

Deteksi Senjata Multi Kelas Menggunakan Metode YOLOv5s Berbasis Gambar

1st Muhamad Syaepul Huda
Fakultas Informatika
Universitas Telkom

Bandung, Indonesia
mshuda@students.telkomuniversity.ac.id

2nd Tjokorda Agung Budi Wirayuda
Fakultas Informatika
Universitas Telkom

Bandung, Indonesia
cokagung@telkomuniversity.ac.id

Abstrak - Kasus kriminalitas di Indonesia tergolong sangat tinggi, tercatat dari tahun 2019 hingga 2022 ada sebanyak 3.335 orang menjadi korban pembunuhan, meskipun telah dilakukan pengawasan, kerap kali lolos karena penglihatan manusia mempunyai keterbatasan terutama jika melakukan penjagaan secara terus menerus, untuk mengurangi tingkat kejahatan menggunakan senjata, deteksi senjata dengan kecerdasan buatan menjadi salah satu kebutuhan yang penting dalam menjaga keamanan masyarakat. Untuk mengatasi masalah ini, maka diperlukan sistem yang dapat mendeteksi senjata otomatis, pada penelitian ini dilakukan pembangunan sistem objek deteksi menggunakan metode *You Only Look Once* (YOLO), metode ini dipilih karena dapat melakukan pendeteksian dengan cepat dan akurat, pengujian dengan mengatur parameter pada YOLOv5s, model YOLOv5s dengan performa terbaik akan dipilih sebagai objek pendeteksian. Dalam tugas akhir ini menggunakan YOLOv5s pengujian menggunakan parameter dengan skenario yang sudah ditentukan, pada dataset (ARI-DaSCI) memiliki nilai tertinggi pada hasil test skenario 7 dengan nilai *Precision* sebesar 0.923, *Recall* sebesar 0.901, *F1 score* sebesar 0.912, dan *mAP* sebesar 0.947. sementara pada dataset (SNEHIL SANYAL) memperoleh nilai tertinggi pada skenario 4 dengan nilai *Precision* sebesar 0.902, *Recall* sebesar 0.837, *F1 score* sebesar 0.868, dan *mAP* sebesar 0.897. Berdasarkan modifikasi parameter tersebut, jumlah parameter yang tinggi akan berpengaruh pada performa dari model YOLOv5s untuk pendeteksian.

Kata kunci - Deteksi Objek, Kecerdasan Buatan, Senjata Api, Senjata Tajam, YOLO.

I. PENDAHULUAN

A. Latar Belakang

Kejahatan yang terjadi di Indonesia terus meningkat, menurut data Badan Pusat Statistik pada tahun 2018 terdapat 1.024 kasus pembunuhan, tahun 2019 terdapat 964 kasus pembunuhan dan pada tahun 2020 terdapat 898 kasus pembunuhan[1]. Belasan hingga puluhan siswa dan mahasiswa telah terlibat dalam kasus pembunuhan. Bahkan, sebanyak 20 pelajar dilaporkan melakukan kejahatan ini sepanjang tahun 2022. Data Januari

hingga Oktober 2022 menunjukkan polisi menindak 472 laporan kasus pembunuhan[2]. Berdasarkan data dari PUSIKNAS Bareskrim Polri selama empat tahun terakhir, ada lebih dari 3.000 orang telah tercatat menjadi korban dalam kejadian pembunuhan pada data yang diperoleh *e-MP Robinopsnal* Bareskrim Polri. Menurut laporan *e-MP Robinopsnal* Bareskrim Polri pada Jumat, 13 Januari 2023, total korban pembunuhan yang tercatat dari tahun 2019 hingga 2022 sebanyak 3.335 orang, dimana mayoritas di antaranya adalah laki-laki[3]. Pada hari pertama tahun baru 2023, Kasat Reskrim Polres Tangerang Selatan, AKP Aldo, sekitar pukul

5.47 WIB, pada Minggu 1 Januari 2023, mendapat laporan dari warga menemukan seorang laki-laki dalam posisi tengkurap di pinggir jalan. Data dari *e-MP* selama periode 1 hingga 9 Januari 2023, Polri menerima 31 laporan kasus pembunuhan setiap harinya[4]. Berdasarkan data dari *e-MP Robinopsnal* Bareskrim Polri menunjukkan peningkatan signifikan dalam kejahatan dan aktivitas premanisme yang melibatkan penggunaan

senjata tajam dari tahun 2021 hingga 2022, meningkat sebesar 36,6 persen. Pada tahun 2021, Polda Sulawesi Selatan menindak 213 kasus, dan pada tahun 2022, jumlah penindakan meningkat menjadi 303 kasus[5].

Sebagai warga negara Indonesia, kita harus ikut serta membela negara seperti bunyi Undang-Undang Republik Indonesia Nomor 56 Tahun 1999 pasal 1 yang bertuliskan <Bela negara adalah sikap dan perilaku warga negara yang dijiwai oleh kecintaannya kepada negara kesatuan Republik Indonesia yang berdasarkan Pancasila dan Undang-Undang Dasar 1945 dalam menjamin kelangsungan hidup bangsa dan negara.>, sehingga kita sebagai rakyat Indonesia wajib ikut serta dalam bela negara, dan oleh karena itu untuk melindungi kedaulatan negara dan keamanan dari segala ancaman, maka harus dibentuk sistem pertahanan agar dapat melindungi warga dari segala ancaman[6]. Banyak cara untuk melakukan pengawasan dalam lingkungan sekitar, salah satunya menggunakan

kamera pengawas, tetapi cara ini ini dapat memakan waktu dalam menemukan pelaku kejahatan, dan juga operator rentan melakukan kesalahan, terutama jika operator harus bekerja sepanjang waktu. Oleh karena itu, dibutuhkan sistem otomatis dengan *artificial intelligence* yang mampu menganalisis rekaman kamera CCTV dan memberi peringatan kepada operator keamanan ketika terdeteksi benda mencurigakan.

Berdasarkan penelitian dengan prinsip *confusion matrix* yang berkaitan untuk meminimalisir kesalahan positif dan negatif, dengan melatih dan menguji pada basis data. Dalam ranah klasifikasi dan pendeteksian yang menggunakan deep learning, model seperti YOLO memiliki performa yang unggul, baik dalam aspek kecepatan maupun ketepatan. Model pilihan yang telah kami latih mampu melakukan prediksi gambar dengan efektif dalam berbagai posisi, sudut pandang, dan penampilan, berhasil mencapai tingkat presisi rata-rata yang tinggi yaitu 91,73% dan skor F1 mencapai 91% pada model YOLO[8]. *You Only Look Once* (YOLO) adalah salah satu model yang populer pada sistem objek pendeteksian, YOLO juga dapat melakukan objek pendeteksian secara cepat dan mempunyai akurasi yang baik[7]. Sistem deteksi senjata yang akan dibangun dengan YOLOv5s, dengan dilakukannya konfigurasi pada model untuk menghasilkan performa terbaik akan menjadi solusi untuk mengatasi atau meminimalisir kasus kejahatan.

B. Topik dan Batasannya

Topik pada penelitian ini adalah membangun sistem objek deteksi yang dapat mendeteksi senjata untuk meminimalisir tindak kejahatan, penelitian ini akan membangun sistem objek pendeteksian menggunakan metode YOLOv5s, selanjutnya akan dilakukan analisis pada hasil *training* pada model YOLOv5s,

Tujuan pada penelitian ini adalah melakukan studi dengan metode YOLOv5s untuk mendeteksi senjata, meliputi bagaimana cara menerapkan model YOLOv5s pada pendeteksian senjata untuk menjaga keamanan, serta meneliti hasil pendeteksian senjata menggunakan metode YOLOv5s, adapun batasan masalah pada penelitian ini yaitu kondisi citra yang akan dilakukan uji coba adalah citra dengan intensitas cahaya terang, keterbatasan objek kecil atau objek yang terpotong, tetap melakukan deteksi senjata walaupun senjata tersebut mainan, jumlah kelas pada deteksi *dataset* 1 meliputi kelas pistol dan *knife*, jumlah kelas pada *dataset* 2 meliputi 3 kelas yaitu kelas *knife*, pistol, dan *heavygun*, keterbatasan dalam mendeteksi objek yang diluar jangkauan penangkapan gambar.

C. Organisasi Tulisan

Pendahuluan merupakan gambaran mengenai masalah yang menjadi topik pembahasan, studi terkait mengulas teori serta tinjauan jurnal atau makalah yang relevan dengan topik, bagian sistem

yang akan dibangun akan dijelaskan rinci, evaluasi menampilkan hasil kinerja sistem yang telah dibangun, kesimpulan berisikan hasil penelitian dan saran untuk penelitian selanjutnya.

II. STUDI TERKAIT

Dalam era modern ini banyak sekali terjadi kejahatan, mulai dari senjata tajam maupun senjata api yang sering digunakan untuk melakukan tindak kriminal, terkadang penjaga pun sering lengah dengan situasi tersebut, senjata tajam maupun senjata api sering kali lolos dari pandangan manusia (penjaga), oleh karena itu diperlukan sistem deteksi senjata otomatis agar meminimalisir tingkat kejahatan yang sering terjadi.

A. Deteksi secara manual oleh manusia

Visual manusia terkadang sering lengah dalam melakukan penjagaan di *mall*, tempat belanja, maupun di area yang sering menjadi objek perampokan, seperti *bank*, dan juga seperti kejadian di masjid *New Zealand*, maka dari itu diperlukan deteksi objek senjata secara otomatis agar tingkat kejahatan dan ancaman berkurang[8].

Keamanan dari tindak kejahatan merupakan fondasi yang esensial untuk menjaga ketertiban sosial dan kesejahteraan masyarakat. Ancaman kejahatan mencakup berbagai bentuk seperti kekerasan, pencurian, dan *cybercrime*, yang dapat merugikan individu dan komunitas secara keseluruhan. Langkah-langkah pencegahan efektif menjadi kunci dalam melindungi masyarakat dari risiko kejahatan. Secara tradisional mekanisme menjaga keamanan suatu tempat dilakukan oleh manusia dengan dukungan alat bantu seperti CCTV atau kamera pengawas.

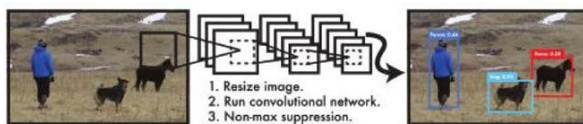
Closed-Circuit Television (CCTV) memiliki peran yang penting dalam meningkatkan keamanan. Pertama CCTV dapat menjadi *deterrent* atau pencegah kejahatan, karena kesadaran bahwa aktivitas dapat tercatat dan diawasi dapat mengurangi kemungkinan tindakan kriminal. Kedua, CCTV berfungsi sebagai alat pengawasan yang efektif, memungkinkan pihak berwenang untuk memantau area tertentu secara *real-time*, mengidentifikasi potensi ancaman, dan merespon dengan cepat terhadap situasi darurat. Namun faktor manusia yang bertindak menjadi pengawas CCTV menjadi salah satu titik lemah dalam mencegah kejahatan. *Visual* manusia terkadang sering lengah dalam melakukan penjagaan di *mall*, tempat belanja, maupun di area yang sering menjadi objek perampokan, seperti *bank*, dan juga seperti kejadian di masjid *New Zealand*. Durasi yang panjang dalam pekerjaan tenaga keamanan yang dapat mencapai 12 jam menjadi salah satu penyebab kelelahan sehingga mengurangi konsentrasi dalam pengawasan[9].

Artificial Intelligence (AI) atau kecerdasan buatan merupakan salah satu teknologi komputer yang dapat meniru kemampuan intelektual atau

kecerdasan manusia. AI memungkinkan komputer dapat belajar dari pengalaman, membuat keputusan, identifikasi pola, dan menyelesaikan tugas yang kompleks dengan cepat dan efisien. AI dapat bekerja dengan menggunakan data yang diinputkan untuk diproses pada sistem sehingga AI dapat mengenali data yang diinputkan[10]. Kecerdasan Buatan (*Artificial Intelligence*) adalah ilmu yang mempelajari kemampuan sistem untuk memahami data dengan benar, mempelajari dan mengaplikasikannya untuk tujuan atau tugas tertentu melalui adaptasi yang fleksibel [11]. Dengan adanya sistem yang menggunakan kecerdasan buatan untuk pendeteksian senjata, diharapkan akan mengurangi tingkat kejahatan yang terjadi.

B. You Only Look Once

Algoritma *You Only Look Once* (YOLO) adalah salah satu varian model yang serupa pada metode *Convolutional Neural Network* (CNN). YOLO merupakan metode yang berfokus dengan pendekatan baru dalam sistem pendeteksian objek, yang mempunyai tujuan atau berfokus pada pemrosesan data secara cepat (*real-time*). YOLO melakukan proses pemrosesan gambar yang sederhana, sistem YOLO hanya mengubah ukuran gambar yang dimasukkan lalu melakukan *single convolutional network* pada gambar, dan dapat menetapkan *setting threshold* untuk mendapatkan hasil deteksi dengan nilai *confidence* pada model, contoh sederhana YOLO dapat dilihat pada gambar 1[12].



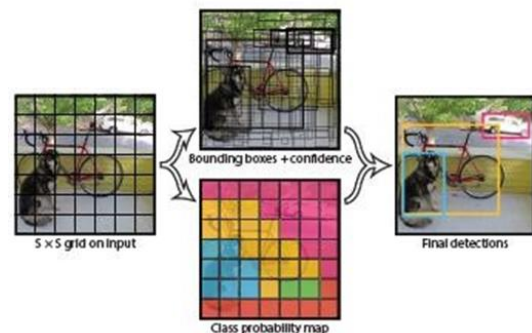
GAMBAR 1.
Arsitektur sederhana pada metode YOLO

Input pada metode YOLO akan dibagi menjadi *grid* $S \times S$. Setiap *sel grid* mempunyai tugas untuk memprediksi objek apa saja yang terdapat di dalamnya. Setiap *sel grid* memprediksi nilai kepercayaan (*Confidence Score*) pada kotak pembatas B dan memprediksi nilai probabilitas kelas C pada *grid*. Setiap kotak prediksi (*bounding box*) B yang dihasilkan dalam proses deteksi menggunakan YOLO akan mengandung lima elemen kunci, yaitu koordinat x , y , lebar w , tinggi h , serta kotak skor kepercayaan (*Confidence Score*). Koordinat pusat tengah diwakili oleh koordinat (x, y) , yang mengarah ke batas-batas kotak *grid*. Sementara untuk kotak (w, h) adalah ukuran dari gambar. Kotak nilai keyakinan (*Box Confidence Score*), menunjukkan seberapa yakin sistem dapat mendeteksi objek pada kotak pembatas B . Prediksi yang dilakukan pada model YOLO memiliki vektor *output* berupa $S \times S (B \times 5 + C)$. Setelah YOLO mengeluarkan nilai kepercayaan (*confidence*) pada kotak B sebagai probabilitas kotak yang mempunyai

objek. Nilai kepercayaan ini kemudian akan dikalikan pada IoU (*intersection over union*) pada kotak prediksi dan *ground truth*, dimana nilai *ground truth* yang didapatkan dengan cara *training*. IoU adalah pengukuran keakuratan pada pendeteksi objek yang terdapat pada suatu *dataset*. Untuk menghitung *box confidence score* atau nilai kepercayaan (*confidence*) [13], persamaan dapat dilihat pada persamaan 1 dan 2.

Penjelasan dari rumus tersebut sebagai berikut :

1. P_r (*object*) adalah probabilitas kotak yang berisikan objek, jika ada objek terdeteksi maka akan bernilai 1, jika tidak ada objek terdeteksi maka bernilai 0.
2. IoU^{truth} adalah IoU antara kotak prediksi dan kebenaran dasar.



GAMBAR 2.
Model dari metode YOLO

Nilai probabilitas yang mendeteksi bahwa objek milik kelas tertentu dikenal sebagai probabilitas pada kelas kondisional C yang diprediksi untuk setiap *sel grid*. Objek yang terdeteksi akan dikondisikan pada setiap kotak yang dianggap berisi objek tertentu. YOLO hanya akan melakukan prediksi pada satu set probabilitas kelas per *sel*, terlepas dari banyak jumlah kotak B yang ada, sehingga hanya akan ada total $S \times S \times C$ dengan probabilitas yang akan dimiliki oleh YOLO, contoh model dapat dilihat pada gambar 2.

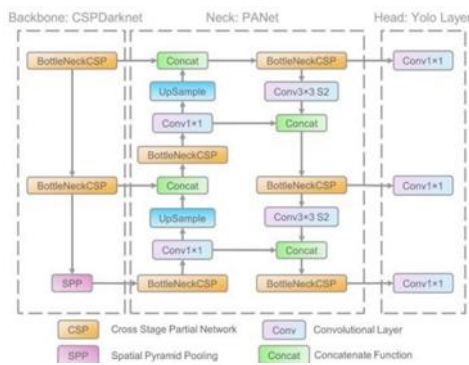


GAMBAR 3.
Contoh pendeteksian senjata menggunakan YOLO

Algoritma *You Only Look Once* (YOLO) telah menjadi sistem deteksi objek *real-time* yang sentral untuk robotika, seperti mobil tanpa pengemudi, dan pemantauan video[13]. Pada model YOLOv4 mempunyai banyak peningkatan daripada generasi sebelumnya, dengan mengimplementasikan arsitektur *Backbone* berhasil meningkatkan nilai *mAP* (*mean Average Precision*) sebanyak 10% dan

jumlah FPS (*Frame per Second*) sebanyak 12%. Selain itu, sekarang lebih mudah untuk melatih jaringan ini pada satu GPU[14]. Metode *transfer learning* juga dapat digunakan pada model YOLO, metode ini dapat meningkatkan performa model, uji coba model memiliki akurasi yang baik saat melakukan objek deteksi secara *real-time* [15]. Pengujian YOLO dengan menggunakan 64 *batch size* pada model, lalu *learning rate* 0.01 dengan *epoch* 75, *learning rate* 0.001 dengan *epoch* 30, dan *learning rate* 0.0001 dengan *epoch* 30, YOLO mampu melakukan objek deteksi secara *real-time*[12].

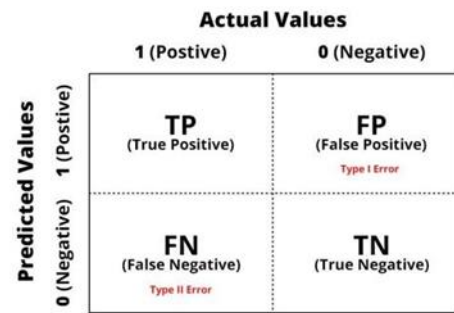
Pada gambar 4 adalah gambar Arsitektur pada YOLOv5 yang terdiri dari *Backbone CSPDarknet*, menggunakan *Neck PANet*, dan *Head* yaitu YOLO *layer*, yang dilakukan YOLO pertama-tama yaitu memasukkan data kedalam *CSPDarknet* untuk melakukan ekstraksi fitur-fitur yang nantinya dibutuhkan, *Spatial Pyramid Pooling* (SPP) ini ekstraksi fitur yang memiliki bentuk seperti piramid, lalu akan disalurkan ke *PANet* yang akan dilakukan beberapa proses yang sudah diatur seperti *convolution*, *upsample*, *concat*, *bottleneckCSP*, setelah itu akan data dilanjutkan ke YOLO *layer* pada tahapan ini YOLO akan mengeluarkan hasil deteksi berdasarkan pada masing-masing *grid*[16].



GAMBAR 4.
Arsitektur model YOLO.

C. Confusion Matrix

Menganalisis kinerja pengklasifikasi biner adalah aspek krusial dari evaluasi model pembelajaran mesin secara kuantitatif. Dalam kumpulan data terbatas yang memiliki label yang telah diketahui, kita dapat menilai efektivitas prediksi dengan menggunakan *confusion matrix* berukuran 2x2. Matriks ini untuk melakukan perhitungan pada model dengan menghitung *True Positive* (TP), *False Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) yang diprediksi model dengan label[23], penelitian dengan menggunakan *confusion matrix* performa model akan menjadi baik dan akurat, *confusion matrix* dapat dilihat pada gambar 5.



GAMBAR 5.
Confusion Matrix

D. Literatur Review

Deteksi objek akan menggunakan metode *You Only Look Once* (YOLO), penelitian ini akan difokuskan untuk mendeteksi senjata pada citra gambar dari beberapa sudut pandang, referensi yang berkaitan pada penelitian ini dapat dilihat dibawah ini.

Dalam paper yang ditulis oleh Bhatti, dkk, 2021[8] penelitian ini bertujuan untuk mengurangi kejahatan dan kriminalitas, pada paper ini melakukan uji coba pada beberapa metode, yaitu YOLOv4, YOLOv3, *Faster R-CNN*, *SDD MobileNetV1* sebagai objek deteksi, dia juga menggunakan metode *sliding window* menggunakan *VGG16*, *InceptionV3*, dan *InceptionV2*, dengan hasil terbaik menggunakan YOLOv4 yang mempunyai *F1 score* 91%, dan presisi sebesar 91.73%.

Dalam paper yang ditulis oleh Iqbal, dkk, 2021[17] ini menggunakan algoritma CNN dengan model *Scaled YOLOv4* untuk mendeteksi senjata tajam, penulis juga menggunakan *MongoDB* sebagai *database NoSQL* untuk mengatasi jika ada masalah dalam memonitoring, sistem deteksi senjata ini berjalan dengan *avg loss* sebesar 0.2514 dan mendapatkan rata-rata 35FPS, hasil dari model YOLO dengan *dataset* UGR memperoleh akurasi 0.89, presisi 0.82, *recall* 0.96, dan *F1 score* 0.90.

Paper yang ditulis oleh Sanam, dkk, 2021[18] melakukan uji coba pada 3 model yaitu YOLOv3, YOLOv2, dan *Faster R-CNN*, hasil dari uji coba mendapatkan tingkat akurasi pada YOLOv3 98.89%, YOLOv2 96.76%, traditional CNN 95%, hasil dari penelitian tersebut model YOLOv3 mempunyai akurasi terbaik dengan nilai 98.89.

Dalam paper yang ditulis oleh Asad, dkk, 2023[19] melakukan uji coba menggunakan 3 metode YOLO, yaitu YOLOv3, YOLOv4, dan YOLOv5, masing-masing metode mempunyai nilai yang cukup tinggi, hasil dari penelitian tersebut membuktikan YOLOv5 dengan model terbaik menghasilkan nilai *Precision* 88%, *Recall* 83%, *F1 score* 86%, *mAP* 87,8%.

Dalam paper yang ditulis oleh Arif, dkk, 2019[20] melakukan objek deteksi dengan model YOLOv3, *dataset* yang digunakan adalah pistol, hasil dari model YOLOv3 yang sudah dibuat menghasilkan 45 *Frame Per Second* (FPS) yang

jauh lebih baik dari model *Faster R-CNN* yang memiliki 8 FPS

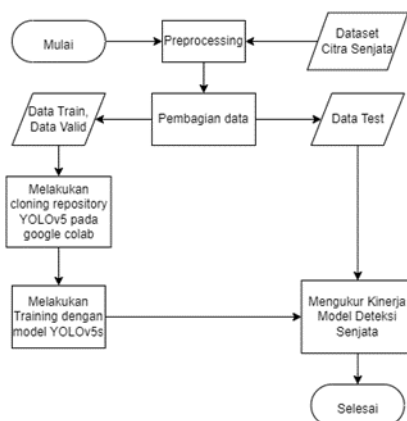
Paper yang ditulis oleh Dharmapuri, dkk, 2023[21] ini melakukan objek deteksi senjata menggunakan model YOLOv4 dengan 2 kelas yaitu pistol dan pisau, lalu melakukan objek deteksi dengan video dari CCTV, hasil dari model YOLOv4 sangat cocok untuk mendeteksi secara cepat dan mempunyai akurasi yang baik.

Dalam paper yang ditulis oleh Abdul, dkk, 2021[22] ini melakukan pendeteksian objek menggunakan *Faster R-CNN* dan YOLOv5s, gambar akan dilakukan resize 416×416 untuk model YOLO dan 240×240 untuk model *Faster R-CNN*, dan *dataset* dibagi menjadi 70% *train*, 20% *valid*, dan 10% *test*, untuk model YOLO dilakukan *epoch* sebesar 100-200 dan *Faster R-CNN* sebesar 60, setelah dilakukan uji coba YOLOv5s lebih cepat untuk pendeteksian dengan kecepatan 0,10s dibandingkan dengan *Faster R-CNN* yang mempunyai kecepatan 0.17s. YOLO-v5s lebih cepat 19 kali daripada *Faster R-CNN* dengan hasil *best results* 99%, *recall* 81%, *precision* pada gambar 93%, *precision* pada video 94%.

III. PERANCANGAN SISTEM

Di dalam bagian ini, akan membahas perancangan sistem deteksi senjata otomatis yang akan dibuat dalam penelitian, termasuk penjelasan dari skema yang sudah dibuat.

A. Gambaran umum rancangan sistem



GAMBAR 6.

Gambaran umum pada sistem yang akan dibuat

Pada gambar 6, rancangan umum yang dibuat akan dijelaskan secara singkat sebagai berikut:

1. tahap awal akan dilakukan pengumpulan data yang diperlukan yaitu senjata, setelah data dikumpulkan lalu diberi *anotasi* atau *label* berupa *bounding box*,
2. selanjutnya setelah *dataset* diberi *label* dan dilakukan *split* data menjadi 3 bagian yaitu 70% data *train*, 20% data *valid*, dan 10% data *test*, tahap ini dilakukan dengan menggunakan *website roboflow*,
3. setelah itu data akan diproses ke dalam *google*

colab, melakukan *cloning repository* YOLOv5,

4. memasukan *dataset* ke dalam *google colab* dengan cara menggunakan *api* yang disediakan oleh *website roboflow*, selanjutnya mengatur *parameter* pada model YOLOv5s yang akan dilakukan *training* pada data yang sudah dimasukan pada *google colab*.

5. setelah melakukan *training* pada model YOLOv5s yang telah diatur *parameternya*, maka akan menghasilkan *file* yang akan digunakan untuk pendeteksian berupa *file weight*, dan juga mengeluarkan matriks evaluasi berupa *recall*, *precision*, dan *mAP*,

6. tahap selanjutnya akan dilakukan analisis pada model yang sudah dibuat, dan menyimpulkan hasilnya.

B. Dataset

Data yang akan digunakan adalah data yang bersifat publik, data diperbolehkan oleh pembuat untuk uji coba maupun akademik, *dataset* pada penelitian ini diperoleh dari *website Andalusian Research Institute in Data Science and Computational Intelligence (ARI-DaSCI)*, *dataset* ini berupa data pisau sebanyak 2078 data dan pistol 3000 data, sedangkan dari kaggle (*SNEHIL SANYAL*) yaitu campuran data pisau, pistol, pedang, pelontar, *smg*, *sniper*, *rifle*, *shotgun*, dan *bazoka* sebanyak 714 data, dan tambahan data gambar dari internet yang diperbolehkan datanya untuk dipakai penelitian, tujuan dari mengambil data dari internet dan memanfaatkan data pistol dan pisau pada *dataset (ARI-DaSCI)* karena data yang digunakan untuk objek deteksi 3 kelas memerlukan data yang seimbang agar mendapatkan hasil yang baik, pada *dataset* ini mempunyai total 1500 data pada masing-masing kelas terdapat 500 data, kedua data tersebut akan dilakukan uji coba pada model objek YOLOv5s sebagai pendeteksian, informasi singkat tentang *dataset* yang digunakan



GAMBAR 7.

Isi folder dataset yang akan digunakan

C. Matriks Evaluasi atau Pengukuran Kinerja

Dalam penelitian ini terdapat evaluasi *confusion matrix* pada model berupa *precision*, *recall*, *F1 Score*, dan *mAP* dari model YOLOv5s yang dilakukan *training* pada *dataset*.

Matriks *Precision* matriks ini berfungsi untuk mengukur seberapa akurat sistem dalam mengidentifikasi objek positif, Selama tidak ada *False Positives*, akurasi model yang didapat adalah (dengan asumsi *True Positives* ≥ 0). Jika

mendapatkan beberapa nilai *False Positives*, ketepatannya akan menurun matrik ini akan memprediksi *True Positives* (TP) terhadap jumlah data yang diprediksi *True Positives* (TP), dan *False Positive* (FP). Matrik ini menilai seberapa akurat model dalam mengidentifikasi hasil yang positif dibandingkan dengan semua hasil yang diklasifikasikan sebagai positif, persamaan *precision* dapat dilihat pada persamaan 3.

$$\text{precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}} \quad (3)$$

Matriks *Recall* salah satu matrik evaluasi klasifikasi yang mengukur kemampuan model untuk mengidentifikasi dari suatu kelas tertentu, *recall* yang juga dikenal sebagai sensitivitas atau tingkat positif sejati, mengukur kemampuan model untuk mengidentifikasi dengan benar *True Positives* (TP) dari semua kasus positif yang sebenarnya. *Recall* dihitung sebagai rasio antara *True Positives* (TP) dengan jumlah total kasus positif sebenarnya, yang merupakan jumlah dari *True Positives* (TP) dan *False Negatives* (FN). Nilai *recall* yang tinggi menunjukkan bahwa model tersebut efektif dalam menemukan kembali informasi yang relevan dan meminimalkan risiko melewatkan kasus positif. Persamaan *recall* dapat dilihat pada persamaan 4.

$$\text{recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}} \quad (4)$$

Matriks *F1 Score* adalah perbandingan nilai rata-rata pada *precision* dan *recall*, jika nilai *precision* dan *recall* memiliki nilai yang baik, maka *F1 Score* juga akan memiliki nilai yang baik, *F1 Score* menggabungkan dua matrik yang saling bersaing, yaitu *precision* dan *recall*, menjadi satu nilai tunggal. Ini sangat berguna dalam situasi di mana kelas-kelas dalam dataset tidak seimbang, dan kita ingin menemukan keseimbangan antara kemampuan model untuk mengidentifikasi kasus positif (*recall*) dan hanya mengidentifikasi kasus positif yang benar (*precision*). Nilai yang terbaik pada *F1 Score* adalah (1.0) dan nilai terburuk adalah (0.0), persamaan *F1 Score* dapat dilihat pada persamaan 5.

$$F1\ Score = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \quad (5)$$

Matriks *mean Average Precision* (*mAP*) adalah matrik yang dapat melakukan evaluasi pada model deteksi, *mAP* memberikan penilaian yang komprehensif terhadap model dengan menghitung rata-rata dari nilai *Average Precision* (AP) untuk setiap kelas dari berbagai batas *IoU* (*Intersection*

over Union), *mAP* dapat dicari dengan mengambil nilai rata-rata pada *average precision* dan *IoU*, $N =$ jumlah indeks, AP_i nilai AP dari kelas i , persamaan *mAP* dapat dilihat pada persamaan 6.

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \quad (6)$$

D. Skenario Pengujian

Pada tahapan ini akan dilakukan skenario pengujian:

1. *Pre-processing* data :

Pada tahapan ini *dataset* pertama (*ARI-DaSCI*) akan dilakukan label menjadi 2 kelas yaitu *knife*, *pistol*, dan *dataset* kedua menjadi 3 label kelas yaitu *knife*, *pistol*, *heavygun*, dan dilakukan *split* data pada kedua *dataset* tersebut sebanyak 70% data *train*, 20% data *valid*, dan 10% data *test*, pada *dataset* kedua akan dilakukan *pre-processing* berupa perubahan *size* menjadi 640x640 karena *size* gambar yang berbeda-beda akan lebih baik jika dilakukan perubahan ukuran, dan dilakukan *auto-oriented* fitur ini berfungsi untuk memperbaiki orientasi gambar yang salah, tahap *pre-processing* ini dilakukan menggunakan *website roboflow*,

2. Konfigurasi model :

Setelah melakukan *cloning repository* YOLOv5 pada *google colab*, selanjutnya model YOLOv5s akan dilakukan pengaturan *parameter*, pengaturan *parameter* dapat dilihat pada tabel 2,

3. Uji model pada *dataset* :

Melakukan *training* pada *dataset* untuk menghasilkan nilai *precision*, *recall*, dan *mAP* pada model deteksi senjata YOLOv5s yang sudah dilakukan *training*, *Dataset* yang akan diuji adalah *dataset 1* (*ARI-DaSCI*), *dataset 2* (*SNEHIL SANYAL*) *original* dengan 3 kelas, dan *dataset 2* (*SNEHIL SANYAL*) yang sudah dilakukan penyeimbangan data pada masing-masing kelas

4. Hasil pengujian :

Menganalisis dan mencari hasil terbaik dari pengujian model YOLOv5s pada *dataset*.

Untuk informasi konfigurasi eksperimen atau uji coba yang akan dilakukan dapat dilihat pada tabel 2 konfigurasi eksperimental, penjelasan pada tabel kombinasi *parameter* :

5. Skenario Model 1: *Image Size* 416, *Batch* 32, *Epoch* 10
6. Skenario Model 2: *Image Size* 416, *Batch* 64, *Epoch* 10
7. Skenario Model 3: *Image Size* 416, *Batch* 32, *Epoch* 20
8. Skenario Model 4: *Image Size* 416, *Batch* 64, *Epoch* 20
9. Skenario Model 5: *Image Size* 512, *Batch* 32, *Epoch* 10

10. Skenario Model 6: *Image Size* 512, *Batch* 64, *Epoch* 10
11. Skenario Model 7: *Image Size* 512, *Batch* 32, *Epoch* 20
12. Skenario Model 8: *Image Size* 512, *Batch* 64, *Epoch* 20

TABEL 2.
Konfigurasi parameter eksperimental

Hyperparameter	Values
<i>Image Size</i>	416, 512
<i>Batch</i>	32, 64
<i>Epoch</i>	10, 20

IV. EVALUASI

Pada bab ini akan menampilkan hasil dari pengujian berdasarkan skenario yang sudah ditentukan pada bab sebelumnya, pengujian menggunakan model YOLOv5s dan dilakukan dengan 2 *dataset* yang berbeda, pada *dataset* (ARI-DaSCI) terdapat 5078 data dengan kelas *Knife* dan *Pistol*, pada *dataset* (SNEHIL SANYAL) terdapat 1500 data dengan kelas *Knife*, *Pistol*, *Heavygun*, pada *training dataset epoch* yang digunakan adalah 10, 20, selain itu akan dilakukan konfigurasi *image size* berukuran 416, 512, dan *batch* 32 dan 64. Nilai yang tertinggi atau mendekati nilai 1.0 pada hasil *training* dan *test* dengan model adalah hasil yang terbaik.

A. Hasil *training* pada *dataset* 1 (ARI-DaSCI)

Pada tabel 3 terdapat hasil pengujian yang telah dilakukan dengan *parameter* yang sebelumnya sudah ditentukan, terdapat 8 skenario pengujian dengan menampilkan hasil *training* berupa *Precision*, *Recall*, dan *mAP*, dari uji *parameter* ini, terlihat bahwa penggunaan jumlah *epoch* yang lebih tinggi 20 cenderung meningkatkan nilai *recall* dan *mAP*. Selain itu, ukuran *batch* yang lebih besar 64 juga menunjukkan hasil yang lebih baik dalam beberapa skenario. skenario 4, dan 7 memiliki nilai yang cukup baik, skenario 4 menyatakan

bahwa model cukup optimal dikarenakan *mAP* lebih tinggi daripada skenario 7, skenario 7 menunjukkan nilai yang dominan cukup tinggi daripada skenario yang lain.

TABEL 3.

Tabel hasil *training* matriks evaluasi keseluruhan pada *dataset* 1 (ARI-DaSCI)

Skenario	<i>Precision</i>	<i>Recall</i>	<i>mAP</i> 50	<i>F1 Score</i>
1	0.575	0.797	0.562	0.669
2	0.572	0.824	0.567	0.676
3	0.588	0.83	0.582	0.687
4	0.588	0.843	0.592	0.694
5	0.569	0.823	0.578	0.672
6	0.579	0.802	0.575	0.673
7	0.588	0.863	0.583	0.699
8	0.565	0.851	0.574	0.680

Analisis nilai *Precision* pada tabel 4 *image size* 416 memberikan hasil yang sedikit lebih baik dalam beberapa skenario. Misalnya, untuk *knife* pada *epoch* 20 (skenario 3 dan 4) hasilnya 0.6 dan 0.59, dibandingkan dengan 0.599 dan 0.56 pada *image size* 512 (skenario 7 dan 8), untuk *pistol*, hasilnya sedikit bervariasi, pada skenario 6 (*image size* 512, *batch* 64, *epoch* 10 dengan nilai *precision* *Pistol* 0.59, yang lebih tinggi dibandingkan dengan beberapa skenario lain dengan *image size* 416. Penggunaan *batch size* 32 cenderung memberikan hasil yang lebih baik pada beberapa skenario, misalnya, pada *image size* 416 dengan *epoch* 10 (skenario 1 dan 2), *batch size* 32 menghasilkan nilai *precision knife* 0.576, sedangkan *batch size* 64 menghasilkan nilai *precision* 0.563. Pada *pistol*, *batch size* 64 sedikit lebih baik pada beberapa skenario seperti skenario 2 dan 6. Meningkatkan jumlah *epoch* juga sangat berpengaruh seperti skenario 1 dan 3 dengan *parameter* yang sama tetapi jumlah *epoch* 20 pada skenario 3 lebih tinggi dibandingkan skenario 1, meningkatkan jumlah *epoch* sangat berpengaruh terhadap kenaikan performa dari model, tetapi pada beberapa skenario kenaikan tidak terlalu signifikan, nilai *precision* tertinggi pada kelas *knife* skenario 3 dengan nilai 0.6, nilai *precision* tertinggi pada kelas *pistol* terdapat pada skenario 6 dengan nilai 0.59. Ini membuktikan bahwa skenario 3 baik dalam mengidentifikasi objek positif pada kelas *knife* sedangkan skenario 6 baik dalam mengidentifikasi objek positif pada kelas *pistol*.

Analisis nilai *Recall* pada tabel 4 jumlah *epoch* membuktikan kenaikan pada skenario 1 dan 3, komposisi *parameter* yang sama pada skenario 3 dengan *epoch* 20 pada kedua kelas terjadi peningkatan nilai *recall* yang dihasilkan daripada skenario 1. Pada ukuran *batch* 64 dengan 32 skenario 2 lebih tinggi daripada skenario 1 ini membuktikan bahwa *batch* berpengaruh juga pada *recall*, *image size* yang tinggi juga berpengaruh pada kenaikan nilai *recall* seperti pada skenario 5 dan 1, *image size* 512 cenderung memiliki nilai yang lebih tinggi jika dilakukan pelatihan dengan jumlah *epoch* 20, *batch* 64 memiliki hasil yang cukup baik dari beberapa skenario seperti skenario 4 namun pada skenario 7 dengan *batch* 32 juga memiliki nilai yang baik, meningkatkan jumlah *epoch* juga sangat berpengaruh terhadap hasil *recall*, nilai *recall* tertinggi pada kelas *knife* terdapat pada skenario 7 dengan nilai 0.89, nilai *recall* tertinggi pada kelas *pistol* terdapat pada skenario 8 dengan nilai 0.846. Ini membuktikan bahwa skenario 7 baik dalam mengukur objek kelas pada kelas *knife* sedangkan skenario 8 baik dalam mengukur objek kelas pada kelas *pistol*.

Analisis nilai *mAP* pada tabel 4 jumlah *epoch* membuktikan kenaikan pada skenario 1 dan 3, komposisi *parameter* yang sama pada skenario 3 dengan *epoch* 20 pada kedua kelas terjadi peningkatan nilai *mAP* yang dihasilkan daripada

skenario 1. Pada ukuran *batch* 64 dengan 32 skenario 2 lebih tinggi daripada skenario 1 ini membuktikan bahwa *batch* berpengaruh juga pada *mAP*, *image size* yang tinggi juga berpengaruh pada kenaikan nilai *mAP* seperti pada skenario 5 dan 1, *image size* 512 cenderung memiliki nilai yang lebih tinggi jika dilakukan pelatihan dengan jumlah *epoch* 20, namun *image size* 416 memiliki nilai yang lebih baik pada kedua kelas pada skenario 4, meningkatkan jumlah *epoch* juga sangat berpengaruh terhadap hasil, nilai *mAP* tertinggi pada kelas *knife* terdapat pada skenario 4 dengan nilai 0.611, nilai *mAP* tertinggi pada kelas pistol terdapat pada skenario 4 dengan nilai 0.572. Ini membuktikan bahwa skenario 4 memiliki model yang optimal dalam objek pendeteksian.

Analisis nilai *F1 score* pada tabel 4 jumlah *epoch* membuktikan kenaikan pada skenario 1 dan 3, komposisi *parameter* yang sama pada skenario 3 dengan *epoch* 20 pada kedua kelas terjadi peningkatan nilai *mAP* yang dihasilkan daripada skenario 1. Pada ukuran *batch* 64 dengan 32 skenario 2 lebih tinggi daripada skenario 1 ini membuktikan bahwa *batch* berpengaruh juga pada *F1 Score*, *image size* yang tinggi juga berpengaruh pada kenaikan nilai *F1 Score* seperti pada skenario 5 dan 1, *image size* 512 cenderung memiliki nilai yang lebih tinggi jika dilakukan pelatihan dengan jumlah *epoch* 20, namun *image size* 416 memiliki nilai yang lebih baik pada kedua kelas pada skenario 4, meningkatkan jumlah *epoch* juga sangat berpengaruh terhadap hasil, nilai *F1 Score* tertinggi pada kelas *knife* terdapat pada skenario 7 dengan nilai 0.717, nilai *F1 Score* tertinggi pada kelas pistol terdapat pada skenario 4 dengan nilai 0.685. Ini membuktikan bahwa skenario 4 memiliki rata-rata nilai yang baik pada objek pendeteksian.

TABEL 4.

Tabel hasil training matriks evaluasi semua kelas pada dataset 1

Skenario	Precision		Recall		mAP		F1 Score	
	Knife	Pistol	Knife	Pistol	Knife	Pistol	Knife	Pistol
1	0.576	0.574	0.812	0.781	0.574	0.55	0.676	0.662
2	0.563	0.58	0.833	0.815	0.583	0.551	0.672	0.679
3	0.6	0.576	0.849	0.81	0.591	0.572	0.705	0.674
4	0.59	0.587	0.857	0.83	0.611	0.572	0.698	0.685
5	0.577	0.561	0.835	0.812	0.59	0.565	0.688	0.664
6	0.567	0.59	0.82	0.783	0.593	0.558	0.672	0.675
7	0.599	0.577	0.89	0.836	0.593	0.572	0.717	0.682
8	0.56	0.57	0.857	0.846	0.584	0.565	0.676	0.680

B. Hasil *training* pada *dataset* 2 (SNEHIL SANYAL)

Pada tabel 5 terdapat hasil pengujian yang telah dilakukan dengan *parameter* yang sebelumnya sudah ditentukan, pada hasil pengujian membuktikan bahwa meningkatkan jumlah *epoch* dari 10 menjadi 20 secara konsisten meningkatkan performa model dalam hal *precision*, *recall*, *mAP*,

dan *F1 Score*, berdasarkan *batch size* menunjukkan penurunan pada *precision* dan *recall* jika jumlah *epoch* tetap, *batch size* dengan jumlah besar dapat menghasilkan performa yang baik, mengubah ukuran gambar 416 menjadi 512 memberikan peningkatan performa pada model jika melakukan *epoch* yang lebih besar, skenario 4 dengan nilai *recall* tertinggi menyatakan bahwa model tersebut memiliki kemampuan menemukan objek yang positif pada *dataset*, skenario 8 memiliki hasil yang dominan tinggi daripada skenario yang lain dengan nilai *precision* sebesar 0.915, *recall* 0.837, *mAP* 0.898, dan *F1 score* 0.875.

TABEL 5.

Tabel hasil training matriks evaluasi keseluruhan pada dataset 2 (SNEHIL SANYAL)

Skenario	Precision	Recall	mAP 50	F1 Score
1	0.869	0.78	0.844	0.821
2	0.779	0.736	0.791	0.759
3	0.904	0.841	0.891	0.870
4	0.883	0.85	0.893	0.867
5	0.803	0.791	0.825	0.797
6	0.804	0.749	0.782	0.774
7	0.886	0.84	0.886	0.861
8	0.915	0.837	0.898	0.875

Analisis nilai *Precision* pada tabel 6 dapat dilihat bahwa kenaikan *epoch* sangat berpengaruh seperti pada skenario 8 yang memiliki hasil lebih tinggi daripada skenario 6, *image size* tinggi tidak selamanya memiliki hasil yang baik pada setiap kelas seperti skenario 8 dan 4 namun skenario 8 memiliki nilai yang dominan pada kelas *knife* dan pistol, *batch size* juga berpengaruh pada performa model seperti pada skenario 8 yang memiliki nilai dominan lebih tinggi daripada skenario 7, nilai *precision* tertinggi pada kelas *heavygun* terdapat pada skenario 8 dengan nilai 0.966, nilai *precision* tertinggi pada kelas *knife* terdapat pada skenario 4 dengan nilai 0.96, nilai *precision* tertinggi pada kelas pistol terdapat pada skenario 8 dengan nilai 0.836. Ini membuktikan bahwa skenario 8 baik dalam mengidentifikasi objek positif pada kelas *heavygun*, skenario 4 baik dalam mengidentifikasi objek positif pada kelas *knife*, dan skenario 8 baik dalam mengidentifikasi objek positif pada kelas pistol, skenario 8 membuktikan bahwa memiliki nilai yang dominan tinggi dan baik dalam mengidentifikasi objek positif.

Analisis nilai *Recall* pada tabel 6 dapat dilihat bahwa kenaikan *epoch* sangat berpengaruh seperti pada skenario 8 yang memiliki hasil lebih tinggi daripada skenario 6, *image size* besar tidak selamanya memiliki hasil yang baik seperti pada skenario 4 dan 8, walau *image size* skenario 8 lebih besar tetapi skenario 4 memiliki nilai rata-rata yang lebih baik daripada skenario 8, *batch size* tinggi tidak selalu memiliki hasil yang baik seperti pada skenario 1 dan 2 *batch size* dengan ukuran 32 lebih dominan daripada 64, nilai *recall* tertinggi pada

kelas *heavygun* terdapat pada skenario 4 dengan nilai 0.926, nilai *recall* tertinggi pada kelas *knife* terdapat pada skenario 3 dengan nilai 0.934, nilai *recall* tertinggi pada kelas pistol terdapat pada skenario 4 dengan nilai 0.793. Ini membuktikan bahwa skenario 4 baik dalam mengukur objek kelas pada kelas *heavygun*, skenario 3 baik dalam mengukur objek kelas pada kelas *knife*, dan skenario 4 baik dalam mengukur objek kelas pada kelas pistol, skenario 4 membuktikan memiliki nilai yang dominan tinggi dan baik dalam mengukur objek kelas.

TABEL 6.

Tabel hasil training matriks evaluasi precision dan recall semua kelas pada dataset 2

Skenario	Precision			Recall		
	Heavygun	Knife	Pistol	Heavygun	Knife	Pistol
1	0.898	0.9	0.809	0.883	0.82	0.639
2	0.879	0.802	0.656	0.848	0.82	0.541
3	0.943	0.96	0.808	0.889	0.934	0.699
4	0.92	0.959	0.77	0.926	0.893	0.793
5	0.852	0.848	0.709	0.852	0.861	0.66
6	0.839	0.893	0.68	0.827	0.828	0.592
7	0.919	0.924	0.815	0.907	0.899	0.714
8	0.966	0.944	0.836	0.873	0.877	0.759

Analisis nilai *mAP* pada tabel 7 dapat dilihat bahwa kenaikan *epoch* sangat berpengaruh seperti pada skenario 8 yang memiliki hasil lebih tinggi daripada skenario 6, *image size* besar tidak selamanya memiliki hasil yang baik seperti pada skenario 8 dan 4, walau *image size* skenario 8 lebih besar tetapi skenario 4 memiliki nilai *mAP* pada kelas *knife* lebih tinggi, *batch size* tinggi tidak selalu memiliki hasil yang baik seperti pada skenario 1 dan 2 *batch size* dengan ukuran 32 pada skenario 1 memiliki nilai *mAP* tertinggi dan dominan daripada skenario 2 dengan *batch size* 64, nilai *mAP* tertinggi pada kelas *heavygun* terdapat pada skenario 8 dengan nilai 0.953, nilai *mAP* tertinggi pada kelas *knife* terdapat pada skenario 3 dengan nilai 0.958, nilai *mAP* tertinggi pada kelas pistol terdapat pada skenario 8 dengan nilai 0.797. Ini membuktikan bahwa skenario 8 optimal pada kelas *heavygun*, skenario 3 optimal pada kelas *knife*, dan skenario 8 optimal pada kelas pistol, skenario 8 membuktikan memiliki nilai yang dominan tinggi dan optimal pada model.

Analisis nilai *F1 score* pada tabel 7 dapat dilihat bahwa kenaikan *epoch* dapat meningkatkan nilai *F1 Score* seperti pada kelas *heavygun* pada skenario 3 dan 1, *batch size* tinggi tidak selalu memberikan nilai yang baik seperti pada skenario 3 dengan nilai lebih tinggi daripada skenario 4, *image size* tinggi baik dalam beberapa kelas dan skenario namun *image size* tinggi tidak selamanya menghasilkan nilai yang baik seperti pada kelas *heavygun* skenario 4 dan 8, skenario 4 memiliki nilai yang lebih baik daripada skenario 8 pada kelas *heavygun*,

nilai *F1 score* tertinggi pada kelas *heavygun* terdapat pada skenario 4 dengan nilai 0.922, nilai *F1 Score* tertinggi pada kelas *knife* terdapat pada skenario 3 dengan nilai 0.947, nilai *F1 score* tertinggi pada kelas pistol terdapat pada skenario 8 dengan nilai 0.796. Ini membuktikan bahwa skenario 3, 4, dan 8 memiliki keunggulan pada masing-masing kelas.

TABEL 7.

Tabel hasil training matriks evaluasi mAP dan F1 score semua kelas pada dataset 2

Skenario	mAP			F1 score		
	Heavygun	Knife	Pistol	Heavygun	Knife	Pistol
1	0.926	0.911	0.697	0.888	0.858	0.714
2	0.893	0.87	0.609	0.861	0.811	0.594
3	0.949	0.958	0.767	0.916	0.947	0.750
4	0.93	0.957	0.793	0.922	0.923	0.782
5	0.899	0.905	0.671	0.852	0.855	0.686
6	0.848	0.882	0.615	0.832	0.860	0.633
7	0.939	0.949	0.768	0.914	0.912	0.761
8	0.953	0.944	0.797	0.916	0.908	0.796

C. Hasil training pada dataset 2 original (SNEHIL SANYAL)

Pada tabel 8 terdapat hasil pengujian yang telah dilakukan dengan *parameter* yang sebelumnya sudah ditentukan, pada hasil pengujian membuktikan bahwa meningkatkan jumlah *epoch* dari 10 menjadi 20 secara konsisten meningkatkan performa model dalam hal *precision*, *recall*, *mAP*, dan *F1 Score*, *batch size* dengan jumlah besar dapat menghasilkan performa yang baik dalam beberapa skenario, *image size* yang lebih besar juga peningkatan performa pada model jika melakukan *epoch* yang lebih besar, skenario 8 dengan nilai *precision* tertinggi menyatakan bahwa model tersebut memiliki kemampuan mengidentifikasi objek positif pada *dataset*, skenario 3 memiliki hasil yang dominan tinggi daripada skenario yang lain dengan nilai *precision* sebesar 0.925, *recall* 0.941, *mAP* 0.973, dan *F1 score* 0.933.

TABEL 8.

Tabel hasil training matriks evaluasi keseluruhan pada dataset 2 original (SNEHIL SANYAL)

Skenario	Precision	Recall	mAP	F1 Score
1	0.824	0.721	0.834	0.769
2	0.426	0.766	0.689	0.548
3	0.925	0.941	0.973	0.933
4	0.942	0.868	0.951	0.903
5	0.694	0.641	0.75	0.666
6	0.47	0.641	0.595	0.542
7	0.935	0.924	0.955	0.929
8	0.96	0.9	0.961	0.929

Analisis nilai *Precision* pada tabel 9 dapat dilihat bahwa kenaikan *epoch* sangat berpengaruh seperti pada skenario 8 yang memiliki hasil lebih tinggi daripada skenario 6, *image size* tinggi tidak

selamanya memiliki hasil yang baik pada setiap kelas seperti skenario 8 dan 4 namun skenario 8 memiliki nilai yang dominan pada kelas *knife* dan pistol, *batch size* juga berpengaruh pada performa model seperti pada skenario 8 yang memiliki nilai dominan lebih tinggi daripada skenario 7, nilai *precision* tertinggi pada kelas *heavygun* terdapat pada skenario 3 dengan nilai 0.959, nilai *precision* tertinggi pada kelas *knife* terdapat pada skenario 8 dengan nilai 0.945, nilai *precision* tertinggi pada kelas pistol terdapat pada skenario 7 dan 8 dengan nilai 1. Ini membuktikan bahwa skenario 3 baik dalam mengidentifikasi objek positif pada kelas *heavygun*, skenario 8 baik dalam mengidentifikasi objek positif pada kelas *knife*, dan skenario 7 dan 8 baik dalam mengidentifikasi objek positif pada kelas pistol, skenario 8 membuktikan bahwa memiliki nilai yang dominan tinggi dan baik dalam mengidentifikasi objek positif.

Analisis nilai *Recall* pada tabel 9 dapat dilihat bahwa kenaikan *epoch* sangat berpengaruh seperti pada skenario 7 yang memiliki hasil lebih tinggi daripada skenario 6, *image size* besar tidak selamanya memiliki hasil yang baik seperti pada skenario 4 dan 8, walau *image size* skenario 8 lebih besar tetapi skenario 4 memiliki nilai kelas *heavygun* yang lebih baik daripada skenario 8, *batch size* tinggi memiliki nilai yang lebih baik seperti pada skenario 6 dan 8, nilai *recall* tertinggi pada kelas *heavygun* terdapat pada skenario 7 dengan nilai 0.936, nilai *recall* tertinggi pada kelas *knife* terdapat pada skenario 3 dengan nilai 0.925, nilai *recall* tertinggi pada kelas pistol terdapat pada skenario 8 dengan nilai 0.985. Ini membuktikan bahwa skenario 7 baik dalam mengukur objek kelas pada kelas *heavygun*, skenario 3 baik dalam mengukur objek kelas pada kelas *knife*, dan skenario 8 baik dalam mengukur objek kelas pada kelas pistol, setiap skenario mempunyai keunggulan masing-masing pada setiap kelas.

TABEL 9.

Tabel hasil training matriks evaluasi precision dan recall semua kelas pada dataset 2 original

Skenario	Precision			Recall		
	Heavygun	Knife	Pistol	Heavygun	Knife	Pistol
1	0.875	0.732	0.864	0.852	0.819	0.491
2	0.48	0.33	0.468	0.936	0.9	0.462
3	0.959	0.89	0.927	0.925	0.925	0.974
4	0.936	0.917	0.973	0.934	0.823	0.846
5	0.866	0.614	0.602	0.784	0.676	0.462
6	0.503	0.33	0.576	0.884	0.725	0.314
7	0.91	0.894	1	0.936	0.847	0.898
8	0.935	0.945	1	0.861	0.854	0.985

Analisis nilai *mAP* pada tabel 10 dapat dilihat bahwa kenaikan *epoch* sangat berpengaruh seperti pada skenario 8 yang memiliki hasil lebih tinggi daripada skenario 6, *image size* besar tidak selamanya memiliki hasil yang baik seperti pada skenario 8 dan 4, walau *image size* skenario 8 lebih

besar tetapi skenario 4 memiliki nilai *mAP* pada kelas *heavygun* lebih tinggi, *batch size* tinggi mempengaruhi performa seperti skenario 1 dan 2, nilai *mAP* tertinggi pada kelas *heavygun* terdapat pada skenario 3 dengan nilai 0.972, nilai *mAP* tertinggi pada kelas *knife* terdapat pada skenario 3 dengan nilai 0.958, nilai *mAP* tertinggi pada kelas pistol terdapat pada skenario 8 dengan nilai 0.995. Ini membuktikan bahwa skenario 3 optimal pada kelas *heavygun* dan *knife*, skenario 8 optimal pada kelas pistol, skenario 3 membuktikan memiliki nilai yang dominan tinggi dan optimal pada model.

Analisis nilai *F1 score* pada tabel 10 dapat dilihat bahwa kenaikan *epoch* dapat meningkatkan nilai *F1 Score* seperti pada kelas *heavygun* pada skenario 3 dan 1, *batch size* tinggi mempengaruhi performa seperti

pada skenario 1 dan 3, *image size* tinggi baik dalam beberapa kelas dan skenario namun *image size* tinggi tidak selamanya menghasilkan nilai yang baik seperti pada kelas *heavygun* skenario 4 dan 8, skenario 4 memiliki nilai yang lebih baik daripada skenario 8 pada kelas *heavygun*, nilai *F1 score* tertinggi pada kelas *heavygun* terdapat pada skenario 3 dengan nilai 0.940, nilai *F1 Score* tertinggi pada kelas *knife* terdapat pada skenario 3 dengan nilai 0.906, nilai *F1 score* tertinggi pada kelas pistol terdapat pada skenario 8 dengan nilai 0.993. Ini membuktikan bahwa skenario 3 memiliki keunggulan dalam nilai *harmonic precision* dan *recall* pada kelas *heavygun* dan *knife*, dan skenario 8 memiliki nilai *harmonic* yang baik pada kelas pistol.

TABEL 10.

Tabel hasil training matriks evaluasi mAP dan F1 score semua kelas pada dataset 2 original

Skenario	<i>mAP</i>			<i>F1 score</i>		
	<i>Heavygun</i>	<i>Knife</i>	<i>Pistol</i>	<i>Heavygun</i>	<i>Knife</i>	<i>Pistol</i>
1	0.931	0.855	0.715	0.863	0.773	0.313
2	0.879	0.705	0.485	0.634	0.483	0.465
3	0.972	0.958	0.99	0.940	0.906	0.935
4	0.966	0.918	0.969	0.934	0.866	0.905
5	0.895	0.721	0.633	0.822	0.644	0.522
6	0.806	0.545	0.434	0.641	0.454	0.407
7	0.962	0.909	0.995	0.922	0.869	0.946
8	0.947	0.942	0.995	0.897	0.896	0.993

D. Hasil Test dengan model pada dataset 1 (*ARI-DaSCI*)

Pada tabel 11 terdapat hasil *test* dari model yang sudah dilakukan *training* sebelumnya, terdapat 8 model berdasarkan skenario sebelumnya, selanjutnya akan dilakukan *test* dengan *file weight* yang sudah dilakukan *training*, hasil *test* menampilkan matriks evaluasi berupa *Precision*, *Recall*, dan *mAP*, dari *file weight* terhadap *folder test* ini, terlihat bahwa penggunaan *parameter* tinggi cenderung meningkatkan nilai *precision*, *recall*, *mAP*, dan *F1 score*. Selain itu, ukuran *batch*

size dan image size yang lebih besar juga menunjukkan hasil yang lebih baik dalam beberapa skenario. skenario 7 dan 8 memiliki nilai yang cukup baik, skenario 8 menyatakan bahwa model lebih baik dalam melakukan mengidentifikasi objek positif dikarenakan nilai Precision lebih tinggi daripada skenario 7, skenario 7 menunjukan nilai yang dominan cukup tinggi daripada skenario yang lain. Skenario 7 dipilih karena memiliki nilai yang dominan cukup tinggi dengan nilai Precision sebesar 0.923, Recall sebesar 0.901, F1 score sebesar 0.912, dan mAP sebesar 0.947

TABEL 11.
Tabel hasil Test matriks evaluasi keseluruhan dengan model pada dataset 1 (ARI-DaSCI)

Skenario	Precision	Recall	mAP	F1 Score
1	0.908	0.851	0.912	0.879
2	0.921	0.817	0.908	0.866
3	0.921	0.817	0.908	0.866
4	0.93	0.892	0.944	0.911
5	0.905	0.854	0.927	0.879
6	0.911	0.858	0.919	0.884
7	0.923	0.901	0.947	0.912
8	0.928	0.886	0.944	0.907

Analisa Precision pada tabel 12 nilai Precision tertinggi untuk deteksi objek Knife terdapat pada skenario 8 dengan nilai 0.964, menunjukkan model memiliki kemampuan yang sangat baik untuk mengidentifikasi objek Knife secara benar tanpa menghasilkan banyak false positives. Skenario 4 juga menunjukkan performa yang baik dengan nilai precision 0.933. Precision tertinggi untuk Pistol terdapat pada skenario 4 dengan nilai 0.928. Hal ini menunjukkan kemampuan yang baik dalam mendeteksi objek Pistol, dengan sedikit false positives.

Analisa Recall pada tabel 12 nilai Recall tertinggi pada kelas Knife terdapat pada skenario 4 dengan nilai 0.935, menunjukkan bahwa model mampu mendeteksi sebagian besar objek Knife yang ada, dengan sedikit false negatives, Recall tertinggi pada pistol terdapat pada skenario 7 dengan nilai 0.873, menunjukkan bahwa model ini mampu mendeteksi sebagian besar objek pistol yang ada, dengan sedikit false negative.

Analisa mAP pada tabel 12 nilai mAP tertinggi pada kelas Knife terdapat pada skenario 7 dengan nilai 0.977, menunjukkan bahwa model ini memiliki performa terbaik secara keseluruhan untuk mendeteksi objek Knife dengan dengan optimal, pada kelas pistol mAP tertinggi terdapat pada skenario 4 dengan nilai 0.923, menunjukkan performa yang baik dalam mendeteksi objek pistol dengan optimal.

Analisa F1 score pada tabel 12 nilai F1 score tertinggi pada kelas Knife terdapat pada skenario 8 dengan nilai 0.947, menunjukkan bahwa model ini memiliki keseimbangan terbaik antara precision

dan recall, nilai F1 score tertinggi pada kelas Pistol terdapat pada skenario 6 dengan nilai 0.898, menunjukkan keseimbangan optimal antara precision dan recall untuk kelas Pistol.

TABEL 12.
Tabel hasil Test matriks evaluasi semua kelas pada dataset 1

Skenario	Precision		Recall		mAP		F1 score	
	Knife	Pistol	Knife	Pistol	Knife	Pistol	Knife	Pistol
1	0.921	0.895	0.8	0.8	0.9	0.9	0.896	0.860
2	0.92	0.921	0.8	0.8	0.9	0.9	0.874	0.841
3	0.92	0.921	0.8	0.8	0.9	0.9	0.874	0.856
4	0.933	0.928	0.9	0.8	0.9	0.9	0.934	0.887
5	0.932	0.877	0.8	0.8	0.9	0.9	0.897	0.859
6	0.906	0.916	0.8	0.8	0.9	0.9	0.895	0.898
7	0.953	0.893	0.9	0.8	0.9	0.9	0.940	0.883
8	0.964	0.893	0.9	0.8	0.9	0.9	0.947	0.866

E. Hasil Test dengan model pada dataset 2 (SNEHIL SANYAL)

Pada tabel 13 terdapat hasil test dari model yang sudah dilakukan training sebelumnya, terdapat 8 model berdasarkan skenario sebelumnya, selanjutnya akan dilakukan test dengan file weight yang sudah dilakukan training, hasil test menampilkan matriks evaluasi berupa Precision, Recall, dan mAP, dari file weight terhadap folder test ini, terlihat bahwa penggunaan parameter tinggi cenderung meningkatkan nilai precision, recall, mAP, dan F1 score. Selain itu, ukuran batch size dan image size yang lebih besar juga menunjukkan hasil yang lebih baik dalam beberapa skenario. skenario 4 dan 7 memiliki nilai yang cukup baik, skenario 7 menyatakan bahwa model lebih baik dalam melakukan mengidentifikasi objek positif dikarenakan nilai Precision lebih tinggi daripada skenario 4, skenario 4 menunjukan nilai yang dominan cukup tinggi daripada skenario yang lain. Skenario 4 dipilih karena memiliki nilai yang dominan cukup tinggi dengan nilai Precision sebesar 0.902, Recall sebesar 0.837, F1 score sebesar 0.868, dan mAP sebesar 0.897.

TABEL 13.
Tabel hasil Test matriks evaluasi keseluruhan dengan model pada dataset 2 (SNEHIL SANYAL)

Skenario	Precision	Recall	mAP	F1 Score
1	0.826	0.764	0.848	0.837
2	0.736	0.734	0.779	0.735
3	0.864	0.79	0.861	0.825
4	0.902	0.837	0.897	0.868
5	0.785	0.743	0.813	0.763
6	0.784	0.661	0.741	0.717
7	0.924	0.784	0.861	0.848
8	0.824	0.793	0.838	0.808

Analisa Precision pada tabel 14 nilai Precision

tertinggi pada setiap kelas terdapat pada skenario 7 dengan nilai 0.911 pada kelas *heavygun*, 0.988 pada kelas *knife*, 0.872 pada kelas *pistol*, menunjukkan bahwa model memiliki kemampuan yang sangat baik untuk mengidentifikasi objek setiap kelas secara benar tanpa menghasilkan banyak *false positives*.

Analisa *Recall* pada tabel 14 nilai *Recall* tertinggi pada kelas *Heavygun* terdapat pada skenario 7 dengan nilai 0.911, menunjukkan kemampuan model dalam mendeteksi sebagian besar objek *Heavygun* yang ada, dengan sedikit *false negatives*, nilai *recall* tertinggi pada kelas *Knife* terdapat pada skenario 3 dengan nilai 0.941, menunjukkan bahwa model ini mampu mendeteksi hampir semua objek *Knife* dengan *false negatives*, nilai *recall* tertinggi pada kelas *Pistol* terdapat pada skenario 4 dengan nilai 0.683, menunjukkan kemampuan terbaik dalam mendeteksi objek *Pistol* dengan sedikit *false negatives*.

TABEL 14.

Tabel hasil Test matriks evaluasi precision dan recall semua kelas pada dataset 2

Skenario	Precision			Recall		
	Heavygun	Knife	Pistol	Heavygun	Knife	Pistol
1	0.837	0.871	0.77	0.843	0.863	0.586
2	0.805	0.746	0.657	0.835	0.843	0.524
3	0.876	0.921	0.795	0.873	0.941	0.556
4	0.9	0.966	0.84	0.908	0.922	0.683
5	0.814	0.84	0.701	0.778	0.863	0.587
6	0.8	0.862	0.691	0.758	0.734	0.492
7	0.911	0.988	0.872	0.911	0.902	0.54
8	0.834	0.967	0.671	0.891	0.902	0.587

Analisa *mAP* pada tabel 15 nilai *mAP* tertinggi pada kelas *Heavygun* terdapat pada skenario 3 dengan nilai 0.947, menunjukkan performa terbaik secara keseluruhan dalam mendeteksi objek *Heavygun* dengan optimal, nilai *mAP* tertinggi pada kelas *Knife* terdapat pada skenario 4 dengan nilai 0.978, nilai *mAP* tertinggi pada kelas *Pistol* terdapat pada skenario 4 dengan nilai 0.769, menunjukkan bahwa model skenario 4 dapat mendeteksi objek *knife* dan *pistol* secara optimal.

Analisa *F1 score* pada tabel 15 nilai *F1 score* tertinggi pada kelas *Heavygun* dan *Knife* terdapat pada skenario 7 dengan nilai 0.911 pada kelas *Heavygun* dan 0.943 pada kelas *Knife*, menunjukkan keseimbangan optimal antara *precision* dan *recall* pada objek *Heavygun* dan *Knife*, nilai *F1 score* tertinggi pada kelas *Pistol* terdapat pada skenario 4 dengan nilai 0.753, menunjukkan performa terbaik dalam mendeteksi objek *Pistol* dengan keseimbangan yang baik antara *precision* dan *recall*.

TABEL 15.

Tabel hasil Test matriks evaluasi mAP dan F1 score semua kelas pada dataset 2

Skenario	mAP			F1 score		
	Heavygun	Knife	Pistol	Heavygun	Knife	Pistol
1	0.927	0.899	0.719	0.839	0.866	0.665

2	0.904	0.827	0.605	0.819	0.790	0.583
3	0.947	0.964	0.672	0.837	0.930	0.654
4	0.945	0.978	0.769	0.904	0.942	0.753
5	0.885	0.891	0.661	0.795	0.849	0.638
6	0.84	0.865	0.516	0.777	0.756	0.574
7	0.942	0.936	0.707	0.911	0.943	0.666
8	0.927	0.927	0.659	0.761	0.932	0.625

F. Hasil Test dengan model pada dataset 2 original (SNEHIL SANYAL)

Pada tabel 16 terdapat hasil *test* dari model yang sudah dilakukan *training* sebelumnya, terdapat 8 model berdasarkan skenario sebelumnya, selanjutnya akan dilakukan *test* dengan *file weight* yang sudah dilakukan *training*, hasil *test* menampilkan matriks evaluasi berupa *Precision*, *Recall*, dan *mAP*, dari *file weight* terhadap *folder test* ini, terlihat bahwa penggunaan *parameter* tinggi cenderung meningkatkan nilai *precision*, *recall*, *mAP*, dan *F1 score*. Selain itu, ukuran *batch size* dan *image size* yang lebih besar juga menunjukkan hasil yang lebih baik dalam beberapa skenario. skenario 3 dan 4 memiliki nilai yang cukup baik, skenario 4 menyatakan bahwa model lebih baik dalam mengukur objek kelas dikarenakan nilai *Recall* lebih tinggi daripada skenario 4, skenario 4 menunjukkan nilai yang dominan cukup tinggi daripada skenario yang lain. Skenario 3 dipilih karena memiliki nilai yang dominan cukup tinggi dengan nilai *Precision* sebesar 0.89, *Recall* sebesar 0.804, *F1 score* sebesar 0.845, dan *mAP* sebesar 0.888.

TABEL 16.

Tabel hasil Test matriks evaluasi keseluruhan dengan model pada dataset 2 original (SNEHIL SANYAL)

Skenario	Precision	Recall	mAP	F1 Score
1	0.641	0.384	0.384	0.480
2	0.669	0.482	0.548	0.560
3	0.89	0.804	0.888	0.845
4	0.807	0.839	0.858	0.823
5	0.627	0.674	0.66	0.650
6	0.669	0.471	0.52	0.553
7	0.841	0.786	0.862	0.813
8	0.77	0.709	0.784	0.738

Analisa *Precision* pada tabel 17 nilai *Precision* tertinggi pada kelas *Heavygun* dan *Knife* terdapat pada skenario 3 dengan nilai 0.922 pada kelas *Heavygun* dan 0.782 pada kelas *knife*, menunjukkan bahwa model ini memiliki kemampuan yang sangat baik untuk mendeteksi objek *Heavygun* dan *Knife* dengan sedikit *false positives*, nilai *Precision* tertinggi pada kelas *Pistol* terdapat pada skenario 2, 6, dan 7 dengan nilai 1.0, ini menunjukkan bahwa model ini sangat efektif dalam mendeteksi objek *Pistol* tanpa menghasilkan *false positives*.

Analisa *Recall* pada tabel 17 nilai *Recall* tertinggi pada kelas *Heavygun* terdapat pada skenario 4 dengan nilai 0.93, menunjukkan bahwa model ini mampu mendeteksi hampir semua objek *Heavygun* dengan sedikit *false negatives*, nilai

Recall tertinggi pada kelas *Knife* terdapat pada skenario 8 dengan nilai 0.85, ini menunjukkan bahwa model ini sangat efektif dalam mendeteksi objek *Knife*, nilai *Recall* tertinggi pada kelas *Pistol* terdapat pada skenario 4 dengan nilai 0.787, ini menunjukkan model memiliki kemampuan terbaik dalam mendeteksi objek *Pistol* dengan sedikit *false negatives*.

TABEL 17.

Tabel hasil Test matriks evaluasi precision dan recall semua kelas pada dataset 2 original

Skenario	Precision			Recall		
	Heavygun	Knife	Pistol	Heavygun	Knife	Pistol
1	0.599	0.446	0.877	0.521	0.35	0.281
2	0.599	0.408	1	0.845	0.6	0
3	0.922	0.782	0.965	0.887	0.8	0.725
4	0.846	0.623	0.953	0.93	0.8	0.787
5	0.576	0.344	0.961	0.831	0.709	0.481
6	0.576	0.432	1	0.764	0.65	0
7	0.825	0.697	1	0.915	0.7	0.742
8	0.761	0.651	0.899	0.807	0.85	0.471

Analisa *mAP* pada tabel 18 nilai *mAP* tertinggi pada kelas *Heavygun* terdapat pada skenario 3 dengan nilai 0.954, menunjukkan performa terbaik secara keseluruhan dalam mendeteksi objek *Heavygun* dengan optimal, nilai *mAP* tertinggi pada kelas *Knife* terdapat pada skenario 8 dengan nilai 0.807, menunjukkan bahwa model ini sangat optimal dalam mendeteksi objek *Knife* secara keseluruhan, nilai *mAP* tertinggi pada kelas *Pistol* terdapat pada skenario 3 dengan nilai 0.916, menunjukkan bahwa skenario ini optimal dalam mendeteksi objek *Pistol*. Analisa *F1 score* pada tabel 18 nilai *F1 score* tertinggi pada kelas *Heavygun* dan *Knife* terdapat pada skenario 3 dengan nilai 0.903 pada *Heavygun* dan 0.790 pada *Knife*, ini menunjukkan keseimbangan optimal antara *precision* dan *recall* pada objek *Heavygun* dan *Knife*, nilai *F1 score* tertinggi pada kelas *Pistol* terdapat pada skenario 4 dengan nilai 0.861, menunjukkan performa terbaik dalam mendeteksi objek *Pistol* dengan keseimbangan yang baik antara *precision* dan *recall*.

TABEL 18.

Tabel hasil matriks evaluasi Test *mAP* dan *F1 score* semua kelas pada dataset 2 original

Skenario	<i>mAP</i>			<i>F1 Score</i>		
	Heavygun	Knife	Pistol	Heavygun	Knife	Pistol
1	0.417	0.321	0.412	0.557	0.392	0.426
2	0.795	0.453	0.397	0.701	0.243	0
3	0.954	0.794	0.916	0.903	0.790	0.826
4	0.914	0.756	0.905	0.878	0.700	0.861
5	0.786	0.511	0.682	0.680	0.631	0.640
6	0.704	0.508	0.347	0.655	0.519	0
7	0.904	0.778	0.905	0.869	0.698	0.851
8	0.827	0.807	0.733	0.783	0.736	0.617

V. KESIMPULAN

Berdasarkan pengujian dan analisis yang dilakukan pada penelitian ini, metode YOLOv5s dengan melakukan modifikasi berupa kombinasi *parameter* telah berhasil dilakukan, konfigurasi *parameter* pada setiap skenario bertujuan untuk mendapatkan performa model dengan hasil yang baik, setelah dilakukan konfigurasi *parameter* dan didapatkan hasil *training* berupa *file weight best.pt* maka *file* tersebut dilakukan *test* pada *folder test* dan menghasilkan matriks evaluasi, hasil tertinggi dari matriks evaluasi tersebut akan dipilih sebagai model yang dapat mendeteksi dengan baik. Pada *dataset* pertama (*ARI-DaSCI*) memperoleh nilai terbaik pada skenario 7 dengan nilai *Precision* sebesar 0.923, *Recall* sebesar 0.901, *F1 score* sebesar 0.912, dan *mAP* sebesar 0.947. Pada *dataset* kedua milik (SNEHIL SANYAL) *dataset* dilakukan *test file original* dengan 3 kelas yaitu kelas *Heavygun*, *Knife*, dan *Pistol* menunjukkan skenario skenario 3 memiliki hasil yang dominan tinggi daripada skenario yang lain dengan nilai *Precision* sebesar 0.925, *Recall* 0.941, *mAP* 0.973, dan *F1 score* 0.933. Pada *dataset* kedua milik (SNEHIL SANYAL) yang dilakukan penyeimbangan data sebanyak 1500 data dan 3 kelas yaitu kelas *Heavygun*, *Knife*, dan *Pistol*, menunjukkan skenario 4 memiliki nilai yang dominan cukup tinggi daripada skenario yang lain dengan nilai *Precision* sebesar 0.902, *Recall* sebesar 0.837, *F1 score* sebesar 0.868, dan *mAP* sebesar 0.897. Berdasarkan penelitian ini menunjukkan bahwa kombinasi *parameter size*, *batch*, dan jumlah *epoch* yang lebih tinggi memberikan performa yang baik untuk model YOLOv5s dalam mendeteksi objek, namun pada beberapa kombinasi *parameter* mempunyai keunggulan pada masing-masing pengujian.

Saran untuk peneliti selanjutnya agar mencoba melakukan penelitian dengan *parameter batch*, *size*, *epoch*, dan *parameter* lain yang lebih kecil atau lebih besar, melakukan *augmentasi* data *train* agar variasi data lebih banyak sehingga meningkatkan performa model, mencoba beberapa variasi kondisi citra agar sistem dapat mengenali berbagai macam kondisi, penelitian ini juga dapat digunakan sebagai perbandingan untuk peneliti selanjutnya agar mendapatkan hasil evaluasi yang lebih optimal.

REFERENSI

- [1] Badan Pusat Statistik, Pusiknas Mabes Polri, kasus pembunuhan per tahun 2018-2020. [Online] Available at : <https://www.bps.go.id/indicator/34/1306/1/jumlah-kasus-kejahatan-pembunuhan-pada-satu-tahun-terakhir.html>. [accessed 5 november 2023]
- [2] PUSIKNAS Bareskrim Polri, Pelajar dan Mahasiswa pun Jadi Terlapor Kasus Pembunuhan. [Online] Available at : https://pusiknas.polri.go.id/detail_artikel/pelajar_dan_mahasiswa_pun_jadi_terlapor_kasus_pembunuhan

- han. [accessed 5 november 2023]
- [3] PUSIKNAS Polri, 3000 kasus pembunuhan dalam 4 tahun. [Online] Available at : https://pusiknas.polri.go.id/detail_artikel/lebih_3.000_orang_tewas_dibunuh_dalam_4_tahun. [accessed 5 november 2023]
- [4] PUSIKNAS Polri, Tindak Kasus Pembunuhan di Awal Tahun 2023. [Online] Available at : https://pusiknas.polri.go.id/detail_artikel/tindak_kasus_pembunuhan_di_awal_tahun. [accessed 5 november 2023]
- [5] PUSIKNAS Polri, Kasus Pembunuhan dengan Sajam. [Online] Available at : https://pusiknas.polri.go.id/detail_artikel/makin_banyak_kejahatan_dengan_menggunakan_senjata_tajam. [accessed 5 november 2023]
- [6] Undang-undang Republik Indonesia Nomor 56 Tahun 1999, tentang bela negara. [Online] Available at : <https://www.dpr.go.id/dokjdi/document/uu/457.pdf>. [accessed 5 november 2023]
- [7] Ahmed Abdullah A. Shareef, Pravin L. Yannawar, Antar Shaddad H. Abdul-Qawiy, Hashem Al-Nabhi, and Ravindra B. Bankar, <Deep Learning Based Model for Fire and Gun Detection=, 2022, AISR 176, pp. 422–430, 2023. DOI : https://doi.org/10.2991/978-94-6463-196-8_32
- [8] M.T. Bhatti, M.G. Khan, M. Aslam, dan M.J. Fiaz, <Weapon Detection in Real-Time CCTV Videos Using Deep Learning=, IEEE Access, vol. 9, pp. 34366–34382, Feb. 2021. DOI: [10.1109/ACCESS.2021.3059170](https://doi.org/10.1109/ACCESS.2021.3059170).
- [9] Wong, K., Chan, A. H., & Ngan, S. C. (2019). The effect of long working hours and overtime on occupational health: a meta-analysis of evidence from 1998 to 2018. *International journal of environmental research and public health*, 16(12), 2102. DOI: [10.3390/ijerph16122102](https://doi.org/10.3390/ijerph16122102)
- [10] Subarna Shakya, "Analysis of Artificial Intelligence based Image Classification Techniques", 2020. DOI: [10.36548/jiip.2020.1.005](https://doi.org/10.36548/jiip.2020.1.005)
- [11] Andreas Kaplan, Haenlein, M., <Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence=, 2018. DOI: <https://doi.org/10.1016/j.bushor.2018.08.004>
- [12] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, "you Only Look Once: Unified, Real-Time Object Detection", 2016. DOI: <https://doi.org/10.48550/arXiv.1506.02640>
- [13] Juan Terven, Diana Cordova-Esparza. <A Comprehensive Review of YOLO: From YOLOv1 and Beyond=, 2023. DOI: <https://doi.org/10.48550/arXiv.2304.00501>
- [14] Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao. <YOLOv4: Optimal Speed and Accuracy of Object Detection.=, (2020). DOI: <https://doi.org/10.48550/arXiv.2004.10934>
- [15] Xiaoning Li, Zhengzhong Wang, Shichao Geng, Lin Wang, Huaxiang Zhang, Li Liu & Donghua Li. <Yolov3-Pruning (transfer): real-time object detection algorithm based on transfer learning.= Journal of Real-Time Image Processing, Volume 19, pages 839–852, 20221
- [16] Renjie Xu, Haifeng Lin, Kangjie Lu, Lin Cao, dan Yunfei Liu. <A Forest Fire Detection System Based on Ensemble Learning.= Forests 12, no. 2 (2021): 217. doi: [10.3390/f12020217](https://doi.org/10.3390/f12020217).
- [17] Dahlan, I. A., Ariateja, D., Arghanie, M. A., Versantariqih, M. A., David, M., & Fatmawati, U. D. (2021), <Sistem Deteksi Senjata Otomatis Menggunakan Deep Learning Berbasis CCTV Cerdas=, Jurnal Sistem Cerdas, 4(2), 126 - 141. DOI: <https://doi.org/10.37396/jsc.v4i2.172>
- [18] Sanam Narejo, Bishwajeet Pandey, Doris Esenarrovargas, Ciro Rodriguez, and M. Rizwan Anjum, "Weapon Detection Using YOLO V3 for Smart Surveillance System", 2021. <https://doi.org/10.1155/2021/9975700>
- [19] Muhammad Asad, Tufail Sajjad Shah Hashmi, Osama Rasheed, "Multiplatform Surveillance System for Weapon Detection using YOLOv5", 2023. DOI: [10.1109/ICET56601.2022.10004690](https://doi.org/10.1109/ICET56601.2022.10004690)
- [20] Arif Warsi, Munaisyah Abdullah, Mohd Nizam Husen, Muhammad Yahya, Sheraz Khan, Nasreen Jawaid, "Gun detection system using YOLOv3", 2019. DOI: [10.1109/ICSIMA47653.2019.9057329](https://doi.org/10.1109/ICSIMA47653.2019.9057329)
- [21] Dharmapuri Siri, Pallo Bhanu Prasad Reddy, KVSL Harika, S.Ritwika, Shivani Sisodia, Karanam Madhavi, "Automated Weapon Detection System in CCTV's Through Image Processing", 2023. DOI: [10.1051/e3sconf/202343001055](https://doi.org/10.1051/e3sconf/202343001055)
- [22] Abdul Hanan Ashraf, Muhammad Imran, Abdulrahman M. Qahtani, Abdulmajeed Alsufyani, Omar Almutiry, Awais Mahmood, Muhammad Attique, Mohamed Habib, <Weapons Detection for Security and Video Surveillance Using CNN and YOLOV5s.= Computers, Materials & Continua, 2022. DOI: <https://doi.org/10.32604/cmc.2022.018785>
- [23] Jonathan Crall, <The MCC approaches the geometric mean of precision and recall as true negatives approach infinity=, arXiv, 2023. DOI : [arXiv.2305.00594](https://arxiv.org/abs/2305.00594)
- [24] ARI-DaSCI (Andalusian Research Institute in Data Science and Computational Intelligence), Dataset Pistol dan Pisau, 2020. [Online] Available at : <https://dasci.es/transferecia/open-data/24705/> , <https://github.com/ari-dasci/OD-WeaponDetection/> [accessed 5 desember 2023]
- [25] SNEHIL SANYAL, Weapon Detection Dataset, kaggle. [Online] Available at : <https://www.kaggle.com/datasets/snehilsanyal/weapon-detection-test/> [accessed 8 desember 2023]