PERBANDINGAN MODEL DEEP LEARNING UNTUK PENERJEMAHAN BAHASA ISYARAT SIBI BERBASIS MOBILE

(*Comparison Of Deep Learning Models For Mobile-Based Sibi Sign Language Translation***)**

I Putu Yoga Indrawan*[1], Christina Purnama Yati[2], Ni Luh Wiwik Sri Rahayu Ginantra[3]
[1,2,3]Informatic, Institut Bisnis dan Teknologi Indonesia
Jl. Tukad Pakerisan no 97a, Denpasar, Bali, INDONESIA
Email: yoga.indrawan@instiki.ac.id, *[*christinapy, wiwik]@ instiki.ac.id

Abstract

Komunikasi antara penyandang tunarungu dan masyarakat umum di Indonesia masih terbatas akibat rendahnya pemahaman terhadap Sistem Bahasa Isyarat Indonesia (SIBI) yang secara resmi diakui oleh pemerintah. Penelitian ini bertujuan untuk mengevaluasi performa tiga model deteksi objek berbasis deep learning MobileNetV2-SSD, MobileNetV2-RetinaNet, dan YOLOv11 dalam mendeteksi bahasa isyarat SIBI, serta merekomendasikan model terbaik untuk implementasi di perangkat Android. Sistem dirancang dengan fokus pada efisiensi agar dapat digunakan secara optimal di perangkat mobile. Hasil evaluasi menunjukkan bahwa MobileNetV2-SSD memberikan performa terbaik dengan mean Average Precision (mAP) sebesar 99,7% dan kecepatan 9 frame per second (FPS). YOLOv11 memperoleh mAP sebesar 89,8% dan 5 FPS, meskipun mengalami fluktuasi pada validation loss. Sementara itu, MobileNetV2-RetinaNet awalnya mencatat mAP sebesar 38,8%, namun meningkat hingga 87,69% pada rasio dataset 70:15:15. Meskipun akurasinya membaik, model ini tetap kurang ideal karena kecepatan inferensi hanya mencapai 2 FPS.Penelitian ini diharapkan dapat menjadi kontribusi awal dalam pengembangan teknologi penerjemah bahasa isyarat yang inklusif dan efisien, guna meningkatkan aksesibilitas komunikasi bagi penyandang tunarungu di Indonesia.
Keywords: *SIBI, Deteksi Objek, Android, Real-time, Computer Vision.*

**Corresponding Author*

1. PENDAHULUAN

Orang tuli menggunakan bahasa isyarat sebagai media komunikasi utama mereka. Di Indonesia, terdapat dua sistem bahasa isyarat, yaitu Sistem Bahasa Isyarat Indonesia (SIBI) dan Bahasa Isyarat Indonesia (BISINDO). Namun, pemerintah secara resmi mengakui SIBI sebagai sistem standar dalam komunikasi [1]. Sayangnya, pemahaman masyarakat umum terhadap SIBI masih terbatas, yang sering kali menyebabkan hambatan komunikasi. Keterbatasan ini berdampak pada aksesibilitas dan inklusivitas sosial bagi komunitas tunarungu. Oleh karena itu, diperlukan solusi yang dapat menjembatani kesenjangan komunikasi ini dengan cara yang efektif dan mudah diakses. [2]

Kemajuan teknologi dalam bidang *computer vision* membuka peluang untuk mengembangkan sistem otomatis yang dapat mengenali dan menerjemahkan bahasa isyarat ke dalam teks atau suara. Sistem ini dapat membantu meningkatkan interaksi antara pengguna bahasa isyarat dengan masyarakat umum, sehingga komunikasi menjadi lebih *inklusif*. Salah satu pendekatan yang banyak digunakan dalam pengembangan sistem ini adalah *deep learning*, yang dikenal memiliki performa tinggi dalam memproses data visual. Namun, tantangan utama dalam penerapan *deep learning* untuk deteksi bahasa isyarat adalah memilih model yang tidak hanya akurat tetapi juga efisien dalam penggunaan sumber daya, terutama untuk implementasi pada perangkat *mobile*.[3]

MobileNetV2-SSD, MobileNetV2-RetinaNet, dan YOLOv11 merupakan model *deep learning* yang populer untuk tugas deteksi objek. MobileNetV2-SSD memiliki keunggulan dalam kecepatan dan efisiensi, menjadikannya pilihan yang tepat untuk perangkat dengan keterbatasan sumber daya seperti Android [4]. Sementara itu, MobileNetV2-RetinaNet mengadopsi Focal Loss untuk meningkatkan akurasi dalam mendeteksi objek yang sulit atau jarang muncul [5]. Model ini dapat mengatasi variasi bentuk tangan atau posisi kompleks dalam bahasa isyarat SIBI. Di sisi lain, YOLOv11 merupakan model terbaru dalam keluarga YOLO yang dikenal memiliki kecepatan deteksi tinggi serta akurasi yang baik dalam berbagai [6].

Pemilihan ketiga model ini dilakukan karena masing-masing memiliki karakteristik arsitektur dan keunggulan teknis yang berbeda, sehingga perbandingan antar model akan memberikan gambaran yang lebih komprehensif mengenai *trade-off* antara akurasi, kecepatan, dan efisiensi. Dengan membandingkan ketiganya, penelitian ini dapat mengidentifikasi model yang paling optimal untuk digunakan pada aplikasi *real-time* di perangkat *mobile*, khususnya dalam konteks deteksi bahasa isyarat SIBI.

Meskipun berbagai penelitian telah dilakukan terkait deteksi bahasa isyarat menggunakan *deep learning*, masih terdapat kesenjangan dalam penelitian yang secara khusus mengevaluasi model *deep learning* untuk bahasa isyarat SIBI. Kebanyakan penelitian sebelumnya lebih berfokus pada BISINDO atau bahasa isyarat internasional, sementara karakteristik bahasa isyarat SIBI memiliki struktur dan aturan yang berbeda. Dalam konteks SIBI, alfabet manual (*fingerspelling*) merupakan komponen *fundamental* yang memiliki struktur tetap, sehingga sangat relevan untuk dijadikan titik awal pengembangan sistem. Selain itu, masih terbatasnya penelitian yang mengkaji performa model *deep learning* pada perangkat *mobile* menjadi tantangan tersendiri.

Namun demikian, pengenalan alfabet saja belum sepenuhnya memadai untuk menghasilkan sistem penerjemahan yang kontekstual dan natural. Bahasa isyarat SIBI juga mencakup *gesture leksikal*, ekspresi non-manual, serta dinamika gerakan yang membentuk makna kompleks. Oleh karena itu, penelitian ini memposisikan pengenalan alfabet sebagai tahap awal yang terukur dan terstandarisasi untuk mengevaluasi performa model *deep learning*, sebelum dikembangkan lebih lanjut menuju deteksi kosakata dan kalimat secara lebih komprehensif. Penelitian ini bertujuan untuk membandingkan performa MobileNetV2-SSD, MobileNetV2-RetinaNet, dan YOLOv11 dalam mendeteksi bahasa isyarat SIBI. Evaluasi akan dilakukan berdasarkan akurasi, efisiensi, serta kesesuaian implementasi pada perangkat Android. Nantinya, sistem ini akan menerima input berupa gambar, kemudian mendeteksi serta menampilkan hasil terjemahan bahasa isyarat SIBI dalam bentuk teks. Diharapkan, penelitian ini dapat berkontribusi dalam meningkatkan aksesibilitas komunikasi bagi komunitas tunarungu di Indonesia melalui solusi berbasis teknologi yang lebih *efisien* dan mudah digunakan.

2. TINJAUAN PUSTAKA

Penelitian sebagai bahan pertimbangan dalam penelitian ini, telah disertakan lima hasil penelitian terdahulu dari berbagai penulis sebagai perbandingan dengan penelitian yang dilakukan, yaitu:

Penelitian pertama berjudul "Deteksi Bahasa Isyarat Sistem Isyarat Bahasa Indonesia Menggunakan Metode Single Shot Multibox Detector" [7]. Tujuan penelitian ini adalah untuk membuat sistem yang memudahkan non-penyandang disabilitas dalam mempelajari Bahasa Isyarat, khususnya Sistem Isyarat Bahasa Indonesia (SIBI), dengan menerapkan metode Single Shot Multibox Detector (SSD) untuk mendeteksi gerakan alfabet SIBI secara *real-time*. Sistem ini menggunakan kamera pada perangkat laptop untuk mendeteksi gerakan dan menampilkan hasilnya pada halaman web. Hasil pengujian menunjukkan performa optimal dengan akurasi sebesar 100%, mAP@.50IoU sebesar 100%, dan AR@100 sebesar 91,79%, menggunakan dataset dengan rasio 90%:10%, learning rate 0,04, epoch 300, batch size 4, dan step 40.000.

Penelitian kedua berjudul "Sistem Deteksi Simbol Pada SIBI (Sistem Isyarat Bahasa Indonesia) Secara *Realtime* Menggunakan Mobilenet-SSD" [8]. Penelitian ini bertujuan untuk mengembangkan sistem deteksi simbol Sistem Isyarat Bahasa Indonesia (SIBI) menggunakan metode Object Detection secara *real-time* dengan *Pre-Trained* MobileNet-SSDv2 untuk membantu menjembatani kesulitan komunikasi yang dihadapi oleh penyandang tunarungu dan tunawicara, serta mempermudah orang non-penyandang dalam memahami bahasa isyarat. Sistem ini dilatih menggunakan dataset sebanyak 600 citra yang terdiri dari 6 simbol SIBI, dengan pembagian 480 citra untuk training dan 120 citra untuk testing. Proses *training* dilakukan dengan tiga konfigurasi *step* berbeda, menghasilkan performa terbaik pada 20.000 step dengan nilai loss sebesar 0,1387 dan akurasi 86,6%.

Penelitian ketiga berjudul "Implementation of Sibi Sign Language Realtime Detection Program (Case Studi At Sekolah Luar Biasa Negeri 1 Tabanan)"[9]. Penelitian ini bertujuan mengimplementasikan algoritma deteksi gerakan bahasa isyarat Sistem Isyarat Bahasa Indonesia (SIBI) secara *real-time* di Sekolah Luar Biasa Negeri 1 Tabanan menggunakan pemrosesan citra dan *deep learning* dengan YoloV8. Tujuannya adalah mempermudah komunikasi antara siswa tunarungu dan orang yang tidak mengerti bahasa isyarat serta meningkatkan inklusivitas dalam pembelajaran. Model YoloV8 yang dilatih dengan dataset berisi 107 kelas kosa kata dan 7 kelas prefiks afiks dapat mengenali gerakan bahasa isyarat secara *real-time* dan menghasilkan teks. Akurasi deteksi optimal tercatat 87,74%, dengan deteksi suboptimal mencapai 58,02%. Meskipun faktor seperti warna pakaian, pencahayaan,

dan kualitas webcam memengaruhi hasil deteksi, sistem ini efektif dalam memfasilitasi komunikasi dan mendukung proses belajar mengajar di sekolah.

Penelitian keempat "Object Detection in Unmanned Aerial Vehicle Imagery" [10]. Penelitian ini bertujuan memanfaatkan data video yang dihasilkan oleh kamera pada drone atau *Unmanned Air Vehicles* (UAV) melalui penerapan *machine learning* dan *computer vision*. Dengan menggunakan model RetinaNet, penelitian ini melatih berbagai model untuk mendeteksi artefak urban seperti pejalan kaki dan pesepeda berdasarkan gambar yang diambil dari area kampus Stanford. Hal ini menunjukkan potensi drone sebagai alat inovatif untuk mengoptimalkan pengumpulan dan analisis data visual dalam berbagai aplikasi, dari ini hasil penelitian mendapatkan akurasi sebesar 63%.

Penelitian kelima "Aplikasi Pengenalan Abjad Sistem Isyarat Bahasa Indonesia (SIBI) Dengan Algoritma Yolov5"[11]. Penelitian ini bertujuan mengembangkan platform pengenalan abjad Sistem Isyarat Bahasa Indonesia (SIBI) menggunakan algoritma YOLOv5 sebagai media pembelajaran bahasa isyarat yang *efektif*. Pengujian dilakukan dalam dua tahap, yaitu pengujian antarmuka aplikasi yang berhasil menampilkan enam halaman pengguna dengan baik dan pengujian manual untuk mendeteksi gerakan SIBI pada 26 kelas. Beberapa kelas seperti 'A', 'E', 'D', dan 'J' memiliki akurasi deteksi rendah karena kemiripan gerakan atau kerumitan gestur. Deteksi secara real-time menggunakan kamera ponsel menunjukkan akurasi sebesar 77% (0,77). Platform ini diharapkan dapat meningkatkan komunikasi antara masyarakat umum dan teman tuli.

Berdasarkan berbagai penelitian terdahulu, dapat terlihat bahwa pendekatan *deep learning* telah banyak digunakan untuk mendeteksi bahasa isyarat, khususnya pada sistem SIBI. Penelitian pertama dan kedua menunjukkan bahwa model berbasis SSD mampu memberikan akurasi yang tinggi dalam mendeteksi alfabet SIBI secara *real-time*. Namun, penelitian tersebut umumnya menggunakan jumlah kelas yang terbatas serta dataset yang relatif kecil sehingga kemampuan *generalisasi* model pada kondisi nyata masih perlu diuji lebih lanjut.

Penelitian selanjutnya mulai mengadopsi arsitektur lain seperti YOLO yang memiliki kemampuan deteksi objek secara cepat pada sistem *real-time*. Studi yang menggunakan YOLO menunjukkan bahwa model ini mampu mendeteksi gesture bahasa isyarat dengan akurasi yang cukup baik, namun masih dipengaruhi oleh faktor lingkungan seperti pencahayaan, kualitas kamera, dan variasi gerakan tangan.

Penelitian lain yang menggunakan arsitektur RetinaNet menunjukkan bahwa model ini memiliki kemampuan yang baik dalam menangani ketidakseimbangan kelas melalui penggunaan focal loss. Namun demikian, implementasi RetinaNet pada perangkat dengan sumber daya terbatas masih menghadapi tantangan dalam hal efisiensi komputasi.

Berdasarkan rangkaian penelitian tersebut, dapat disimpulkan bahwa setiap arsitektur memiliki kelebihan dan keterbatasan masing-masing dalam hal akurasi, kecepatan deteksi, serta efisiensi komputasi. Oleh karena itu, penelitian ini melakukan perbandingan terhadap tiga arsitektur *deep learning* yaitu MobileNetV2-SSD, MobileNetV2-RetinaNet, dan YOLOv11 untuk mengetahui model yang paling optimal dalam mendeteksi bahasa isyarat SIBI khususnya pada implementasi perangkat *mobile*.

2.2 Teori Penunjang

2.2.1 SIBI

Sistem Isyarat Bahasa Indonesia (SIBI) merupakan Bahasa isyarat yang resmi di Indonesia dan sudah diresmikan dalam Undang-Undang No.2 Tahun 1989. SIBI sendiri mengadaptasi Bahasa isyarat *America Sign Language* (ASL) yang hanya menggunakan gerakan satu tangan, oleh karena itu SIBI dan ASL memiliki kemiripan. Sistem Isyarat Bahasa Indonesia (SIBI) lebih sering digunakan pada kegiatan acara resmi seperti milik pemerintah dan kegiatan di sekolah [7].

2.2.2 MobileNetV2

MobileNetV2 merupakan *deep neural network* ringan yang menggunakan depthwise separable convolution untuk mengurangi jumlah parameter dibandingkan regular convolution. Arsitektur MobileNetV2 terdiri dari 53 convolution layer dan memperkenalkan fitur *linear bottleneck* serta *shortcut* antar *bottleneck layer* (inverted residual). Terdapat dua jenis convolution layer yang digunakan, yaitu 1×1 convolution dan 3×3 depthwise convolution [8].

Depthwise separable convolution terdiri dari depthwise convolution yang memproses setiap channel input secara independen, serta *pointwise convolution* dengan kernel 1×1 untuk menggabungkan informasi antar-channel. MobileNetV2 juga menerapkan batch normalization dan fungsi aktivasi ReLU untuk meningkatkan stabilitas dan performa model [12]. *Depthwise convolution* dengan satu filter

per channel input dapat dinyatakan dengan rumus sebagai berikut:

$$\hat{G}_{\{k,l,m\}} = \sum_{\{i,j\}} \hat{K}_{\{i,j,m\}} \cdot F_{\{k+i-1,l+j-1,m\}} \quad (1)$$

Keterangan :

a. $\hat{G}_{\{k,l,m\}}$: Nilai pada output feature map di lokasi (k,l) untuk channel ke-m. Hasil ini adalah nilai dari fitur yang diekstraksi setelah *depthwise convolution* diterapkan.
b. $\hat{K}_{\{i,j,m\}}$: Kernel filter *depthwise convolution* berukuran $D_k \times D_k$ yang diterapkan hanya pada channel ke-m dari input. Filter ini bekerja secara independen untuk setiap channel input.
c. $F_{\{k+i-1,l+j-1,m\}}$: Nilai dari input feature map F di lokasi spatial $(k+i-1, l+j-1)$ pada channel ke-m. Ini adalah elemen input yang akan dikalikan dengan filter.
d. $\sum_{\{i,j\}}$: Operasi penjumlahan di seluruh elemen kernel i,ji,ji,j untuk menghitung hasil *convolution* pada lokasi tersebut.

2.2.3 SSD

Single Shot Multibox Detector (SSD) merupakan algoritma pendeteksian objek populer yang mudah diimplementasikan, memiliki akurasi yang baik dengan kebutuhan komputasi relatif rendah, serta mampu bekerja secara real-time. SSD menggunakan feed-forward convolutional network untuk menentukan bounding boxes berukuran tetap dan menghitung skor kelas pada setiap objek. Arsitektur SSD terdiri dari VGG-16 sebagai *base network* dan *extra feature layers*, yaitu enam lapisan konvolusi tambahan untuk memprediksi objek dengan berbagai aspek rasio [4].

Untuk meningkatkan akurasi prediksi, SSD menggunakan fungsi loss yang menggabungkan Localization Loss untuk mengukur ketepatan posisi bounding box dan Confidence Loss untuk mengukur ketepatan klasifikasi objek. Secara matematis, fungsi loss SSD dapat dituliskan sebagai berikut:

$$\boldsymbol{L}(x,c,l,g) = \frac{1}{N}\left(L_{\{conf\}(x,c)} + \alpha L_{\{loc\}(x,l,g)}\right) \quad (2)$$

Di mana N adalah jumlah default boxes yang berhasil dicocokkan. Jika N=0, maka nilai loss akan diatur menjadi 0. Localization loss dihitung dengan Smooth L1 loss antara prediksi bounding box (l) dan bounding box ground truth (g) [13].

2.2.3 RetinaNet

RetinaNet merupakan model deteksi objek yang terdiri dari jaringan ekstraksi fitur, *Feature Pyramid Network* (FPN), serta dua sub-network untuk klasifikasi objek dan regresi bounding box. Jaringan ekstraksi fitur umumnya berbasis ResNet, sedangkan FPN berfungsi menggabungkan fitur tingkat rendah dan tinggi agar menghasilkan representasi fitur yang lebih kaya. Dengan arsitektur ini, RetinaNet mampu menghasilkan deteksi objek yang akurat dan efisien, terutama pada kasus ketidakseimbangan kelas pada deteksi satu tahap [14]. Pada penelitian ini, jaringan ekstraksi fitur yang digunakan adalah MobileNetV2 karena bersifat ringan dan efisien untuk perangkat dengan sumber daya terbatas.

Untuk mengatasi ketidakseimbangan kelas antara foreground dan background, RetinaNet menerapkan focal loss, yaitu modifikasi dari *cross-entropy loss* yang menekankan pembelajaran pada contoh sulit dan jarang, sehingga meningkatkan kinerja deteksi objek. Secara matematis, focal loss dapat dituliskan sebagai berikut:

$$FL(p_t) = -\alpha_t(1 - p_t)^{\gamma}\left(log(p_t)\right) \quad (3)$$

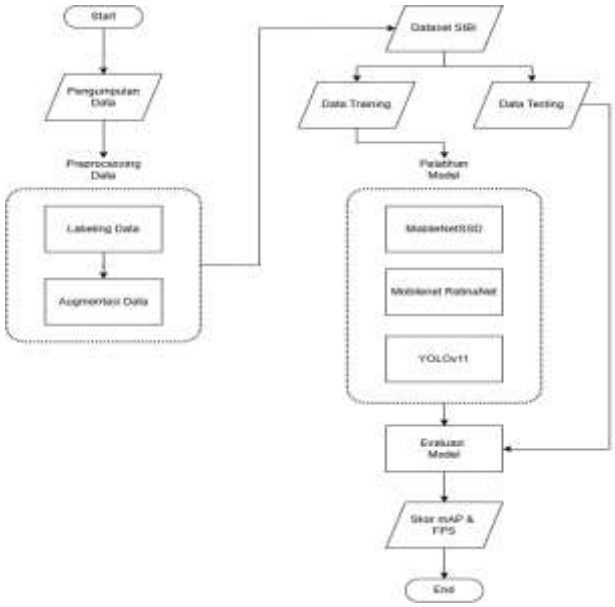
Di mana p_t adalah probabilitas yang diprediksi untuk kelas yang benar, α_t adalah faktor penyeimbang, dan γ mengontrol pengaruh contoh mudah. Dengan ini, RetinaNet dapat meningkatkan deteksi objek meskipun terjadi ketidakseimbangan kelas [15].

2.2.4 YOLOv11

YOLOv11 merupakan terasi terbaru dari YOLO yang membangun kekuatan dari pendahulunya sambil memperkenalkan kemampuan baru yang memperluas kegunaannya di berbagai aplikasi *computer vision* (CV). YOLOv11 membedakan dirinya melalui peningkatan *adaptabilitas*, mendukung berbagai tugas CV yang lebih luas selain deteksi objek tradisional. Beberapa di antaranya adalah estimasi postur dan segmentasi instance, yang memperluas penerapan model ini di berbagai domain. Desain YOLOv11 fokus pada keseimbangan antara kekuatan dan kepraktisan, dengan tujuan untuk mengatasi tantangan spesifik di berbagai industri dengan akurasi dan efisiensi yang lebih tinggi. Model terbaru ini menunjukkan evolusi berkelanjutan dari teknologi deteksi objek real-time, mendorong batasan-batasan apa yang mungkin dicapai dalam aplikasi CV. Fleksibilitas dan peningkatan kinerjanya menjadikan YOLOv11 sebagai kemajuan signifikan di bidang ini, yang berpotensi membuka jalan baru untuk implementasi dunia nyata di berbagai sektor [16].

3. METODE PENELITIAN

Penelitian ini dilakukan melalui beberapa tahapan, dimulai dengan pengumpulan data berupa foto bahasa isyarat SIBI dari huruf A-Z, dilanjutkan dengan preprocessing data, pelatihan model untuk pengenalan SIBI dan diakhiri dengan melakukan evaluasi terhadap model yang telah dilatih.



Gambar 1. Tahapan Penelitian

3.1 Pengumpulan Data

Proses pengumpulan data ini, teknik dokumentasi visual dilakukan menggunakan kamera ponsel. Setiap kata dalam bahasa isyarat SIBI dikumpulkan sebanyak 30 gambar untuk memastikan variasi dalam data yang diperoleh. Dengan adanya 26 huruf dalam alfabet isyarat SIBI, total gambar yang dihasilkan mencapai 780 gambar. Proses ini diawali dengan pemilihan dan penentuan daftar gerakan yang akan difoto, termasuk huruf A-Z. Kamera ponsel dengan resolusi tinggi digunakan untuk memastikan gambar memiliki kualitas yang baik dan detail yang jelas. Pengambilan gambar dilakukan di lokasi dengan pencahayaan yang cukup agar gerakan tangan dapat terlihat tanpa bayangan yang mengganggu. Kamera juga diatur agar tetap fokus pada tangan dengan latar belakang polos untuk memperjelas objek yang difoto.

3.2 Labeling Data

Data berupa foto bahasa isyarat SIBI dari huruf A-Z yang telah dikumpulkan kemudian melalui proses pelabelan. Proses pelabelan ini dilakukan menggunakan aplikasi Label Studio dengan memberikan label pada setiap gambar dengan cara menambahkan anotasi bounding box pada huruf yang diwakili oleh gerakan tangan dalam bahasa isyarat SIBI. Setiap foto dianalisis untuk memastikan bahwa gerakan tangan dalam gambar sesuai dengan huruf atau kata tertentu, dan label yang tepat diberikan pada gambar tersebut. Misalnya, foto yang menggambarkan gerakan tangan untuk huruf "A" akan diberi label "A, dan seterusnya untuk setiap huruf dari A-Z.

Gambar 2. Pelabelan Data

3.3 Augmentasi Data

Penelitian terdahulu, deteksi bahasa isyarat masih mengalami kendala pada kondisi pencahayaan rendah. Oleh karena itu, dilakukan augmentasi data untuk meningkatkan ketahanan dan generalisasi model pada dataset bahasa isyarat SIBI [9]. Teknik augmentasi yang digunakan meliputi *horizontal flip* untuk menambah variasi orientasi posisi tangan, serta penyesuaian kecerahan (*brightness adjustment*) dengan menaikkan atau menurunkan intensitas cahaya guna mensimulasikan berbagai kondisi pencahayaan [6]. Pemilihan kedua teknik ini didasarkan pada pertimbangan bahwa variasi orientasi tangan dan pencahayaan merupakan dua faktor dominan yang paling sering ditemui dalam skenario penggunaan nyata. Oleh karena itu, augmentasi difokuskan pada aspek yang secara langsung berdampak pada kemampuan generalisasi model, tanpa menambahkan kompleksitas transformasi yang berpotensi mengubah karakteristik utama gestur alfabet SIBI.

Gambar 3. Penyesuaian Kontras Gambar Asli

Gambar 4. Penyesuaian Kontras Gambar yang Dibalik

3.4 Pelatihan Model

Pelatihan model yang dilakukan dalam penelitian ini menggunakan tiga jenis model yaitu: MobileNetV2-SSD, MobileNetV2-RetinaNet dan YOLOv11. Dataset yang digunakan akan memiliki rasio training, validasi dan testing sebesar 8:1:1 dan nantinya kedua model MobileNetV2-SSD dan MobileNetV2-RetinaNet memanfaatkan MobileNetV2 sebagai *pre-trained backbone*, yang memberikan keuntungan dalam hal efisiensi komputasi dan akurasi, terutama untuk aplikasi di perangkat dengan keterbatasan sumber daya. MobileNetV2 sebagai *backbone* memungkinkan kedua model ini untuk mendeteksi objek dengan kecepatan tinggi dan menggunakan lebih sedikit daya, menjadikannya cocok untuk aplikasi deteksi bahasa isyarat secara *real-time*. Sementara itu, YOLOv11, meskipun tidak menggunakan MobileNetV2 sebagai *backbone*, karena kemampuannya dalam mendeteksi objek dalam berbagai ukuran dan kecepatan tinggi, menjadikannya pilihan yang kuat untuk deteksi objek dalam video atau gambar yang lebih kompleks. Ketiga model ini dapat dilatih untuk mendeteksi gerakan dan simbol dalam bahasa isyarat SIBI, menerjemahkannya ke dalam teks.

3.5 Evaluasi Model

Evaluasi dilakukan dengan menganalisis hasil perbandingan mAP dan FPS dari ketiga model yang diuji, yaitu MobileNetV2 SSD, MobileNetV2 RetinaNet, dan YOLOv11. Jika suatu model memiliki mAP tinggi tetapi FPS rendah, maka model tersebut mungkin lebih akurat tetapi kurang *efisien* untuk sistem *real-time*. Sebaliknya, jika sebuah model memiliki FPS tinggi tetapi mAP rendah, maka model tersebut lebih cepat tetapi kurang dapat diandalkan dalam mendeteksi isyarat dengan benar. Oleh karena itu, trade-off antara akurasi dan kecepatan menjadi faktor utama dalam pemilihan model terbaik.

4. HASIL DAN PEMBAHASAN

Tahap *preprocessing*, citra dalam dataset di-*resize* ke ukuran 416×416 piksel. Penggunaan resolusi 416×416 mendukung stabilitas proses pelatihan karena mempertahankan rasio aspek yang seragam serta mempermudah optimasi *batch processing* dalam GPU maupun perangkat terbatas. Setelah dilakukan preprocessing, seluruh data training dikenai augmentasi untuk meningkatkan keragaman data menggunakan library Albumentations. Dua jenis augmentasi diterapkan secara konsisten terhadap semua citra, yaitu *Horizontal Flip*, di mana setiap citra dibalik secara *horizontal* untuk menghasilkan versi tangan kiri dari tangan kanan (atau sebaliknya) agar model dapat mengenali gestur dalam orientasi terbalik, dan *Brightness Adjustment*, yaitu penyesuaian kecerahan citra (peningkatan atau penurunan) menggunakan fungsi *Random Brightnes Contrast* untuk mensimulasikan variasi pencahayaan di lingkungan nyata. Dari proses augmentasi ini meningkatkan jumlah dataset *training* awal sebesar 624 data (80% dari total dataset), menjadi 3744 data.

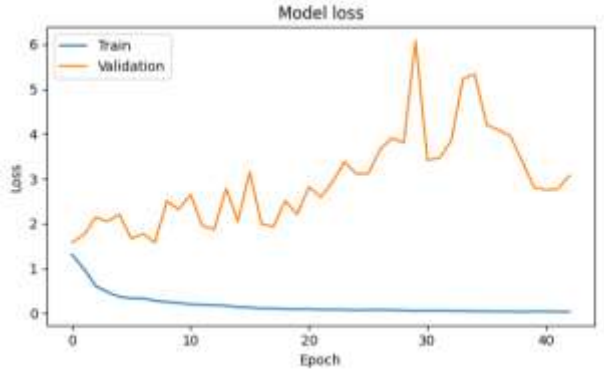
Dalam proses pelatihan model *deep learning*, terdapat dua istilah penting yaitu *epoch* dan *step*. *Epoch* merupakan satu siklus penuh ketika seluruh data pelatihan telah diproses oleh model sebanyak satu kali. Sementara itu, *step* merupakan jumlah pembaruan parameter model yang dilakukan selama proses pelatihan, yang biasanya terjadi setelah model memproses satu batch data. Dengan demikian, dalam satu *epoch* dapat terdapat beberapa *step* tergantung pada ukuran dataset dan batch size yang digunakan.

Perbedaan penggunaan epoch dan step pada penelitian ini disebabkan oleh perbedaan framework pelatihan yang digunakan oleh masing-masing model. Pada pelatihan YOLOv11 dan RetinaNet, proses training dikontrol berdasarkan jumlah epoch. Sedangkan pada pelatihan MobileNetV2-SSD yang menggunakan *TensorFlow Object Detection API*, proses training dikontrol berdasarkan jumlah step.

Tahap ini, dilakukan proses persiapan pelatihan untuk tiga model deteksi objek yang akan dibandingkan, yaitu MobileNetV2-RetinaNet, MobileNetV2-SSD, dan YOLOv11. Setiap model memerlukan pendekatan konfigurasi dan *preprocessing* data yang sedikit berbeda sesuai dengan arsitektur dan format input masing-masing.

4.1 Analisis dan Perbandingan Akurasi Model

Pada tahap awal pelatihan, model RetinaNet dilatih dengan rasio dataset 80% untuk pelatihan, 10% untuk *validasi* dan 10% untuk *testing*.

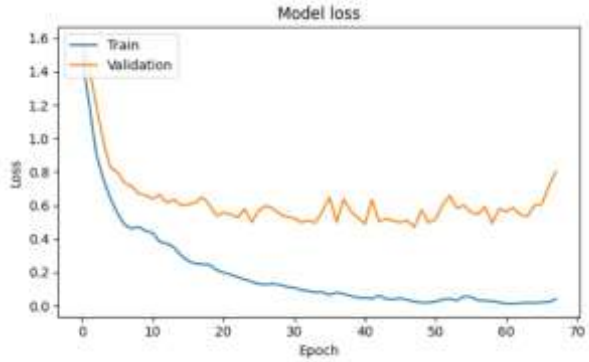


Gambar 5. Grafik Loss Rasio Dataset 80:10:10 dengan MobileNetV2 – RetinaNet

Hasil evaluasi menunjukkan bahwa pada metrik mAP dengan ambang batas *Intersection over Union* (IoU) sebesar 0.50, yang dihasilkan hanya sebesar 38,8%, sementara training loss terus menurun dan *validation loss* cenderung stagnan dan naik. Kondisi ini mengindikasikan bahwa model mengalami *overfitting*, yaitu model terlalu menyesuaikan diri dengan data pelatihan namun tidak mampu melakukan generalisasi terhadap data validasi. *Overfitting* pada model *deep learning* dapat dipengaruhi oleh berbagai faktor, seperti keterbatasan jumlah data, kompleksitas model yang tinggi, serta kurangnya teknik regularisasi selama proses pelatihan.

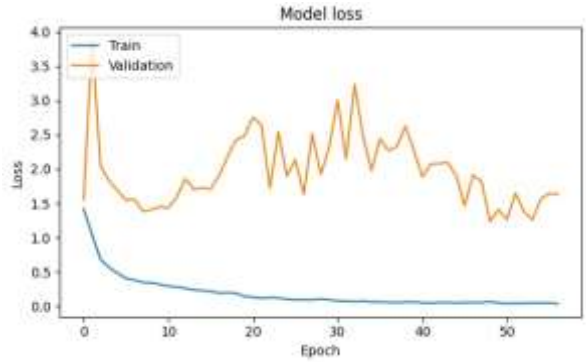
Perubahan rasio pembagian dataset dalam penelitian ini tidak dimaksudkan sebagai solusi langsung terhadap overfitting, melainkan sebagai bagian dari eksperimen untuk melihat pengaruh distribusi data terhadap stabilitas proses pelatihan. Secara umum, penanganan overfitting pada model deep learning biasanya dilakukan melalui beberapa pendekatan seperti augmentasi data, penggunaan teknik regularisasi, penerapan dropout layer, serta penggunaan early stopping selama proses pelatihan.

Dilakukan beberapa eksperimen ulang dengan mengubah rasio pembagian dataset menjadi 50:25:25, 60:20:20, dan 70:15:15 untuk *training*, validation, dan testing.



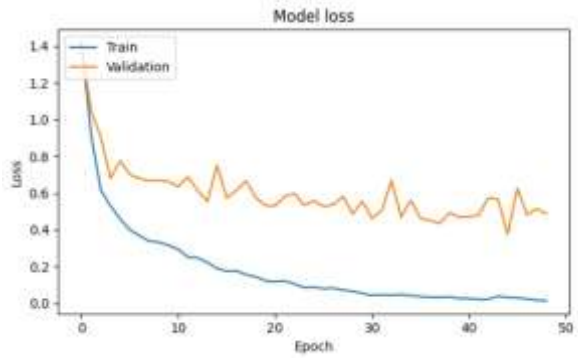
Gambar 6. Grafik Loss Rasio Dataset 50:25:25 dengan MobileNetV2 – RetinaNet

Hasil dari rasio dataset 50:25:25 menunjukkan peningkatan performa yang signifikan, dengan mAP@50 meningkat hingga 79,44%. Validation loss mengalami penurunan yang stabil, meskipun sedikit *overfitting* masih terlihat hingga akhir *epoch*.



Gambar 7. Grafik Loss Rasio Dataset 60:20:20 dengan MobileNetV2 – RetinaNet

Rasio dataset 60:20:20, performa model justru menurun, dengan mAP@50 hanya mencapai 59%. Validation loss mengalami *fluktuasi* hingga akhir pelatihan, yang mengindikasikan model mengalami *overfitting*.

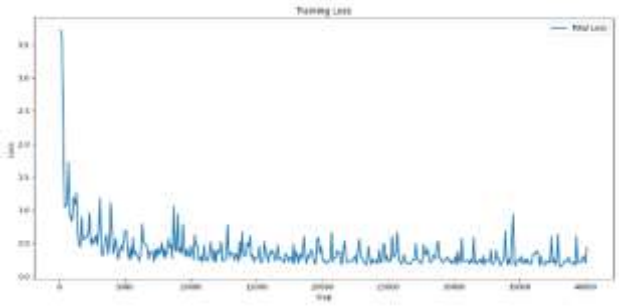


Gambar 8. Grafik Loss Rasio Dataset 70:15:15 dengan MobileNetV2 – RetinaNet

Rasio dataset 70:15:15, performa model kembali meningkat dengan mAP@50 mencapai 87,69%. Validation loss menunjukkan penurunan yang cukup stabil meskipun terdapat sedikit *fluktuasi* hingga akhir *epoch*.

Selanjutnya dilakukan proses pelatihan dengan MobileNetV2 – SSD, proses pelatihan dilakukan berdasarkan jumlah langkah (*step*), sehingga grafik hasil pelatihan menampilkan hubungan antara training loss terhadap *step*. Berdasarkan grafik yang dihasilkan, nilai training loss pada awal pelatihan menunjukkan angka yang cukup tinggi, yaitu 3.3. Namun seiring berjalannya proses pelatihan, loss secara bertahap menurun dan mencapai nilai yang lebih stabil di kisaran 0.5 hingga 0.2 setelah 10.000 langkah. *Fluktuasi* kecil tetap terlihat sepanjang proses pelatihan. Berbeda dengan model lain yang digunakan, model ini tidak terdapat validation loss dalam proses ini karena *TensorFlow Model Research*

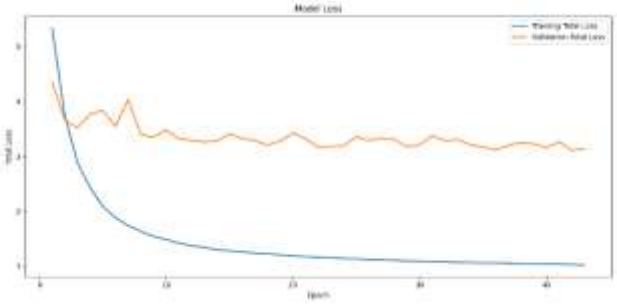
API secara *default* tidak menyertakan proses validasi selama pelatihan.



Gambar 9. Grafik Loss Pelatihan MobileNetV2-SSD

Hasil pelatihan model dan diperoleh hasil evaluasi pada mAP@50, yaitu sebesar 99.70%. Nilai ini menunjukkan bahwa model memiliki performa yang sangat baik dalam mendeteksi objek secara akurat pada dataset yang digunakan. Selain itu, hasil evaluasi juga menunjukkan bahwa model mampu mendeteksi objek berukuran sedang dan besar dengan baik, meskipun tidak ada data yang cukup untuk mengukur performa pada objek berukuran kecil. Secara keseluruhan, hasil ini mengindikasikan bahwa kombinasi MobileNetV2 dan SSD (MobileNetV2-SSD) cukup efektif dan efisien untuk digunakan dalam tugas deteksi objek pada dataset SIBI yang digunakan dalam penelitian ini.

Terakhir dilakukan pelatihan Model YOLOv11 menunjukkan penurunan total *loss* pada data *training* dengan sangat baik dan konsisten selama 43 *epoch*, yang berarti proses pelatihan berhasil memperbaiki kemampuan model dalam mengenali dan mengklasifikasi objek pada data training. Namun, total validation loss yang cenderung lebih tinggi dan fluktuatif mengindikasikan bahwa model mungkin mengalami sedikit *overfitting* dan belum sepenuhnya generalisasi dengan baik pada data validasi.



Gambar 10. Grafik Loss Pelatihan YOLOv11

Model ini lalu diuji pada dataset testing ini menunjukkan performa yang cukup baik pada metrik mAP@50, yaitu sebesar 89,80% pada keseluruhan kelas. Nilai ini mengindikasikan bahwa model mampu mendeteksi objek dengan tingkat akurasi yang tinggi pada IoU 0.50.

4.2 Evaluasi Model

Evaluasi dilakukan terhadap tiga model deteksi objek MobileNetV2+RetinaNet, MobileNetV2+SSD, dan YOLOv11 berdasarkan dua aspek utama: mAP @ IoU 0.50 dan performa *inferensi* FPS di perangkat *mobile*. Hasil evaluasi mAP dan FPS di atas, hasil yang diperoleh disajikan pada tabel I berikut:

TABEL I. HASIL EVALUASI MODEL

Model	mAP (%)	FPS
MobileNetV2-SSD	99,70%	9
MobileNetV2-RetinaNet	87,67%	2
YOLOv11	89,80%	5

5. KESIMPULAN DAN SARAN

Hasil evaluasi terhadap tiga model deteksi objek yang dilatih dapat disimpulkan MobileNetV2-SSD menunjukkan performa terbaik dengan mAP sebesar 99,70%, kecepatan 9 FPS, dan ukuran model yang ringan. YOLOv11 berada di posisi kedua dengan mAP 89,80s% dan kecepatan 5 FPS, meskipun menunjukkan sedikit *overfitting* selama pelatihan. Sementara itu, MobileNetV2-RetinaNet pada awalnya menunjukkan hasil rendah (mAP 38,8%), namun mengalami peningkatan signifikan hingga 87,69% pada rasio dataset terbaik (70:15:15). Meskipun akurasinya meningkat, kecepatan inferensinya tetap rendah (2 FPS) akibat kompleksitas model yang tinggi. Berdasarkan hasil keseluruhan, MobileNetV2-SSD merupakan model yang paling sesuai untuk implementasi sistem penerjemahan bahasa isyarat SIBI di perangkat Android secara *real-time*. Kombinasi antara akurasi tinggi, kecepatan deteksi stabil, dan efisiensi model menjadikannya ideal untuk perangkat dengan keterbatasan sumber daya. YOLOv11 masih berpotensi dikembangkan lebih lanjut, terutama dengan optimalisasi untuk mengurangi *overfitting*. Sementara itu, MobileNetV2-RetinaNet kurang direkomendasikan untuk skenario *real-time* karena keterbatasannya dalam *efisiensi* meskipun akurasi dapat ditingkatkan dengan rasio data pelatihan tertentu.

Berdasarkan hasil dan temuan dalam penelitian ini, terdapat beberapa saran yang dapat diajukan untuk mendukung pengembangan penelitian lebih lanjut serta peningkatan kualitas sistem deteksi bahasa isyarat SIBI di masa mendatang, yaitu sebagai berikut:

Penerapan teknik regularisasi, *augmentasi* data dan jumlah data perlu ditingkatkan, mengingat beberapa model, seperti MobileNetV2-RetinaNet, mengalami *overfitting* selama pelatihan. Penelitian selanjutnya disarankan untuk melakukan eksplorasi lebih mendalam terhadap pengaruh rasio pembagian dataset terhadap performa model, mengingat hasil eksperimen tambahan menunjukkan bahwa variasi rasio dapat memengaruhi akurasi dan kestabilan pelatihan. Penelitian lanjutan dapat diarahkan untuk mengembangkan sistem yang mampu mengenali gestur kontinu atau dinamis. Hal ini penting karena dalam komunikasi bahasa isyarat yang sebenarnya, *gestur* tidak selalu bersifat statis, melainkan sering kali berupa rangkaian gerakan yang saling berkesinambungan. Sistem yang mampu mengenali gestur dinamis akan lebih *representatif* terhadap penggunaan bahasa isyarat dalam situasi nyata.

UCAPAN TERIMA KASIH

Penulis ingin mengucapkan terima kasih kepada Institut Bisnis dan Teknologi Indonesia atas dukungan atas pendanaan hibah internal IRDP 2026.

DAFTAR PUSTAKA

[1] A. R. Syulistyo, D. S. Hormansyah, and P. Y. Saputra, "SIBI (Sistem Isyarat Bahasa Indonesia) translation using Convolutional Neural Network (CNN)," in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, Jan. 2020. doi: 10.1088/1757-899X/732/1/012082.

[2] S. Daniels, N. Suciati, and C. Fathichah, "Indonesian Sign Language Recognition using YOLO Method," IOP Conf. Ser.: Mater. Sci. Eng., vol. 1077, no. 1, p. 012029, Feb. 2021, doi: 10.1088/1757-899x/1077/1/012029.

[3] P. Patel, S. Pampaniya, A. Ghosh, R. Raj, K. Deepa, and S. Kandasamy, "Enhancing Accessibility Through Machine Learning: A Review on Visual and Hearing Impairment Technologies," 2025, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2025.3539081.

[4] S. Apendi and M. W. Paryasto, "Deteksi Bahasa Isyarat Sistem Isyarat Bahasa Indonesia Menggunakan Metode Single Shot Multibox Detector,"*e-Proceding of Engineering*, Vol.10, No1, Feb. 2023.

[5] N. Wulandari, I. Ardiyanto, and H. A. Nugroho, "A Comparison of Deep Learning Approach for Underwater Object Detection," *Jurnal RESTI*, vol. 6, no. 2, pp. 252–258, Apr. 2022, doi: 10.29207/resti.v6i2.3931.

[6] M. A. R. Alif, "YOLOv11 for Vehicle Detection: Advancements, Performance, and Applications in Intelligent Transportation Systems," Oct. 2024, [Online]. Available: http://arxiv.org/abs/2410.22898

[7] Hidayahtullah Mahardi Akbar, "Sistem Deteksi Simbol Pada Sibi (Sistem Isyarat Bahasa Indonesia) Secara Realtime Menggunakan Mobilenet-Ssd,"Tugas akhir, Universitas Dinamika. 2022.

[8] Y. Kristian Suyitno, I. Wayan Sudiarsa, I. Nyoman Buda Hartawan, and I. Dewa Putu Gede Wiyata Putra, "Implementation of Sibi Sign Language Realtime Detection Program (Case Studi At Sekolah Luar Biasa Negeri 1 Tabanan)," *Architecture and High Performance Computing*, vol. 6, no. 3, 2024, doi: 10.47709/cnapc.v6i3.4405.

[9] A. Kailany, | Moayed, and D. Daneshyari, "International Journal of Emerging Trends in Science and Technology Object Detection in Unmanned Aerial Vehicle Imagery", doi: 10.18535/ijetst/v8i9.01.

[10] D. Permana and J. Sutopo, "Aplikasi Pengenalan Abjad Sistem Isyarat Bahasa Indonesia (Sibi) Dengan Algoritma Yolov5 Mobile Application Alphabet Recognition Of Indonesian Language Sign System (Sibi) Using Yolov5 Algorithm," vol. 11, no. 2, 2023.

[11] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," Apr. 2017, [Online]. Available: http://arxiv.org/abs/1704.04861

[12] W. Liu *et al.*, "SSD: Single Shot MultiBox Detector," *European conference on computer vision.* Dec. 2016, doi: 10.1007/978-3-319-46448-0_2.

[13] H. Zhang, J. Zhao, Y. Chen, S. Jiang, D. Wang, and H. Du, "Intelligent bird's nest hazard detection of transmission line based on RetinaNet model," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Aug. 2021. doi: 10.1088/1742-6596/2005/1/012235.

[14] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," Feb. 2018, [Online]. Available: http://arxiv.org/abs/1708.02002

[15] R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," Oct. 2024, [Online]. Available: http://arxiv.org/abs/2410.17725.