

TESIS
OPTIMASI MULTI-OBJEKTIF YOLOV8 DENGAN BAYESIAN
HYPERPARAMETER TUNING UNTUK DETEKSI ALFABET
BISINDO DALAM PENDIDIKAN INKLUSIF



disusun oleh

YOPY TRI BUANA

24.51.1565

Konsentrasi : Business Intelligence

FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA

2026

TESIS
OPTIMASI MULTI-OBJEKTIF YOLOV8 DENGAN BAYESIAN
HYPERPARAMETER TUNING UNTUK DETEKSI ALFABET
BISINDO DALAM PENDIDIKAN INKLUSIF

MULTI-OBJECTIVE OPTIMIZATION OF YOLOV8 USING
BAYESIAN HYPERPARAMETER TUNING FOR BISINDO
ALPHABET DETECTION IN INCLUSIVE EDUCATION

Diajukan untuk memenuhi salah satu syarat mencapai derajat Pascasarjana
Program Studi **Business Intelligence**



disusun oleh

YOPY TRI BUANA

24.51.1565

Konsentrasi : Business Intelligence

FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA

2026

HALAMAN PERSETUJUAN

**OPTIMASI MULTI-OBJEKTIF YOLOV8 DENGAN BAYESIAN
HYPERPARAMETER TUNING UNTUK DETEKSI ALFABET
BISINDO DALAM PENDIDIKAN INKLUSIF**

**MULTI-OBJECTIVE OPTIMIZATION OF YOLOV8 USING
BAYESIAN HYPERPARAMETER TUNING FOR BISINDO
ALPHABET DETECTION IN INCLUSIVE EDUCATION**

yang disusun dan diajukan oleh

Yopy Tri Buana

24.51.1565

telah disetujui oleh Dosen Pembimbing Tesis
pada tanggal 5 Januari 2026

Dosen Pembimbing,



Dhani Ariamanto, S.Kom., M.Kom., Ph.D.
NIK. 190302197

HALAMAN PENGESAHAN

OPTIMASI MULTI-OBJEKTIF YOLOV8 DENGAN BAYESIAN
HYPERPARAMETER TUNING UNTUK DETEKSI ALFABET BISINDO
DALAM PENDIDIKAN INKLUSIF

MULTI-OBJECTIVE OPTIMIZATION OF YOLOV8 USING BAYESIAN
HYPERPARAMETER TUNING FOR BISINDO ALPHABET DETECTION
IN INCLUSIVE EDUCATION

yang disusun dan diajukan oleh

Yopy Tri Buana

24.51.1565

Telah dipertahankan di depan Dewan Penguji
pada tanggal 5 Januari 2026

Susunan Dewan Penguji

Nama Penguji

Dr. Andi Sunyoto, M.Kom.
NIK. 190302052

Dr. Kumara Ari Yuana, S.T., M.T.
NIK. 190302575

Dhani Ariatmanto, S.Kom., M.Kom., Ph.D.
NIK. 190302197

Tanda Tangan



Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer
Tanggal 5 Januari 2026

DEKAN FAKULTAS ILMU KOMPUTER



Prof. Dr. Kusriani, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Yopy Tri Buana
NIM : 24.51.1565

Menyatakan bahwa Tesis dengan judul berikut:

OPTIMASI MULTI-OBJEKTIF YOLOV8 DENGAN BAYESIAN HYPERPARAMETER TUNING UNTUK DETEKSI ALFABET BISINDO DALAM PENDIDIKAN INKLUSIF

Dosen Pembimbing : Dhani Ariatmanto, S.Kom., M.Kom., Ph.D.

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya.
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Dosen Pembimbing.
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini.
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta.
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi.

Yogyakarta, 5 Januari 2026

Yang Menyatakan,

Yopy Tri Buana

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tesis ini dengan baik. Tesis ini disusun sebagai salah satu persyaratan untuk memperoleh gelar Magister pada program studi yang penulis tempuh. Proses penyusunan tesis ini tidak terlepas dari bantuan, bimbingan, serta dukungan dari berbagai pihak.

Pada kesempatan ini penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. Mohammad Suyanto, M.M. selaku Rektor yang telah memberikan kesempatan kepada penulis untuk menempuh pendidikan pada jenjang magister.
2. Bapak Prof. Dr. Kusri, M.Kom. selaku Dekan Fakultas Ilmu Komputer yang telah memberikan dukungan dalam proses akademik selama penulis menempuh studi.
3. Bapak Dhani Ariatmanto, S.Kom., M.Kom., Ph.D. selaku dosen pembimbing yang telah memberikan bimbingan, arahan, motivasi, serta masukan yang sangat berharga selama proses penelitian dan penyusunan tesis ini.
4. Bapak I Made Artha Agastya, S.T., M.Eng., PhD dan Bapak Tonny Hidayat, S.Kom., M.Kom., Ph.D. selaku dosen penguji pada Seminar Proposal Tesis yang telah memberikan berbagai saran, kritik, serta masukan yang konstruktif demi penyempurnaan penelitian ini.
5. Bapak Dr. Andi Sunyoto, M.Kom. dan Bapak Dr. Kumara Ari Yuana, S.T., M.T. selaku dosen penguji pada Seminar Hasil dan Ujian Tesis yang telah memberikan berbagai masukan, arahan, serta evaluasi yang sangat bermanfaat dalam penyempurnaan tesis ini.
6. Kedua orang tua dan keluarga penulis yang senantiasa memberikan doa, dukungan, serta motivasi sehingga penulis dapat menyelesaikan studi dan penyusunan tesis ini dengan baik.
7. Semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyusunan tesis ini yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari bahwa tesis ini masih memiliki keterbatasan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa yang akan datang. Penulis berharap tesis ini dapat memberikan manfaat bagi pengembangan ilmu pengetahuan, khususnya di bidang teknologi informasi.

Yogyakarta, 5 Januari 2026

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN KEASLIAN TESIS	iv
KATA PENGANTAR	v
DAFTAR ISI	vi
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
INTISARI	xi
ABSTRACT	xii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	6
BAB 2 TINJAUAN PUSTAKA	8
2.1 Tinjauan Pustaka	8
2.2 Keaslian Penelitian	12
2.3 Landasan Teori	14
2.3.1 Pengantar Bahasa Isyarat dan Pengenalan Alfabet BISINDO	14
2.3.2 Computer Vision dan Deep Learning	14
2.3.3 Convolutional Neural Network (CNN)	15
2.3.4 Evolusi Arsitektur YOLO dari v1 sampai v8	15
2.3.5 YOLOv8n untuk Deteksi Real-time Huruf BISINDO.....	18
2.3.6 Konsep Hyperparameter Tuning dan Optimasi Model	20
2.3.7 Optimasi Pelatihan Model Deteksi Objek	21

2.3.8 Optimasi Hyperparameter Berbasis Bayesian	24
2.3.9 Metrik Evaluasi Kinerja Deteksi Objek	26
2.3.10 Deteksi Objek Real-Time	27
2.3.11 Composite Score untuk Optimasi Multi-Objektif	28
2.3.12 Ablation Study sebagai Metode Evaluasi Komponen Sistem	30
2.3.13 Kerangka Konseptual FEAMO-BO untuk Optimasi Multi-Objektif..31	
BAB 3 METODE PENELITIAN	35
3.1 Jenis, Sifat, dan Pendekatan Penelitian	35
3.2 Metode Pengumpulan Data	35
3.3 Metode Analisis Data	36
3.4 Alur Penelitian	37
3.4.1 Perumusan Masalah dan Studi Literatur	39
3.4.2 Pengumpulan Dataset	39
3.4.3 Split Dataset	39
3.4.4 Pemodelan dengan YOLOv8	40
3.4.5 Hyperparameter Tuning dengan Optuna	40
3.4.6 Evaluasi Model	41
BAB 4 HASIL PENELITIAN DAN PEMBAHASAN	42
4.1 Pengumpulan dan Persiapan Dataset	42
4.2 Pembagian Dataset	43
4.3 Analisis Hasil Pelatihan dan Evaluasi Model YOLOv8n Default	45
4.4 Hyperparameter Tuning dengan Optuna-FEAMO-BO	51
4.4.1 Konfigurasi dan Desain Eksperimen Optimasi	51
4.4.2 Mekanisme Bayesian Optimization dalam FEAMO-BO	52
4.4.3 Perhitungan Composite Score dan Formulasi Fungsi Objektif	56
4.4.4 Analisis Hasil Optimasi dan Pemilihan Model Terbaik	58
4.4.5 Analisis Kausal Hasil Optimasi FEAMO-BO	62

4.4.6 evaluasi akurasi model terbaik (mAP50-95)	64
4.4.7 Analisis Konvergensi dan Stabilitas Pelatihan	65
4.4.8 Analisis Efisiensi Inferensi	69
4.5 Perbandingan Model Default dan Model Hasil Optimasi	73
4.5.1 Analisis Confusion Matrix	75
4.5.2 Perbandingan dengan Penelitian Sebelumnya	79
4.6 Ablation Study dan Validasi Pemilihan Parameter	81
4.7 Evaluasi Keseluruhan dan Pembahasan Umum	86
BAB 5 PENUTUP	88
5.1 Kesimpulan	88
5.2 Saran	90
DAFTAR PUSTAKA	92



DAFTAR TABEL

Tabel 2.1 Matriks literatur review dan posisi penelitian	12
Tabel 2.2 Karakteristik Varian Model YOLOv8	17
Tabel 4.1 Sampel dataset	43
Tabel 4.2 Hasil Evaluasi Model Default YOLOv8n	48
Tabel 4.3 Search Space Hyperparameter	52
Tabel 4.4 Bobot Komponen Composite score FEAMO-BO	56
Tabel 4.5 Top 10 Trial Terbaik Hasil Optimasi FEAMO-BO	60
Tabel 4.6 Evaluasi mAP50-95 pada Berbagai Ambang IoU	64
Tabel 4.7 Perbandingan Model Baseline dan Model Hasil Optimasi	73
Tabel 4.8 Perbandingan Hasil Penelitian Ini dengan Penelitian Munandar	80
Tabel 4.9 Hasil Ablation Komponen FEAMO-BO	83
Tabel 4.10 Ringkasan Ablation Skema Pembobotan Composite Score	84

DAFTAR GAMBAR

Gambar 2.1 Arsitektur YOLOv8	17
Gambar 3.1 Alur Penelitian	38
Gambar 4.1 Distribusi Dataset Per-kelas	45
Gambar 4.2 Proporsi Split Dataset	45
Gambar 4.3 Kurva loss pelatihan dan validasi model YOLOv8n default	46
Gambar 4.4 Kurva mAP model YOLOv8n default selama pelatihan	47
Gambar 4.5 Kurva Konvergensi Composite Score pada 30 Trial	66
Gambar 4.6 Perbandingan Composite Score pada Top-10 Trial	68
Gambar 4.7 Distribusi FPS pada Top-10 Trial	70
Gambar 4.8 Perbandingan Latency Top-10 Trial	71
Gambar 4.9 Confusion Matrix pada Test Set	76
Gambar 4.10 salah prediksi pada Test Set	78



INTISARI

Penelitian ini bertujuan mengoptimalkan hyperparameter model YOLOv8n untuk deteksi alfabet Bahasa Isyarat Indonesia (BISINDO) secara akurat, efisien, dan stabil dalam konteks real-time guna mendukung pendidikan dan komunikasi inklusif. Penelitian dibatasi pada deteksi alfabet BISINDO berbasis citra statis dengan variabel utama meliputi akurasi deteksi (mAP50-95), kecepatan inferensi (FPS dan latency), serta stabilitas pelatihan antar-epoch. Metode yang digunakan adalah pendekatan FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization (FEAMO-BO) dengan algoritma Tree-structured Parzen Estimator (TPE) melalui pustaka Optuna. Dataset yang digunakan terdiri atas 11.500 citra alfabet BISINDO beresolusi 640×640 piksel yang telah melalui pre-augmentation dan dibagi menjadi 80% data latih, 10% data validasi, dan 10% data uji. Proses eksperimen dilakukan dengan melatih model YOLOv8n default sebagai pembandingan awal, kemudian melakukan optimasi hyperparameter menggunakan FEAMO-BO, di mana setiap trial dievaluasi melalui composite score yang mengintegrasikan mAP50-95, FPS, latency, dan stabilitas pelatihan. Hasil penelitian menunjukkan bahwa model terbaik hasil optimasi FEAMO-BO mampu mencapai mAP50-95 sebesar 0,9350, kecepatan inferensi 133,73 FPS, dan latency 7,44 ms dengan pola konvergensi yang stabil, serta memberikan peningkatan signifikan pada efisiensi inferensi dibandingkan model YOLOv8n default dan penelitian terdahulu berbasis YOLOv5s tanpa penurunan akurasi yang berarti. Dengan demikian, pendekatan FEAMO-BO terbukti efektif sebagai kerangka optimasi multi-objektif dan layak digunakan sebagai dasar pengembangan sistem pendukung komunikasi dan pendidikan inklusif bagi penyandang Tuli di Indonesia.

Kata kunci: YOLOv8, BISINDO, Bayesian Optimization, FEAMO-BO, pendidikan inklusif.

ABSTRACT

This study aims to optimize the hyperparameters of the YOLOv8n model for accurate, efficient, and stable real-time detection of the Indonesian Sign Language (BISINDO) alphabet to support inclusive education and communication. The research is limited to static image-based BISINDO alphabet detection, with key variables including detection accuracy (mAP50–95), inference speed (FPS and latency), and inter-epoch training stability. The proposed method adopts a FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization (FEAMO-BO) approach using the Tree-structured Parzen Estimator (TPE) algorithm implemented through the Optuna framework. The dataset consists of 11,500 BISINDO alphabet images with a resolution of 640×640 pixels that have undergone pre-augmentation and are split into 80% training, 10% validation, and 10% testing sets. The experimental process begins by training the YOLOv8n default model as a baseline, followed by hyperparameter optimization using FEAMO-BO, where each trial is evaluated through a composite score integrating mAP50–95, FPS, latency, and training stability. The results show that the best model obtained through FEAMO-BO achieves a mAP50–95 of 0.9350, an inference speed of 133.73 FPS, and a latency of 7.44 ms with a stable convergence pattern, while providing a significant improvement in inference efficiency compared to the YOLOv8n default model and previous studies based on YOLOv5s without a meaningful loss in accuracy. Therefore, the proposed FEAMO-BO approach is proven effective as a multi-objective optimization framework and can serve as a foundation for developing computer vision-based inclusive communication and educational systems for the Deaf community in Indonesia.

Keywords: YOLOv8, BISINDO, Bayesian Optimization, FEAMO-BO, inclusive education.

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Bahasa isyarat merupakan sarana komunikasi utama bagi individu dengan gangguan pendengaran dan/atau bicara. Menurut World Health Organization (WHO), lebih dari 430 juta orang di seluruh dunia mengalami gangguan pendengaran, termasuk sekitar 34 juta anak-anak [1]. Kondisi ini menegaskan pentingnya penyediaan sistem komunikasi yang inklusif guna menjamin hak individu Tuli dalam mengakses pendidikan dan informasi secara setara.

Di Indonesia, terdapat dua sistem bahasa isyarat yang digunakan, yaitu Bahasa Isyarat Indonesia (BISINDO) dan Sistem Isyarat Bahasa Indonesia (SIBI). SIBI diperkenalkan pada tahun 1970-an sebagai upaya standarisasi bahasa isyarat dalam pendidikan formal. Namun, SIBI kerap dikritik karena bersifat struktural dan kurang merepresentasikan pola komunikasi alami komunitas Tuli. Sebaliknya, BISINDO berkembang secara organik melalui interaksi sosial komunitas Tuli dan dinilai lebih natural serta inklusif dalam praktik komunikasi sehari-hari [2], [3]. Oleh karena itu, BISINDO lebih banyak digunakan dalam konteks komunikasi nyata, khususnya di lingkungan sosial dan pendidikan inklusif.

Dalam praktik pendidikan di Indonesia, keterbatasan kemampuan pendidik dan tenaga kependidikan dalam menggunakan bahasa isyarat masih menjadi hambatan utama bagi proses pembelajaran siswa Tuli. Tantangan komunikasi ini berpotensi menghambat pemahaman materi, interaksi di kelas, serta partisipasi aktif

siswa. Dalam konteks kebijakan pendidikan yang menekankan kesetaraan akses dan inklusivitas, kebutuhan akan solusi teknologi pendukung komunikasi bagi penyandang disabilitas komunikasi menjadi semakin mendesak [4]. Tanpa dukungan sistem bantu yang memadai, siswa Tuli berisiko mengalami kesenjangan pembelajaran dibandingkan dengan siswa non-disabilitas.

Perkembangan kecerdasan buatan, khususnya pada bidang visi komputer (computer vision), membuka peluang besar dalam mendukung komunikasi inklusif. Visi komputer memungkinkan sistem untuk menganalisis dan memahami data visual seperti citra dan video, sehingga mampu mengenali pola gerakan tangan yang membentuk bahasa isyarat. Penelitian yang dilakukan oleh Vidhyasagar menyebutkan bahwa pendekatan visi komputer dapat digunakan untuk mengenali pola visual pada bahasa isyarat secara efektif [5]. Berbagai metode deep learning berbasis Convolutional Neural Network (CNN) serta algoritma You Only Look Once (YOLO) telah banyak digunakan karena kemampuannya dalam mendeteksi objek secara real-time dengan tingkat akurasi yang tinggi.

Sejumlah penelitian terdahulu telah menunjukkan potensi penggunaan teknologi deteksi objek untuk alfabet BISINDO. Munandar mengimplementasikan YOLOv5s menggunakan dataset citra alfabet BISINDO hasil augmentasi dan memperoleh nilai precision sebesar 85,2%, recall 89,3%, F1-score 87,2%, serta mAP sebesar 87,1% [6]. Penelitian lain oleh Renaningtias merancang sistem deteksi alfabet BISINDO berbasis YOLOv7 dengan input video dan menunjukkan bahwa sistem tersebut mampu melakukan klasifikasi alfabet secara real-time serta

efektif dalam mendukung komunikasi Tuli tanpa penerjemah manusia secara langsung [7].

Meskipun penelitian-penelitian tersebut menunjukkan performa deteksi yang baik, evaluasi yang dilakukan umumnya menitikberatkan pada metrik akurasi sesuai dengan tujuan masing-masing penelitian. Aspek lain yang berperan penting dalam penerapan sistem real-time, seperti kecepatan inferensi (FPS), latency, serta konsistensi performa pelatihan antar-epoch, belum menjadi fokus utama dalam proses pemilihan konfigurasi model. Padahal, dalam konteks penggunaan nyata di lingkungan pendidikan inklusif, sistem deteksi bahasa isyarat tidak hanya dituntut akurat, tetapi juga responsif dan stabil selama penggunaan.

Berdasarkan kondisi tersebut, penelitian ini mengembangkan model deteksi alfabet BISINDO berbasis YOLOv8n yang dioptimasi menggunakan pendekatan FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization (FEAMO-BO). Pendekatan ini dirancang untuk menyeimbangkan tiga aspek utama performa, yaitu akurasi deteksi, efisiensi komputasi (FPS dan latency), serta stabilitas pelatihan antar-epoch dalam proses pemilihan konfigurasi model yang terintegrasi. Dataset yang digunakan berjumlah sekitar 11.500 citra alfabet BISINDO (A-Z) dengan resolusi 640×640 piksel yang dibagi ke dalam data pelatihan, validasi, dan pengujian. Dengan demikian, penelitian ini diarahkan untuk menghasilkan model deteksi alfabet BISINDO yang tidak hanya akurat, tetapi juga efisien dan stabil, sehingga lebih relevan untuk diterapkan pada sistem komunikasi real-time dalam pendidikan inklusif.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, rumusan masalah penelitian ini dapat disusun sebagai berikut:

1. Bagaimana pendekatan FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization (FEAMO-BO) dapat diterapkan untuk mengoptimalkan hyperparameter model YOLOv8n dalam deteksi alfabet BISINDO agar memenuhi kebutuhan akurasi, efisiensi inferensi, dan stabilitas pelatihan pada skenario real-time?
2. Sejauh mana peningkatan performa model hasil optimasi FEAMO-BO dibandingkan model YOLOv8n default, serta dibandingkan dengan penelitian terdahulu berbasis YOLOv5s pada aspek akurasi deteksi (mAP50-95), dan bagaimana karakteristik performa tambahan model hasil optimasi FEAMO-BO ditinjau dari FPS, latency, dan stabilitas konvergensi?
3. Bagaimana pengaruh variasi konfigurasi hyperparameter terhadap nilai composite score serta hubungan trade-off antar-metrik (akurasi, FPS, latency, dan stabilitas pelatihan) dalam proses pemilihan model menggunakan Bayesian Optimization dan analisis ablation?

1.3 Batasan Masalah

Untuk menjaga fokus penelitian dan memastikan kelayakan pelaksanaan eksperimen, ruang lingkup penelitian ini dibatasi pada aspek-aspek berikut:

- a. Penelitian hanya mencakup 26 huruf alfabet BISINDO (A-Z) dengan gestur statis dan tidak melibatkan angka, kata, atau kalimat. Sistem Isyarat Bahasa Indonesia (SIBI) juga tidak termasuk dalam cakupan penelitian ini.
- b. Dataset yang digunakan terdiri atas ±11.500 citra dari dataset publik (Kaggle) alfabet BISINDO. Seluruh citra diproses pada resolusi 640×640 piksel guna menjaga keseimbangan antara akurasi deteksi dan efisiensi komputasi.

- c. Penelitian ini hanya menggunakan YOLOv8n sebagai model baseline sekaligus objek optimasi. Pemilihan varian ini didasarkan pada karakteristik dataset yang ringan (26 kelas statis) serta keterbatasan sumber daya GPU NVIDIA RTX 2060 (6 GB VRAM). YOLOv8n dipilih karena menawarkan trade-off terbaik antara kecepatan inferensi real-time dan konsumsi sumber daya.
- d. Proses optimasi hyperparameter dilakukan menggunakan Bayesian Optimization dengan algoritma Tree-structured Parzen Estimator (TPE) yang diimplementasikan melalui pustaka Optuna.
- e. Evaluasi model dilakukan pada perangkat desktop GPU dengan metrik mAP50-95, precision, recall, FPS, latency, dan stabilitas pelatihan. Implementasi langsung pada perangkat edge atau mobile tidak dilakukan dan menjadi peluang penelitian lanjutan.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk mengoptimalkan kinerja model YOLOv8n dalam mendeteksi alfabet BISINDO melalui pendekatan FEAMO-BO (FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization) dengan mempertimbangkan akurasi deteksi, efisiensi komputasi, dan stabilitas pelatihan antar-epoch. Tujuan khusus penelitian ini adalah:

- a. Mengevaluasi efektivitas pendekatan FEAMO-BO dalam mengoptimalkan model YOLOv8n agar mampu mendeteksi 26 huruf alfabet BISINDO secara akurat, efisien, dan real-time, dengan mempertimbangkan stabilitas konvergensi antar-epoch selama proses pelatihan.

- b. Menganalisis peningkatan akurasi model setelah dilakukan hyperparameter tuning berbasis FEAMO-BO, serta membandingkan hasilnya dengan model baseline YOLOv8n dan penelitian terdahulu (YOLOv5s = 87,1 %), guna menilai kontribusi pendekatan multi-objektif terhadap performa deteksi.
- c. Menilai pengaruh optimasi hyperparameter berbasis FEAMO-BO terhadap efisiensi komputasi, yang meliputi peningkatan frames per second (FPS), penurunan latency, serta pengurangan waktu pelatihan per epoch, untuk menilai kelayakan performa model sebagai prasyarat penerapan pada sistem deteksi bahasa isyarat real-time dan memiliki potensi adaptasi pada perangkat edge atau mobile melalui teknik optimasi lanjutan seperti quantization (FP16/INT8).

1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat pada berbagai aspek, baik akademik, praktis, institusional, maupun industri. Setiap aspek berikut merepresentasikan kontribusi langsung dari penerapan kerangka FEAMO-BO (FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization) dalam pengembangan sistem deteksi alfabet Bahasa Isyarat Indonesia (BISINDO) berbasis YOLOv8n.

a. Aspek Akademik

Memberikan kontribusi ilmiah dalam bidang optimasi machine learning melalui pendekatan multi-objektif yang mengintegrasikan akurasi deteksi, efisiensi komputasi, dan stabilitas pelatihan antar-epoch.

b. Aspek Praktis

Menyediakan sistem deteksi alfabet BISINDO yang berpotensi digunakan sebagai alat bantu komunikasi dan pembelajaran bagi siswa Tuli di lingkungan pendidikan inklusif.

c. Aspek Institusional

Mendukung penguatan aksesibilitas pendidikan berbasis teknologi kecerdasan buatan di Indonesia.

d. Aspek Industri

Membuka peluang pengembangan produk teknologi pendidikan dan komunikasi inklusif berbasis AI yang adaptif dan efisien.



BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Penelitian mengenai deteksi dan pengenalan bahasa isyarat telah berkembang seiring dengan kemajuan teknologi computer vision dan deep learning. Tujuan utama dari penelitian-penelitian tersebut adalah memungkinkan sistem komputer mengenali gerakan tangan manusia dan menerjemahkannya menjadi karakter alfabet atau simbol yang bermakna. Dalam konteks Indonesia, Bahasa Isyarat Indonesia (BISINDO) merupakan sarana komunikasi utama bagi penyandang disabilitas rungu dan wicara, sehingga pengembangan sistem pengenalan dan deteksi BISINDO menjadi bidang penelitian yang penting dan relevan.

Pendekatan awal dalam pengenalan alfabet BISINDO umumnya menggunakan metode klasifikasi berbasis Convolutional Neural Network (CNN). Dwijayanti et al. (2021) mengusulkan arsitektur CNN untuk mengenali alfabet BISINDO menggunakan dataset berskala besar yang terdiri dari 39.455 citra[8]. Model yang dikembangkan menunjukkan performa yang sangat tinggi dengan akurasi mencapai 98,3% dan F1-score sebesar 99,3%. Hasil tersebut menunjukkan kemampuan CNN dalam mengekstraksi pola visual alfabet BISINDO secara efektif. Namun, pendekatan ini masih terbatas pada klasifikasi citra statik dan belum mendukung kebutuhan sistem deteksi objek berbasis bounding box maupun pemrosesan real-time.

Pendekatan CNN berbasis BISINDO juga dikaji oleh Nizhamuddin Ahmad et al. (2023) melalui pengembangan model CNN sederhana untuk pengenalan alfabet BISINDO [9]. Penelitian ini melaporkan akurasi sebesar 82,56% dengan precision 84,76% dan F1-score 82,30%. Meskipun performanya lebih rendah dibandingkan CNN berskala besar, penelitian ini berperan sebagai baseline yang menunjukkan keterbatasan pendekatan klasifikasi statik dalam menangani variasi posisi tangan, latar belakang, serta kebutuhan sistem real-time.

Seiring dengan meningkatnya kebutuhan akan sistem yang mampu bekerja secara real-time dan lebih fleksibel terhadap variasi posisi objek, pendekatan berbasis deteksi objek mulai diterapkan pada pengenalan BISINDO. Munandar et al. (2024) mengembangkan sistem deteksi alfabet BISINDO berbasis YOLOv5 dengan memanfaatkan dataset lebih dari 11.500 citra hasil augmentasi [6]. Evaluasi performa menunjukkan nilai mAP@IoU 0.5 sebesar 87,1% dengan precision 85,2% dan recall 89,3%. Penelitian ini menunjukkan bahwa YOLO memiliki potensi sebagai algoritma deteksi objek yang cepat dan cukup akurat untuk BISINDO, meskipun evaluasi masih menitikberatkan pada metrik akurasi deteksi sesuai dengan tujuan penelitian tersebut.

Pengembangan lebih lanjut pada deteksi BISINDO berbasis YOLO dilakukan oleh Ningsih et al. (2024) dengan menggunakan arsitektur YOLOv5-NAS-S dan pendekatan fine-tuning [10]. Penelitian ini melaporkan performa yang sangat tinggi dengan mAP@IoU 0.5 sebesar 97,26% dan recall 99,68% pada 47 kelas alfabet BISINDO, serta mendukung penggunaan pada citra, video, dan webcam. Meskipun demikian, pemilihan konfigurasi model pada penelitian ini dilakukan berdasarkan

eksperimen dan evaluasi metrik yang relevan dengan fokus penelitian, tanpa menggabungkan aspek efisiensi inferensi ke dalam satu kerangka pemilihan model yang terintegrasi.

Pendekatan deteksi BISINDO berbasis video ditunjukkan oleh Renaningtias et al. (2025) yang menggunakan YOLOv7 untuk mendeteksi alfabet BISINDO pada data video [7]. Hasil penelitian menunjukkan performa yang sangat tinggi dengan mAP@IoU 0.5 sebesar 0,995 serta nilai precision dan recall mencapai 1,00 pada kondisi dataset yang terkontrol. Penelitian ini menunjukkan kemampuan YOLO dalam menangani skenario video, namun evaluasi difokuskan pada performa deteksi akhir dan belum menitikberatkan pada analisis efisiensi inferensi maupun konsistensi performa pelatihan.

Kajian terbaru terkait deteksi alfabet BISINDO dilakukan oleh Hayati et al. (2025) melalui evaluasi algoritma YOLO pada berbagai kondisi pencahayaan dan latar belakang [11]. Penelitian ini melaporkan nilai Average Precision mendekati 0,99 dan Average Recall mendekati 1,00, dengan akurasi yang tetap tinggi pada kondisi indoor maupun outdoor. Hasil tersebut menegaskan pentingnya robustness model dalam skenario dunia nyata. Namun, pemilihan konfigurasi model masih didasarkan pada evaluasi performa yang relevan dengan skenario pengujian, tanpa mekanisme formal untuk menggabungkan beberapa metrik dalam satu proses seleksi model.

Berdasarkan perkembangan penelitian yang telah dibahas, arah state-of-the-art (SOTA) pada deteksi alfabet BISINDO menunjukkan pergeseran dari pendekatan klasifikasi citra statik berbasis CNN menuju sistem deteksi objek

berbasis YOLO yang mendukung pemrosesan real-time serta memiliki ketahanan (robustness) terhadap variasi lingkungan. Penelitian-penelitian terkini telah mencapai tingkat akurasi yang tinggi dan mulai mengeksplorasi penerapan pada data video serta kondisi pencahayaan yang beragam. Namun demikian, meskipun arah pengembangan tersebut telah terlihat jelas, proses pemilihan konfigurasi model pada penelitian-penelitian terdahulu umumnya masih didasarkan pada evaluasi metrik secara individual, dengan fokus utama pada akurasi deteksi, tanpa mempertimbangkan secara terintegrasi aspek efisiensi inferensi dan stabilitas pelatihan yang krusial dalam skenario penggunaan real-time. Berdasarkan tinjauan pustaka tersebut, dapat diidentifikasi tiga research gap utama, yaitu belum adanya pendekatan optimasi multi-objective yang secara simultan mengintegrasikan akurasi deteksi, efisiensi inferensi, dan stabilitas pelatihan; pemilihan konfigurasi model yang masih berbasis metrik tunggal sehingga berpotensi menghasilkan solusi yang suboptimal untuk aplikasi real-time; serta belum tersedianya mekanisme formal yang memastikan kelayakan real-time secara kuantitatif dalam proses seleksi model. Oleh karena itu, penelitian ini berada pada kelanjutan arah SOTA tersebut dengan mengembangkan pendekatan pemilihan konfigurasi model yang lebih terstruktur dan terukur dalam konteks deteksi alfabet BISINDO.

2.2 Keaslian Penelitian

Tabel 2.1 Matriks literatur review dan posisi penelitian
OPTIMASI MULTI-OBJEKTIF YOLOV8 DENGAN BAYESIAN HYPERPARAMETER TUNING UNTUK DETEKSI ALFABET BISINDO DALAM PENDIDIKAN INKLUSIF

No	Judul Penelitian	Nama Peneliti, Tahun, Index	Metode Penelitian	Hasil	Keunggulan dan Kelemahan	Perbandingan
1	Indonesian Sign Language (BISINDO) Alphabet Detection System Using YOLO (You Only Look Once) Algorithm	Andra Munandar et al., 2024	YOLOv5 (Object Detection)	mAP@IoU 0.5 = 87.1%, Precision = 85.2%, Recall = 89.3%, F1-score = 87.2%	Keunggulan: Menunjukkan potensi YOLO untuk deteksi alfabet BISINDO. Kelemahan: Optimasi hyperparameter belum dilakukan dan efisiensi inferensi belum dievaluasi secara kuantitatif.	Penelitian ini menggunakan YOLOv8n dan melakukan optimasi hyperparameter berbasis Bayesian dengan mempertimbangkan akurasi, FPS, latency, dan stabilitas pelatihan secara terintegrasi.
2	Deteksi Bahasa Isyarat Indonesia (BISINDO) pada Video dengan YOLOv7	Renaningtias et al., 2025	YOLOv7 (Object Detection – Video)	mAP@IoU 0.5 = 0.995, Precision = 1.00, Recall = 1.00, F1-score = 1.00	Keunggulan: Performa sangat tinggi pada data video BISINDO. Kelemahan: Dataset terbatas dan evaluasi efisiensi inferensi serta stabilitas pelatihan belum dilakukan.	Penelitian ini mengevaluasi efisiensi inferensi dan stabilitas pelatihan sebagai bagian dari proses pemilihan model, bukan hanya pada performa akhir.
3	Sign Language Detection System Using YOLOv5 Algorithm to Promote Communication Equality People with Disabilities	Maylinna R. Ningsih et al., 2024	YOLOv5-NAS-S + Fine-Tuning	mAP@IoU 0.5 = 97.26%, Recall = 99.68%, 47 kelas BISINDO	Keunggulan: Performa sangat tinggi pada data video BISINDO. Kelemahan: Dataset terbatas dan evaluasi efisiensi inferensi serta stabilitas pelatihan belum dilakukan. Kelemahan: Belum diimplementasikan pada model deteksi objek seperti YOLO sehingga kontribusinya masih bersifat konseptual.	Penelitian ini memformalkan pemilihan konfigurasi model menggunakan optimasi multi-objective sehingga tidak bergantung pada konfigurasi manual.

No	Judul Penelitian	Nama Peneliti, Tahun, Index	Metode Penelitian	Hasil	Keunggulan dan Kelemahan	Perbandingan
4	Optimizing YOLO-Based Algorithms for Real-Time BISINDO Alphabet Detection Under Varied Lighting and Background Conditions in Computer Vision Systems	Lilis Nur Hayati et al., 2025	YOLO-based Object Detection + Evaluasi Kondisi Lingkungan	Average Precision (AP) = 0.99, Average Recall (AR) ≈ 1.00; akurasi indoor 0.889–0.978, outdoor 0.933–0.959	Keunggulan: Mengevaluasi robustness YOLO pada berbagai kondisi pencahayaan dan latar belakang BISINDO. Kelemahan: Optimasi belum menggunakan pendekatan probabilistik dan belum mengintegrasikan stabilitas pelatihan.	Penelitian ini menambahkan optimasi Bayesian multi-objective untuk memilih konfigurasi yang optimal secara sistematis di bawah berbagai batasan performa.
5	Indonesian Sign Language Recognition using Convolutional Neural Network	Suci Dwijayanti et al., 2021	CNN (Custom Architecture vs AlexNet & VGG-16)	Accuracy = 98.3%, Precision = 98.3%, Recall = 98.4%, F1-score = 99.3%	Keunggulan: Akurasi sangat tinggi pada citra statik BISINDO. Kelemahan: Tidak mendukung deteksi objek dan skenario real-time. Kelemahan: Tidak dirancang untuk pemrosesan real-time sehingga kurang aplikatif pada sistem deteksi berbasis video.	Penelitian ini menggunakan deteksi objek berbasis YOLO yang lebih sesuai untuk aplikasi real-time dibandingkan klasifikasi statik.
6	Transforming Sign Language using CNN Approach based on BISINDO Dataset	Nizamuddin Ahmad et al., 2023	CNN (Vision-based Classification)	Accuracy = 82.56%, Precision = 84.76%, Recall = 82.56%, F1-score = 82.30%	Keunggulan: CNN sederhana dan mudah direproduksi pada dataset BISINDO. Kelemahan: Akurasi relatif lebih rendah dan bersifat klasifikasi statik.	Penelitian ini berfokus pada pemilihan model deteksi objek yang optimal untuk skenario real-time, bukan pada klasifikasi citra statik.

2.3 Landasan Teori

2.3.1 Pengantar Bahasa Isyarat dan Pengenalan Alfabet BISINDO

Bahasa isyarat merupakan sistem komunikasi visual-spasial yang digunakan oleh komunitas Tuli untuk menyampaikan makna melalui gestur tangan, ekspresi wajah, dan gerak tubuh [8]. Di Indonesia terdapat dua sistem utama, yaitu SIBI dan BISINDO. SIBI disusun secara formal berdasarkan struktur bahasa Indonesia, sedangkan BISINDO berkembang secara alami di komunitas Tuli dan lebih ekspresif dalam komunikasi sehari-hari.

BISINDO kini menjadi fokus banyak penelitian karena potensinya dalam pendidikan inklusif dan transformasi digital komunikasi antar komunitas. Penelitian [12] menunjukkan bahwa sistem deteksi alfabet BISINDO berbasis deep learning dapat memperkuat pembelajaran interaktif di sekolah inklusi.

2.3.2 Computer Vision dan Deep Learning

Computer Vision (CV) adalah bidang kecerdasan buatan yang memungkinkan komputer memahami isi gambar dan video. Melalui tahapan feature extraction, object localization, dan classification, CV menjadi dasar dari deteksi isyarat berbasis kamera.

Kemajuan deep learning khususnya CNN dan YOLO meningkatkan akurasi serta efisiensi pengenalan visual. Penelitian Maulidina et al. (2024) menunjukkan model ResNet dan YOLO mampu mendeteksi huruf BISINDO dengan mAP > 90% dan FPS di atas 60 [13].

2.3.3 Convolutional Neural Network (CNN)

CNN berfungsi mengekstraksi fitur spasial dari citra melalui lapisan konvolusi dan pooling yang meniru cara kerja sistem visual manusia. CNN terdiri atas lapisan konvolusi untuk mendeteksi pola, lapisan pooling untuk reduksi dimensi, dan lapisan fully-connected untuk klasifikasi. Dalam konteks BISINDO, CNN digunakan untuk menerjemahkan bentuk tangan menjadi huruf atau kata. Namun CNN konvensional bekerja secara statis dan tidak mengidentifikasi lokasi spasial isyarat secara langsung, sehingga perlu integrasi dengan algoritma deteksi seperti YOLO untuk menghasilkan sistem real-time [14].

2.3.4 Evolusi Arsitektur YOLO dari v1 sampai v8

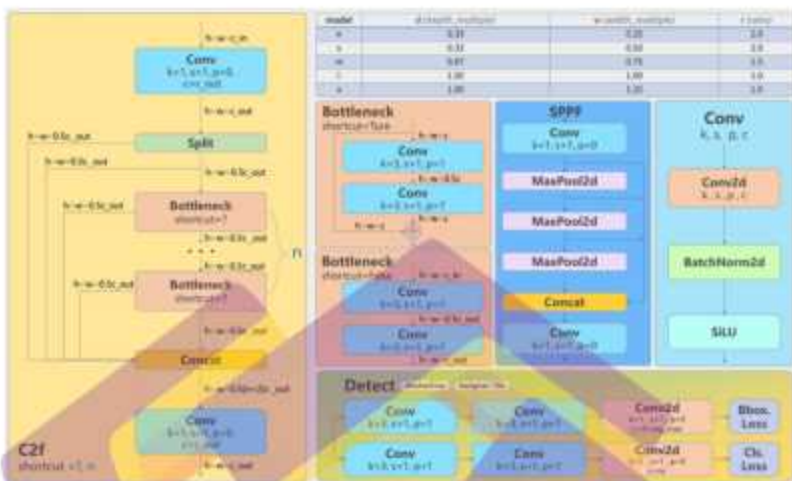
Algoritma You Only Look Once (YOLO) pertama kali diperkenalkan oleh Redmon et al. (2016) sebagai pendekatan deteksi objek satu tahap (single-stage detector) yang mereformulasikan proses deteksi sebagai regresi langsung dari citra ke koordinat bounding box dan kelas objek. Dengan memproses citra hanya satu kali melalui sebuah jaringan Convolutional Neural Network (CNN) terpadu, YOLOv1 mampu mencapai kecepatan inferensi real-time hingga 45 FPS pada GPU Titan X, menjadikannya pionir dalam deteksi objek real-time. Namun demikian, pendekatan awal ini masih memiliki keterbatasan dalam hal akurasi dan presisi lokalisasi objek, sehingga nilai mean Average Precision (mAP) yang diperoleh masih lebih rendah dibandingkan metode dua tahap seperti Faster R-CNN [15].

Pengembangan berlanjut pada YOLOv2 dan YOLOv3, yang meningkatkan performa secara signifikan. YOLOv2 memperkenalkan anchor boxes, training multiskala, serta backbone Darknet-19 yang lebih dalam, menghasilkan mAP 78,6% pada VOC 2007 dengan 40 FPS [16]. Sementara itu, YOLOv3 menggunakan

backbone Darknet-53 dengan residual connections dan deteksi multi-skala yang meningkatkan deteksi objek kecil dan menengah secara bersamaan [17].

Evolusi berikutnya pada YOLOv4 dan YOLOv5 menekankan optimalisasi efisiensi komputasi dan generalisasi model. YOLOv4 mengintegrasikan CSPDarknet53, PANet, serta Spatial Pyramid Pooling (SPP) untuk meningkatkan aliran informasi fitur dan akurasi deteksi pada dataset skala besar. YOLOv5, yang dikembangkan oleh Ultralytics dan diimplementasikan menggunakan framework PyTorch, menghadirkan berbagai varian ukuran model (nano hingga extra-large) serta kemudahan integrasi dan pelatihan. Meskipun YOLOv5 tidak dipublikasikan melalui jurnal peer-review tradisional, model ini banyak digunakan dalam praktik industri karena fleksibilitas, efisiensi, dan performanya yang kompetitif [18].

Model-model generasi terbaru seperti YOLOv6, YOLOv7, dan YOLOv8 terus menyempurnakan arsitektur deteksi objek untuk memenuhi kebutuhan aplikasi modern. YOLOv6 dioptimalkan untuk kebutuhan industri dengan fokus pada inferensi sangat cepat dan pendekatan anchor-free label assignment. YOLOv7 memperkenalkan konsep dual-head dan re-parameterization untuk meningkatkan efisiensi pelatihan dan inferensi. YOLOv8, sebagai rilis terbaru dari Ultralytics, mengadopsi arsitektur yang lebih modular dengan decoupled detection head, pendekatan anchor-free, serta dukungan untuk berbagai tugas visi komputer seperti deteksi, segmentasi, dan pose estimation secara real-time. Pendekatan ini menghasilkan model yang lebih stabil, efisien, dan fleksibel untuk berbagai skenario penggunaan, termasuk sistem deteksi bahasa isyarat berbasis citra dan video [19].



Gambar 2.1 Arsitektur YOLOv8

Gambar 2.1 memperlihatkan struktur umum arsitektur YOLOv8 yang terdiri dari backbone untuk ekstraksi fitur, neck untuk penggabungan informasi multi-skala, serta decoupled detection head yang memisahkan proses klasifikasi dan regresi. Struktur ini memberikan fleksibilitas yang lebih tinggi dan stabilitas pelatihan yang lebih baik dibandingkan generasi YOLO sebelumnya, sehingga menjadi dasar pemilihan YOLOv8 sebagai model deteksi utama dalam penelitian ini.

Tabel 2.2 Karakteristik Varian Model YOLOv8

Varian	Kompleksitas Model	Ukuran Relatif	Kecepatan Inferensi	Akurasi Potensial	Konsumsi Komputasi	Kesesuaian untuk Real-Time
YOLOv8n	Sangat rendah	Sangat kecil	Sangat tinggi	Tinggi	Sangat rendah	Sangat sesuai
YOLOv8s	Rendah	Kecil	Tinggi	Lebih tinggi	Rendah	Sesuai
YOLOv8m	Menengah	Sedang	Sedang	Tinggi	Menengah	Cukup sesuai
YOLOv8l	Tinggi	Besar	Lebih rendah	Sangat tinggi	Tinggi	Kurang sesuai
YOLOv8x	Sangat tinggi	Sangat besar	Rendah	Maksimum	Sangat tinggi	Tidak sesuai

Tabel 2.2 menyajikan perbandingan karakteristik utama dari berbagai varian model YOLOv8, mulai dari YOLOv8n hingga YOLOv8x. Informasi pada tabel tersebut disusun berdasarkan dokumentasi resmi Ultralytics dan laporan teknis YOLOv8, yang menjelaskan perbedaan relatif antar varian model dalam hal kompleksitas arsitektur, ukuran model, kebutuhan komputasi, serta kecenderungan performa inferensi dan akurasi [20]. Variasi ini dirancang untuk memenuhi kebutuhan aplikasi yang berbeda, mulai dari sistem real-time berbasis perangkat terbatas hingga lingkungan komputasi dengan sumber daya besar.

Berdasarkan karakteristik tersebut, varian YOLOv8n memiliki kompleksitas dan ukuran model paling rendah, sehingga menawarkan kecepatan inferensi yang tinggi dan konsumsi sumber daya yang minimal. Sebaliknya, varian YOLOv8l dan YOLOv8x memiliki kapasitas representasi yang lebih besar dan potensi akurasi yang lebih tinggi, namun memerlukan sumber daya komputasi yang jauh lebih besar. Dalam konteks aplikasi real-time, dokumentasi Ultralytics menunjukkan bahwa pemilihan varian model perlu mempertimbangkan trade-off antara akurasi dan efisiensi komputasi agar sistem tetap responsif dan stabil [21]. Oleh karena itu, penelitian ini memfokuskan penggunaan dan optimasi pada varian YOLOv8n sebagai model dasar, karena paling sesuai dengan kebutuhan sistem deteksi alfabet BISINDO yang menuntut kinerja real-time dan efisiensi komputasi.

2.3.5 YOLOv8n untuk Deteksi Real-time Huruf BISINDO

Deteksi huruf BISINDO secara real-time membutuhkan model yang ringan, cepat, dan tetap akurat untuk mendeteksi gerakan tangan yang kompleks dalam rentang waktu singkat. YOLOv8n (nano), varian terkecil dari keluarga YOLOv8,

dirancang khusus untuk efisiensi komputasi tinggi. Dengan ukuran hanya sekitar 3 juta parameter, model ini memiliki ukuran file kurang lebih 3 MB, jauh lebih kecil dibandingkan varian YOLOv8 lainnya atau detektor dua tahap seperti Faster R-CNN [19], [21]. Keunggulan ukuran kecil ini menghasilkan latency rendah dan frame rate tinggi, sangat ideal untuk aplikasi berbasis video.

Studi oleh Dong et al. (2025) menunjukkan bahwa model berbasis YOLOv8n, dengan modifikasi lebih lanjut (GPC-YOLO), dapat mencapai kecepatan hingga 201 FPS pada input video resolusi mobile. Meskipun angka ini dihasilkan dari varian yang dioptimalkan, hasil tersebut tetap menunjukkan bahwa YOLOv8n memiliki margin performa cukup besar untuk menembus ambang minimum real-time 30 FPS bahkan pada perangkat dengan sumber daya terbatas seperti laptop GPU atau sistem embedded [21].

Lebih lanjut, meskipun kecil, YOLOv8n tetap menunjukkan akurasi yang kompetitif dalam mendeteksi objek kecil hingga sedang — fitur penting untuk pengenalan alfabet isyarat tangan. Jia & Li (2024) dalam studi komparatifnya melaporkan bahwa YOLOv8 memberikan akurasi tertinggi di antara YOLOv5, YOLOv6, dan YOLOv7 untuk deteksi huruf bahasa isyarat, sambil mempertahankan kecepatan inferensi tertinggi. Oleh karena itu, YOLOv8n dipandang sebagai titik seimbang antara kecepatan dan presisi, menjadikannya pilihan alami untuk sistem penerjemah bahasa isyarat real-time [19].

Selain faktor teknis, dukungan dokumentasi yang lengkap, API Python yang sederhana, serta komunitas pengguna aktif dari Ultralytics juga menjadi keunggulan YOLOv8n. Proses pelatihan dan fine-tuning dapat dilakukan lebih

cepat, memungkinkan eksperimen hyperparameter yang lebih luas. Kombinasi efisiensi teknis dan dukungan implementasi menjadikan YOLOv8n cocok sebagai core detector untuk sistem deteksi huruf BISINDO yang dikembangkan dalam penelitian ini.

2.3.6 Konsep Hyperparameter Tuning dan Optimalisasi Model

Dalam pengembangan model deep learning seperti YOLOv8n, pemilihan hyperparameter yang tepat menjadi faktor penting yang menentukan kualitas pelatihan dan hasil akhir model. Hyperparameter seperti learning rate, batch size, jumlah epoch, weight decay, hingga arsitektur layer harus ditentukan sebelum proses pelatihan dimulai. Tidak seperti parameter model yang dipelajari langsung dari data, hyperparameter ditetapkan dari awal dan memiliki dampak besar terhadap konvergensi, akurasi, dan kemampuan generalisasi model. Oleh karena itu, proses tuning hyperparameter merupakan bagian penting dari pipeline pelatihan model yang tidak bisa diabaikan [21], [22].

Hyperparameter tuning adalah metode sistematis untuk mencari konfigurasi terbaik yang memaksimalkan performa model, biasanya diukur berdasarkan metrik validasi seperti akurasi atau loss. Strategi tuning yang populer meliputi grid search dan random search. Grid search mencoba semua kombinasi yang mungkin dalam rentang nilai tertentu, tetapi metode ini cepat menjadi tidak efisien karena jumlah kombinasi meningkat secara eksponensial seiring bertambahnya jumlah hyperparameter. Sebaliknya, random search memilih kombinasi secara acak dan sering kali lebih efisien karena memiliki peluang lebih besar menjelajahi area penting dalam ruang pencarian [22].

Untuk mengatasi keterbatasan metode pencarian konvensional, pendekatan lanjutan seperti Bayesian Optimization (BO) diperkenalkan. BO menggunakan model probabilistik sebagai surrogate dari fungsi objektif (misalnya validasi loss terhadap kombinasi hyperparameter), lalu memilih titik selanjutnya berdasarkan estimasi akurasi terbaik. Salah satu varian paling populer adalah Tree-structured Parzen Estimator (TPE) yang terbukti efisien dalam berbagai eksperimen deep learning. Keunggulan BO terletak pada kemampuannya mengombinasikan eksplorasi (mencoba kombinasi baru) dan eksploitasi (memperkuat pencarian di area terbaik), serta mengurangi jumlah percobaan yang dibutuhkan dibandingkan random atau grid search [23].

Dalam konteks penelitian ini, optimasi model difokuskan pada tuning hyperparameter YOLOv8n menggunakan pendekatan Bayesian Optimization. Upaya ini bertujuan meningkatkan akurasi deteksi huruf BISINDO sambil menjaga efisiensi komputasi. Hal ini sejalan dengan studi terbaru yang menunjukkan bahwa hyperparameter tuning mampu memberikan peningkatan performa signifikan dibanding hanya mengubah arsitektur model. Sebagai tambahan, karena pelatihan deep learning bersifat mahal dan memakan waktu, maka digunakan strategi tambahan berupa terminasi otomatis (early stopping) untuk mempersingkat durasi eksperimen dan mencegah overfitting [24].

2.3.7 Optimasi Pelatihan Model Deteksi Objek

Dalam pelatihan model deep learning, pemilihan hyperparameter yang tepat memegang peranan penting dalam menentukan kecepatan konvergensi, stabilitas pelatihan, serta performa akhir model. Pada sistem deteksi objek berbasis

Convolutional Neural Network (CNN), beberapa hyperparameter dinilai sangat berpengaruh karena secara langsung mengendalikan proses optimasi fungsi loss. Sejumlah penelitian menunjukkan bahwa kombinasi hyperparameter tertentu dapat secara signifikan meningkatkan akurasi deteksi sekaligus menjaga efisiensi komputasi dan stabilitas pelatihan model, khususnya pada arsitektur YOLO generasi terbaru, sebagaimana ditunjukkan pada studi optimasi YOLOv8 yang melaporkan peningkatan konsisten pada mAP dan stabilitas pelatihan setelah pengaturan hyperparameter secara sistematis [25].

Optimizer berfungsi sebagai mekanisme pembaruan bobot jaringan selama proses pelatihan. Berbagai optimizer seperti Stochastic Gradient Descent (SGD), Adam, dan AdamW memiliki karakteristik yang berbeda dalam hal kecepatan konvergensi, stabilitas gradien, serta kemampuan generalisasi. SGD dikenal menghasilkan generalisasi yang lebih baik dan pelatihan yang relatif stabil, meskipun memerlukan pengaturan learning rate yang cermat. Sebaliknya, Adam dan AdamW mengadopsi mekanisme adaptif berbasis momen gradien yang memungkinkan konvergensi lebih cepat pada fase awal pelatihan, namun performanya sangat dipengaruhi oleh protokol tuning hyperparameter yang digunakan. Studi empiris berskala besar menunjukkan bahwa perbandingan kinerja antar optimizer sangat sensitif terhadap ruang pencarian hyperparameter, dan bahwa optimizer adaptif tidak pernah berada pada kondisi inferior ketika dilakukan tuning yang adil dan memadai [26].

Learning rate merupakan hyperparameter paling sensitif dalam pelatihan jaringan saraf dalam karena mengatur besar langkah pembaruan bobot pada setiap

iterasi. Nilai learning rate yang terlalu besar dapat menyebabkan proses pelatihan menjadi tidak stabil atau gagal mencapai konvergensi, sedangkan nilai yang terlalu kecil dapat memperlambat proses pembelajaran dan meningkatkan risiko terjebaknya model pada solusi suboptimal. Sensitivitas ini juga dipengaruhi oleh interaksi antara learning rate dan mekanisme optimasi, khususnya pada optimizer adaptif, di mana perubahan kecil pada learning rate dapat berdampak signifikan terhadap dinamika pelatihan dan kualitas solusi akhir [26].

Sementara itu, weight decay berperan sebagai mekanisme regularisasi yang membatasi kompleksitas model dengan memberikan penalti terhadap bobot yang bernilai besar. Penelitian terkini menunjukkan bahwa peran weight decay dalam modern deep learning tidak semata-mata sebagai regularisasi eksplisit, tetapi juga sebagai pengendali dinamika optimasi yang berkontribusi pada stabilitas pelatihan dan konsistensi konvergensi, terutama pada jaringan dalam yang dilatih menggunakan stochastic gradient descent maupun optimizer adaptif [27]. Selain itu, pendekatan decoupled weight decay yang diperkenalkan pada AdamW terbukti mampu memisahkan pengaruh learning rate dan weight decay, sehingga mempermudah proses tuning serta meningkatkan kemampuan generalisasi model dibandingkan penggunaan L2 regularisasi konvensional [28]. Oleh karena itu, kombinasi optimizer, learning rate, dan weight decay secara luas digunakan sebagai search space utama dalam optimasi hyperparameter karena pengaruhnya yang signifikan terhadap keseimbangan antara akurasi, stabilitas pelatihan, dan efisiensi sistem secara keseluruhan.

2.3.8 Optimasi Hyperparameter Berbasis Bayesian

Bayesian Optimization (BO) dikenal sebagai metode yang powerful dalam pencarian hyperparameter optimal pada model deep learning karena efisiensinya dalam mengeksplorasi ruang parameter yang luas. Namun, dalam konteks pelatihan model deep learning yang memerlukan waktu lama seperti YOLOv8n, BO murni sering kali belum cukup efisien. Salah satu strategi yang terbukti efektif adalah menggabungkan BO dengan mekanisme terminasi otomatis atau pruning, yang secara cerdas menghentikan proses pelatihan hyperparameter yang tidak menjanjikan lebih awal. Strategi ini memungkinkan eksplorasi lebih banyak kombinasi hyperparameter dalam waktu pelatihan yang sama, menjadikannya lebih hemat sumber daya [29].

Teknik terminasi otomatis populer seperti Successive Halving Algorithm (SHA) dan Hyperband telah digunakan secara luas untuk adaptive resource allocation. Li et al. (2018) menjelaskan bahwa Hyperband memulai pelatihan dengan banyak konfigurasi hyperparameter namun hanya memberikan sedikit epoch, kemudian hanya konfigurasi terbaik yang dilanjutkan untuk pelatihan tambahan secara bertahap. Hal ini meningkatkan efisiensi karena percobaan yang berkinerjanya buruk segera dihentikan, menghindari pemborosan waktu komputasi [30]. Bahkan pendekatan early stopping agresif seperti SHA dapat menyaingi atau mengungguli BO tradisional dalam praktik tuning hyperparameter model deep learning.

Falkner et al. (2018) memperkenalkan algoritma gabungan bernama BOHB (Bayesian Optimization with Hyperband), yang menggabungkan kemampuan pemodelan probabilistik BO dengan efisiensi Hyperband dalam alokasi resource.

BOHB membangun surrogate model (menggunakan Tree-structured Parzen Estimator, TPE) untuk memprediksi konfigurasi yang menjanjikan, dan menilai performanya dengan strategi pruning berbasis Hyperband. BOHB berhasil menunjukkan efisiensi eksplorasi setara dengan Hyperband dan akurasi pencarian mendekati BO murni, terutama karena menggunakan pendekatan evaluasi awal berbasis epoch-aware optimization[31].

Pendekatan epoch-aware evaluation menjadi kunci dalam pruning. Dengan memantau metrik kinerja model seperti akurasi validasi atau loss setelah sejumlah epoch awal, sistem dapat memutuskan apakah suatu konfigurasi hyperparameter layak diteruskan atau dihentikan. Asumsinya, konfigurasi yang baik akan menunjukkan tren perbaikan sejak awal pelatihan (misalnya penurunan loss yang stabil atau akurasi yang meningkat lebih cepat). Banyak framework tuning modern seperti Optuna mengadopsi strategi ini melalui modul pruner yang memantau nilai intermediate dan memotong trial yang dianggap tidak prospektif [32]. Dengan demikian, pencarian hyperparameter tidak hanya dipandu oleh probabilitas hasil terbaik, tetapi juga disaring secara real-time berdasarkan kinerja awal setiap trial. Konsep-konsep hyperparameter tuning, Bayesian Optimization, dan terminasi otomatis yang dibahas pada subbab ini menjadi fondasi metodologis dalam perancangan kerangka FEAMO-BO, di mana evaluasi performa model tidak hanya dilakukan pada akhir pelatihan, tetapi dimanfaatkan sejak epoch awal untuk memandu proses optimasi secara adaptif.

2.3.9 Metrik Evaluasi Kinerja Deteksi Objek

Evaluasi performa pada sistem deteksi objek memerlukan metrik yang mampu mengukur ketepatan lokasi objek sekaligus akurasi klasifikasinya. Intersection over Union (IoU) digunakan untuk mengukur tingkat kesesuaian antara bounding box hasil prediksi dan ground truth dengan menghitung rasio luas irisan terhadap luas gabungan kedua kotak. IoU menjadi dasar penentuan apakah suatu prediksi dikategorikan sebagai deteksi yang benar dan telah ditetapkan sebagai komponen utama dalam protokol evaluasi deteksi objek modern, khususnya pada dataset Microsoft COCO [33].

Berdasarkan ambang IoU tersebut, performa model selanjutnya dianalisis menggunakan Precision dan Recall. Precision merepresentasikan proporsi prediksi yang benar dari seluruh prediksi positif, sedangkan Recall menunjukkan kemampuan model dalam mendeteksi seluruh objek yang benar-benar ada di dalam citra. Hubungan antara Precision dan Recall mencerminkan perilaku model dalam menyeimbangkan kesalahan false positive dan false negative, sebagaimana dijelaskan secara sistematis dalam panduan evaluasi deteksi objek modern [34], [35].

Untuk memperoleh evaluasi performa yang lebih komprehensif, digunakan metrik Average Precision (AP), yaitu luas area di bawah kurva Precision-Recall pada setiap kelas objek. Nilai AP kemudian dirata-ratakan untuk seluruh kelas sehingga diperoleh Mean Average Precision (mAP). Dalam protokol evaluasi Microsoft COCO, mAP dihitung pada rentang ambang IoU dari 0.50 hingga 0.95 dengan interval 0.05, yang dikenal sebagai mAP50-95 [33]. Pendekatan ini dirancang untuk menilai kualitas deteksi tidak hanya pada kondisi ambang yang

longgar, tetapi juga pada ambang yang lebih ketat sehingga menuntut ketepatan lokasi objek yang tinggi. Penjelasan teknis mengenai mAP dan implementasinya dalam evaluasi deteksi objek modern juga banyak dibahas dalam literatur praktis dan akademik, yang menegaskan bahwa mAP50–95 telah menjadi standar evaluasi utama karena mampu merepresentasikan performa detektor secara lebih objektif dan dapat dibandingkan lintas model maupun konfigurasi pelatihan [35], [36]. Oleh karena itu, penelitian ini menggunakan mAP50–95 sebagai metrik utama untuk mengevaluasi dan membandingkan kinerja model YOLOv8 sebelum dan sesudah proses optimasi hyperparameter.

2.3.10 Deteksi Objek Real-Time

Pada sistem deteksi objek berbasis visi komputer, konsep real-time detection merujuk pada kemampuan model untuk melakukan inferensi secara cepat dan responsif sehingga hasil deteksi dapat digunakan secara langsung tanpa jeda yang mengganggu alur sistem. Dua metrik utama yang digunakan untuk merepresentasikan kemampuan real-time adalah Frames Per Second (FPS) dan latency. FPS menunjukkan jumlah frame yang dapat diproses model dalam satu detik, sedangkan latency merepresentasikan waktu yang dibutuhkan untuk memproses satu frame sejak input diterima hingga output dihasilkan. Berbagai penelitian menunjukkan bahwa sistem deteksi yang ditujukan untuk aplikasi lapangan seperti UAV, edge device, dan sistem monitoring industri umumnya mensyaratkan FPS yang stabil dan latency yang rendah agar mampu merespons dinamika lingkungan secara efektif [37]. Penelitian pada sistem UAV berbasis Edge AI juga menegaskan bahwa keterlambatan inferensi dapat mengurangi keandalan

sistem, terutama pada skenario bergerak dan berisiko tinggi, sehingga optimasi kecepatan inferensi menjadi aspek krusial selain akurasi [38].

Sejumlah studi real-time detection menunjukkan bahwa pencapaian FPS tinggi harus disertai dengan pengendalian latency agar sistem benar-benar layak digunakan secara operasional. Penelitian mengenai deteksi objek berlatensi rendah pada lingkungan hutan menunjukkan bahwa meskipun akurasi model tinggi, latency yang besar dapat menghambat pemanfaatan sistem secara langsung, terutama pada aplikasi monitoring berbasis video streaming [39]. Studi lain pada deteksi cacat berbasis YOLO di perangkat edge menunjukkan bahwa peningkatan kecepatan inferensi secara signifikan dapat dicapai melalui optimasi arsitektur dan penyesuaian pipeline inferensi, sehingga sistem mampu memenuhi kebutuhan real-time tanpa mengorbankan akurasi secara drastis. Berdasarkan temuan-temuan tersebut, penelitian ini mengadopsi FPS dan latency sebagai metrik evaluasi utama untuk menilai kelayakan model YOLOv8 dalam konteks deteksi alfabet BISINDO secara real-time, khususnya pada skenario penggunaan yang menuntut respons cepat dan stabilitas inferensi.

2.3.11 Composite Score untuk Optimasi Multi-Objektif

Evaluasi performa model pada permasalahan optimasi modern, khususnya dalam konteks deep learning dan deteksi objek real-time, tidak lagi dapat bergantung pada satu metrik tunggal. Dalam banyak kasus, beberapa tujuan yang saling berkompetisi perlu dipertimbangkan secara bersamaan, seperti akurasi prediksi, kecepatan inferensi, efisiensi komputasi, dan stabilitas sistem. Pendekatan multi-objective evaluation digunakan untuk menangkap trade-off antar tujuan

tersebut secara komprehensif. Salah satu pendekatan yang umum digunakan adalah menggabungkan beberapa metrik kinerja ke dalam satu nilai agregat yang disebut *composite score*, sehingga proses perbandingan dan pemilihan model dapat dilakukan secara lebih sistematis dan terukur. Dalam kerangka optimasi multi-objektif, konsep ini selaras dengan pendekatan Pareto optimality, di mana solusi terbaik bukan ditentukan oleh satu tujuan dominan, melainkan oleh keseimbangan antar berbagai tujuan yang relevan [40].

Dalam literatur optimasi berbasis Bayesian, evaluasi multi-objektif umumnya dilakukan menggunakan indikator hypervolume atau pendekatan agregasi berbobot untuk merepresentasikan preferensi relatif antar objektif. Daulton et al. menegaskan bahwa penggabungan beberapa objektif ke dalam satu fungsi evaluasi memungkinkan proses optimasi menjadi lebih efisien dalam ruang pencarian berdimensi tinggi, sekaligus tetap menjaga representasi trade-off antar tujuan yang kompleks [40]. Pendekatan serupa juga dikemukakan dalam studi multi-objective Bayesian optimization berdimensi tinggi, di mana agregasi performa digunakan untuk menyederhanakan proses pengambilan keputusan tanpa menghilangkan karakteristik konflik antar objektif utama seperti akurasi dan latency [41]. Pembobotan pada *composite score* dalam penelitian ini tidak dimaksudkan sebagai standar universal, melainkan sebagai skema preferensi yang disesuaikan dengan kebutuhan aplikasi penerjemah alfabet BISINDO secara real-time. Penetapan bobot dilakukan berdasarkan pertimbangan domain, di mana akurasi deteksi tetap menjadi prioritas utama, sementara performa real-time (FPS dan latency) serta stabilitas pelatihan dipertimbangkan sebagai faktor kelayakan operasional sistem. Untuk

memastikan bahwa skema pembobotan tersebut tidak bersifat arbitrer, validasi dilakukan melalui ablation study yang dibahas pada Bab IV, sehingga kontribusi masing-masing komponen terhadap performa akhir model dapat dianalisis secara empiris.

2.3.12 Ablation Study sebagai Metode Evaluasi Komponen Sistem

Ablation study merupakan pendekatan eksperimental yang digunakan untuk menganalisis kontribusi individual dari setiap komponen atau parameter dalam suatu model pembelajaran mesin dengan cara menghilangkan, memodifikasi, atau menonaktifkan komponen tersebut secara sistematis, kemudian mengamati perubahan kinerja yang dihasilkan. Dalam konteks deep learning modern yang memiliki kompleksitas tinggi, ablation study berperan penting untuk membedakan peningkatan performa yang benar-benar berasal dari komponen tertentu dan peningkatan yang bersifat kebetulan atau akibat interaksi tidak langsung antarparameter. Pendekatan ini umum digunakan untuk memvalidasi arsitektur, strategi pelatihan, maupun konfigurasi hyperparameter, karena hasil evaluasi akhir saja sering kali tidak cukup menjelaskan mengapa suatu model bekerja lebih baik dibandingkan baseline. Studi ablation memungkinkan peneliti memperoleh pemahaman kausal mengenai pengaruh masing-masing elemen terhadap akurasi, stabilitas, dan generalisasi model [42].

Dalam penelitian deep learning kontemporer, ablation study juga diposisikan sebagai metode validasi ilmiah terhadap proses pemilihan parameter, terutama ketika model melibatkan optimasi kompleks seperti Bayesian Optimization atau pendekatan multi-objective. Dengan membandingkan performa model hasil

optimasi penuh terhadap beberapa konfigurasi parsial, ablation study dapat menunjukkan bahwa parameter terpilih memang berkontribusi signifikan terhadap peningkatan performa, bukan sekadar hasil eksplorasi ruang pencarian. Penelitian terbaru juga menunjukkan bahwa ablation study semakin penting seiring meningkatnya penggunaan otomatisasi dan kecerdasan buatan dalam proses desain model, karena membantu menjaga transparansi dan reproduisibilitas hasil eksperimen. Pendekatan ini memastikan bahwa keputusan desain model dapat dipertanggungjawabkan secara ilmiah dan tidak bersifat arbitrer, sehingga memperkuat validitas metodologi penelitian yang diusulkan [43].

2.3.13 Kerangka Konseptual FEAMO-BO untuk Optimasi Multi-Objektif

FEAMO-BO (FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization) merupakan kerangka optimasi yang dirancang pada penelitian ini untuk meningkatkan efisiensi pelatihan model YOLOv8n dalam konteks sistem penerjemah alfabet BISINDO secara real-time. Pendekatan ini memadukan kekuatan Bayesian Optimization (BO) dalam eksplorasi hyperparameter dengan evaluasi performa multi-dimensi yang mencakup akurasi deteksi, kecepatan inferensi (FPS), latency per frame, serta stabilitas antar-epoch. Evaluasi berbasis multi-komponen ini penting dalam konteks real-time, karena performa model tidak dapat dinilai dari akurasi semata, melainkan harus mempertimbangkan trade-off antara kualitas prediksi dan efisiensi komputasi pada perangkat sesungguhnya.

Penelitian-penelitian terbaru pada domain UAV, robot humanoid, autonomous vehicle, dan edge computing menegaskan bahwa sistem deteksi objek modern wajib mempertimbangkan kombinasi akurasi, FPS, latency, dan efisiensi

komputasi sebagai metrik utama kelayakan deployment, bukan hanya akurasi akhir [44], [45], [46]. Pendekatan multi-metrik ini sejalan dengan kerangka multi-objective yang diterapkan pada AutoDistill, yang menggunakan Bayesian Optimization berbasis beberapa tujuan (accuracy-latency) dalam memilih konfigurasi terbaik [47]. Selain itu, fokus pada real-time juga didorong oleh studi yang menunjukkan pentingnya latency rendah dan throughput tinggi untuk menjamin pengalaman interaksi langsung antara sistem AI dan pengguna di dunia nyata [44], [48].

Dalam penelitian ini, setiap trial Optuna menghasilkan empat nilai utama $mAP_{0.5}$, FPS, latency, dan stability yang kemudian digabung melalui skor komposit dengan pembobotan multi-objective sesuai dengan rumus (2.1) :

$$\text{CompositeScore} = 0.45 \times mAP_{0.5} + 0.25 \times FPS_{\text{trial}} + 0.15 \times \text{Latency}_{\text{trial}} + 0.15 \times \text{Stability}_{\text{trial}} \quad (2.1)$$

Bobot 0.45:0.25:0.15:0.15 ditetapkan berdasarkan kebutuhan aplikasi penerjemah bahasa isyarat real-time, di mana akurasi tetap menjadi prioritas utama, namun performa real-time (FPS dan latency) secara bersama-sama diberi bobot 40% karena sangat menentukan kelayakan operasional sistem. Pembobotan tersebut selaras dengan hasil penelitian UAV, robot humanoid, dan autonomous vehicle yang menggunakan kombinasi metrik serupa untuk menentukan model terbaik dan menolak model yang walaupun akurat tetapi tidak stabil atau terlalu lambat untuk tugas real-time.

Kecepatan inferensi dinilai menggunakan FPS yang dihitung dari total waktu pemrosesan satu frame (preprocessing + inference + postprocessing), dengan fungsi normalisasi sesuai dengan rumus (2.2):

$$FPS_{i\%e\%} = \min\left(\frac{FPS}{70}, 1.0\right). \quad (2.2)$$

Nilai 70 FPS dipilih sebagai anchor performa optimal karena berada di atas standar minimum real-time (30–60 FPS) dan sesuai dengan hasil pengujian YOLOv8n pada perangkat Jetson Orin NX yang mencapai 52–65 FPS setelah proses quantization pada pengujian UAV [44]. Sementara itu, latency dinormalisasi menggunakan fungsi penalti linier sesuai dengan persamaan (2.3):

$$Latency_{i\%e\%} = 1.0 - \min\left(7 \frac{L_{i\%e\%} - 1}{1}, 1\right) \cdot 0.8. \quad (2.3)$$

Batas <20 ms dipilih berdasarkan ambang persepsi pengguna dan praktik yang umum diterapkan pada AV dan robotika, di mana latency tinggi dapat berdampak langsung pada keselamatan operasi [46]. Model dengan FPS di bawah ambang minimum dikenakan penalti otomatis melalui persamaan (2.4):

$$Composite_score = Composite_score \times 0.7 \quad (2.4)$$

sebagai mekanisme untuk mencegah dominasi model yang memiliki akurasi tinggi tetapi tidak layak secara respons waktu, sehingga model demikian tidak dianggap unggul.

Untuk menilai konsistensi dan stabilitas proses pelatihan, digunakan Stability Score yang menggabungkan training efficiency dan loss convergence antar-epoch melalui formula (2.5):

$$Stability_{i\%e\%} = 0.5 \times TrainingEfficiency_{i\%e\%} + 0.5 \times Convergence_{i\%e\%} \quad (2.5)$$

$$TrainingEfficiency_{i\%e\%} = 1.0 - \min\left(\frac{t_{i\%e\%}}{300}, 1.0\right). \quad (2.6)$$

Batas 300 detik per epoch ditetapkan sebagai ambang praktis efisiensi eksplorasi seperti formula (2.6), selaras dengan studi AutoDistill (2023) yang

menekankan optimasi waktu eksperimen sebagai faktor krusial [47]. Aspek konvergensi loss dinilai berdasarkan rasio variansi:

$$\text{Convergence}_{\text{loss}} = 1 - \frac{\sigma_{t+1}^+}{\sigma_{t+1}^-} \quad (2.7)$$

Pada formula (2.7) yang mencerminkan kestabilan pembelajaran melalui fluktuasi loss antar-epoch, mengacu pada studi analisis perilaku loss dalam deep learning [49], [50].

Optimasi hyperparameter pada FEAMO-BO dilaksanakan menggunakan Optuna dengan pendekatan Tree-structured Parzen Estimator (TPE) untuk memilih konfigurasi terbaik, disertai automatic pruning menggunakan MedianStoppingRule dan Successive Halving guna menghemat biaya komputasi tanpa mengurangi kualitas evaluasi [51]. Dengan demikian, FEAMO-BO menawarkan pendekatan optimasi yang adaptif, efisien, dan realistis, yang mempertimbangkan seluruh aspek penting pada sistem penerjemah BISINDO real-time: akurasi, kecepatan inferensi, latency, efisiensi komputasi, dan stabilitas pelatihan antar-epoch.

Pendekatan multi-objective yang diimplementasikan pada penelitian ini sejalan dengan tren perkembangan terbaru implementasi model deteksi objek berbasis YOLO pada perangkat terbatas, yang menilai performa bukan berdasarkan satu metrik dominan, tetapi berdasarkan kombinasi metrik operasional yang menentukan kelayakan dunia nyata.

BAB 3

METODE PENELITIAN

3.1 Jenis, Sifat, dan Pendekatan Penelitian

Penelitian ini termasuk dalam jenis penelitian eksperimental kuantitatif, karena bertujuan untuk menguji performa model deteksi objek berdasarkan perlakuan tertentu terhadap variabel bebas (hyperparameter) dan mengamati pengaruhnya terhadap variabel terikat (akurasi, FPS, latency, dan stabilitas pelatihan).

Sifat penelitian ini adalah terapan (applied research), karena hasilnya diharapkan dapat memberikan solusi praktis terhadap masalah komunikasi inklusif, khususnya melalui penerapan sistem deteksi alfabet Bahasa Isyarat Indonesia (BISINDO) berbasis Computer Vision.

Pendekatan yang digunakan adalah pendekatan eksperimen berbasis kecerdasan buatan, di mana penelitian dilakukan dengan menerapkan model deteksi objek YOLOv8 yang dioptimasi menggunakan metode Bayesian Optimization berbasis Optuna dengan skema Epoch-Aware Multi-Objective Optimization (FEAMO-BO). Pendekatan ini menggabungkan dimensi akurasi, efisiensi komputasi, dan stabilitas pelatihan sebagai dasar penentuan konfigurasi model terbaik.

3.2 Metode Pengumpulan Data

Metode pengumpulan data dilakukan dengan memanfaatkan dataset publik alfabet Bahasa Isyarat Indonesia (BISINDO) yang tersedia di platform Kaggle.

Dataset tersebut disediakan dalam bentuk citra JPG/PNG dan telah melalui proses pre-augmentation oleh pembuat dataset, sehingga setiap kelas alfabet memiliki variasi pose tangan, kondisi pencahayaan, sudut pandang, serta latar belakang yang beragam. Keragaman ini memberikan keuntungan dalam meningkatkan kemampuan generalisasi model tanpa perlu melakukan pengambilan data citra tambahan secara manual.

Tahapan pengolahan data mencakup proses penataan ulang dan penyesuaian dataset agar sesuai dengan kebutuhan model deteksi objek. Seluruh citra diverifikasi menggunakan format anotasi YOLO (.txt) yang memuat informasi koordinat bounding box dan label kelas objek. Karena proses augmentasi dilakukan sebelum pembagian data dan bukan pada tingkat eksperimen peneliti, maka pembagian dataset dilakukan berdasarkan identitas file untuk meminimalkan risiko terjadinya data leakage antar subset.

Dataset selanjutnya dibagi menjadi tiga subset utama, yaitu 80% data untuk training, 10% untuk validation, dan 10% untuk testing. Pembagian ini mengikuti praktik umum dalam pelatihan model deep learning agar proses pelatihan, validasi, dan pengujian dapat dilakukan secara proporsional serta menghasilkan evaluasi performa model yang objektif.

3.3 Metode Analisis Data

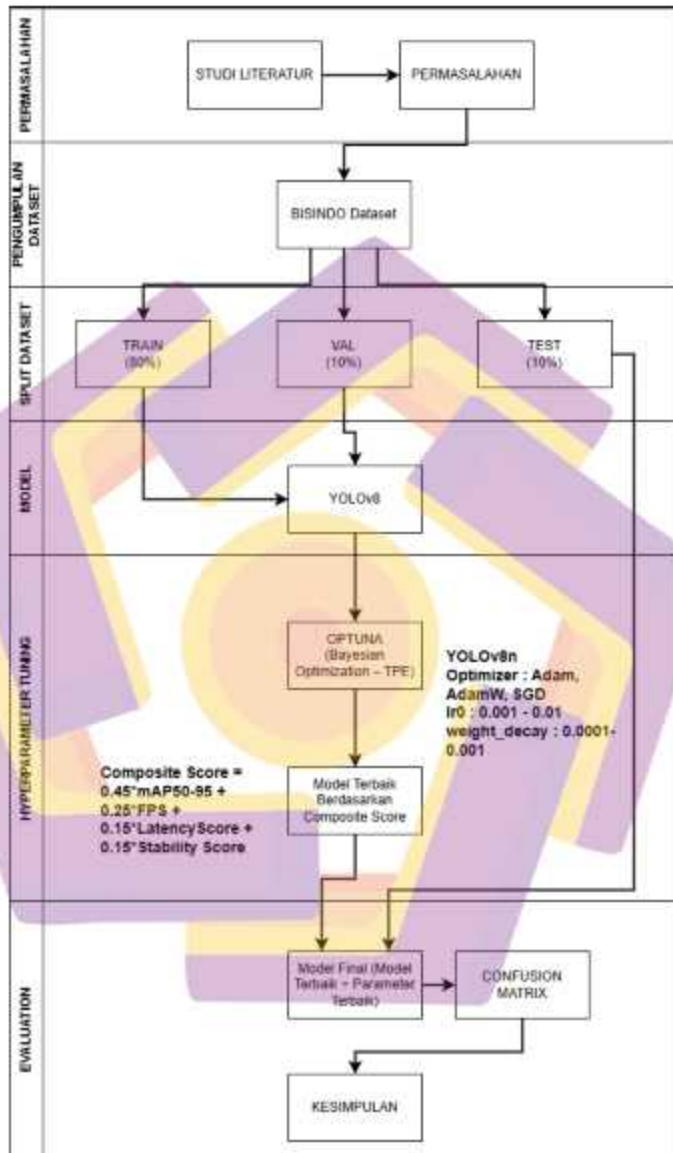
Analisis data dalam penelitian ini dilakukan secara kuantitatif dan eksperimental untuk mengevaluasi performa model deteksi alfabet BISINDO secara objektif. Proses dimulai dengan pelatihan model YOLOv8 menggunakan data training yang telah melalui tahap preprocessing dan augmentasi guna

mempelajari pola visual setiap huruf secara akurat. Selanjutnya dilakukan optimasi hyperparameter dengan Optuna berbasis Bayesian Optimization untuk menemukan kombinasi parameter terbaik seperti learning rate, optimizer, momentum, dan weight decay yang memberikan keseimbangan antara akurasi, efisiensi, dan stabilitas pelatihan.

Setiap trial hasil tuning dievaluasi menggunakan pendekatan FEAMO-BO (FPS-Constrained Epoch-Aware Multi-Objective Bayesian Optimization) dengan mempertimbangkan tiga aspek utama: akurasi (mAP50-95), efisiensi komputasi (FPS dan latency), serta stabilitas antar-epoch. Model dengan composite score tertinggi dipilih sebagai best trial dan kemudian diuji menggunakan testing set untuk memperoleh metrik akhir seperti precision, recall, mAP50-95, dan confusion matrix. Hasil evaluasi ini menjadi dasar penilaian efektivitas metode FEAMO-BO dalam meningkatkan performa deteksi alfabet BISINDO secara menyeluruh

3.4 Alur Penelitian

Alur penelitian ini menggambarkan seluruh tahapan yang dilakukan secara sistematis mulai dari perumusan masalah hingga penarikan kesimpulan. Setiap tahap dirancang agar penelitian dapat menghasilkan model deteksi alfabet Bahasa Isyarat Indonesia (BISINDO) yang optimal, efisien, serta stabil. Tahapan-tahapan tersebut ditunjukkan pada Gambar 3.1 dan dijelaskan secara rinci sebagai berikut.



Gambar 3.1 Alur Penelitian

3.4.1 Perumusan Masalah dan Studi Literatur

Tahap awal penelitian diawali dengan identifikasi permasalahan terkait keterbatasan akses teknologi komunikasi bagi penyandang tunarungu, khususnya kebutuhan akan sistem penerjemah alfabet Bahasa Isyarat Indonesia (BISINDO) yang mampu bekerja secara real-time dengan akurasi tinggi. Berdasarkan kondisi tersebut, penelitian ini mengusulkan pengembangan model deteksi alfabet BISINDO berbasis YOLOv8 yang dioptimasi untuk mencapai kinerja real-time.

Selanjutnya, dilakukan studi literatur untuk menelaah penelitian terdahulu mengenai deteksi bahasa isyarat, perkembangan arsitektur YOLO, serta metode optimasi hyperparameter berbasis Bayesian Optimization. Studi ini menjadi dasar bagi pengembangan pendekatan FEAMO-BO (FPS-Constrained Epoch-Aware Multi-Objective Bayesian Optimization) yang menggabungkan aspek akurasi, efisiensi komputasi, dan stabilitas pelatihan.

3.4.2 Pengumpulan Dataset

Dataset alfabet BISINDO yang digunakan pada penelitian ini berasal dari platform Kaggle dengan total 11.500 citra yang telah melalui proses augmentasi oleh penyedia dataset (pre-augmented dataset). Dataset dikumpulkan dalam format gambar (.jpg/.png) dan anotasi sesuai format YOLO, sehingga siap digunakan untuk proses pelatihan model tanpa memerlukan tahap preprocessing tambahan.

3.4.3 Split Dataset

Dataset dibagi menjadi tiga subset menggunakan pembagian proporsional berdasarkan identitas file untuk menghindari risiko data leakage antar subset, yaitu:

- Training set sebesar 80% digunakan untuk melatih model.

- Validation set sebesar 10% digunakan untuk mengukur performa selama proses pelatihan dan mencegah overfitting.
- Testing set sebesar 10% digunakan untuk evaluasi akhir guna menilai kemampuan model dalam mengenali data baru.

Proporsi ini dipilih agar keseimbangan antara jumlah data pelatihan dan data pengujian tetap terjaga, sehingga hasil evaluasi dapat mencerminkan performa model secara objektif.

3.4.4 Pemodelan dengan YOLOv8

Tahap berikutnya adalah pemodelan (modeling) dengan menggunakan arsitektur YOLOv8 sebagai model deteksi objek utama. YOLOv8 dipilih karena memiliki kemampuan mendeteksi objek secara cepat dengan akurasi tinggi, serta efisiensi komputasi yang baik untuk aplikasi real-time.

Model dilatih menggunakan dataset hasil preprocessing dengan parameter awal (default hyperparameters) dari Ultralytics sebagai titik awal sebelum dilakukan tuning. Proses pelatihan dijalankan pada GPU agar efisien dan mampu memproses data dalam jumlah besar dengan waktu yang optimal.

3.4.5 Hyperparameter Tuning dengan Optuna

Pada tahap hyperparameter tuning, setiap trial yang dihasilkan oleh Optuna dievaluasi menggunakan composite score sebagai dasar pemilihan model terbaik. Composite score dirancang untuk menggabungkan berbagai metrik performa ke dalam satu nilai agregat yang merepresentasikan kualitas model secara multidimensional. Pendekatan ini dipilih karena akurasi tinggi saja tidak cukup untuk menjamin kemampuan model beroperasi dalam konteks real-time; diperlukan

keseimbangan antara akurasi, efisiensi komputasi, dan stabilitas pelatihan. Oleh karena itu, composite score mengintegrasikan beberapa metrik utama, yaitu:

- mAP50-95 untuk mengukur akurasi deteksi secara keseluruhan,
- FPS untuk mengukur kecepatan inferensi dalam mendukung performa real-time,
- Latency score untuk mengukur waktu pemrosesan per frame,
- Training time per epoch sebagai bagian dari evaluasi efisiensi dan stabilitas pelatihan, serta
- Stability score untuk mengukur konsistensi performa model antar-epoch selama proses pelatihan.

Melalui pendekatan ini, proses optimasi hyperparameter tidak hanya diarahkan untuk memperoleh model dengan performa akurasi tertinggi, tetapi juga mempertimbangkan kelayakan operasional model dalam skenario penggunaan nyata, khususnya pada sistem deteksi alfabet BISINDO berbasis real-time.

3.4.6 Evaluasi Model

Evaluasi dilakukan dengan membandingkan model final hasil tuning dengan model baseline, menggunakan test set independen untuk memastikan objektivitas hasil. Metrik evaluasi yang digunakan meliputi mAP50-95, precision, recall, FPS, latency, stability, serta visualisasi performa per kelas menggunakan confusion matrix.

Hasil evaluasi menjadi dasar untuk menarik kesimpulan mengenai efektivitas pendekatan optimasi FEAMO-BO terhadap kinerja YOLOv8 dalam konteks deteksi alfabet BISINDO real-time.

BAB 4 HASIL PENELITIAN DAN PEMBAHASAN




4.1 Pengumpulan dan Persiapan Dataset

Dataset alfabet Bahasa Isyarat Indonesia (BISINDO) yang digunakan dalam penelitian ini diperoleh dari platform publik Kaggle melalui tautan <https://www.kaggle.com/datasets/agungmrf/indonesian-sign-language-bisindo>.

Dataset tersebut terdiri dari 11.500 citra tangan alfabet BISINDO dengan resolusi 640×640 piksel yang telah dianotasi dalam format bounding box YOLO, sehingga secara langsung kompatibel dengan proses pelatihan dan evaluasi model deteksi objek berbasis YOLOv8. Seluruh citra pada dataset ini telah melalui proses augmentasi oleh penyedia dataset (pre-augmented), yang mencakup variasi sudut pengambilan gambar, pencahayaan, latar belakang, rotasi pose tangan, serta kondisi visual lainnya untuk meningkatkan keragaman data.

Karakteristik dataset yang variatif tersebut menjadikan dataset ini relevan untuk pengembangan model deteksi alfabet BISINDO yang robust terhadap variasi visual. Dataset yang digunakan dalam penelitian ini seluruhnya berupa citra statik (still image) alfabet BISINDO, tanpa melibatkan data video, sehingga seluruh proses pelatihan dan pengujian dilakukan pada skenario inferensi berbasis citra. Dengan pengaturan ini, evaluasi performa model dapat dilakukan secara konsisten dan terkontrol, khususnya dalam pengukuran metrik akurasi deteksi, kecepatan inferensi, dan latency yang merepresentasikan karakteristik performa real-time. Contoh sampel dataset yang digunakan ditunjukkan pada Tabel 4.1.

Tabel 4.1 Sampel dataset

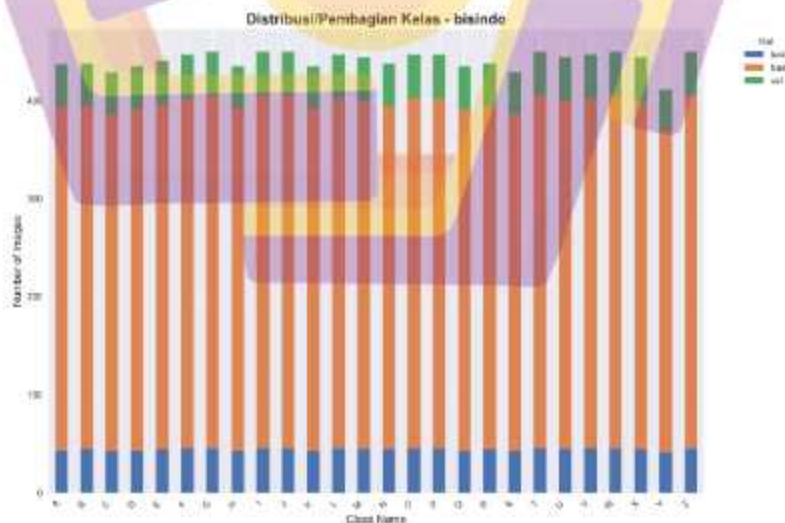
No	Sampel Gambar	Kelas
1		A
2		B
3		C
...
5		Z

4.2 Pembagian Dataset

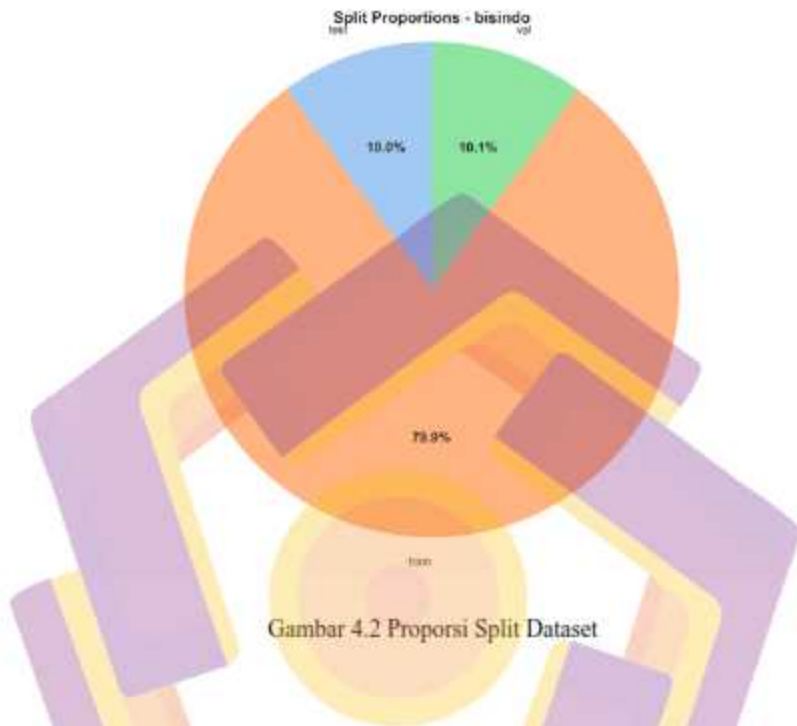
Setelah dataset BISINDO diperoleh, dilakukan proses pembagian dataset (dataset splitting) menjadi tiga subset utama dengan proporsi 80% untuk training, 10% untuk validation, dan 10% untuk testing. Pembagian dilakukan berdasarkan identitas nama file agar tidak terjadi duplikasi citra pada subset berbeda, sehingga menghindari risiko data leakage yang dapat menyebabkan evaluasi menjadi tidak objektif. Training set digunakan dalam proses pelatihan model, validation set berfungsi untuk memonitor performa model selama pelatihan dan mencegah overfitting, sedangkan test set digunakan sebagai data independen untuk evaluasi akhir hasil pelatihan dan tuning hyperparameter.

Pembagian dataset dilakukan secara seimbang pada setiap kelas alfabet A–Z sehingga tidak terdapat ketidakseimbangan jumlah gambar (class imbalance) antar kelas. Distribusi yang merata memastikan bahwa model mempelajari tiap kelas secara proporsional dan memperoleh kemampuan generalisasi yang baik pada seluruh kategori. Selain itu, proporsi 80–10–10 dipilih agar jumlah data pelatihan cukup besar untuk menghasilkan pembelajaran representatif, sementara test set tetap memiliki ukuran yang memadai untuk menghasilkan evaluasi performa model yang dapat dipertanggungjawabkan secara ilmiah.

Visualisasi pembagian dataset ditampilkan pada Gambar 4.1 yang menunjukkan distribusi jumlah citra per kelas pada subset train, validation, dan test, serta Gambar 4.2 yang memperlihatkan proporsi split dataset secara keseluruhan. Berdasarkan visualisasi tersebut, terlihat bahwa pembagian data telah dilakukan secara konsisten dan seimbang pada seluruh kelas alfabet BISINDO.



Gambar 4.1 Distribusi Dataset Per-kelas

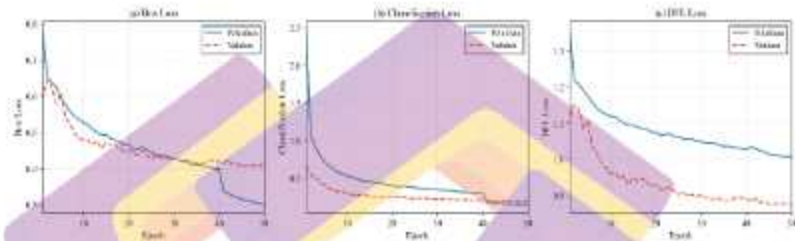


Gambar 4.2 Proporsi Split Dataset

4.3 Analisis Hasil Pelatihan dan Evaluasi Model YOLOv8n Default

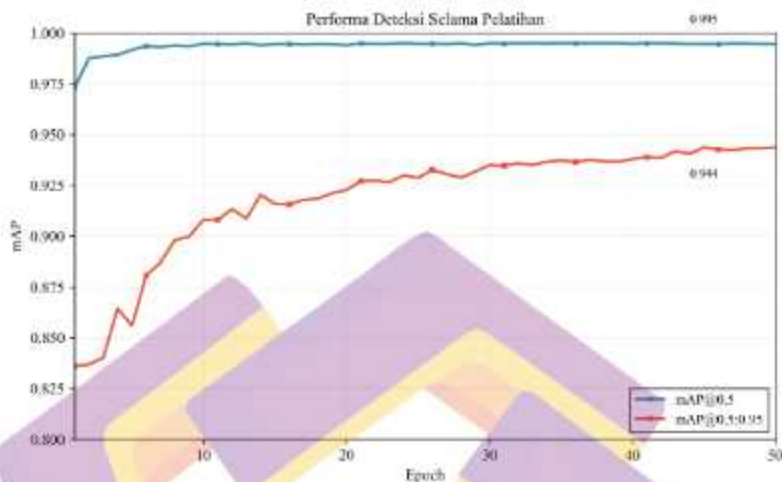
Tahap awal eksperimen dilakukan dengan melatih model YOLOv8n default menggunakan konfigurasi parameter bawaan dari Ultralytics. Proses pelatihan menggunakan 80% data sebagai data latih dan 10% data sebagai data validasi, dengan seluruh eksperimen dijalankan pada lingkungan GPU selama 8.142 detik (\pm 2 jam 15 menit). Proses pelatihan berlangsung hingga model mencapai kondisi konvergen, yang ditunjukkan oleh penurunan nilai loss secara konsisten pada seluruh komponen pelatihan. Kurva box loss, classification loss, dan distribution focal loss (DFL) pada data pelatihan dan validasi memperlihatkan tren menurun

yang stabil serta tidak menunjukkan divergensi signifikan antar-epoch. Pola ini mengindikasikan bahwa proses pembelajaran berjalan secara stabil dan terkendali, tanpa indikasi overfitting maupun ketidakstabilan gradien selama pelatihan, seperti pada Gambar 4.3.



Gambar 4.3 Kurva loss pelatihan dan validasi model YOLOv8n default

Secara kuantitatif, stabilitas proses pelatihan juga tercermin dari nilai stability score sebesar 0.9691, yang menunjukkan bahwa dinamika pembelajaran antar-epoch berlangsung konsisten. Nilai ini menguatkan observasi visual pada kurva loss, di mana fluktuasi nilai loss relatif kecil dan perbedaan antara data pelatihan dan validasi tetap terjaga dalam rentang yang wajar. Stabilitas tersebut menunjukkan bahwa model mampu mempelajari representasi fitur gesture alfabet BISINDO secara bertahap dan berkelanjutan, sehingga menghasilkan proses pelatihan yang konvergen dan reliabel.



Gambar 4.4 Kurva mAP model YOLOv8n default selama pelatihan

Perkembangan performa deteksi selama proses pelatihan ditunjukkan melalui kurva mean Average Precision (mAP), sebagaimana ditampilkan pada Gambar 4.4. Nilai mAP@0.5 meningkat secara cepat sejak tahap awal pelatihan dan mencapai kondisi stabil pada epoch-epoch selanjutnya, yang menandakan bahwa model mampu mempelajari pola visual gesture secara efektif. Sementara itu, mAP@0.5:0.95 menunjukkan peningkatan yang lebih gradual, mencerminkan kemampuan model dalam melakukan lokalisasi bounding box yang semakin presisi seiring bertambahnya epoch. Pola konvergensi pada kedua metrik tersebut mengindikasikan bahwa model tidak hanya mencapai tingkat akurasi yang tinggi, tetapi juga mempertahankan konsistensi performa deteksi sepanjang proses pelatihan.

Tabel 4.2 Hasil Evaluasi Model Default YOLOv8n

Metrik Evaluasi	Nilai
mAP50-95 (test)	0.9419
mAP50	0.994
Precision	0.9903
Recall	0.9951
FPS	81.53 FPS
Total Latency per frame	12.26 ms
Stability Score	0.9691

Evaluasi kuantitatif terhadap data uji disajikan pada Tabel 4.2, yang menunjukkan bahwa model YOLOv8n default mampu mencapai nilai mAP50-95 sebesar 0.9419 dan mAP50 sebesar 0.994, dengan nilai precision sebesar 0.9903 dan recall sebesar 0.9951. Selain itu, model menghasilkan kecepatan inferensi sebesar 81.53 FPS dengan total latency sebesar 12.26 ms per frame, yang menunjukkan bahwa model telah memenuhi kriteria pemrosesan real-time. Kombinasi antara akurasi yang tinggi, stabilitas pelatihan yang baik, serta efisiensi inferensi tersebut menjadikan YOLOv8n default sebagai baseline yang kuat dan representatif untuk tahap eksperimen selanjutnya. Oleh karena itu, penelitian ini melanjutkan proses optimasi menggunakan pendekatan FEAMO-BO guna memperoleh konfigurasi hyperparameter yang lebih optimal dan seimbang antara akurasi deteksi, kecepatan inferensi, dan stabilitas pelatihan.

Secara umum, model menunjukkan performa yang sangat baik sejak tahap awal, ditandai dengan nilai stability score sebesar 0.9691 yang mencerminkan bahwa dinamika pembelajaran berjalan stabil antar-epoch. Selain itu, model mampu mencapai kecepatan inferensi sebesar 81.53 FPS dengan total latency 12.26 ms/frame, yang menunjukkan kelayakan untuk penggunaan real-time meskipun masih memungkinkan peningkatan lebih lanjut melalui optimasi.

Evaluasi performa pada test set dipaparkan pada Tabel 4.2, yang memperlihatkan bahwa YOLOv8n default menghasilkan mAP50-95 sebesar 0.9419, mAP50 sebesar 0.994, precision 0.9903, dan recall 0.9951. Nilai-nilai ini menunjukkan bahwa model mampu mendeteksi dan mengklasifikasikan gesture alfabet BISINDO dengan tingkat akurasi yang sangat tinggi bahkan tanpa tuning tambahan.

Performa tinggi yang ditunjukkan pada Tabel 4.2 dapat dijelaskan oleh beberapa faktor teknis yang saling berkaitan. Pertama, dataset BISINDO yang digunakan pada penelitian ini telah melalui proses augmentasi oleh penyedia dataset, sehingga menghadirkan variasi pose, pencahayaan, orientasi tangan, dan latar belakang yang variatif. Variasi dataset ini memberikan kondisi pembelajaran yang kuat dan memperkaya kemampuan model dalam mengenali pola visual secara konsisten. Akibatnya, proses pelatihan tidak hanya mencapai akurasi tinggi, tetapi juga menghasilkan generalisasi yang baik ketika diuji pada data baru.

Kedua, arsitektur YOLOv8n itu sendiri berkontribusi besar terhadap pencapaian akurasi dan kecepatan tersebut. Sebagai varian ringan dari YOLOv8, model ini menggunakan jumlah parameter yang relatif kecil sehingga proses

pembaruan bobot lebih stabil dan efisien. Desain decoupled head yang memisahkan proses regresi bounding box dari klasifikasi kelas memungkinkan model mempelajari representasi fitur secara lebih terfokus, sehingga meningkatkan kemampuan deteksi detail tangan yang halus seperti bentuk jari atau perbedaan kecil antar gesture. Kecilnya kompleksitas arsitektural model juga menjelaskan nilai stability score yang tinggi, karena model tidak mengalami fluktuasi gradien berlebihan selama proses pelatihan.

Ketiga, pipeline inferensi Ultralytics yang telah dioptimalkan untuk lingkungan GPU modern membuat proses pemrosesan per frame berlangsung cepat dan efisien. Kombinasi preprocessing yang ringan, inference yang teroptimasi, dan postprocessing melalui Non-Maximum Suppression (NMS) yang efisien menghasilkan latency rendah sebesar 12.26 ms, yang menjadikan YOLOv8n default mampu mencapai lebih dari 80 FPS. Kecepatan ini sudah melampaui standar minimum real-time yang umumnya berada pada kisaran 25–30 FPS.

Berdasarkan keseluruhan hasil tersebut, model YOLOv8n default dapat dianggap sebagai titik awal eksperimen yang kuat dan representatif. Namun demikian, karena aplikasi penerjemahan alfabet BISINDO menuntut performa real-time yang lebih cepat, stabil, dan efisien, penelitian ini melanjutkan proses optimasi dengan pendekatan FEAMO-BO untuk memperoleh konfigurasi hyperparameter yang lebih optimal dan seimbang antara akurasi, kecepatan inferensi, serta stabilitas pelatihan.

4.4 Hyperparameter Tuning dengan Optuna-FEAMO-BO

Tahap hyperparameter tuning dilakukan untuk memperoleh konfigurasi model YOLOv8n yang paling optimal dalam mendeteksi alfabet BISINDO secara real-time. Proses optimasi dilakukan menggunakan pendekatan Bayesian Optimization melalui framework Optuna, yang dirancang untuk melakukan eksplorasi ruang hyperparameter secara adaptif dan efisien berdasarkan informasi hasil trial sebelumnya. Metode ini dipilih karena lebih efektif daripada pendekatan grid search atau random search yang cenderung boros sumber daya dan tidak memiliki mekanisme eksplorasi-eksploitasi yang cerdas.

Seluruh model hasil tuning dievaluasi menggunakan Composite Score FEAMO-BO, yaitu metrik agregat multidimensi yang mempertimbangkan akurasi deteksi (mAP50-95), kecepatan inferensi (FPS), latency komputasi per frame, waktu rata-rata pelatihan per epoch, serta stabilitas performa antar-epoch. Pemilihan composite score sebagai metrik seleksi utama merupakan langkah penting untuk memastikan bahwa model yang dipilih tidak hanya unggul pada satu indikator, tetapi benar-benar ideal untuk implementasi operasional real-time.

4.4.1 Konfigurasi dan Desain Eksperimen Optimasi

Proses optimasi dilakukan melalui 30 trial, dengan masing-masing trial menjalankan pelatihan penuh model YOLOv8n menggunakan kombinasi hyperparameter yang berbeda. Parameter yang dioptimalkan meliputi optimizer, learning rate awal (lr_0), dan weight decay, yang merupakan komponen paling berpengaruh terhadap dinamika gradien, konvergensi pelatihan, serta risiko overfitting. Ruang pencarian hyperparameter dirangkum dalam Tabel 4.3.

Tabel 4.3 Search Space Hyperparameter

Hyperparameter	Tipe	Range	Distribusi	Justifikasi
Optimizer	Categorical	SGD, Adam, AdamW	Uniform	Tiga algoritma optimasi paling umum pada model pre-trained; membedakan kecepatan dan stabilitas konvergensi.[26], [28]
Learning Rate (lr)	Float	0.001 – 0.01	Log-uniform	Rentang optimal untuk fine-tuning pre-trained models; distribusi log memungkinkan eksplorasi mendetail pada nilai kecil [25].
Weight Decay	Float	0.0001 – 0.001	Log-uniform	Rentang regulasi L2 yang mencegah overfitting tanpa membatasi kemampuan belajar[27], [28].

Pemilihan tiga hyperparameter tersebut didasarkan pada analisis sensitivitas dari penelitian-penelitian terdahulu dan karakteristik transfer learning YOLOv8. Optimizer SGD dikenal stabil namun lambat; Adam memiliki adaptasi learning rate otomatis yang mempercepat konvergensi; sedangkan AdamW menawarkan regulasi yang lebih baik melalui decoupled weight decay sehingga sering menghasilkan generalisasi yang lebih tinggi.

Distribusi log-uniform digunakan untuk learning rate dan weight decay agar proses pencarian lebih sensitif pada skala kecil yang sering menjadi titik optimum pada fine-tuning models. Rancangan ini memungkinkan Optuna mengevaluasi kombinasi parameter secara probabilistik, sehingga setiap trial mendekati konfigurasi optimal dengan jumlah iterasi yang efisien.

4.4.2 Mekanisme Bayesian Optimization dalam FEAMO-BO

Pada penelitian ini, proses optimasi hyperparameter dilakukan menggunakan pendekatan Bayesian Optimization yang diimplementasikan melalui framework Optuna dengan algoritma Tree-structured Parzen Estimator (TPE). Pendekatan ini dipilih karena mampu melakukan pencarian ruang hyperparameter secara adaptif

dan efisien, dengan memanfaatkan informasi dari hasil eksperimen sebelumnya untuk mengarahkan proses pencarian menuju konfigurasi yang lebih optimal.

Berbeda dengan metode pencarian konvensional seperti grid search atau random search yang mengevaluasi kombinasi hyperparameter secara independen dan tidak terarah, Bayesian Optimization membangun model probabilistik dari fungsi objektif yang sedang dioptimasi. Model ini digunakan untuk menyeimbangkan dua fase utama dalam proses optimasi, yaitu eksplorasi (pencarian ruang parameter secara luas) dan eksploitasi (pemfokusan pencarian pada wilayah parameter dengan performa tinggi).

Dalam Optuna, mekanisme tersebut direalisasikan melalui algoritma TPE yang memisahkan hasil trial ke dalam dua kelompok, yaitu trial dengan performa baik dan trial dengan performa kurang baik, berdasarkan nilai fungsi objektif. Pada penelitian ini, fungsi objektif yang digunakan bukan hanya berbasis akurasi tunggal, melainkan Composite Score FEAMO-BO yang mengintegrasikan beberapa metrik performa penting, meliputi mAP₉₅ FPS, latency, efisiensi pelatihan, dan stabilitas antar-epoch. Dengan demikian, proses optimasi diarahkan untuk menemukan konfigurasi hyperparameter yang seimbang antara akurasi, efisiensi komputasi, dan stabilitas pelatihan.

Secara umum, mekanisme optimasi dimulai dengan pendefinisian objective function yang menerima satu trial sebagai input. Pada setiap trial, Optuna melakukan sampling hyperparameter dari ruang pencarian yang telah ditentukan, yaitu jenis optimizer, learning rate awal (lr_0), dan weight decay. Kombinasi hyperparameter tersebut kemudian diterapkan pada model YOLOv8n, yang

selanjutnya dilatih menggunakan data training selama jumlah epoch yang telah ditetapkan. Setelah proses pelatihan selesai, model dievaluasi menggunakan data validasi untuk memperoleh metrik-metrik performa yang relevan.

Nilai-nilai metrik hasil evaluasi tersebut kemudian diproses menggunakan skema FEAMO-BO Composite Score, sehingga setiap trial direpresentasikan oleh satu nilai agregat yang mencerminkan performa multidimensi model. Nilai composite score ini menjadi dasar utama dalam menentukan kualitas trial serta mengarahkan proses pembentukan distribusi probabilitas pada algoritma TPE.

Algoritma TPE bekerja dengan membangun dua distribusi probabilitas, yaitu distribusi $l(x)$ yang merepresentasikan kelompok trial dengan performa baik, dan distribusi $g(x)$ yang merepresentasikan kelompok trial dengan performa kurang baik. Pemisahan kedua kelompok ini dilakukan berdasarkan parameter kuantil γ (gamma), yang pada penelitian ini ditetapkan sebesar 0,15. Nilai gamma tersebut menentukan proporsi trial terbaik yang digunakan sebagai referensi dalam proses eksploitasi. Dengan membandingkan rasio $l(x)/g(x)$, Optuna memilih kombinasi hyperparameter yang memiliki peluang terbesar untuk menghasilkan peningkatan composite score pada trial berikutnya.

Proses optimasi berlangsung secara iteratif hingga jumlah trial yang ditetapkan tercapai. Pada fase awal, Optuna cenderung melakukan eksplorasi untuk memperoleh gambaran umum ruang solusi. Seiring bertambahnya jumlah trial, proses optimasi secara bertahap beralih ke fase eksploitasi, di mana pencarian difokuskan pada kombinasi hyperparameter yang terbukti memberikan performa tinggi berdasarkan composite score FEAMO-BO.

Untuk memperjelas alur kerja mekanisme optimasi yang diterapkan, Algoritma 4.1 menyajikan pseudocode proses Bayesian Optimization berbasis Optuna-TPE yang digunakan dalam penelitian ini, mulai dari inialisasi trial, proses sampling hyperparameter, pelatihan dan evaluasi model YOLOv8n, hingga pemilihan konfigurasi hyperparameter terbaik berdasarkan nilai composite score.

Algoritma 4.1 Mekanisme Optuna Bekerja

```

Mulai
1  Definiskan objective_function(trial):
2  Pilih hyperparameter dari search space:
3  optimizer  $\in$  {SGD, Adam, AdamW}
4  lr0  $\in$  [0.001, 0.01] (log-uniform)
5  weight_decay  $\in$  [0.0001, 0.001] (log-uniform)
6  Bangun model YOLOv8n
7  Terapkan hyperparameter ke konfigurasi pelatihan
8  Latih model pada data latih selama E epoch (misal: 50 epoch)
9  Evaluasi pada data validasi
10 Ekstraksi metrik evaluasi :
11  mAP50-95
12  FPS
13  Latency
14  Stability (berbasis variansi performa pada epoch akhir)
15 Hitung FEAMO_Composite_Score dari kombinasi metrik tersebut
16 Kembalikan FEAMO_Composite_Score
17 Inialisasi:
18 daftar_trial  $\leftarrow$  kosong
19 jumlah_trial  $\leftarrow$  N
20 gamma  $\leftarrow$  0.15 # kuantil pemisah trial baik-buruk (sesuai TPE)
21 trial_awal  $\leftarrow$  beberapa trial awal (eksplorasi)
22 Untuk setiap trial ke-i:
23 Jika i  $\leq$  trial_awal:
24 # fase eksplorasi awal
25 Pilih hyperparameter secara acak dari search space
26 Jika i > trial_awal:
27 # fase Tree-structured Parzen Estimator (TPE)
28 Unutkan daftar_trial berdasarkan FEAMO_Composite_Score
29 Tentukan:
30 n_baik  $\leftarrow$  gamma * jumlah_trial sebelumnya
31 Kelompok_BAIK  $\leftarrow$  n_baik trial terbaik
32 Kelompok_BURUK  $\leftarrow$  sisa trial
33 Bangun distribusi probabilitas:
34 l(x) dari Kelompok_BAIK
35 g(x) dari Kelompok_BURUK
36 Untuk setiap kandidat hyperparameter x:
37 Hitung rasio  $R(x) = l(x) / g(x)$ 
38 Pilih hyperparameter x dengan nilai R(x) terbesar

```

39	<i>Jalankan objective_function(trial)</i>
40	<i>Simpan hasil trial ke daftar_trial</i>
41	<i>Akhir untuk</i>
42	<i>Pilih hyperparameter dengan FEAMO_Composite_Score terbaik</i>
43	<i>Latih ulang model terbaik</i>
44	<i>Evaluasi pada data uji</i>
45	<i>Selesai</i>

Melalui mekanisme ini, proses hyperparameter tuning tidak hanya diarahkan untuk memaksimalkan akurasi deteksi, tetapi juga memastikan bahwa model yang terpilih memiliki kinerja real-time yang tinggi serta pola pelatihan yang stabil. Integrasi Bayesian Optimization dengan skema evaluasi multi-objektif FEAMO-BO menjadi fondasi utama dalam pemilihan model terbaik yang siap digunakan pada skenario penerjemahan alfabet BISINDO berbasis visi komputer secara real-time.

4.4.3 Perhitungan Composite Score dan Formulasi Fungsi Objektif

Dalam penelitian ini, pemilihan konfigurasi model optimal dilakukan menggunakan composite score yang menggabungkan beberapa metrik performa utama dalam satu fungsi evaluasi terintegrasi. Penyusunan composite score didasarkan pada kebutuhan sistem deteksi alfabet BISINDO berbasis real-time, yang tidak hanya menuntut akurasi tinggi, tetapi juga efisiensi inferensi dan stabilitas pelatihan. Komponen penyusun composite score, bobot, serta target masing-masing metrik ditunjukkan pada Tabel 4.4.

Tabel 4.4 Bobot Komponen Composite score FEAMO-BO

Komponen	Bobot	Target	Justifikasi
mAP _{0.5} Score	45 %	Maksimal	Prioritas utama pada akurasi deteksi; mean Average Precision (mAP _{0.5}) dipilih karena mengukur kinerja model pada rentang IoU 0.5–0.95, sesuai standar COCO Detection Challenge (Lin et al., 2014).
FPS Score	25 %	≥ 70 FPS	Menjamin kemampuan real-time inference; 70 FPS dipilih agar model dapat memproses stream video 60 fps dengan margin komputasi yang aman.

Komponen	Bobot	Target	Justifikasi
Latency Score	15 %	< 20 ms	Menilai respons sistem per frame; latency di bawah 20 ms penting agar keterlambatan tidak dirasakan oleh pengguna dalam konteks real-time.
Training Efficiency	10 %	< 300 s / epoch	Menggambarkan efisiensi waktu pelatihan, penting untuk mempercepat siklus experimentation loop dalam pengembangan model.
Convergence Bonus	5 %	Stabil	Memberi penghargaan tambahan untuk model yang menunjukkan stabilitas antar-epoch yang tinggi; bertujuan menghindari overfitting dan pelatihan yang fluktuatif.

Metrik yang digunakan dalam perhitungan composite score meliputi $mAP_{0.5}$, FPS, latency inferensi, efisiensi waktu pelatihan, dan stabilitas konvergensi antar-epoch. Metrik $mAP_{0.5}$ diberikan bobot terbesar sebesar 45% karena merepresentasikan akurasi deteksi secara komprehensif pada rentang Intersection over Union (IoU) 0.5 hingga 0.95 sesuai dengan standar evaluasi COCO Detection Challenge. FPS dan latency masing-masing diberi bobot 25% dan 15% untuk menjamin kemampuan inferensi real-time dan respons sistem yang rendah terhadap keterlambatan. Efisiensi pelatihan dan stabilitas konvergensi diberikan bobot lebih kecil, yaitu 10% dan 5%, sebagai faktor pendukung dalam pemilihan model yang stabil dan efisien selama proses optimasi.

Untuk memungkinkan penggabungan metrik dengan skala yang berbeda, setiap metrik dinormalisasi terlebih dahulu ke dalam rentang [0,1]. Metrik yang bersifat benefit (semakin besar semakin baik), seperti $mAP_{0.5}$ dan FPS, dinormalisasi menggunakan persamaan (4.1):

$$x^2 = \frac{x - x_{/3}}{x_{/0} - x_{/3}} \quad (4.1)$$

Sementara itu, metrik yang bersifat cost (semakin kecil semakin baik), seperti latency dan waktu pelatihan per epoch, dinormalisasi menggunakan Persamaan (4.2):

$$x^z = 1 - \frac{x - x_{(3)}}{x_{(10)} - x_{(3)}} \quad (4.2)$$

Setelah seluruh metrik dinormalisasi, nilai composite score dihitung sebagai kombinasi berbobot dari seluruh komponen menggunakan Persamaan (4.3):

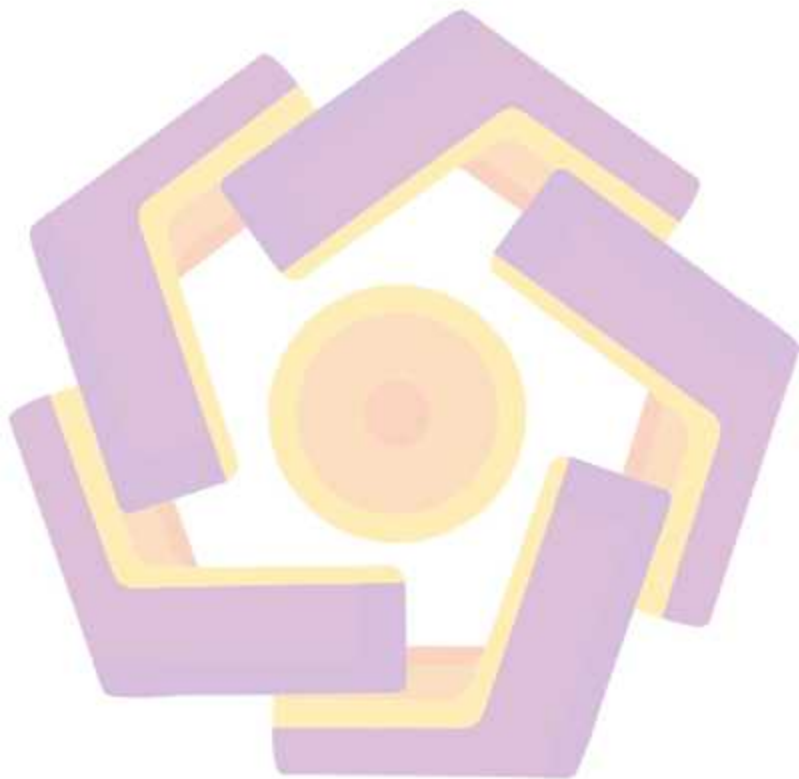
$$\text{CompositeScore} = 0.45 \cdot \text{mAP}_{0.5}^z + 0.25 \cdot \text{FPS}^z + 0.15 \cdot \text{Latency}^z + 0.10 \cdot \text{TrainingEff}^z + 0.05 \cdot \text{Convergence}^z \quad (4.3)$$

dengan tanda (') menyatakan nilai metrik yang telah dinormalisasi. Nilai composite score ini digunakan sebagai fungsi objektif utama dalam proses FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization (FEAMO-BO). Konfigurasi hyperparameter dengan nilai composite score tertinggi dipilih sebagai model optimal karena merepresentasikan keseimbangan terbaik antara akurasi deteksi, efisiensi inferensi, dan stabilitas pelatihan pada skenario evaluasi yang digunakan.

4.4.4 Analisis Hasil Optmiasi dan Pemilihan Model Terbaik

Proses hyperparameter tuning menggunakan pendekatan FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization (FEAMO-BO) menghasilkan sejumlah konfigurasi model (trial) dengan karakteristik performa yang berbeda-beda. Seluruh trial yang dihasilkan dievaluasi menggunakan composite score yang dihitung berdasarkan formulasi yang sama sebagaimana dijelaskan pada Subbab 4.4.3, dengan mengintegrasikan metrik akurasi deteksi ($\text{mAP}_{0.5}$), performa inferensi real-time (FPS dan latency), efisiensi waktu pelatihan, serta stabilitas konvergensi antar-epoch. Dengan pendekatan ini,

pemilihan model tidak dilakukan berdasarkan satu metrik tunggal, melainkan melalui evaluasi keseimbangan performa multidimensi.



Dari total 30 trial yang dievaluasi, algoritma Bayesian Optimization mampu mengarahkan proses pencarian hyperparameter ke wilayah solusi dengan performa tinggi secara konsisten. Nilai composite score dari sepuluh konfigurasi terbaik berada dalam rentang yang relatif sempit, yaitu antara 0.8811 hingga 0.8862, yang menunjukkan bahwa proses optimasi telah mencapai fase eksploitasi yang stabil. Ringkasan sepuluh trial dengan nilai composite score tertinggi ditampilkan pada Tabel 4.5 berikut:

Tabel 4.5 Top 10 Trial Terbaik Hasil Optimasi FEAMO-BO

Rank	Trial	Optimizer	lr_0	Weight Decay	Composite Score	mAP50-95	FPS	Latency (ms)	Stability
1	7	Adam	0.00652	0.0004403	0.8862	0.9350	133.73	7.44	0.854
2	3	AdamW	0.00619	0.0005987	0.8846	0.9394	136.63	7.37	0.915
3	1	AdamW	0.00399	0.0005106	0.8843	0.9378	139.66	7.23	0.907
4	8	Adam	0.00931	0.0003061	0.8840	0.9355	132.83	7.50	0.851
5	4	AdamW	0.00976	0.0008980	0.8825	0.9382	134.68	7.45	0.897
6	5	SGD	0.00983	0.0001511	0.8822	0.9398	133.13	7.58	0.945
7	18	AdamW	0.00459	0.0008358	0.8821	0.9403	127.16	7.86	0.917
8	0	Adam	0.00397	0.0001432	0.8819	0.9411	141.13	7.16	0.904
9	10	Adam	0.00549	0.0007958	0.8818	0.9351	134.61	7.39	0.849
10	9	Adam	0.00182	0.0005468	0.8811	0.9399	132.38	7.56	0.898

Berdasarkan Tabel 4.5, Trial #7 memperoleh nilai composite score tertinggi sebesar 0.8862 dan menempati peringkat pertama di antara seluruh konfigurasi yang diuji. Konfigurasi ini menunjukkan keseimbangan yang baik antara akurasi deteksi ($mAP_{0.5}$ 0.9350), performa inferensi real-time (FPS = 133.73 dan latency = 7.44 ms), serta stabilitas pelatihan yang memadai (stability score = 0.854). Nilai composite score tersebut merupakan nilai maksimum global yang diperoleh dari seluruh trial, sehingga Trial #7 dipilih sebagai kandidat model optimal dalam penelitian ini.

Perlu dicatat bahwa tidak seluruh komponen penyusun composite score ditampilkan sebagai kolom terpisah pada Tabel 4.5. Metrik efisiensi waktu pelatihan telah diintegrasikan secara langsung ke dalam perhitungan composite score sesuai dengan formulasi pada Persamaan (4.3), sedangkan stabilitas konvergensi direpresentasikan melalui kolom Stability. Dengan demikian, Tabel 4.5 menampilkan metrik utama yang paling relevan secara operasional, sementara evaluasi lengkap seluruh komponen tetap tercermin dalam nilai composite score.

Sebagai perbandingan, Trial #0 menunjukkan performa yang sangat menonjol pada aspek kecepatan inferensi dan latency, dengan nilai FPS tertinggi (141.13 FPS) serta latency terendah (7.16 ms), sekaligus mencatat nilai $mAP_{0.5}$ tertinggi sebesar 0.9411. Namun demikian, ketika seluruh metrik dievaluasi secara terintegrasi menggunakan composite score, Trial #0 hanya menempati peringkat kedelapan dengan nilai 0.8819. Hal ini menunjukkan bahwa keunggulan pada satu atau dua metrik tertentu tidak secara otomatis menghasilkan performa multidimensi

terbaik ketika aspek efisiensi pelatihan dan stabilitas konvergensi turut dipertimbangkan.

Analisis lebih lanjut terhadap pola konvergensi pelatihan menunjukkan bahwa Trial #0 memiliki karakteristik pembelajaran yang lebih agresif dan cenderung kurang stabil dibandingkan Trial #7. Sebaliknya, Trial #7 menunjukkan pola konvergensi yang lebih konsisten antar-epoch, sehingga lebih sesuai dengan tujuan penelitian yang berorientasi pada pemilihan satu konfigurasi model yang andal, stabil, dan efisien untuk penggunaan jangka panjang.

Berdasarkan keseluruhan hasil evaluasi tersebut, Trial #7 dipilih sebagai model terbaik (best trial) dalam penelitian ini. Pemilihan ini menegaskan bahwa pendekatan FEAMO-BO tidak hanya mengutamakan pencapaian nilai akurasi atau kecepatan tertinggi secara individual, tetapi menekankan keseimbangan performa multidimensi sebagai dasar pemilihan konfigurasi model deteksi alfabet BISINDO yang layak digunakan pada skenario real-time secara stabil dan berkelanjutan.

4.4.5 Analisis Kausal Hasil Optimasi FEAMO-BO

Hasil optimasi hyperparameter menggunakan pendekatan FEAMO-BO menunjukkan bahwa konfigurasi dengan optimizer Adam dan AdamW secara konsisten mendominasi peringkat teratas berdasarkan nilai composite score. Dominasi optimizer Adam dapat dijelaskan secara kausal melalui mekanisme adaptif learning rate per-parameter yang dimilikinya. Pada dataset alfabet BISINDO yang bersifat statis dengan variasi pose tangan dan latar belakang yang relatif terbatas, Adam mampu mempercepat proses konvergensi gradien pada fase awal pelatihan tanpa menyebabkan fluktuasi loss yang signifikan. Hal ini

menghasilkan stabilitas pelatihan yang baik serta efisiensi komputasi yang lebih tinggi dibandingkan optimizer SGD, yang meskipun stabil, cenderung membutuhkan iterasi lebih banyak untuk mencapai titik konvergensi optimal.

Selain itu, hasil eksperimen menunjukkan adanya penurunan nilai mAP50-95 yang relatif kecil pada model hasil optimasi dibandingkan model baseline, namun diikuti oleh peningkatan FPS yang signifikan. Fenomena ini mencerminkan adanya trade-off yang wajar antara akurasi dan efisiensi inferensi. Penyesuaian hyperparameter seperti learning rate dan weight decay yang lebih agresif mendorong model menuju konfigurasi bobot yang lebih sederhana dan efisien secara komputasi, sehingga mempercepat proses inferensi per frame. Meskipun demikian, penurunan akurasi yang terjadi bersifat marginal dan tidak berdampak signifikan terhadap kemampuan model dalam mengenali kelas alfabet BISINDO secara keseluruhan, sehingga tetap memenuhi kebutuhan fungsional sistem.

Pemilihan composite score sebagai dasar seleksi model terbaik menjadi relevan karena metrik ini secara eksplisit memformalkan keseimbangan antara akurasi deteksi, kecepatan inferensi, latency, dan stabilitas pelatihan dalam satu nilai agregat. Pendekatan ini mencegah bias pemilihan model yang hanya unggul pada satu metrik tunggal, misalnya mAP yang tinggi tetapi tidak layak untuk penggunaan real-time karena latency yang besar. Dengan demikian, konfigurasi model dengan composite score tertinggi—seperti Trial #7—dapat dipandang sebagai solusi paling rasional dan aplikatif untuk skenario deteksi alfabet BISINDO berbasis real-time, karena mencerminkan kompromi optimal antara kualitas deteksi dan efisiensi sistem secara menyeluruh.

4.4.6 evaluasi akurasi model terbaik (mAP50–95)

Untuk memenuhi standar evaluasi deteksi objek yang umum digunakan pada COCO Detection Challenge, performa akurasi model terbaik (Trial #7) dievaluasi menggunakan metrik mean Average Precision (mAP) pada rentang Intersection over Union (IoU) 0.50 hingga 0.95 dengan interval 0.05. Evaluasi ini dilakukan untuk memastikan bahwa nilai mAP50–95 yang dilaporkan benar-benar merepresentasikan performa model pada berbagai tingkat ketelitian lokalisasi.

Hasil pengujian menunjukkan bahwa nilai mAP pada setiap ambang IoU bervariasi sesuai dengan tingkat ketatnya kriteria tumpang tindih antara prediksi dan ground truth. Nilai mAP tertinggi diperoleh pada IoU 0.85 dengan nilai 0.9804, sedangkan nilai mAP terendah terjadi pada IoU 0.70 sebesar 0.9100. Rata-rata nilai mAP pada rentang IoU 0.50–0.95 menghasilkan nilai mAP50–95 sebesar 0.9371, yang digunakan sebagai metrik utama akurasi deteksi dalam penelitian ini. Ringkasan hasil evaluasi mAP pada setiap ambang IoU disajikan pada Tabel 4.6.

Tabel 4.6 Evaluasi mAP50–95 pada Berbagai Ambang IoU

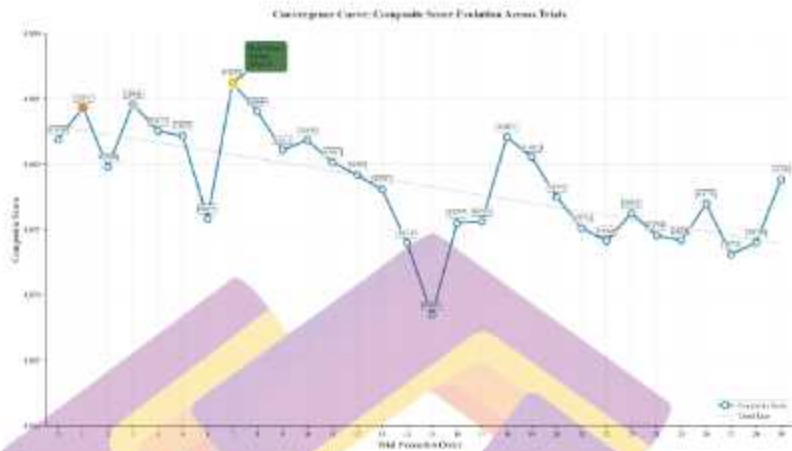
IoU Threshold	Label mAP	Nilai mAP
0.5	mAP50	0.9604
0.55	mAP55	0.9663
0.6	mAP60	0.9472
0.65	mAP65	0.9743
0.7	mAP70	0.91
0.75	mAP75	0.9479
0.8	mAP80	0.9569
0.85	mAP85	0.9804
0.9	mAP90	0.9197
0.95	mAP95	0.9457
Rata-rata (0.50–0.95)	mAP50–95	0.9371

Hasil ini menunjukkan bahwa model hasil optimasi FEAMO-BO tidak hanya memiliki performa yang tinggi pada IoU rendah, tetapi juga mampu

mempertahankan akurasi deteksi yang stabil pada ambang IoU yang lebih ketat. Dengan demikian, nilai mAP50-95 yang diperoleh mencerminkan kemampuan model dalam melakukan deteksi alfabet BISINDO secara akurat dan konsisten, sesuai dengan standar evaluasi deteksi objek modern.

4.4.7 Analisis Konvergensi dan Stabilitas Pelatihan

Analisis kurva konvergensi dilakukan untuk mengevaluasi dinamika pelatihan model selama proses hyperparameter tuning serta menilai sejauh mana stabilitas performa antar-epoch dicapai oleh konfigurasi terbaik. Setiap trial mencatat perkembangan metrik kinerja model pada titik epoch ke-5, 10, 20, 30, 40, dan 50. Pendekatan epoch-aware monitoring ini memberikan gambaran evolusi kinerja selama proses pelatihan, memungkinkan identifikasi indikasi overfitting, underfitting, maupun fluktuasi yang tidak stabil pada kurva pelatihan. Pola konvergensi merupakan aspek penting dalam proses optimasi karena model dengan performa akurasi atau FPS tinggi belum tentu dapat dipertahankan secara stabil sepanjang pelatihan, dan model dengan fluktuasi berlebihan cenderung tidak optimal untuk deployment pada sistem real-time yang bergantung pada konsistensi output.



Gambar 4.5 Kurva Konvergensi Composite Score pada 30 Trial

Kurva konvergensi pada Gambar 4.5 memperlihatkan perkembangan nilai Composite Score sepanjang proses optimasi hyperparameter yang dilakukan selama 30 trial (trial 1 hingga trial 30) menggunakan pendekatan Bayesian Optimization melalui Optuna. Pada fase awal, yaitu trial 1 hingga trial 10, nilai Composite Score menunjukkan fluktuasi yang relatif besar dengan rentang variasi skor yang cukup lebar. Pola ini mencerminkan fase eksplorasi, di mana algoritma secara aktif melakukan pencarian yang luas pada ruang hyperparameter untuk mengidentifikasi konfigurasi awal yang berpotensi menghasilkan performa model yang optimal.

Memasuki trial 11 hingga trial 18, proses optimasi mulai berada pada fase transisi dari eksplorasi menuju eksploitasi. Pada rentang ini, meskipun fluktuasi nilai Composite Score masih terjadi, perbedaan skor antar trial mulai menunjukkan kecenderungan yang lebih sempit dibandingkan fase sebelumnya. Kondisi ini mengindikasikan bahwa algoritma mulai memanfaatkan informasi dari hasil trial terdahulu untuk memfokuskan pencarian pada kombinasi hyperparameter yang

lebih menjanjikan, sembari tetap mempertahankan eksplorasi terbatas guna menghindari terjadinya konvergensi prematur.

Selanjutnya, pada trial 19 hingga trial 22, variasi nilai Composite Score terlihat semakin berkurang secara konsisten. Tidak lagi ditemukan lonjakan maupun penurunan skor yang ekstrem sebagaimana pada fase eksplorasi awal, yang menandakan bahwa ruang pencarian hyperparameter telah semakin menyempit. Fase ini dapat diinterpretasikan sebagai tahap awal menuju konvergensi, di mana algoritma semakin terfokus pada area solusi yang stabil dan memiliki performa yang relatif optimal.

Pada fase akhir, yaitu trial 23 hingga trial 30, kurva menunjukkan pola yang relatif stabil dengan fluktuasi nilai Composite Score yang kecil antar trial. Tidak terdapat peningkatan performa yang signifikan dibandingkan skor terbaik yang telah diperoleh pada trial sebelumnya, meskipun nilai yang dihasilkan tetap berada pada kisaran yang kompetitif. Pola kurva yang semakin merata pada tahap ini mengindikasikan bahwa proses optimasi telah mencapai kondisi konvergensi, di mana mekanisme eksploitasi mendominasi dan peluang untuk menemukan konfigurasi hyperparameter yang secara signifikan lebih unggul menjadi semakin kecil.

Berdasarkan hasil tersebut, dapat disimpulkan bahwa pelaksanaan optimasi hyperparameter dengan 30 trial (trial 1–30) telah memadai untuk memperoleh konfigurasi hyperparameter yang optimal dan stabil. Penambahan jumlah trial diperkirakan tidak akan memberikan peningkatan performa yang signifikan,

sehingga konfigurasi terbaik yang diperoleh pada proses ini dapat digunakan secara representatif untuk tahap evaluasi dan pengujian model selanjutnya.



Gambar 4.6 Perbandingan Composite Score pada Top-10 Trial

Gambar 4.6 memperlihatkan distribusi Composite Score untuk sepuluh trial terbaik. Selisih skor yang sangat tipis antara trial terbaik (0.8862) dan trial ke-10 (0.8811) menggambarkan bahwa performa tiap konfigurasi sangat kompetitif serta berada dalam zona konsistensi tinggi. Kondisi ini memperkuat kesimpulan bahwa pemilihan model terbaik pada akhirnya ditentukan oleh keseimbangan antar metrik, bukan hanya nilai maksimal pada satu indikator performa. Kemiripan skor antar trial membuktikan bahwa proses tuning telah menghasilkan himpunan solusi kandidat yang sama-sama layak, namun perbedaan aspek seperti stabilitas kurva pelatihan, kecepatan inferensi, dan latency menjadi penentu final.

Pola distribusi ini juga menunjukkan karakteristik optimasi multi-objektif: jika evaluasi hanya berbasis mAP50-95, Trial #8 atau Trial #3 mungkin dipilih sebagai model terbaik, tetapi ketika aspek real-time dihitung (FPS dan latency) serta

stabilitas pelatihan diperhatikan, Trial #7 menjadi pilihan paling seimbang dan paling cocok untuk penggunaan dunia nyata. Dengan kata lain, pola konvergensi memperlihatkan bagaimana Composite Score memberikan keputusan lebih komprehensif dibanding single-metric evaluation.

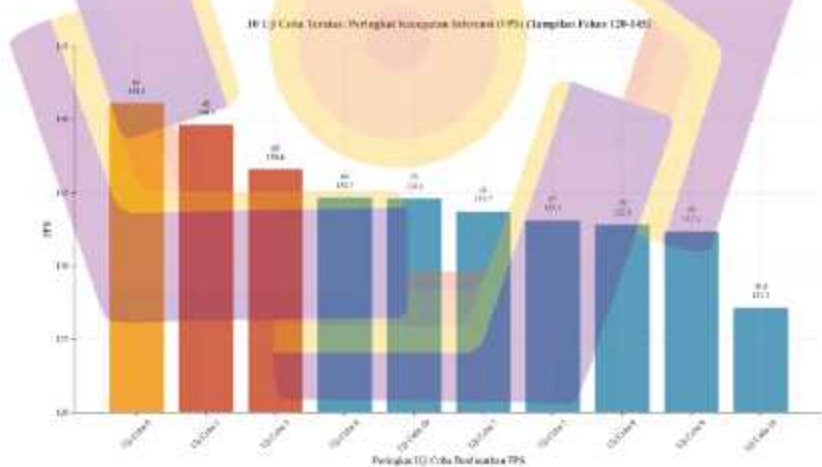
Hasil analisis kurva konvergensi menunjukkan bahwa proses optimasi berjalan stabil dan efektif, serta berhasil menemukan parameter optimal tanpa tanda-tanda ketidakstabilan pelatihan atau divergensi. Dinamika kurva memperlihatkan transisi yang jelas dari eksplorasi menuju eksploitasi, dengan hasil akhir yang sangat konsisten di antara model-model terbaik. Model terbaik dari Trial #7 bukan dipilih karena memiliki nilai maksimal pada satu metrik, tetapi karena menunjukkan keseimbangan performa dan pola konvergensi yang paling stabil. Hal ini memperkuat argumen bahwa pendekatan multi-objektif FEAMO-BO mampu menghasilkan hasil tuning yang robust, realistis, dan siap untuk kebutuhan implementasi real-time deteksi alfabet BISINDO.

4.4.8 Analisis Efisiensi Inferensi

Analisis efisiensi inferensi dilakukan untuk mengevaluasi kemampuan model dalam menghasilkan prediksi secara cepat dan konsisten, yang merupakan aspek krusial pada sistem penerjemah alfabet BISINDO berbasis visi komputer. Efisiensi inferensi diukur menggunakan dua metrik utama, yaitu Frames Per Second (FPS) yang merepresentasikan throughput pemrosesan, serta latency per frame (ms) yang menunjukkan waktu respons dari input hingga keluaran prediksi. Perlu ditegaskan bahwa pengukuran FPS pada penelitian ini dilakukan berdasarkan proses inferensi model terhadap dataset citra statik, bukan pada pemrosesan video real-time atau

input kamera langsung. Dengan demikian, nilai FPS merepresentasikan kemampuan throughput inferensi model dalam kondisi komputasi terkontrol.

Hasil pengujian menunjukkan bahwa konfigurasi model terbaik pada Trial #7 mencapai performa inferensi yang tinggi dengan nilai FPS sebesar 133.73 dan latency sebesar 7.44 ms. Nilai tersebut melampaui batas minimum pemrosesan real-time yang umumnya berada pada kisaran 25–30 FPS, serta memenuhi kebutuhan sistem interaksi gestural yang menargetkan performa di atas 60 FPS untuk memastikan respons yang cepat dan nyaman bagi pengguna. Selain itu, rentang nilai FPS dan latency pada sepuluh trial terbaik relatif sempit, yang mengindikasikan bahwa proses optimasi hyperparameter berhasil menghasilkan konfigurasi model dengan efisiensi inferensi yang stabil dan konsisten.



Gambar 4.7 Distribusi FPS pada Top-10 Trial

Gambar 4.7 memperlihatkan distribusi FPS pada sepuluh trial terbaik. Trial #0 mencatatkan nilai FPS tertinggi sebesar 141.13, yang menunjukkan throughput pemrosesan yang sangat cepat. Namun, konfigurasi tersebut menunjukkan

kompromi pada aspek stabilitas pelatihan dan keseimbangan performa secara keseluruhan. Beberapa trial lain, seperti Trial #3 dan Trial #1, juga menunjukkan FPS yang kompetitif, namun tidak memberikan keseimbangan performa yang optimal ketika seluruh metrik dievaluasi secara terintegrasi. Temuan ini menunjukkan bahwa FPS yang tinggi tidak selalu identik dengan model terbaik secara multidimensi.



Gambar 4.8 Perbandingan Latency Top-10 Trial

Gambar 4.8 menyajikan perbandingan latency inferensi pada sepuluh trial terbaik. Seluruh trial menghasilkan latency dalam rentang 7.16–7.86 ms, yang menegaskan efisiensi komputasi inferensi pada seluruh konfigurasi hasil optimasi. Trial #0 memperoleh latency terendah sebesar 7.16 ms, namun tidak dipilih sebagai model terbaik karena memiliki stabilitas pelatihan dan konsistensi performa yang lebih rendah dibandingkan Trial #7. Trial #7 berada pada titik kompromi yang

optimal dengan latency rendah serta kestabilan performa yang lebih baik secara keseluruhan.

Selain memengaruhi FPS, resolusi input juga berpengaruh terhadap aspek lain seperti akurasi deteksi, latency inferensi, dan stabilitas prediksi model. Resolusi yang lebih tinggi berpotensi meningkatkan detail fitur visual sehingga mendukung ketepatan deteksi, namun di sisi lain meningkatkan beban komputasi dan waktu inferensi. Dalam penelitian ini, resolusi input dikunci pada ukuran 640×640 piksel untuk seluruh eksperimen guna menjaga konsistensi skenario inferensi dan memastikan bahwa perbedaan performa yang diamati berasal dari konfigurasi hyperparameter dan proses optimasi, bukan dari variasi resolusi input. Dengan pengendalian ini, pengaruh resolusi terhadap performa di luar FPS tidak menjadi faktor bias dalam proses pemilihan model.

Secara keseluruhan, analisis efisiensi inferensi menunjukkan bahwa model terbaik pada Trial #7 memberikan keseimbangan yang paling optimal antara kecepatan inferensi dan kestabilan performa. FPS yang tinggi dan latency yang rendah memastikan model mampu memproses gestur tangan secara cepat tanpa keterlambatan yang signifikan, sementara kestabilan hasil inferensi mendukung keandalan sistem dalam penggunaan berkelanjutan. Konsistensi performa pada sepuluh trial terbaik juga mengonfirmasi efektivitas pendekatan FEAMO-BO dalam menghasilkan konfigurasi model yang efisien tanpa kompromi ekstrem terhadap akurasi maupun stabilitas pelatihan.

4.5 Perbandingan Model Default dan Model Hasil Optimasi

Setelah proses hyperparameter tuning menggunakan kerangka FEAMO-BO menghasilkan konfigurasi model terbaik pada Trial #7, tahap berikutnya adalah melakukan perbandingan kinerja antara model default YOLOv8n dan model hasil optimasi pada testing set independen. Evaluasi pada testing set dipilih untuk memastikan performa model mencerminkan kemampuan generalisasi terhadap data baru yang tidak pernah ditemui selama proses pelatihan maupun validasi.

Model default dilatih menggunakan bawaan hyperparameters Ultralytics tanpa optimasi tambahan, sedangkan model hasil optimasi memanfaatkan konfigurasi terbaik yang dihasilkan Optuna dengan parameter optimizer Adam, learning rate 0.00652, dan weight decay 0.0004403. Kedua model diuji dengan konfigurasi inference yang sama serta platform komputasi identik untuk menjamin fairness perbandingan.

Tujuan utama evaluasi ini bukan hanya untuk mengetahui apakah tuning meningkatkan akurasi, tetapi juga untuk mengukur kualitas trade-off antara akurasi deteksi dan kinerja inferensi real-time, sesuai dengan fokus utama penelitian ini. Perbandingan mencakup lima metrik kinerja utama: mAP50-95, precision, recall, FPS, dan latency yang di tunjukkan pada table Tabel 4.7.

Tabel 4.7 Perbandingan Model Baseline dan Model Hasil Optimasi

Model	mAP50-95	Precision	Recall	FPS	Latency (ms)
YOLOv8n Baseline	0.9419	0.9903	0.9951	81.53	12.26
FEAMO-BO (Trial #7)	0.9370	0.9860	0.9970	132.69	7.54

Berdasarkan hasil pada Tabel 4.7, terlihat bahwa model hasil optimasi FEAMO-BO berhasil mencapai peningkatan signifikan pada aspek efisiensi

inferensi, dengan peningkatan FPS sebesar +62.8% (dari 81.53 FPS menjadi 132.69 FPS) dan penurunan latency sebesar 38.5% (dari 12.26 ms menjadi 7.54 ms). Peningkatan tersebut mengindikasikan bahwa model hasil optimasi jauh lebih efisien dan responsif untuk penggunaan real-time, terutama pada aplikasi komunikasi gestural yang membutuhkan pemrosesan seketika tanpa jeda yang mengganggu pengalaman pengguna.

Nilai mAP50-95 model hasil tuning mengalami penurunan tipis sebesar 0.0049 dibanding baseline. Penurunan ini masih berada dalam rentang variasi performa yang lazim terjadi pada proses fine-tuning multi-objektif, khususnya ketika model baseline telah mencapai kondisi mendekati saturasi akurasi. Fakta bahwa model hasil tuning masih sanggup mempertahankan akurasi yang hampir identik sambil mencapai peningkatan besar pada FPS dan latency menunjukkan keberhasilan strategi multi-objective optimization dalam menciptakan model paling efisien untuk deployment operasional.

Kenaikan FPS dan penurunan latency dicapai tanpa mengorbankan stabilitas pelatihan, dibuktikan oleh nilai stability score tinggi pada Trial #7 dalam proses tuning sebelumnya. Hasil ini mengkonfirmasi bahwa pendekatan FEAMO-BO bukan hanya meningkatkan performa runtime, tetapi memastikan keandalan model pada jangka panjang.

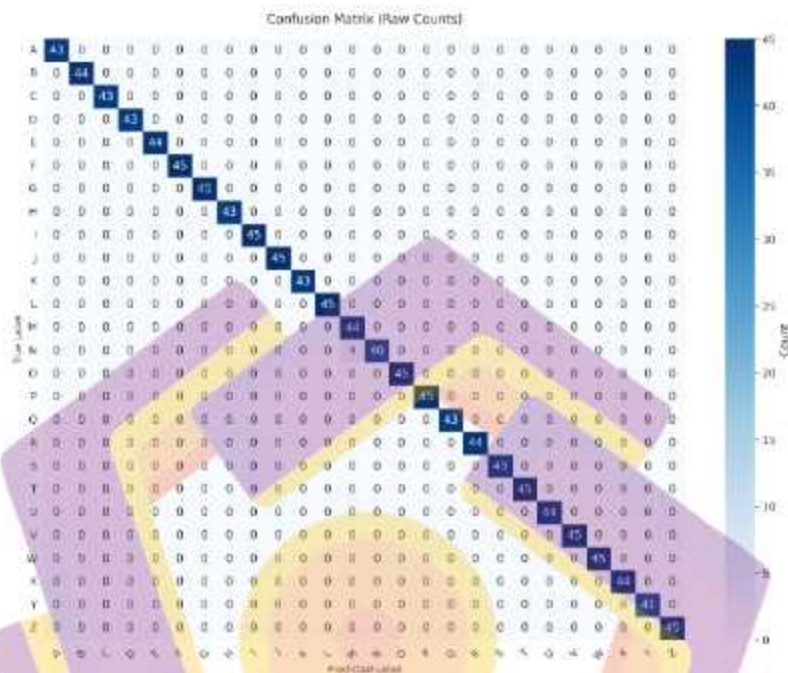
Perbandingan model menunjukkan bahwa FEAMO-BO mampu menghasilkan peningkatan performa komputasi yang sangat signifikan dengan penurunan akurasi yang tidak berarti, sehingga model hasil optimasi dinilai lebih layak digunakan dalam implementasi nyata penerjemah alfabet BISINDO berbasis

visi komputer. Model baseline tetap memiliki nilai akademik sebagai pembanding, namun tidak mampu memenuhi kebutuhan real-time yang menjadi fokus utama penelitian ini. Dengan demikian, penelitian ini berhasil membuktikan bahwa pendekatan multi-objective hyperparameter tuning berbasis Bayesian Optimization dapat menghasilkan model YOLOv8 yang lebih efisien, stabil, dan dapat dioperasikan secara real-time.

4.5.1 Analisis Confusion Matrix

Confusion matrix digunakan untuk mengevaluasi kinerja model dalam mengklasifikasikan setiap kelas alfabet BISINDO secara rinci pada data uji. Pada konteks deteksi objek berbasis YOLOv8, confusion matrix ini tidak digunakan untuk menilai ketepatan lokalisasi bounding box, melainkan untuk mengevaluasi ketepatan penentuan label kelas setelah objek target berhasil terdeteksi. Analisis ini memungkinkan identifikasi pola prediksi benar maupun kesalahan klasifikasi pada tingkat kelas individual.





Jumlah sampel pada data uji untuk setiap kelas alfabet BISINDO bervariasi pada rentang 41 hingga 45 citra per kelas, sebagai konsekuensi dari proses pembagian dataset yang dilakukan secara proporsional sesuai dengan distribusi data awal. Variasi jumlah sampel tersebut tercermin secara langsung pada nilai total prediksi pada setiap baris dan kolom dalam confusion matrix, serta tidak menunjukkan adanya data yang terlewat atau kegagalan proses inferensi yang di tunjukkan pada Gambar 4.9.

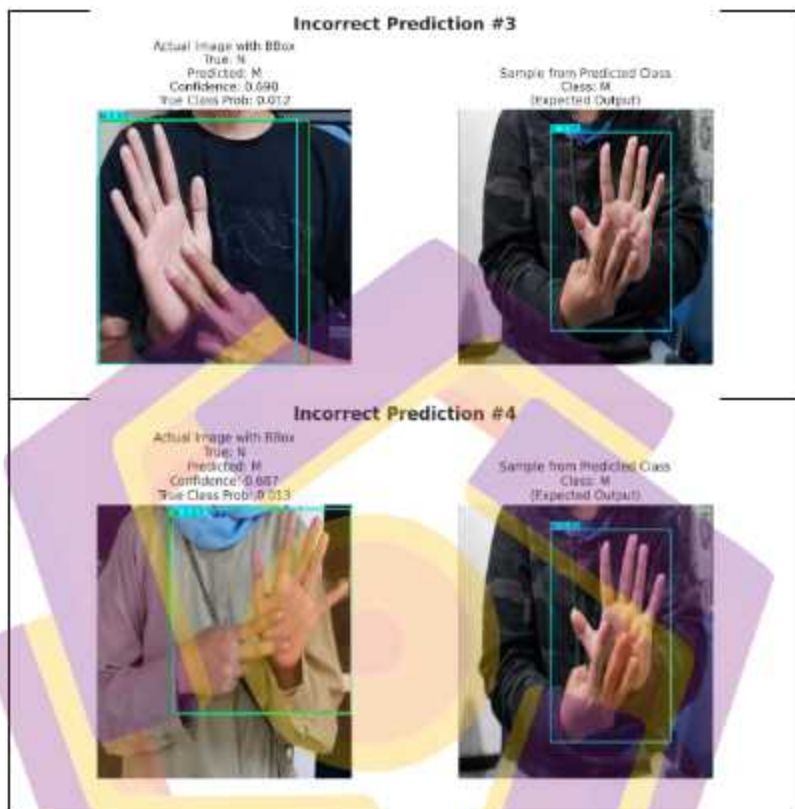


Gambar 4.9 Confusion Matrix pada Test Set

Berdasarkan confusion matrix yang ditunjukkan pada Gambar 4.9, performa klasifikasi model menunjukkan hasil yang sangat baik dengan dominasi nilai pada diagonal utama. Sebagian besar kelas alfabet BISINDO berhasil diklasifikasikan dengan benar pada seluruh sampel uji yang tersedia. Secara kuantitatif, 25 dari 26 kelas alfabet BISINDO mencapai tingkat akurasi klasifikasi sebesar 100%, sementara satu kelas, yaitu huruf N, menunjukkan akurasi sebesar 90,91%, dengan 40 prediksi benar dari total 44 sampel uji. Kesalahan klasifikasi pada kelas tersebut terjadi ketika beberapa sampel huruf N diprediksi sebagai huruf M.

Contoh misklasifikasi yang terjadi pada kelas huruf N ditampilkan pada Gambar 4.10. Pada gambar tersebut terlihat bahwa beberapa sampel huruf N diprediksi sebagai huruf M oleh model.

Sample Misklasifikasi	Output yang diharapkan
Incorrect Prediction #1	
<p data-bbox="362 477 506 547">Actual Image with BBox True: N Predicted: M Confidence: 0.896 True Class Prob: 0.006</p> 	<p data-bbox="706 505 879 547">Sample from Predicted Class Class: M (Expected Output)</p> 
Incorrect Prediction #2	
<p data-bbox="362 866 506 937">Actual Image with BBox True: N Predicted: M Confidence: 0.828 True Class Prob: 0.007</p> 	<p data-bbox="706 895 879 937">Sample from Predicted Class Class: M (Expected Output)</p> 



Gambar 4.10 salah prediksi pada Test Set

Pola kesalahan ini dapat dijelaskan oleh kemiripan konfigurasi gestur tangan antara huruf M dan N dalam alfabet BISINDO, khususnya pada posisi serta lipatan jari tengah dan jari manis. Perbedaan visual yang relatif halus pada citra statik menyebabkan kedua kelas tersebut memiliki tingkat ambiguitas yang lebih tinggi. Fenomena serupa juga banyak dilaporkan dalam penelitian bahasa isyarat lain, seperti American Sign Language (ASL) dan British Sign Language (BSL), di mana kelas dengan konfigurasi jari yang mirip cenderung menghasilkan tingkat kesalahan klasifikasi yang lebih tinggi dibandingkan kelas lainnya.

Meskipun terdapat misklasifikasi pada kelas huruf N, kondisi ini tidak berdampak signifikan terhadap performa keseluruhan model. Jumlah misklasifikasi hanya terjadi pada empat sampel dari keseluruhan data uji, sementara mayoritas kelas menunjukkan akurasi sempurna. Hal ini menunjukkan bahwa model memiliki tingkat konsistensi dan keandalan yang tinggi dalam mengenali alfabet BISINDO pada tingkat kelas individual.

Secara keseluruhan, analisis confusion matrix menegaskan bahwa model hasil optimasi FEAMO-BO mampu mempertahankan performa klasifikasi yang sangat baik meskipun diarahkan untuk mencapai efisiensi inferensi yang tinggi. Dominasi nilai diagonal serta minimnya misklasifikasi menunjukkan bahwa pendekatan optimasi multi-objektif yang diterapkan tidak mengorbankan kemampuan model dalam membedakan kelas alfabet BISINDO, sehingga model dinilai layak dan andal untuk digunakan dalam sistem deteksi alfabet BISINDO berbasis visi komputer.

4.5.2 Perbandingan dengan Penelitian Sebelumnya

Sebagai bagian dari validasi eksternal terhadap performa model yang dikembangkan dalam penelitian ini, dilakukan perbandingan dengan penelitian sebelumnya yang dilakukan oleh Munandar et al. (2024). Penelitian tersebut menggunakan model YOLOv5s untuk mendeteksi alfabet BISINDO dan menjadi salah satu acuan penting dalam menentukan state of the art (SOTA) pada domain ini. Meskipun penelitian tersebut berhasil mencapai tingkat akurasi yang kompetitif, evaluasinya masih terbatas pada metrik akurasi dan belum mencakup

metrik real-time seperti FPS, latency, dan stabilitas pelatihan, yang pada dasarnya merupakan komponen kritis dalam pengembangan sistem deteksi bahasa isyarat.

Perbandingan performa antara penelitian sebelumnya dan hasil penelitian ini ditampilkan pada Tabel 4.8, yang memuat metrik-metrik utama seperti mAP50–95, precision, recall, kecepatan inferensi, serta latency. Tabel tersebut menunjukkan bahwa model YOLOv8n Default yang digunakan dalam penelitian ini memberikan peningkatan akurasi yang signifikan dibandingkan model YOLOv5s. Selain itu, model hasil optimasi dengan pendekatan FEAMO-BO menawarkan peningkatan efisiensi komputasi melalui FPS yang lebih tinggi dan latency yang lebih rendah, sehingga lebih sesuai untuk implementasi sistem deteksi berbasis real-time seperti pada Tabel 4.8.

Tabel 4.8 Perbandingan Hasil Penelitian Ini dengan Penelitian Munandar

Model	mAP50–95	Precision	Recall	FPS	Latency	Stability
YOLOv5s (Munandar, 2024)	0.871	0.852	0.893	–	–	–
YOLOv8n Default	0.9419	0.9903	0.9951	81.53 FPS	12.26 ms	0.9691
YOLOv8n FEAMO-BO	0.937	0.989	0.993	132.69 FPS	7.54 ms	0.854

Hasil pada Tabel 4.8 menunjukkan bahwa YOLOv8n Default memberikan peningkatan akurasi yang cukup besar dibandingkan penelitian Munandar, yaitu peningkatan mAP50–95 sebesar 0,0709 poin atau sekitar 8.1% secara relatif. Precision dan recall juga meningkat secara signifikan, mencerminkan kemampuan model dalam mendeteksi dan mengklasifikasikan gesture alfabet BISINDO dengan tingkat kesalahan yang lebih kecil. Peningkatan ini dapat dijelaskan oleh evolusi arsitektur YOLOv8 yang lebih efisien, terutama pada mekanisme decoupled head dan pemrosesan fitur yang lebih matang, sehingga mampu menangani variasi pose dan kondisi pencahayaan secara lebih baik.

Sementara itu, model hasil optimasi FEAMO-BO mengalami sedikit penurunan mAP50-95 dibandingkan model default, namun penurunan tersebut terjadi sebagai konsekuensi dari proses optimasi multi-objektif yang menekankan keseimbangan antara akurasi, kecepatan inferensi, dan stabilitas pelatihan. Meskipun akurasi sedikit menurun, model FEAMO-BO memberikan peningkatan signifikan pada FPS, yaitu dari 81.53 menjadi 132.69 FPS, serta penurunan latency dari 12.26 ms menjadi 7.54 ms, yang secara substansial meningkatkan kelayakan sistem untuk penggunaan real-time. Dengan demikian, model hasil optimasi memberikan kompromi yang lebih tepat antara akurasi dan efisiensi komputasi dibandingkan model default.

Perbandingan ini sekaligus menunjukkan bahwa penelitian sebelumnya belum dapat divalidasi secara penuh untuk penggunaan real-time karena tidak melaporkan FPS, latency, maupun stabilitas pelatihan. Sebaliknya, penelitian ini menghadirkan kerangka evaluasi yang lebih komprehensif yang tidak hanya menilai akurasi deteksi, tetapi juga mengukur kecepatan dan responsivitas sistem. Dengan demikian, pendekatan FEAMO-BO yang diterapkan dalam penelitian ini menawarkan kontribusi penting terhadap pengembangan model deteksi alfabet BISINDO yang lebih sesuai untuk kebutuhan operasional dunia nyata.

4.6 Ablation Study dan Validasi Pemilihan Parameter

Ablation study pada penelitian ini bertujuan untuk menjawab dua pertanyaan utama:

1. Se jauh mana komponen tambahan pada FEAMO-BO (penalti FPS dan stability) benar-benar mengubah keputusan optimasi dibanding Bayesian Optimization standar (BO only), dan
2. Apakah skema pembobotan Composite Score yang diusulkan cukup robust atau sangat sensitif terhadap perubahan bobot.

Untuk menjawab pertanyaan pertama, dilakukan analisis ablation berbasis post-hoc re-scoring terhadap 30 trial hasil Bayesian Optimization, dengan menghitung ulang composite score untuk setiap trial berdasarkan variasi komponen fungsi objektif: BO only (weighted sum), BO + FPS penalty, BO + Stability, dan BO + FPS penalty + Stability (FEAMO-BO full). Ringkasan hasil seleksi model terbaik untuk masing-masing varian ditunjukkan pada Tabel 4.9.



Tabel 4.9 Hasil Ablation Komponen FEAMO-BO

Eksperimen	Komponen	Best Trial	Composite	mAP50-95	FPS	Latency (ms)	Stability
optuna_bo	BO only (weighted sum)	0	0.8861	0.9411	141.13	5.45	0.904
feamo_no_stability	BO + FPS penalty	0	0.8861	0.9411	141.13	5.45	0.904
feamo_no_fps_penalty	BO + Stability	7	0.8862	0.9350	133.73	5.76	0.854
feamo_full	BO + FPS penalty + Stability (FEAMO-BO)	7	0.8862	0.9350	133.73	5.76	0.854

Dari Tabel 4.9 terlihat bahwa Bayesian Optimization standar (optuna_bo) maupun varian yang hanya menambahkan penalti FPS (feamo_no_stability) secara konsisten memilih Trial #0 sebagai solusi terbaik. Trial ini memang menunjukkan performa yang sangat agresif, dengan nilai mAP50-95 tinggi (0.9411), FPS mencapai 141.13, dan latency hanya 5.45 ms. Namun, pemilihan ini sepenuhnya didorong oleh kombinasi akurasi dan kecepatan tanpa kendali eksplisit terhadap stabilitas pelatihan.

Sebaliknya, ketika komponen stabilitas dimasukkan ke dalam fungsi objektif (feamo_no_fps_penalty dan feamo_full), keputusan seleksi solusi optimal bergeser dari Trial #0 ke Trial #7. Pergeseran ini terjadi meskipun Trial #7 memiliki FPS yang sedikit lebih rendah (133.73) dan nilai mAP50-95 yang turun tipis menjadi 0.9350. Hal ini menunjukkan bahwa kehadiran komponen stabilitas secara eksplisit mengubah keputusan seleksi model terbaik (best trial) pada hasil proses optimisasi.

Perubahan solusi optimal ini mengindikasikan bahwa tanpa mempertimbangkan stabilitas, proses seleksi cenderung memilih model yang sangat cepat dan akurat tetapi belum tentu paling andal untuk penggunaan jangka panjang. Meskipun Trial #0 memiliki nilai stability score yang relatif tinggi secara numerik, integrasi komponen stabilitas dalam composite score yang juga mempertimbangkan pola konvergensi antar-epoch menghasilkan preferensi terhadap Trial #7 yang menunjukkan karakteristik pelatihan lebih konsisten sesuai kriteria stabilitas yang digunakan. Ketika stabilitas dijadikan bagian dari fungsi objektif dan diberi bobot tersendiri, solusi optimasi bergeser ke Trial #7 yang menunjukkan pola konvergensi lebih halus dan variasi performa antar-epoch yang lebih terkendali. Dengan demikian, hasil ablation pada level komponen menunjukkan secara empiris bahwa stabilitas berperan penting dalam keputusan seleksi model dan secara nyata memengaruhi arah seleksi solusi optimal.

Untuk menjawab pertanyaan kedua, dilakukan ablation terhadap skema pembobotan composite score sebagai bentuk analisis sensitivitas. Beberapa skenario pembobotan diuji, meliputi `baseline`, `accuracy_priority`, `realtime_priority`, `stability_priority`, `map_only`, dan `baseline_no_penalty`. Ringkasan hasilnya ditampilkan pada Tabel 4.10.

Tabel 4.10 Ringkasan Ablation Skema Pembobotan Composite Score

Skenario	Best Trial	mAP50-95	FPS	Latency (ms)	Stability	Composite	Spearman vs Baseline
baseline	7	0.9350	133.73	5.76	0.854	0.8862	1.00
accuracy_priority	7	0.9350	133.73	5.76	0.854	0.8941	0.98
realtime_priority	7	0.9350	133.73	5.76	0.854	0.8783	0.99
stability_priority	7	0.9350	133.73	5.76	0.854	0.8679	0.90

map_only	26	0.9427	123.59	6.27	0.950	0.9427	-0.05
baseline_no_penalty	7	0.9350	133.73	5.76	0.854	0.8862	1.00

Berdasarkan Tabel 4.10, terlihat bahwa pada seluruh skenario multi-objective yaitu baseline, accuracy_priority, realtime_priority, stability_priority, dan baseline_no_penalty Trial #7 secara konsisten terpilih sebagai model terbaik. Meskipun nilai composite score berubah mengikuti bobot yang digunakan, koefisien korelasi Spearman terhadap skenario baseline berada pada rentang 0.90–1.00, yang menunjukkan bahwa urutan kualitas model relatif stabil meskipun preferensi objektif digeser.

Sebaliknya, pada skenario ekstrem map_only, di mana seleksi model hanya didasarkan pada mAP50–95, solusi terbaik bergeser ke Trial #26 dengan nilai mAP50–95 sebesar 0.9427. Namun, nilai korelasi Spearman terhadap baseline turun drastis menjadi -0.05, yang mengindikasikan bahwa peringkat model menjadi tidak selaras dengan tujuan sistem ketika aspek performa real-time dan stabilitas diabaikan. Temuan ini menegaskan bahwa seleksi model berbasis satu metrik saja menghasilkan keputusan yang kurang sesuai untuk sistem deteksi real-time.

Secara keseluruhan, ablation study baik pada level komponen FEAMO-BO maupun pada level pembobotan composite score mendukung klaim bahwa pendekatan yang diusulkan bersifat valid dan robust. Konsistensi Trial #7 sebagai konfigurasi optimal pada hampir seluruh skenario multi-objective menunjukkan bahwa desain composite score dan mekanisme seleksi model FEAMO-BO selaras dengan kebutuhan praktis sistem deteksi alfabet BISINDO berbasis real-time inference.

4.7 Evaluasi Keseluruhan dan Pembahasan Umum

Hasil eksperimen secara konsisten menunjukkan bahwa model hasil optimasi FEAMO-BO mampu memberikan peningkatan kinerja yang signifikan pada aspek real-time inference dibanding model baseline YOLOv8n default. Peningkatan FPS lebih dari $\pm 62\%$ dan penurunan latency hingga $\pm 38\%$ dicapai tanpa kehilangan akurasi yang berarti (selisih hanya $\pm 0,005$ poin pada mAP50-95), membuktikan bahwa efisiensi komputasi dapat ditingkatkan tanpa mengorbankan kualitas prediksi. Temuan ini memperkuat hipotesis bahwa pendekatan multi-objective optimization lebih efektif dibanding optimasi akurasi tunggal, serta sejalan dengan tren penelitian modern pada domain UAV, autonomous robotics, dan real-time HCI yang menempatkan kecepatan dan stabilitas sebagai syarat utama implementasi praktis.

Dari perspektif akademik, pendekatan composite score yang diusulkan terbukti robust berdasarkan ablation study dan analisis korelasi Spearman, di mana ranking model tetap stabil pada berbagai skenario pembobotan, dengan nilai korelasi $> 0,90$. Hal ini menunjukkan bahwa konfigurasi model terbaik bukan sekadar hasil kebetulan numerik, melainkan kombinasi parameter yang benar-benar optimal secara multidimensional. Pendekatan ini menjadi kontribusi metodologis yang relevan dalam literatur, karena penelitian-penelitian BISINDO sebelumnya seperti Munandar (2024) maupun penelitian dataset gesture lain umumnya hanya mengevaluasi akurasi tanpa mempertimbangkan FPS, latency, dan stabilitas konvergensi.

Posisi penelitian ini terhadap penelitian terkini (state of the art) menunjukkan kontribusi nyata dalam menjembatani celah penelitian terdahulu: model YOLOv8n FEAMO-BO tidak hanya melampaui performa baseline YOLOv5s penelitian sebelumnya, tetapi juga memperkenalkan framework evaluasi baru yang lebih komprehensif dan sesuai kebutuhan real-time deployment. Dalam konteks novelty, penelitian ini mengusulkan integrasi Multi-Objective Bayesian Optimization dengan evaluasi berbasis FEAMO-BO yang menggabungkan akurasi, kecepatan inferensi, efisiensi pelatihan, dan stabilitas epoch-aware ke dalam satu skema pemilihan model yang belum pernah diterapkan dalam penelitian deteksi alfabet BISINDO sebelumnya.

Klaim kesiapan implementasi pada penelitian ini merujuk pada kelayakan performa komputasi dan stabilitas model yang ditunjukkan melalui evaluasi FPS, latency, serta konsistensi pelatihan, meskipun pengujian langsung pada perangkat edge atau embedded berada di luar ruang lingkup penelitian ini. Dengan demikian, penelitian ini memberikan kontribusi baru yang signifikan, baik dari sisi metodologi optimasi, standar evaluasi performa real-time, maupun implikasi praktis untuk pengembangan sistem komunikasi inklusif berbasis visi komputer bagi penyandang tunarungu di Indonesia.

BAB 5 PENUTUP

5.1 Kesimpulan

Bab ini menyajikan rangkuman akhir dari keseluruhan penelitian yang telah dilakukan. Kesimpulan diperoleh dari hasil analisis dan pembahasan pada bab sebelumnya, serta difokuskan untuk menjawab rumusan masalah dan mencapai tujuan penelitian yang telah ditetapkan. Dengan demikian, kesimpulan ini mencerminkan efektivitas pendekatan yang diusulkan, pencapaian kinerja model, serta kontribusi penelitian terhadap bidang deteksi bahasa isyarat dan optimasi model deep learning.

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan pada Bab III dan Bab IV, diperoleh kesimpulan sebagai berikut

1. Pendekatan FEAMO-BO (FPS-Constrained, Epoch-Aware Multi-Objective Bayesian Optimization) terbukti efektif dalam mengoptimalkan hyperparameter model YOLOv8n untuk deteksi alfabet BISINDO dengan mempertimbangkan aspek akurasi deteksi, efisiensi inferensi, dan stabilitas pelatihan. Pendekatan ini mengintegrasikan evaluasi multiobjektif berbasis composite score yang mampu menyeimbangkan kinerja deteksi dan performa inferensi secara simultan. Model hasil optimasi menunjukkan pola konvergensi antar-epoch yang stabil dengan variansi loss yang rendah, sehingga mencerminkan proses pembelajaran yang konsisten dan mendukung pengembangan sistem deteksi berorientasi real-time.

2. Model YOLOv8n hasil optimasi menggunakan pendekatan FEAMO-BO menunjukkan peningkatan performa yang signifikan dibandingkan model YOLOv8n baseline, terutama pada aspek efisiensi inferensi. Model terbaik (Trial #7) mencapai nilai mAP50-95 sebesar 0,9350, FPS 133,73, dan latency 7,44 ms, yang merepresentasikan peningkatan kecepatan inferensi dan penurunan latency tanpa penurunan akurasi yang substansial dibandingkan baseline. Dalam konteks perbandingan dengan penelitian terdahulu berbasis YOLOv5s, model yang diusulkan menunjukkan performa akurasi deteksi yang lebih tinggi berdasarkan metrik yang dilaporkan, sekaligus menghadirkan evaluasi performa yang lebih komprehensif melalui pengukuran FPS, latency, dan stabilitas konvergensi, yang belum dibahas secara mendalam pada penelitian sebelumnya.
3. Variasi konfigurasi hyperparameter terbukti berpengaruh terhadap nilai composite score serta hubungan trade-off antar-metrik performa model, meliputi akurasi deteksi, FPS, latency, dan stabilitas pelatihan. Hasil analisis ablation dan evaluasi korelasi menunjukkan bahwa mekanisme composite score dalam pendekatan FEAMO-BO bersifat robust terhadap variasi skenario pembobotan objektif, sehingga pemilihan model terbaik tidak bersifat kebetulan numerik, melainkan merepresentasikan solusi yang konsisten secara metodologis. Konfigurasi hyperparameter hasil optimasi mampu menghasilkan keseimbangan performa yang optimal antara akurasi, efisiensi inferensi, dan stabilitas pelatihan, yang menegaskan pentingnya evaluasi multidimensi dalam proses optimasi berbasis Bayesian Optimization.

Secara keseluruhan, penelitian ini telah berhasil menjawab seluruh rumusan masalah dan memenuhi tujuan penelitian. Pendekatan FEAMO-BO menawarkan kontribusi metodologis berupa kerangka optimasi multiobjektif yang dapat digunakan untuk pengembangan model deteksi real-time di berbagai domain seperti sign language recognition, UAV, robotics, dan human-computer interaction. Model hasil penelitian ini tidak hanya akurat dan cepat, tetapi juga layak untuk diimplementasikan pada sistem penerjemah alfabet BISINDO berbasis AI untuk mendukung pendidikan dan komunikasi inklusif di Indonesia.

5.2 Saran

Bagian ini memuat rekomendasi yang disusun berdasarkan hasil penelitian, keterbatasan yang ditemui, serta peluang pengembangan di masa mendatang. Saran berikut diharapkan dapat menjadi acuan bagi peneliti selanjutnya dalam mengembangkan sistem deteksi bahasa isyarat yang lebih komprehensif dan adaptif terhadap kebutuhan nyata.

1. Perluasan Lingkup Dataset dan Variasi Gestur

Penelitian selanjutnya disarankan untuk memperluas cakupan dataset dengan memasukkan gestur dinamis (kata dan kalimat), lebih banyak variasi individu, serta kondisi lingkungan yang lebih beragam (pencahayaan, latar belakang, dan sudut pengambilan gambar). Langkah ini penting untuk meningkatkan kemampuan generalisasi model terhadap situ.

2. Implementasi pada Perangkat Edge AI

Model hasil optimasi FEAMO-BO memiliki ukuran relatif kecil dan kinerja inferensi sangat cepat (133 FPS; 7,44 ms latency), sehingga berpotensi tinggi untuk

diimplementasikan pada perangkat edge computing seperti Jetson Nano, Raspberry Pi, Android/iOS, maupun microcontroller berbasis AI accelerator. Penelitian lanjutan disarankan untuk mengeksplorasi teknik quantization (FP16/INT8), pruning, atau distillation guna mencapai performa optimal pada perangkat dengan sumber daya terbatas.

3. Integrasi dengan Aplikasi Real-Time untuk Pendidikan Inklusif

Model yang dihasilkan dapat diintegrasikan dengan aplikasi berbasis web atau mobile sebagai alat bantu komunikasi dua arah antara pengguna Tuli dan pengajar. Pengembangan antarmuka interaktif akan memperluas manfaat model dalam konteks pendidikan inklusif.

4. Pengembangan Evaluasi Multiobjektif FEAMO-BO yang Lebih Luas

Kerangka FEAMO-BO dapat diperluas dengan menambahkan metrik objektif tambahan seperti konsumsi daya GPU, penggunaan memori saat inference, dan efisiensi energi komputasi sebagai bagian dari evaluasi multiobjektif. Hal ini akan memperkuat relevansi penelitian terhadap kebutuhan implementasi skala besar dan pengembangan sistem hemat daya (green computing).

5. Validasi Lapangan dan Pelibatan Pengguna Akhir

Penelitian selanjutnya disarankan untuk melibatkan lembaga pendidikan Tuli atau komunitas pengguna BISINDO sebagai lokasi uji coba langsung. Pendekatan partisipatif (user-centered evaluation) akan menghasilkan insight berbasis pengguna yang lebih akurat untuk menilai efektivitas sistem, kenyamanan penggunaan, serta rekomendasi perbaikan dari perspektif pengguna akhir.

DAFTAR PUSTAKA

- [1] "Deafness and hearing loss." Accessed: Jan. 15, 2025. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss>
- [2] A. Taupiq, M. Wildan Fajri, and Dannylee, "Identification of Indonesian Sign Language System Using Deep Learning in Yolo-based," *Media J. Gen. Comput. Sci.*, vol. 1, no. 2, pp. 40–47, 2024, doi: 10.62205/mjgcs.v1i2.22.
- [3] A. S. Nugraheni, A. P. Husain, and H. Unayah, "Optimalisasi Penggunaan Bahasa Isyarat Dengan Sibi Dan Bisindo Pada Mahasiswa Difabel Tunarungu Di Prodi Pgmi Uin Sunan Kalijaga," *J. Holistika*, vol. 5, no. 1, p. 28, 2023, doi: 10.24853/holistika.5.1.28-33.
- [4] M. Paramansyah, Arman; Rifhaulipasya Parojai, *Pendidikan Inklusif dalam Era Digital*. 2024.
- [5] B. S. Vidhyasagar, A. S. Lakshmanan, M. K. Abishek, and S. Kalimuthu, "Video Captioning Based on Sign Language Using YOLOV8 Model," *IFIP Adv. Inf. Commun. Technol.*, vol. 683 AICT, pp. 306–315, 2024, doi: 10.1007/978-3-031-45878-1_21.
- [6] S. R. Andra, Munandar, Zara Yunizar, "Indonesian Sign Language (BISINDO) Alphabet Detection Using the You Only Look Once (YOLO) Algorithm Version 8," *Int. Conf. Comput. Control. Informatics its Appl. IC3INA*, vol. 00001, no. 2024, pp. 388–393, 2024, doi: 10.1109/IC3INA64086.2024.10732209.
- [7] N. Renaningtias, F. Putra Utama, A. Nur, and A. Sobri, "Deteksi Bahasa Isyarat Indonesia (BISINDO) Pada Video dengan YOLOv7," *JSAI J. Sci. Appl. Informatics*, vol. 8, no. 1, pp. 1–8, 2025.
- [8] S. Dwijayanti, Hermawati, S. I. Taqiyyah, H. Hikmarika, and B. Y. Suprpto, "Indonesia Sign Language Recognition using Convolutional Neural Network," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 10, pp. 415–422, 2021, doi: 10.14569/IJACSA.2021.0121046.
- [9] N. Ahmad, E. S. Wijaya, C. Tjoaquin, H. Lucky, and I. A. Iswanto, "Transforming Sign Language using CNN Approach based on BISINDO Dataset," in *2023 International Conference on Informatics, Multimedia, Cyber and Informations System (ICIMCIS)*, Jakarta: IEEE, Nov. 2023, pp. 543–548. doi: 10.1109/ICIMCIS60089.2023.10349011.
- [10] M. R. Ningsih *et al.*, "Sign Language Detection System Using YOLOv5 Algorithm to Promote Communication Equality People with Disabilities," *Sci. J. Informatics*, vol. 11, no. 2, pp. 549–558, 2024, doi: 10.15294/sji.v11i2.6007.
- [11] L. N. Hayati, A. N. Handayani, W. S. G. Irianto, R. A. Asmara, D. Indra, and N. S. Damanhuri, "Optimizing YOLO-Based Algorithms for Real-Time BISINDO Alphabet Detection Under Varied Lighting and

- Background Conditions in Computer Vision Systems,” *Int. J. Eng. Sci. Inf. Technol.*, vol. 5, no. 3, pp. 285–294, 2025, doi: 10.52088/ijesty.v5i3.948.
- [12] R. Setiawan, Y. Yunita, F. F. Rahman, and H. Fahmi, “BISINDO (Bahasa Isyarat Indonesia) Sign Language Recognition Using Deep Learning,” *IT Soc.*, vol. 9, no. 1, pp. 16–21, 2024, doi: 10.33021/itfs.v9i1.5076.
- [13] M. N. Eriyadi, R. Ilyas, G. Abdillah, and A. I. Hadiana, “Object Detection of BISINDO Sign Language Letters Using Residual Network,” *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 10, no. 1, pp. 88–95, 2024, doi: 10.23917/khif.v10i1.3670.
- [14] A. Kinanti and D. Maulana, “Convolutional Neural Network Implementation in BISINDO Alphabet Sign Language Recognition System Using Flask,” vol. 11, no. 1, pp. 16–26, 2024.
- [15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.
- [16] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 6517–6525, 2017, doi: 10.1109/CVPR.2017.690.
- [17] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” 2018, [Online]. Available: <http://arxiv.org/abs/1804.02767>
- [18] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” 2020, [Online]. Available: <http://arxiv.org/abs/2004.10934>
- [19] W. Jia and C. Li, “SLR-YOLO: An improved YOLOv8 network for real-time sign language recognition,” *J. Intell. Fuzzy Syst.*, vol. 46, no. 1, pp. 1663–1680, 2024, doi: 10.3223/JIFS-235132.
- [20] Ultralytics, “YOLOv8: Ultralytics Real-Time Object Detection,” 2023, [Online]. Available: <https://docs.ultralytics.com/>
- [21] Y. Dong *et al.*, “GPC-YOLO: An Improved Lightweight YOLOv8n Network for the Detection of Tomato Maturity in Unstructured Natural Environments,” *Sensors*, vol. 25, no. 5, pp. 1–20, 2025, doi: 10.3390/s25051502.
- [22] J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, 2012.
- [23] F. Hutter, *Automated Machine Learning*, vol. 19, 2019, doi: 10.1142/9789814630146_0014.
- [24] P. N. Srinivasu, J. G. Sivasai, M. F. Ijaz, A. K. Bhoi, W. Kim, and J. J. Kang, “Classification of skin disease using deep learning neural networks with mobilenet v2 and lstm,” *Sensors*, vol. 21, no. 8, pp. 1–27, 2021, doi: 10.3390/s21082852.

- [25] L. Ramos, E. Casas, E. Bendek, C. Romero, and F. Rivas-Echeverría, "Hyperparameter optimization of YOLOv8 for smoke and wildfire detection: Implications for agricultural and environmental safety," *Artif. Intell. Agric.*, vol. 12, pp. 109–126, 2024, doi: 10.1016/j.aiaa.2024.05.003.
- [26] D. Choi, C. J. Shallue, Z. Nado, J. Lee, C. J. Maddison, and G. E. Dahl, "On Empirical Comparisons of Optimizers for Deep Learning," no. 1, 2020, [Online]. Available: <http://arxiv.org/abs/1910.05446>
- [27] F. D'Angelo, M. Andriushchenko, A. Varre, and N. Flammarion, "Why Do We Need Weight Decay in Modern Deep Learning?," *Adv. Neural Inf. Process. Syst.*, vol. 37, no. NeurIPS, 2024, doi: 10.52202/079017-0730.
- [28] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *7th International Conference on Learning Representations, ICLR 2019*, New Orleans, LA, USA, 2019, doi: 10.48550/arXiv.1711.05101.
- [29] O. G. Ajayi, P. O. Ibrahim, and O. S. Adegboyege, "Effect of Hyperparameter Tuning on the Performance of YOLOv8 for Multi Crop Classification on UAV Images," *Appl. Sci.*, vol. 14, no. 13, 2024, doi: 10.3390/app14135708.
- [30] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *J. Mach. Learn. Res.*, vol. 18, pp. 1–52, 2018.
- [31] S. Falkner, A. Klein, and F. Hutter, "BOHB: Robust and Efficient Hyperparameter Optimization at Scale," *35th Int. Conf. Mach. Learn. ICML 2018*, vol. 4, pp. 2323–2341, 2018.
- [32] Optuna Contributors, "optuna.pruners — Optuna 4.5.0 documentation," 2025. Accessed: May 20, 2025. [Online]. Available: <https://optuna.readthedocs.io/en/stable/reference/pruners.html>
- [33] T. Y. Lin *et al.*, "Microsoft COCO: Common objects in context," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8693 LNCS, no. PART 5, pp. 740–755, 2014, doi: 10.1007/978-3-319-10602-1_48.
- [34] A. Rosebrock, "Mean Average Precision (mAP) for Object Detection," 2020. Accessed: Dec. 14, 2025. [Online]. Available: <https://learnopencv.com/mean-average-precision-map-object-detection-model-evaluation-metric/>
- [35] Roboflow, "Object Detection Metrics," 2021. Accessed: Dec. 14, 2025. [Online]. Available: <https://blog.roboflow.com/object-detection-metrics/>
- [36] Securade.ai, "Object Detection Metrics: A Comprehensive Guide," 2023. Accessed: Dec. 14, 2025. [Online]. Available: <https://securade.ai/blog/technology/object-detection-metrics-guide.html>
- [37] L. Bermoy and J. Sanchez, "Edge AI Drone : Lightweight MobileNetV3-SSD for Real-Time Detection of Abandoned Weapons in Outdoor

- Terrains,” vol. XIV, no. Xi, pp. 715–730, 2025, doi: 10.51583/IJLTEMAS.
- [38] A. Hussain, B. Barua, A. Osman, R. Abozariba, and A. T. Asyhari, “Low Latency and Non-Intrusive Accurate Object Detection in Forests,” *2021 IEEE Symp. Ser. Comput. Intell. SSCI 2021 - Proc.*, 2021, doi: 10.1109/SSCI50451.2021.9660175.
- [39] X. Shen, H. Wang, B. Wei, and J. Cao, “Real-time scene classification of unmanned aerial vehicles remote sensing image based on Modified GhostNet,” *PLoS One*, vol. 18, no. 6 June, pp. 1–17, 2023, doi: 10.1371/journal.pone.0286873.
- [40] S. Daulton, M. Balandat, and E. Bakshy, “Parallel Bayesian Optimization of Multiple Noisy Objectives with Expected Hypervolume Improvement,” *Adv. Neural Inf. Process. Syst.*, vol. 3, no. 7, pp. 2187–2200, 2021.
- [41] S. Daulton, D. Eriksson, M. Balandat, and E. Bakshy, “Multi-Objective Bayesian Optimization over High-Dimensional Search Spaces,” *Proc. Mach. Learn. Res.*, vol. 180, no. Uai, pp. 507–517, 2022.
- [42] R. Meyses, M. Lu, C. W. de Puiseau, and T. Meisen, “Ablation Studies in Artificial Neural Networks,” pp. 1–19, 2019, [Online]. Available: <http://arxiv.org/abs/1901.08644>
- [43] S. Sheikholeslami, H. Ghasemirahni, A. H. Payberah, T. Wang, J. Dowling, and V. Vlassov, *Utilizing Large Language Models for Ablation Studies in Machine Learning and Deep Learning*, vol. 1, no. 1. Association for Computing Machinery, 2025, doi: 10.1145/3721146.3721957.
- [44] L. Rey *et al.*, “A Performance Analysis of You Only Look Once Models for Deployment on Constrained Computational Edge Devices in Drone Applications,” *Electron.*, vol. 14, no. 3, pp. 1–25, 2025, doi: 10.3390/electronics14030638.
- [45] N. Pottier and M. C. Lau, “A Modular Object Detection System for Humanoid Robots Using YOLO,” no. 1, pp. 1–16, 2025, [Online]. Available: <http://arxiv.org/abs/2510.13625>
- [46] S. Sali *et al.*, “Real-time Object Detection and Associated Hardware Accelerators Targeting Autonomous Vehicles: A Review,” Sep. 2025, [Online]. Available: <http://arxiv.org/abs/2509.04173>
- [47] X. Zhang, Z. Zhou, D. Chen, and Y. E. Wang, “AutoDistill: an End-to-End Framework to Explore and Distill Hardware-Efficient Language Models,” 2022, [Online]. Available: <http://arxiv.org/abs/2201.08539>
- [48] L. Boyle, J. Moosmann, N. Baumann, S. Heo, and M. Magno, “DSORT-MCU: Detecting Small Objects in Real Time on Microcontroller Units,” *IEEE Sens. J.*, vol. 24, no. 24, pp. 40231–40239, 2024, doi: 10.1109/JSEN.2024.3425904.
- [49] C. Wang, J. Sun, W. Xu, and X. Chen, “Depth learning standard deviation loss function,” *J. Phys. Conf. Ser.*, vol. 1176, no. 3, 2019, doi:

10.1088/1742-6596/1176/3/032050.

- [50] Z. Zhang, L. Shi, and D.-X. Zhou, "Optimal Convergence Rates of Deep Neural Network Classifiers," no. 12171039, 2025, [Online]. Available: <http://arxiv.org/abs/2506.14899>
- [51] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, pp. 2623–2631, 2019, doi: 10.1145/3292500.3330701.

