model merespons situasi baru dari data uji, kita dapat menentukan sejauh mana model dapat diterapkan dalam kondisi nyata. Analisis terhadap hasil pengujian ini memberikan wawasan yang berharga untuk memperbaiki dan mengoptimalkan model, serta mengeksplorasi cara-cara untuk meningkatkan kinerja model dalam menghadapi variasi data yang lebih luas. Dalam konteks pengembangan model TensorFlow, pengujian dengan data uji dari video percobaan merupakan langkah krusial dalam memvalidasi kemampuan adaptasi dan generalisasi dari model yang telah dilatih.

### 2.7 Integrasi Telegram

Salah satu aspek menarik dalam pengembangan model deteksi adalah integrasi hasil deteksi dengan platform eksternal, seperti Telegram, untuk mengirimkan notifikasi secara real-time ketika terjadi deteksi objek. Tahapan integrasi ini memungkinkan hasil deteksi yang diperoleh dari model untuk dikomunikasikan dengan platform pesan instan, seperti Telegram, guna memberikan informasi secara cepat kepada pengguna terkait adanya objek yang dideteksi. Proses ini melibatkan pengaturan modul khusus yang bertugas menyampaikan hasil deteksi ke platform Telegram, sehingga pengguna atau pengelola dapat menerima notifikasi secara langsung saat terjadi deteksi objek, dalam konteks ini, orang yang lewat.

Implementasi integrasi antara hasil deteksi dengan layanan pesan instan seperti Telegram memungkinkan pengguna untuk segera menanggapi kejadian yang terdeteksi oleh model. Dengan menerima notifikasi secara real-time, pengguna memiliki kesempatan untuk bertindak cepat dan merespons situasi yang terjadi. Selain itu, integrasi ini juga memberikan kemungkinan untuk mengadaptasi model secara dinamis berdasarkan umpan balik yang diterima



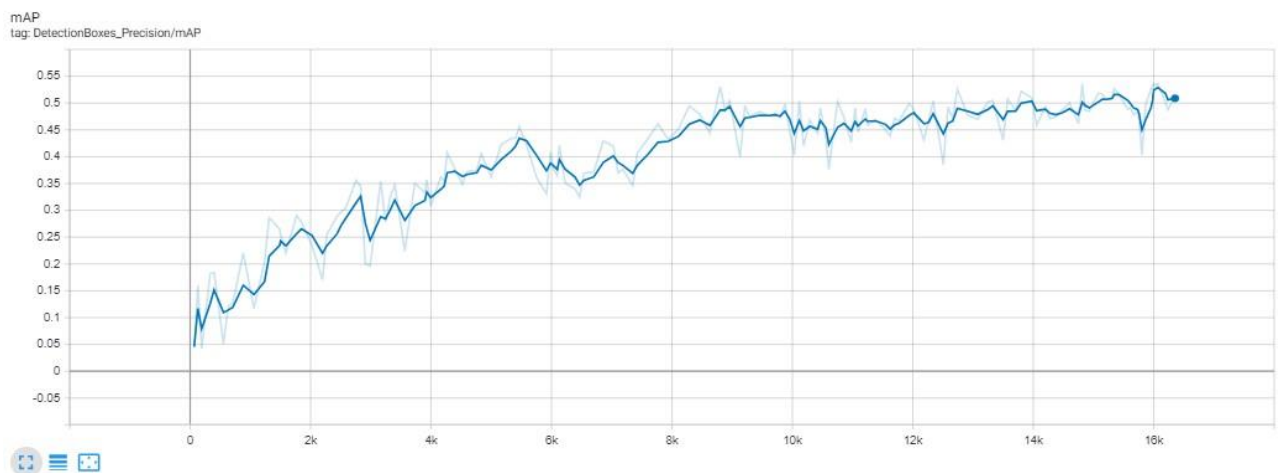
dari pengguna, sehingga memungkinkan peningkatan kontinu terhadap akurasi dan responsivitas model dalam mendeteksi objek tertentu, dalam hal ini, orang yang lewat. Tahap integrasi hasil deteksi dengan platform pesan instan seperti Telegram merupakan langkah penting dalam mengembangkan sistem deteksi yang tidak hanya memiliki kemampuan akurasi, tetapi juga memberikan respons real-time kepada pengguna terkait kejadian yang terdeteksi.

### 3. HASIL DAN PEMBAHASAN

#### 1. Hasil Nilai Precision

Hasil dari evaluasi model deteksi, terutama dalam mengukur presisi kotak deteksi (detection box precision) dengan menggunakan Mean Average Precision (MAP), menunjukkan pencapaian yang signifikan dengan skor akhir sebesar 0.5131. Skor ini menggambarkan seberapa baik model dapat mengidentifikasi dan menempatkan kotak deteksi dengan tepat pada objek yang ada dalam gambar. Dengan skor MAP sebesar ini, model berhasil menunjukkan kemampuan yang konsisten dalam menemukan dan mengelompokkan objek secara presisi, memvalidasi keandalan serta keakuratannya dalam melakukan tugas deteksi. Pentingnya skor MAP sebesar 0.5131 ini juga tercermin dari total langkah (steps) yang diambil oleh model selama proses pelatihan, dengan total 16.35K langkah, yang menunjukkan kompleksitas dan kerumitan dari proses pembelajaran yang diperlukan untuk mencapai tingkat presisi yang tinggi.

Hasil akhir sebesar 0.5131 pada metrik MAP memberikan gambaran yang positif tentang kemampuan model deteksi dalam mengenali objek secara presisi. Skor yang diperoleh tersebut merupakan hasil dari proses iteratif yang melibatkan ribuan langkah dalam melatih model, menegaskan komitmen untuk mencapai tingkat presisi yang tinggi. Dengan demikian, total step sebesar 16.35K yang ditempuh oleh model merupakan refleksi dari upaya yang intensif dalam melatih model untuk mencapai kinerja yang optimal dalam tugas deteksi objek, dan sekaligus menjadi indikator kesungguhan dalam meningkatkan presisi model deteksi. Hasil akhir metrik MAP seperti yang terlihat pada Gambar 5. dibawah ini.



**Gambar 5.** mAP Detection Boxes Precision

Intersection over Union (IoU) adalah sebuah metrik yang digunakan dalam evaluasi performa pada tugas deteksi objek dalam pengolahan citra. IoU mengukur sejauh mana kotak deteksi yang dihasilkan oleh model tumpang tindih atau saling bersinggungan dengan kotak deteksi yang menjadi acuan atau ground truth, umumnya pada dataset yang sudah dianotasikan oleh manusia.

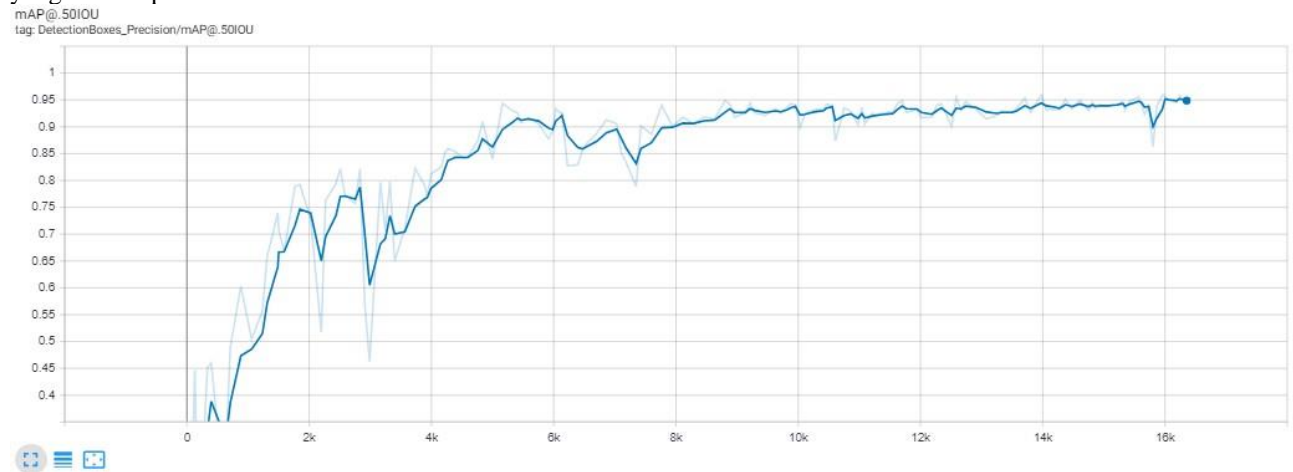
IoU dihitung dengan membandingkan luasan area yang tumpang tindih antara dua kotak deteksi (kotak prediksi dan ground truth) dengan luasan total gabungan dari kedua kotak tersebut. Persentase dari area tumpang tindih ini memberikan informasi tentang seberapa baik posisi dan ukuran dari kotak deteksi yang dihasilkan oleh model jika dibandingkan dengan kotak deteksi yang sebenarnya. Semakin tinggi nilai IoU, semakin baik kotak deteksi yang dihasilkan oleh model sesuai dengan kotak deteksi yang sebenarnya, yang mengindikasikan performa yang lebih baik dalam mendeteksi objek.

Jadi, IoU digunakan sebagai metrik untuk mengevaluasi seberapa baik model dalam menghasilkan kotak deteksi yang tepat dan sesuai dengan objek yang sebenarnya dalam gambar, memberikan informasi tentang seberapa akurat dan presisi model dalam tugas deteksi objek. Hasil evaluasi model deteksi dengan metrik Detection Box Precision



MAP @.50IOU menampilkan performa yang sangat mengesankan, dengan mencapai skor akhir sebesar 0.9433. Skor ini menunjukkan kemampuan model dalam menghasilkan kotak deteksi dengan tingkat presisi yang tinggi ketika diberlakukan batasan Intersection over Union (IOU) sebesar 0.50. Keakuratan dan ketepatan dalam mengidentifikasi objek pada gambar tercermin dengan jelas melalui skor MAP yang mencapai level yang sangat tinggi. Hasil akhir yang mencapai skor 0.9433 pada metrik MAP @.50IOU ini memberikan indikasi bahwa model mampu mengenali objek dengan presisi yang luar biasa, memberikan keandalan yang sangat tinggi dalam tugas-tugas deteksi yang mengedepankan ketelitian.

Skor akhir sebesar 0.9433 pada metrik Detection Box Precision MAP @.50IOU merupakan pencapaian yang luar biasa, menegaskan kualitas unggul model deteksi yang telah dikembangkan. Tingkat presisi yang sangat tinggi ini menjadi indikasi kemajuan yang signifikan dalam pengenalan objek dengan tingkat keakuratan yang sangat tinggi pada batasan IOU 0.50. Dengan demikian, total langkah sebesar 16.35K yang ditempuh oleh model mencerminkan komitmen yang besar dalam melatih model untuk mencapai kinerja yang superior dalam tugas deteksi objek, khususnya dalam menghasilkan kotak deteksi dengan presisi yang tinggi pada tingkat IOU yang ditentukan seperti yang terlihat pada Gambar 6. dibawah ini.

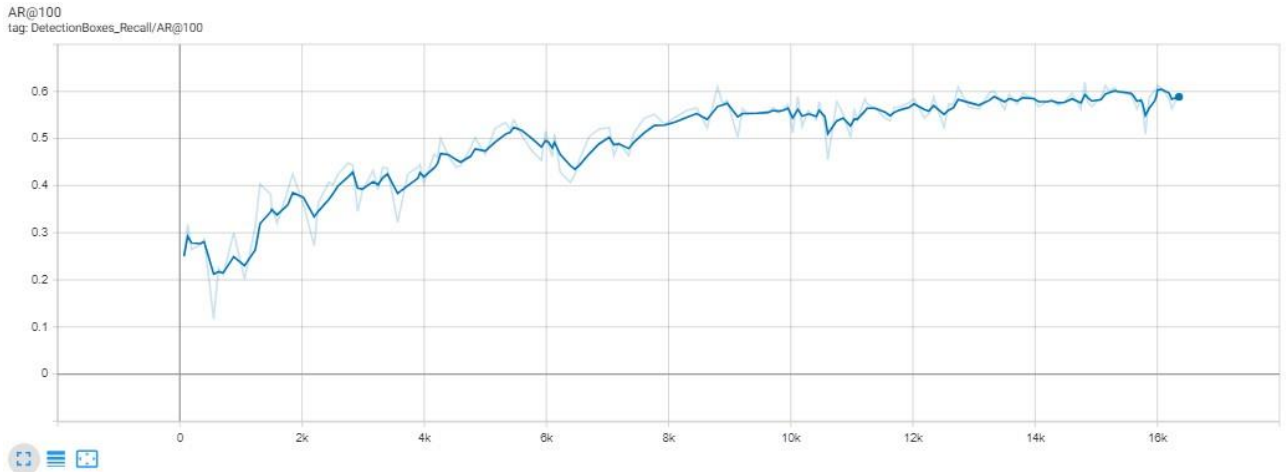


**Gambar 6.** mAP dengan 0.50 IOU

## 2. Hasil Skor Recall

Hasil evaluasi model deteksi menggunakan metrik Detection Box Recall @100 menunjukkan pencapaian yang penting, dengan skor akhir mencapai 0.5963. Skor ini mencerminkan kemampuan model dalam mengenali sejumlah besar objek yang sebenarnya ada pada tingkat kepercayaan (confidence) sebesar 100%. Skor Recall yang mencapai 0.5963 mengindikasikan kemampuan model untuk mengidentifikasi sejumlah besar objek yang benar-benar ada dalam gambar, menjelaskan sejauh mana model mampu menemukan objek-objek yang sebenarnya pada tingkat kepercayaan tertentu. Dengan pencapaian skor Recall yang signifikan ini, model mampu memberikan jaminan dalam mengenali sebagian besar objek yang ada dengan tingkat kepercayaan yang tinggi.

Skor akhir sebesar 0.5963 pada metrik Detection Box Recall @100 menggambarkan kualitas unggul dari model deteksi yang telah dikembangkan. Capaian ini menunjukkan kemampuan model dalam mengingat atau menemukan kembali sejumlah besar objek yang sebenarnya ada pada level kepercayaan 100%. Proses pelatihan sebanyak 16.35K langkah menandakan dedikasi yang tinggi dalam melatih model untuk menghasilkan kinerja yang optimal dalam tugas deteksi objek pada tingkat kepercayaan yang tinggi. Dengan skor Recall sebesar 0.5963 ini, model memberikan keyakinan yang kuat dalam kemampuannya untuk mengenali sebagian besar objek yang ada dalam gambar dengan tingkat kepercayaan yang tinggi. metrik Detection Box Recall @100 seperti yang terlihat pada Gambar 7. dibawah ini.

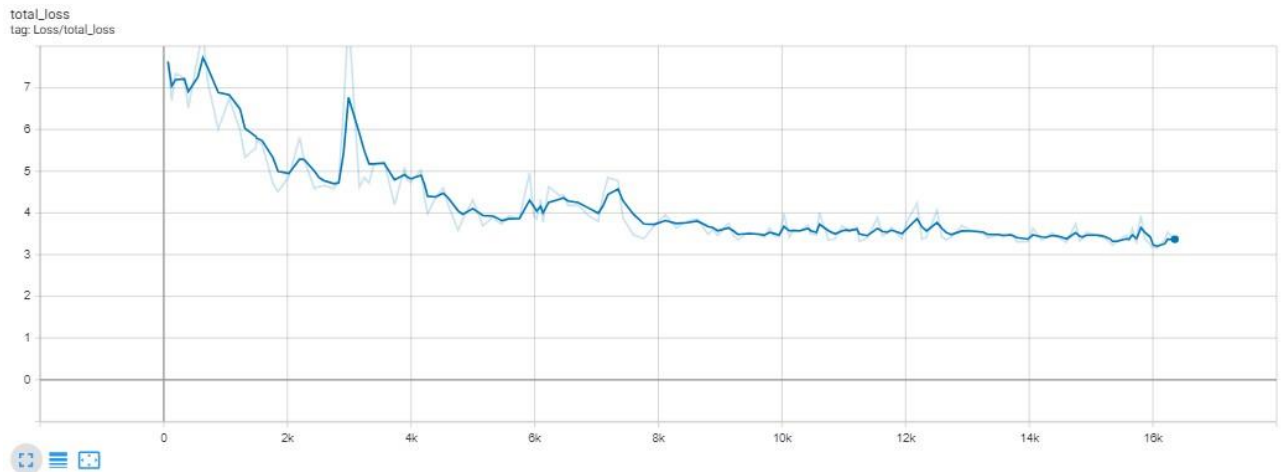


**Gambar 7.** Detection Boxes Recall @100

**3. Hasil Skor Loss**

Total loss adalah metrik yang mencakup keseluruhan kerugian atau kesalahan yang diukur oleh model selama proses pelatihan. Hal ini mencakup kombinasi dari berbagai jenis loss function yang digunakan dalam proses pelatihan model. Total loss memberikan gambaran menyeluruh tentang seberapa baik atau buruk performa model dalam menyesuaikan diri dengan data pelatihan yang diberikan. Pentingnya total loss dalam laporan terletak pada kemampuannya sebagai indikator utama dalam mengukur kemajuan selama pelatihan. Pengamatan dan analisis terhadap perubahan total loss dari iterasi ke iterasi dapat memberikan wawasan yang berharga tentang apakah model sedang mengalami konvergensi, apakah ada overfitting, atau apakah diperlukan penyesuaian lain dalam arsitektur atau parameter model.

Total loss menjadi indikator kritis dalam mengevaluasi kinerja dan keefektifan model selama proses pelatihan. Selain sebagai penunjuk kemajuan, total loss juga menjadi bahan evaluasi apakah model telah mencapai tingkat kestabilan yang diinginkan atau masih memerlukan peningkatan. Pengamatan terhadap total loss membantu dalam menyesuaikan proses pelatihan, seperti mengatur parameter optimasi, memilih atau menyesuaikan fungsi loss yang digunakan, atau bahkan melakukan pre-processing terhadap data. Dengan total loss sebesar 3.362 pada total 16.35K langkah yang ditempuh oleh model, nilai ini memberikan informasi terkait performa model secara keseluruhan selama proses pelatihan, yang nantinya dapat memberikan panduan untuk perbaikan atau peningkatan model di masa depan. Grafik total loss dapat dilihat pada Gambar 8. dibawah ini.



**Gambar 8.** Total Loss

**4. Hasil Pengujian Model**



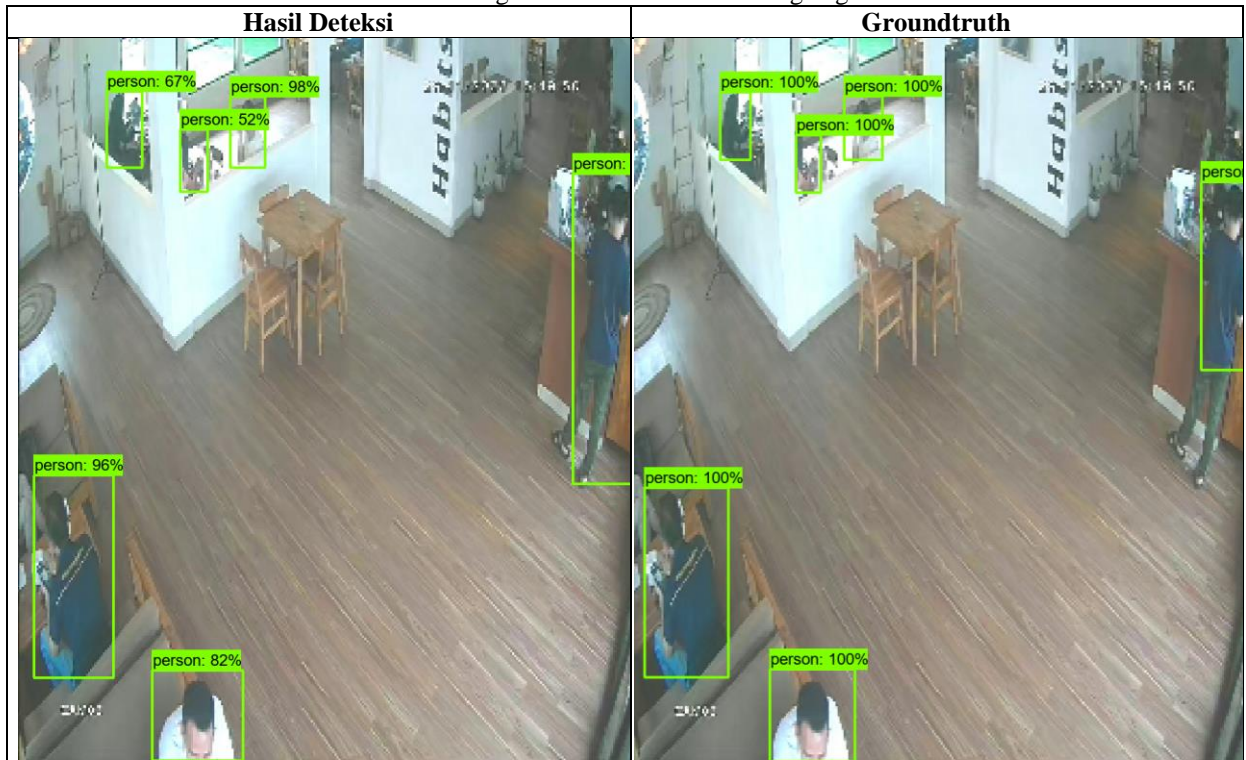


Evaluasi model menggunakan gambar uji (test image) adalah langkah krusial dalam menilai performa deteksi objek yang dihasilkan oleh model. Dalam proses ini, gambar-gambar uji yang telah diambil dari dataset diuji dengan model untuk mengidentifikasi apakah model dapat mengenali dan menempatkan kotak deteksi secara tepat di sekitar objek yang ada pada gambar. Detections\_Left menunjukkan hasil deteksi yang diberikan oleh model, sementara Groundtruth\_Right merupakan kotak deteksi yang dianggap sebagai referensi atau 'ground truth' yang dihasilkan oleh manusia. Perbandingan antara hasil deteksi model dengan kotak deteksi ground truth ini memberikan gambaran seberapa baik model mampu mengenali dan menempatkan kotak deteksi pada objek yang sebenarnya ada dalam gambar.

Hasil evaluasi model yang ditampilkan dalam Detections\_Left dan Groundtruth\_Right memberikan informasi yang penting tentang kemampuan model dalam mengenali objek. Ketika terjadi kesesuaian antara kotak deteksi model dengan kotak deteksi ground truth, menandakan bahwa model berhasil mengidentifikasi objek dengan baik. Namun, ketidaksesuaian antara kedua kotak deteksi tersebut dapat memberikan indikasi tentang area di mana model mungkin perlu perbaikan atau peningkatan dalam pengenalan objek. Evaluasi ini menjadi dasar untuk menganalisis performa model dalam konteks deteksi objek pada gambar, yang nantinya akan memberikan panduan untuk perbaikan dan pengembangan lebih lanjut.

Hasil dari detection dan groundtruth seperti yang terlihat pada Tabel 1. dibawah ini.

**Tabel 1.** Perbandingan antara hasil deteksi dengan groundtruth



**5. Hasil Integrasi AI dan Arduino**

Pencapaian sistem dalam mendeteksi objek, khususnya orang, dan kemudian menginisiasi aktuasi alarm melalui trigger ke perangkat Arduino merupakan langkah signifikan dalam implementasi deteksi objek untuk tujuan keamanan. Dengan deteksi objek yang berhasil, sistem telah berhasil mengenali kehadiran orang dalam suatu ruang atau area tertentu, dan kemudian meneruskan informasi ini untuk memicu respons yang diinginkan, seperti pengaktifan alarm keamanan. Proses trigger yang diarahkan ke Arduino menandakan integrasi yang berhasil antara sistem deteksi objek dan teknologi perangkat keras, memungkinkan adanya respons yang cepat dan terukur terhadap kehadiran orang dalam lingkungan yang termonitor.

Keberhasilan sistem dalam mengaktifkan alarm melalui trigger ke Arduino setelah mendeteksi kehadiran orang memberikan dasar untuk penerapan deteksi objek dalam skenario keamanan. Respons yang dihasilkan oleh sistem memungkinkan tindakan cepat dalam situasi yang memerlukan kehadiran atau pergerakan orang yang tidak



diinginkan atau mencurigakan. Integrasi antara deteksi objek dan aktivasi alarm oleh perangkat keras seperti Arduino menunjukkan potensi sistem untuk meningkatkan sistem keamanan dengan memberikan respons otomatis dalam mendeteksi situasi tertentu, meningkatkan tingkat keselamatan dan keamanan lingkungan yang terpasang sistem deteksi ini.

#### **6. Hasil Notifikasi Telegram**

Integrasi sistem keamanan dengan deteksi objek yang dikombinasikan dengan pengiriman notifikasi melalui platform pesan instan, seperti Telegram, merupakan langkah progresif dalam memperkuat sistem keamanan modern. Dalam konteks ini, sistem deteksi objek yang telah diimplementasikan dalam sistem keamanan memungkinkan identifikasi dan pemantauan aktivitas yang mencurigakan atau tidak diinginkan, seperti deteksi pergerakan manusia atau objek tertentu dalam suatu area. Ketika deteksi tersebut terjadi, sistem secara otomatis mengirimkan notifikasi langsung melalui Telegram, memberikan informasi cepat kepada pengguna atau pemilik sistem tentang kejadian yang terdeteksi. Integrasi ini memberikan tingkat responsivitas yang tinggi, memungkinkan tindakan cepat atau respons yang diperlukan dalam situasi keamanan tertentu.

Keberhasilan integrasi antara sistem keamanan yang dilengkapi dengan deteksi objek dan pengiriman notifikasi ke Telegram memberikan dampak positif dalam meningkatkan responsivitas dan efektivitas sistem keamanan. Notifikasi yang langsung dikirimkan ke platform pesan instan seperti Telegram memberikan kemudahan akses kepada pengguna, memungkinkan mereka untuk segera mengetahui kejadian yang terdeteksi tanpa keterlambatan. Hal ini tidak hanya mempercepat respons terhadap situasi keamanan yang terjadi tetapi juga meningkatkan kemampuan untuk mengelola dan mengawasi situasi-situasi yang memerlukan perhatian khusus. Integrasi yang berhasil ini menandakan evolusi sistem keamanan menuju respons yang lebih real-time dan terkoordinasi dalam mendeteksi serta menanggapi situasi keamanan yang berpotensi membahayakan.

## **4. KESIMPULAN**

Dalam penelitian ini, telah dilakukan serangkaian langkah yang menunjukkan evolusi sistem deteksi objek dan integrasi dengan teknologi keamanan modern. Dari integrasi deteksi objek dengan sistem keamanan yang dapat mengaktifkan alarm hingga pengiriman notifikasi real-time melalui Telegram saat terjadi deteksi, penelitian ini menunjukkan kemajuan signifikan dalam pengembangan sistem keamanan yang responsif dan terintegrasi. Langkah-langkah ini menggambarkan potensi besar dalam meningkatkan respons terhadap situasi keamanan dengan mendeteksi keberadaan objek tertentu, seperti manusia, dan memberikan notifikasi segera kepada pengguna melalui platform pesan instan.

Penerapan deteksi objek dalam sistem keamanan telah membawa kontribusi yang signifikan dalam meningkatkan keselamatan lingkungan. Respons otomatis seperti pengaktifan alarm dan notifikasi real-time telah memberikan kesempatan untuk mengatasi potensi ancaman atau situasi mencurigakan dengan cepat. Melalui integrasi ini, sistem keamanan telah menemukan cara untuk memberikan informasi yang tepat waktu kepada pengguna, memungkinkan pengelolaan yang lebih efektif dalam mengawasi dan merespons situasi yang memerlukan perhatian khusus. Dengan demikian, penelitian ini tidak hanya membuktikan kehandalan deteksi objek dalam sistem keamanan, tetapi juga menegaskan peran penting teknologi komunikasi instan dalam meningkatkan responsivitas dan efektivitas sistem keamanan modern.

## **DAFTAR PUSTAKA**

- [1]. B. P. Statistik, "Statistik Kriminal 2019," BPS, Jakarta, 2019.
- [2]. S. W. Rahmian dan A. Hernawan, "Human Detection in Aerial Images using Deep Learning Techniques," *Journal of Robotics and Control (JRC)* Volume 2, Issue 6, November 2021, pp. 462-468, 2022.
- [3]. T. HAPSARI dan A. FITRIANSYAH, "PENERAPAN SISTEM OTOMATISASI RUMAH DENGAN," *Faktor Exacta*, pp. 323-332, 2017.
- [4]. H. Widiyanatha, "Sistem Kontrol Keamanan Rumah Jarak Jauh Berbasis Web Menggunakan Raspberry Pi," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek.*, p. 1-12, 2016.
- [5]. M. I. KURNIAWAN, . U. SUNARYA dan . R. TULLOH, "Internet of Things : Sistem Keamanan Rumah," *ELKOMIKA*, pp. 1 - 15, 2018.
- [6]. S. A. Ramadhan dan L. B. Handoko, "RANCANG BANGUN SISTEM KEAMANAN RUMAH BERBASIS," *Techno.COM*, Vol. 15, No. 2, pp. 117- 124, 2016.



- [7]. D. NATALIANA, S. ANWARI dan M. S. AKBAR, "Implementasi Prototype Sistem Home security dengan Pemanfaatan Kode Akses berbasis Arduino Mega," Jurnal ELKOMIKA, no. Teknik Elektro Institut Teknologi Nasional (ITENAS) Bandung, pp. 119 - 137, 2017.
- [8]. Fikrieabdillah, "Penggunaan Deep Learning untuk Prediksi Churn pada Jaringan Telekomunikasi Mobile," dalam e-Proceeding of Engineering : Vol.3, No.2 Agustus, 2016.
- [9]. T. Nurhikmat, "Implementasi Deep Learning Untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Wayang Golek," Jurnal Teknik ITS. Vol. 5, No. 1, 2020.
- [10]. L. Deng dan D. Yu, "Deep Learning: Methods and Applications," Foundations and Trends® in Signal Processing: Vol. 7 No. 3-4, pp. 197-387, 2021.
- [11]. Y. LeCun, Y. Bengio dan G. Hinton, Deep learning, Neural Network, 2019.
- [12]. S. N. Mashita, "IMPLEMENTASI DEEP LEARNING OBJECT DETECTION RAMBU K3 PADA VIDEO MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN) DENGAN TENSORFLOW," 2020.61
- [13]. F. Chollet, Deep Learning with Python. Shelter Island, New York: Manning Publications Co, 2018.
- [14]. Tensorflow, "Tensorflow Release," [Online]. Available: <https://www.tensorflow.org/?hl=id>.
- [15]. Taufiq, "Deep Learning for Detection Vehicle Number Signs Using Convolutional Neural Network Algoritm Using Python and Tensorflow.," 2018.
- [16]. F. Djuandi, Pengenalan Arduino, Jakarta: Penerbit Elexmedia, 2011.
- [17]. W. M. Fajar dan H. , Mudah Belajar Mikrokontroler Arduino, Bandung: Informatika Bandung, 2017.
- [18]. Artanto, Interaksi Arduino dan LabView, Jakarta: Elex Media Komputindo, 2012.
- [19]. E. J. Istiyanto, Pengantar Elektronika dan Istrumentasi Pendekatan Project Arduino dan Android, Yogyakarta: Penerbit Andi, 2014.
- [20]. S. Muhammad, Panduan Mudah Simulasi dan Praktik: Mikrokontroler Arduino, Yogyakarta: Andi Publisher, 2013.
- [21]. Arduino, "Arduino Get Started," 2018. [Online]. Available: <https://arduinogetstarted.com/tutorials/arduino-button-relay>.
- [22]. R. Rosdian dan A. Wibowo, "Monitoring Suhu Ruangan Server Dengan Fuzzy Logic Metode Sugeno Menggunakan Arduino Dan Sms," SWABUMI. vol 1, pp. 1-2, 2021.
- [23]. M. Syarif dan W. , "Deteksi Kedipan Mata Dengan Haar Cascade Classifier Dan Contour Untuk Password Login," Techno.com 14(4), p. 242-249, 2021.
- [24]. C. Mauko dan S. Tunliu, "Kontrol Arah Gerak Web Kamera (Webcam) Berbasis Web," Jurnal Ilmiah FLASH Volume 2 Nomor 2, pp. 107-115, 2021.
- [25]. R. Putri, "Pengelolaan Citra Dengan Menggunakan Webcam pada Kendaraan Bergerak di Jalan Raya," JIPI (Jurnal Ilmiah Pendidikan Informatika) Volume 1, Nomor 1, pp. 1-6, 2019.
- [26]. R. D. Arifin, "Pengertian Telegram | Sejarah, Fitur, Kelebihan, Fungsi," Jurnal EKSIS Vol 06, pp. 10-20, 2021.
- [27]. A. Maula, " Mengenal Bot Telegram," Jurnal SIMETRIS, Vol 8 No 2., pp. 102-137, 2020.62
- [28]. D. Mulyanto, "Pemanfaatan Bot Telegram Untuk Media Informasi Penelitian," Jurnal Ilmu Komputer dan Teknologi Informasi Volume 12, No. 1, pp. 49-54 , 2020.
- [29]. Y. P. Atmojo, "Bot Alert Snort dengan Telegram Bot API pada Instrusion Detection System: Studi Kasus IDS pada Server Web," Seminar Nasional Sistem Informasi dan Teknologi Informasi 2018, pp. 16-29, 2018.
- [30]. D. Mulyanto, "Pemanfaatan Bot Telegram Untuk Media Informasi Penelitian," Jurnal Ilmu Komputer dan Teknologi Informasi Volume 12, No. 1, pp. 49-54 , 2020.
- [31]. Muhardian, "Membuat Bot Telegram Tanpa Menulis Kode Program (Coding)," petanikode.com (online), 2016.
- [32]. Mauko dan S. Tunliu, "Kontrol Arah Gerak Web Kamera (Webcam) Berbasis Web," Jurnal Ilmiah FLASH Volume 2 Nomor 2, pp. 107-115, 2018