



Pendeteksian Senjata Api pada Manusia dalam Situasi Real-Time Menggunakan Model YOLOv4-Tiny

* Dimas Aliftha Sulthoni ¹, Thoyyibah ²

^{1,2} Teknik Informatika, Program Pascasarjana, Universitas Pamulang, Tangerang Selatan, Banten

Email: ¹dimasaliftasulthoni@gmail.com

ABSTRACT

This research aims to develop a real-time human firearm detection system using the YOLOv4-tiny method. The system is implemented and tested on public security CCTV cameras to enhance responses to potential security threats. The research results indicate that the developed detection system achieves an accuracy level of approximately 95%. Real-time testing successfully detects various types of firearms, including rifles, shotguns, and handguns. This success demonstrates the potential of YOLOv4-tiny as an effective solution for improving public safety with fast and accurate firearm detection. The research makes a significant contribution to security technology development, offering an efficient means to prevent violent incidents and protect communities effectively.

Keywords: CNN; YOLOV4-tiny; RTSP; Object Detection

ABSTRAK

Penelitian ini bertujuan mengembangkan sistem deteksi senjata api pada manusia dalam situasi real-time dengan menggunakan metode YOLOv4-tiny. Sistem ini diimplementasikan dan diuji pada kamera CCTV keamanan publik untuk meningkatkan respons terhadap potensi ancaman keamanan. Hasil penelitian menunjukkan bahwa sistem deteksi yang dikembangkan mencapai tingkat akurasi sekitar 95%. Uji coba dalam situasi real-time berhasil mendeteksi berbagai jenis senjata api, termasuk rifle, shotgun, dan handgun. Keberhasilan ini membuktikan potensi YOLOv4-tiny sebagai solusi efektif dalam meningkatkan keamanan publik dengan deteksi senjata api yang cepat dan akurat. Penelitian ini memberikan kontribusi penting dalam pengembangan teknologi keamanan, yang dapat digunakan untuk mencegah kejadian kekerasan dan melindungi masyarakat secara efisien.

Kata kunci: CNN; YOLOV4-tiny; RTSP; Deteksi Objek

1. PENDAHULUAN

Keamanan publik merupakan salah satu hal yang penting dalam kehidupan masyarakat. Salah satu upaya untuk meningkatkan keamanan publik adalah dengan menggunakan sistem deteksi senjata api. Sistem deteksi senjata api dapat digunakan untuk mendeteksi keberadaan senjata api pada manusia, sehingga dapat mencegah terjadinya tindak kriminalitas yang menggunakan senjata api [1]. Tindak kriminalitas yang menggunakan senjata api merupakan salah satu masalah yang serius di berbagai negara, termasuk Indonesia. Menurut data dari Badan Pusat Statistik (BPS), jumlah kasus

kejahatan dengan senjata api di Indonesia pada tahun 2022 mencapai 10.000 kasus. Jumlah ini meningkat sebesar 10% dari tahun sebelumnya.

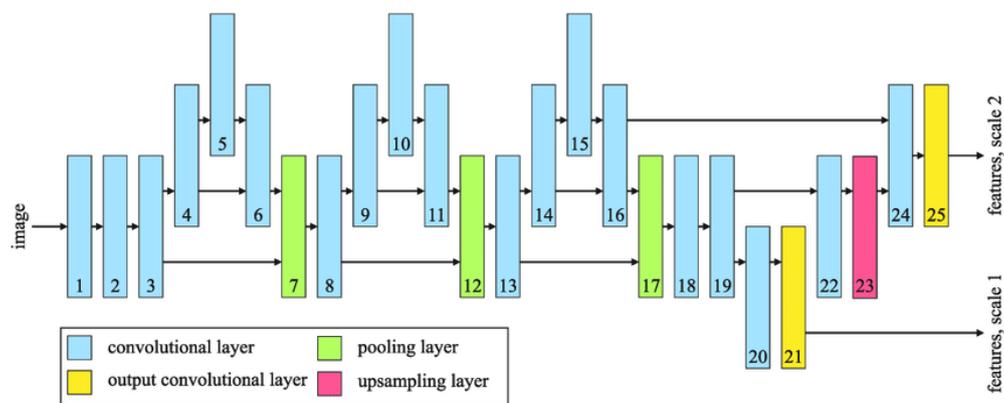
Salah satu faktor yang menyebabkan meningkatnya kasus kejahatan dengan senjata api adalah mudahnya akses terhadap senjata api. Di Indonesia, kepemilikan senjata api masih diperbolehkan untuk keperluan tertentu, seperti olahraga, berburu, dan pengamanan pribadi. Namun, kepemilikan senjata api ini tidak selalu disertai dengan pengawasan yang ketat [2]. Untuk mengatasi masalah ini, perlu dilakukan upaya untuk meningkatkan keamanan publik, salah satunya dengan menggunakan sistem deteksi senjata api. Sistem deteksi senjata api dapat digunakan untuk mendeteksi keberadaan senjata api pada manusia secara realtime. Dengan demikian, sistem ini dapat membantu mencegah terjadinya tindak kriminalitas yang menggunakan senjata api.

Salah satu metode yang dapat digunakan untuk mendeteksi senjata api adalah menggunakan metode deteksi objek berbasis kecerdasan buatan (AI). Metode ini memiliki beberapa kelebihan, seperti dapat mendeteksi objek dengan cepat dan akurat, serta dapat digunakan dalam berbagai kondisi, termasuk dalam situasi realtime. Dalam penelitian ini, kami menggunakan metode YOLOv4-tiny untuk mendeteksi senjata api pada manusia dalam situasi realtime. YOLOv4-tiny merupakan metode deteksi objek yang memiliki ukuran model yang kecil, sehingga dapat diimplementasikan pada perangkat keras dengan sumber daya yang terbatas, seperti Raspberry Pi [3].

CNN adalah salah satu jenis jaringan saraf tiruan (neural network) yang dirancang untuk memproses data yang memiliki topologi seperti grid, misalnya gambar. CNN terdiri dari beberapa layer [4], yaitu: (1) Convolution layer: Layer ini digunakan untuk mengekstrak fitur dari gambar. Fitur-fitur ini direpresentasikan sebagai filter yang diterapkan pada gambar input; (2) Pooling layer: Layer ini digunakan untuk mengurangi ukuran gambar output dari convolution layer. Hal ini dilakukan untuk mengurangi jumlah parameter yang perlu dipelajari oleh jaringan saraf tiruan. (3) Fully connected layer: Layer ini digunakan untuk mengintegrasikan fitur-fitur yang telah diekstrak dari convolution layer.

Cara kerja CNN adalah dengan memodifikasi perhitungan matematika untuk memproyeksikan gambar. Jika neural network biasa mengubah input melalui hidden layer, maka CNN bisa menggunakan komponen convolutional. Komponen convolutional

ini berfungsi untuk mengekstrak fitur-fitur dari gambar input. CNN telah terbukti sangat efektif untuk berbagai tugas pengenalan gambar, seperti klasifikasi gambar, deteksi objek, dan segmentasi gambar. CNN juga telah digunakan untuk berbagai aplikasi lainnya, seperti pengenalan wajah, pengenalan tulisan tangan, dan pengolahan bahasa alami. YOLO adalah *Model* object detection yang bekerja dengan membagi gambar menjadi grid, lalu memprediksi bounding box dan probabilitas untuk masing-masing grid. Model YOLO pertama kali diusulkan oleh Joseph Redmon dan Ross Girshick pada tahun 2015. YOLOv4-Tiny adalah *Model* deteksi objek yang sangat efisien dan memiliki arsitektur yang lebih ringan dibandingkan dengan varian penuh YOLOv4 [5].



Gambar 1. Backbone CNN YOLOv4-tiny [6]

Arsitektur ini dirancang untuk mendeteksi objek dalam gambar dengan cepat dan akurat. Arsitektur YOLOv4tiny terdiri dari 10 lapisan konvolusi, 2 lapisan pooling, dan 1 lapisan output konvolusi. Lapisan konvolusi pertama hingga ketujuh menggunakan kernel 3x3, sedangkan lapisan konvolusi kedelapan dan kesembilan menggunakan kernel 1x1. Lapisan output konvolusi menggunakan kernel 1x1 untuk menghasilkan 3 bounding box dan 1 kelas probabilitas untuk setiap sel grid. Arsitektur YOLOv4tiny memiliki dua skala, yaitu skala 1 dan skala 2. Skala 1 digunakan untuk mendeteksi objek berukuran besar, sedangkan skala 2 digunakan untuk mendeteksi objek berukuran kecil [7].

RTSP adalah sebuah protokol komunikasi jaringan yang dirancang khusus untuk mendukung streaming media real-time. Protokol ini memungkinkan pengguna untuk mengendalikan pemutaran, navigasi, dan interaksi dengan konten multimedia yang disediakan oleh server streaming. RTSP memiliki peran penting dalam memungkinkan layanan streaming video, audio, dan konten multimedia lainnya yang dapat diakses

dengan mudah oleh pengguna akhir. Dalam era konektivitas yang semakin berkembang, integrasi perangkat elektronik menjadi hal yang semakin umum. Salah satu contoh yang menonjol adalah penggunaan protokol RTSP (Real Time Streaming Protocol) untuk membentuk jembatan komunikasi yang efisien antara Raspberry Pi (Raspi) dan sistem Closed-Circuit Television (CCTV). Kombinasi ini membuka pintu bagi berbagai aplikasi yang berkaitan dengan pemantauan real-time, pengawasan, dan pengelolaan keamanan. Raspberry Pi, sebagai platform perangkat keras yang ringkas dan serbaguna, telah memainkan peran yang semakin penting dalam banyak konteks teknologi. Di sisi lain, sistem CCTV memberikan solusi keamanan yang andal dengan kemampuan pemantauan video yang terdistribusi. Integrasi kedua teknologi ini melalui RTSP membuka potensi baru dalam bidang pengawasan keamanan dan pemantauan, serta aplikasi lain yang berkaitan.

Protokol RTSP memungkinkan Raspi untuk berkomunikasi dengan sistem CCTV secara real-time, mengambil aliran video dari kamera-kamera CCTV, dan mengendalikan aliran ini sesuai kebutuhan. Ini memberikan kontrol yang lebih besar kepada pengguna, termasuk kemampuan untuk mengakses feed video, merekam, memutar mundur, dan mengambil tindakan segera ketika diperlukan. Integrasi ini juga memungkinkan penggunaan Raspi sebagai platform pusat untuk mengintegrasikan data sensor, perangkat lain, dan sistem keamanan yang lebih besar.

2. METODE

Penelitian ini dilakukan pada perusahaan Konsultan IT di Tangerang Selatan, Banten pada Bulan Oktober 2023. Adapun tahap Desain sistem yang dilakukan pada penelitian ini, berikut adalah paparan Desain Sistem.

2.1. Persiapan Dataset Gambar

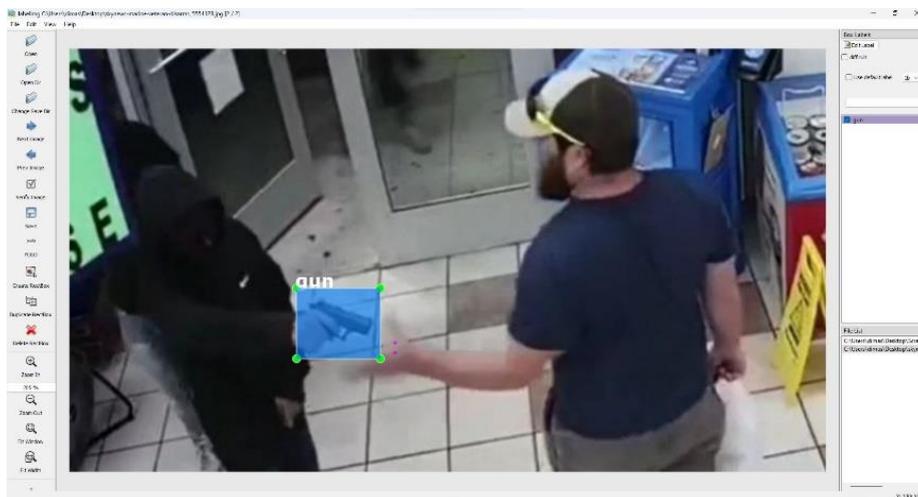
Tahap Persiapan Dataset gambar dilakukan dengan pengambilan gambar dari situs www.kaggle.com dan pengambilan mandiri oleh peneliti. Gambar tersebut selanjutnya dilakukan labeling untuk mendapatkan data dalam bentuk (.txt).

Tabel 1. Jumlah Gambar setiap Kelas

Kelas	Jumlah
Gun	53101
Stair	1790

2.2. Labeling Dataset

Labeling dataset, juga disebut anotasi, adalah proses memberikan informasi batas kotak pada senjata api yang ada di gambar. Untuk mencapai ground-truth boundary, yang akan dibandingkan dengan kotak yang diprediksi, setiap kotak gambar yang dikumpulkan diberi label satu per satu. Nilai Intersection over Union (IoU) akan diperoleh dengan membandingkan kedua ground box tersebut. Labeling dilakukan melalui source code python yang dikompilasi menjadi sebuah aplikasi labeling. Labeling hasil akan diexport ke format YOLO, yaitu.txt, sehingga dapat diolah selama instruksi.



Gambar 2. Proses *Labeling* menggunakan aplikasi *labelimg.py*

2.3. Training Dataset

Dataset yang telah dilabeli pada tahap sebelumnya akan dilatih untuk membentuk pola yang akan menghasilkan berat. *Weights* ini akan digunakan untuk menemukan objek pada gambar. Pendidikan akan dilakukan dengan YOLOV4-tiny. Dalam proses pengolahan data pelatihan, Google Colaboratory digunakan. Berdasarkan diagram pada Gambar 2, YOLO akan menerima input gambar atau video. Keyakinan, nama kelas, dan posisi koordinat bounding box manusia yang terdeteksi oleh sistem akan diberikan sebagai output.

Tabel 2. Spesifikasi *Hardware* untuk Training

<i>Hardware</i>	<i>Spesifikasi</i>
Prosesor	Intel(R) Xeon(R) CPU @ 2.20GHz
RAM	83.5 GB
GPU	NVIDIA A100
Penyimpanan	166.8GB

- Pengolahan Input: YOLO membutuhkan gambar dengan ukuran piksel berkelipatan 32 untuk mendeteksi. Ukuran gambar yang lebih besar akan menghasilkan hasil deteksi yang lebih akurat, tetapi pengolahan akan membutuhkan waktu yang lebih lama. Hal ini juga berlaku sebaliknya. Akibatnya, ukuran piksel gambar yang dimasukkan disesuaikan dengan perangkat yang digunakan untuk pelatihan dan deteksi.
- YOLOv4 menggunakan feature extraction CSPDarknet53-tiny sebagai bagian dari backbone-nya. Feature extraction ini memiliki tiga belas lapisan convolutional berukuran 3×3 , 725×725 field reseptif, dan parameter berukuran 5,45 M. Dengan demikian, CSPDarknet53-tiny adalah model yang ideal untuk mengekstraksi fitur atau backbone detektor.
- Pengolahan Output: YOLO menghasilkan prediksi yang terdiri dari kotak batas (x, y, w, h), keyakinan 55, dan kemungkinan kelas. Pusat kotak dibandingkan dengan sisi sel diwakili oleh koordinat (x,y). Di sisi lain, tinggi dan lebar gambar secara keseluruhan diprediksi oleh koordinat (w,h). Nilai kepercayaan biasanya digambarkan sesuai dengan persamaan 6 dan merupakan representasi dari Intersection over Union (IoU) antara kotak kebenaran dasar dan kotak prediksi. Untuk membuat prediksi akhir, faktor penentunya adalah nilai kepercayaan kelas yang diperoleh, yang didasarkan pada probabilitas kondisional kelas dan nilai kepercayaan kotak. Nilai kepercayaan kelas spesifik untuk setiap kotak memberikan nilai kepercayaan kelas, yang mengkodekan kemungkinan kelas yang muncul di dalam kotak dan seberapa sesuai kotak yang diprediksi dengan objek. Nilai keyakinan adalah nol jika tidak ada objek yang terdeteksi.

$$\text{Confidences Score} = \text{Pr}(\text{Class}) \times \text{IoU}_{\text{truth}}^{\text{Pred}} \quad (1)$$

Intersection over Union (IoU) dapat dihitung dengan membandingkan kotak penghalang kebenaran dan prediksi, yang dapat ditulis dengan persamaan:

$$\text{IoU} = \frac{\text{area of Overlap}}{\text{area of Union}} \quad (2)$$

Selain nilai IoU, diperoleh nilai rata-rata IoU, juga dikenal sebagai ketepatan rata-rata rata-rata (mAP). Pada penelitian ini, mAP@IoU membutuhkan nilai treshold yang

melebihi 0,5 untuk diakui sebagai keberhasilan; jika nilai treshold kurang dari 0,5, hasil dapat dianggap salah. Untuk ketetapan ambang 0,5, diketahui jika:

- Jika $IoU \geq 0,5$ maka klasifikasi objek sebagai True Positive (TP).
- Jika $IoU < 0,5$ maka klasifikasi objek sebagai False Positive (FP).
- Jika ground truth menampilkan gambar dan Model gagal mendeteksi objek maka klasifikasikan sebagai False Negative (FN).

Setiap elemen gambar yang tidak dapat dideteksi disebut True Negative (TN).

Untuk menghitung ketepatan dan recall, nilai di atas dapat digunakan.:

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

Hasil perhitungan IoU menghasilkan nilai Positif Benar, Positif Benar, dan Negatif Benar. Nilai ketepatan dan recall dari hasil pendeteksian senjata dihitung dengan nilai-nilai ini. Selanjutnya, kurva precision-recall menunjukkan nilai precision dan recall. Nilai Average Precision (AP) sendiri didapat dari hasil penghitungan area di bawah kurva untuk setiap kelas yang diidentifikasi oleh sistem. Nilai Average Precision Mean (mAP) sendiri diperoleh dengan merata-ratakan nilai AP untuk seluruh kelas yang diidentifikasi.

2.4. Transfer Learning

Penggunaan transfer learning merupakan strategi yang sangat berguna ketika ingin mengembangkan *Model* deep learning dengan sumber daya terbatas pada perangkat Raspberry Pi setelah melatih *Model* secara komprehensif di platform Google Colab. Dalam konteks ini, *Model* awal dapat dilatih dengan sumber daya yang kuat dan GPU yang tersedia di Google Colab, memungkinkan kita untuk memanfaatkan keunggulan sumber daya yang lebih kuat. Setelah *Model* awal berhasil dilatih, langkah selanjutnya adalah menerapkan transfer learning, di mana pengetahuan yang telah diperoleh dari *Model* awal digunakan untuk fine-tuning *Model* yang lebih ringan dan cocok untuk Raspberry Pi. Pendekatan ini memungkinkan penghematan waktu dan sumber daya berharga, serta memberikan hasil yang lebih efisien saat menerapkan *Model* deep learning pada perangkat keras terbatas seperti Raspberry Pi.

2.5. Implementasi Model

Pada tahap ini adalah tahap akhir dimana Program Transfer Learning ini diolah pada sebuah mikroprosesor yang terhubung dengan CCTV menggunakan protokol RTSP.

3. HASIL DAN PEMBAHASAN

Pada penelitian ini, metode YOLO digunakan untuk mendeteksi manusia. Pertama, dataset yang diperoleh dilabelkan dalam satu kelas, yaitu objek manusia. Setelah labeling selesai, masuk ke tahap pelatihan dan membuat model.

3.1. Pembuatan Dataset

Tujuan pembuatan dataset adalah untuk menyiapkan data yang akan digunakan untuk pelatihan. Data berupa gambar orang bersenjata, baik individu maupun kelompok bersenjata di keramaian, yang diambil secara manual atau melalui internet. Tabel 3 menunjukkan bahwa data yang diperoleh berjumlah 102025 dibagi menjadi 91821 train gambar dan 10204 test gambar.

Tabel 3. Pembagian Dataset

Dataset	Jumlah
Train	91821
Val	10204

3.2. Training Data

Tujuan dari pelatihan data adalah untuk menghasilkan metrik yang akan digunakan dalam proses deteksi manusia. Training data menggunakan YOLOv4-tiny dan dilakukan menggunakan Google Colaboratory. Spesifikasi komputer ditunjukkan pada Tabel 1, yang menunjukkan proses pembelajaran dengan 100000 batch maksimum, ukuran gambar 416, dan beberapa hyper parameter dengan dua kelas. Gambar 3 menunjukkan hasil pelatihan YOLOv4-tiny; mAP terbaik yang diperoleh YOLOv4-tiny sebesar 95,00%.k protokol RTSP, dan pengujian dilakukan dengan berbagai jenis senjata api, seperti pistol, senapan, dan senapan lipat.

Tabel 4. Custom Hyperparameter

Hyperparameter	Value
batch	64
subdivisons	4
width	416
height	416
channels	3
momentum	0.9
decay	0.0005

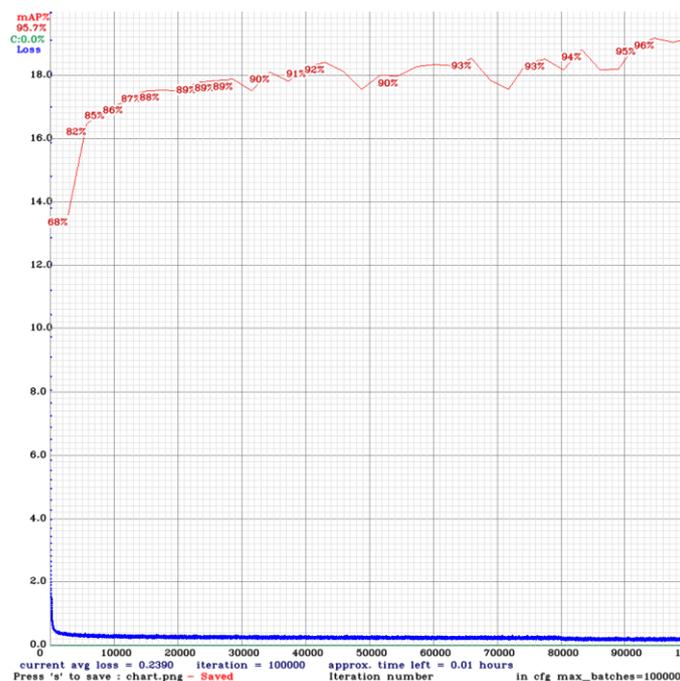
Hyperparameter	Value
angle	0
saturation	1.5
exposure	1.5
hue	.1
learning rate	0.00261
burn in	1000
max batches	100000
policy	steps
steps	80000,90000
scales	.1,.1
random	1

Tabel 5. Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	5022	463
Actual Negative	255	40

Tabel 6. Confusion Matrix 2

Conf.	Value
Conf thresh	0.25
Precision	0.95
Recall	0.92
F1-Score	0.93
Avg. IoU	0.76
mAP	0.95
Avg. Loss	0.24



Gambar 3. Kurva Hasil Training Model YOLOV4-tiny pada 10000 iterasi

3.3. Hasil Deteksi Senjata

Deteksi Manusia menggunakan data yang dikumpulkan secara real-time di lokasi, seperti yang ditunjukkan pada Gambar 4,5, dan 6. Pada Gambar 6, bounding box dari hasil deteksi terlihat.

3.4. Pengujian Performa YOLOV4-tiny

Tujuan pengujian performa adalah untuk menentukan tingkat keakurasian YOLOv4. Dataset uji yang telah diberi anotasi—dataset ini sebelumnya belum pernah digunakan sebagai dataset pelatihan—dibutuhkan untuk uji performa. Dataset ini sekitar 10% dari dataset pelatihan total. Tabel 5 dan 6 menunjukkan matriks kebingungan setelah anotasi data dan kelas dilakukan. Selanjutnya, rumus yang diberikan pada desain sistem digunakan untuk menghitung nilai untuk Recall, Average Precision (AP), skor F1, Intersection over Union (IoU), dan Mean Average Precision. Untuk menguji kinerja, masing-masing Model menggunakan Google Colaboratory, dan spesifikasi hardware untuk masing-masing model ditunjukkan dalam Tabel 1. Di Kantor Keamanan Publik, pengujian dengan berbagai jenis senjata api seperti senjata api api, pistol, dan pistol dilakukan secara real-time melalui CCTV dan protokol RTSP.



Gambar 4. Hasil Deteksi Senjata Handgun



Gambar 5. Hasil Deteksi Senjata Rifle



Gambar 6. Hasil Deteksi Senjata Shotgun

4. KESIMPULAN DAN SARAN

4.1. Kesimpulan

Penulis berhasil menerapkan sistem deteksi objek senjata pada manusia setelah menerapkan dan menguji sistem yang sudah ada. Untuk lebih detail, berikut adalah beberapa kesimpulan:

1. Modul deteksi senjata dalam penelitian ini berhasil mendeteksi senjata pada gambar CCTV.
2. Berdasarkan hasil uji performa YOLOv4-tiny, diperoleh nilai mAP sebesar 95,00%.
3. Pengujian deteksi secara real-time berhasil mendeteksi berbagai jenis senjata api, seperti senjata api api, senapan, dan pistol.

4.2. Saran

Dalam penelitian ini, peneliti menyadari banyak kekurangan yang mungkin bisa lebih dikembangkan lagi. Berikut saran yang di berikan penulis penelitian yaitu menggunakan kamera CCTV yang meliliki sensor infra merah, sehingga ketika di waktu malam hari memberikan hasil video dengan detail objek yang baik.

5. DAFTAR PUSTAKA

- [1] I. A. Dahlan, D. Ariateja, M. A. Arghanie, M. A. Versantariqh, M. David, and U. D. Fatmawati, "Sistem Deteksi Senjata Otomatis Menggunakan Deep Learning Berbasis CCTV Cerdas," *J. Sist. Cerdas*, vol. 4, no. 2, pp. 126–141, Aug. 2021, doi: 10.37396/jsc.v4i2.172.
- [2] R. Parengkuan, D. Antouw, and F. Pongkorung, "Penegakan Hukum Oleh Kepolisian Republik Indonesia Terhadap Penyalahgunaan Kepemilikan Ilegal

- Senjata Api,” *LEX Crim.*, vol. 11, no. 4, pp. 1–13, 2022.
- [3] J. S. W. Hutauruk, T. Matulatan, and N. Hayaty, “Deteksi Kendaraan secara Real Time menggunakan Metode YOLO Berbasis Android,” *J. Sustain. J. Has. Penelit. dan Ind. Terap.*, vol. 9, no. 1 SE-Articles, pp. 8–14, May 2020, doi: 10.31629/sustainable.v9i1.1401.
- [4] J. Du, “Understanding of Object Detection Based on CNN Family and YOLO,” *J. Phys. Conf. Ser.*, vol. 1004, no. 1, p. 12029, 2018, doi: 10.1088/1742-6596/1004/1/012029.
- [5] A. M. Soto i Serrano, Albert López Peña, “YOLO Object Detector for Onboard Driving Images,” *Eng. Informàtica*, 2017, [Online]. Available: <https://ddd.uab.cat/record/181557>.
- [6] R. Pilipović, V. Risojević, J. Božič, P. Bulić, and U. Lotrič, “An Approximate GEMM Unit for Energy-Efficient Object Detection,” *Sensors*, vol. 21, no. 12, 2021, doi: 10.3390/s21124195.
- [7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.