P-ISSN: 1411-3635
E-ISSN: 2962-5645

Perbandingan Kinerja YOLOv8 dan SSD MobileNet untuk Deteksi Kelengkapan Atribut Seragam Siswa

[1]Argi Ginanjar, [2]Adhi Kusnadi
[1,2]Program Magister Informatika, Universitas Nusa Putra, Sukabumi

E-mail: [1]argi.ginanjar_mif24@nusaputra.ac.id, [2]adhikusnadi@nusaputra.ac.id

ABSTRAK

Pemeriksaan atribut seragam siswa membutuhkan ketelitian karena beberapa komponen, seperti epolet, *name tag*, sabuk, sepatu hitam, dan topi, tidak selalu tampak jelas pada citra. Penelitian ini membandingkan kinerja YOLOv8n dan SSD MobileNet untuk mendeteksi kelengkapan atribut seragam siswa menggunakan *dataset* mandiri. *Dataset* terdiri atas 1.532 citra setelah augmentasi, lima kelas objek, dan 5.294 anotasi *bounding box* berformat YOLO. Eksperimen dilakukan melalui penyusunan *dataset*, anotasi objek, pembagian data *train*, *validation*, dan *test*, pelatihan *model*, serta evaluasi akhir pada *test set*. Metrik yang digunakan meliputi *precision*, *recall*, *mAP@50*, *mAP@50-95*, *inference time*, dan FPS. Pada *test set*, YOLOv8n memperoleh *precision* 0,8966, *recall* 0,8689, *mAP@50* 0,9276, *mAP@50-95* 0,6723, *inference time* 10,36 ms, dan FPS 96,47. SSD MobileNet memperoleh *precision* 0,5140, *recall* 0,5169, *mAP@50* 0,6034, *mAP@50-95* 0,3171, *inference time* 11,00 ms, dan FPS 90,87. Hasil tersebut memperlihatkan bahwa YOLOv8n lebih unggul pada skenario pengujian yang digunakan.

Kata kunci: ***YOLOv8, SSD MobileNet, deteksi objek, computer vision, deep learning, mAP***

ABSTRACT

Inspecting the completeness of student uniform attributes is challenging because several items, such as epaulettes, name tags, belts, black shoes, and caps, may appear small, partially occluded, or visually unclear. This study compares YOLOv8n and SSD MobileNet for detecting student uniform attributes using a self-collected dataset. The dataset contains 1,532 augmented images, five object classes, and 5,294 YOLO-format bounding box annotations. The experimental procedure included dataset preparation, object annotation, train-validation-test splitting, model training, and final evaluation on the test set. The evaluation used precision, recall, mAP@50, mAP@50-95, inference time, and FPS. On the test set, YOLOv8n achieved a precision of 0.8966, recall of 0.8689, mAP@50 of 0.9276, mAP@50-95 of 0.6723, inference time of 10.36 ms, and 96.47 FPS. SSD MobileNet achieved a precision of 0.5140, recall of 0.5169, mAP@50 of 0.6034, mAP@50-95 of 0.3171, inference time of 11.00 ms, and 90.87 FPS. These findings indicate that YOLOv8n performed better in the tested dataset and experimental setting.

Keyword: YOLOv8, SSD MobileNet, object detection, computer vision, deep learning, mAP

1. PENDAHULUAN

Kelengkapan atribut seragam menjadi bagian penting dalam pengamatan kedisiplinan siswa di lingkungan sekolah. Atribut seperti epolet, *name tag*, sabuk, sepatu hitam, dan topi sering muncul dengan ukuran, posisi, dan tingkat keterlihatan yang berbeda pada setiap citra. Perbedaan kondisi visual tersebut membuat pemeriksaan manual tidak selalu mudah dilakukan, terutama ketika objek berukuran kecil atau sebagian tertutup.

Pemanfaatan *computer vision* dalam penelitian ini diarahkan untuk membantu membaca informasi visual pada citra siswa secara lebih sistematis. Pada kasus atribut seragam, *model* tidak hanya perlu mengenali keberadaan suatu atribut, tetapi juga perlu memahami posisi objek pada gambar karena letak dan skala

objek dapat berubah antar citra (Szeliski, 2022).

Pendekatan *object detection* dipilih karena satu citra dapat memuat beberapa atribut sekaligus. Berbeda dari klasifikasi citra yang hanya memberikan label umum, deteksi objek menghasilkan informasi kelas dan lokasi objek melalui *bounding box*. Karakteristik ini sesuai dengan kebutuhan penelitian karena atribut seperti *name tag*, epolet, sabuk, sepatu hitam, dan topi harus dikenali sekaligus dilokalisasi pada citra (Zhao et al., 2019).

Pada deteksi objek modern, YOLO dan SSD termasuk *model* satu tahap yang banyak digunakan karena proses prediksinya relatif ringkas. YOLO memformulasikan deteksi sebagai prediksi langsung terhadap kelas objek dan koordinat *bounding box* (Redmon et al., 2016). SSD menggunakan *default box* pada beberapa *feature map* untuk memperkirakan lokasi dan kelas objek dalam satu proses prediksi (Liu et al., 2016).

Penelitian ini menggunakan YOLOv8n sebagai *model* utama dan SSD MobileNet sebagai *model* pembanding. Pemilihan YOLOv8n didasarkan pada kebutuhan untuk melihat keseimbangan antara akurasi dan kecepatan pada *dataset* mandiri. Dokumentasi Ultralytics menjelaskan bahwa YOLOv8 tersedia dalam beberapa varian yang dapat dipilih sesuai kebutuhan performa dan efisiensi komputasi (Ultralytics, 2026).

Dengan latar tersebut, penelitian ini diarahkan untuk membandingkan kinerja YOLOv8n dan SSD MobileNet pada tugas deteksi atribut seragam siswa. Perbandingan dilakukan menggunakan *precision*, *recall*, *mAP@50*, *mAP@50-95*, *inference time*, dan FPS. Fokus kajian bukan untuk menggeneralisasi *model* terbaik pada seluruh kasus deteksi objek, melainkan menentukan *model* yang lebih sesuai untuk *dataset* atribut seragam siswa yang digunakan dalam eksperimen ini.

2. LANDASAN TEORI

2.1 *Object Detection*

Object detection merupakan tugas *computer vision* yang menghasilkan dua keluaran utama, yaitu kategori objek dan posisi objek pada citra. Dalam penelitian ini, kemampuan tersebut diperlukan karena atribut seragam tidak cukup hanya dikenali sebagai kelas tertentu, tetapi juga harus diketahui letaknya melalui koordinat *bounding box* (Zhao et al., 2019).

Perkembangan deteksi objek berbasis *deep learning* melahirkan dua kelompok pendekatan besar. *Model* berbasis *region proposal*, seperti Fast R-CNN dan Faster R-CNN, mengutamakan tahapan pencarian kandidat wilayah objek (Girshick, 2015). Sebaliknya, *model* satu tahap seperti YOLO dan SSD menyederhanakan proses deteksi dengan menghasilkan prediksi kelas dan *bounding box* secara secara langsung dalam satu tahap pemrosesan (Redmon et al., 2016); (Liu et al., 2016).

Pada konteks atribut seragam siswa, *object detection* digunakan karena objek yang diamati memiliki ukuran dan bentuk yang tidak seragam. *Name tag* dan epolet dapat tampak kecil pada citra, sedangkan sabuk memiliki bentuk memanjang dan dapat tertutup sebagian oleh pakaian. Kondisi ini membuat evaluasi tidak hanya menilai ketepatan kelas, tetapi juga kualitas lokalisasi *bounding box* dan kecepatan inferensi.

Deep learning memungkinkan *model* mempelajari pola visual secara bertingkat dari citra. Representasi tersebut membantu *model* membedakan atribut seragam dari latar belakang maupun

P-ISSN: 1411-3635
E-ISSN: 2962-5645

bagian pakaian lain. Oleh karena itu, variasi citra, ukuran objek, serta kualitas anotasi menjadi faktor yang berpengaruh terhadap keberhasilan deteksi (Goodfellow et al., 2016).

2.2 YOLOv8

YOLO atau You Only Look Once merupakan pendekatan deteksi objek yang memproses citra dalam satu alur prediksi untuk menghasilkan kelas objek dan koordinat *bounding box*. Konsep ini diperkenalkan untuk mendukung deteksi *real-time* dengan menyederhanakan proses deteksi menjadi persoalan regresi langsung (Redmon et al., 2016). Perkembangan YOLO selanjutnya terus diarahkan pada keseimbangan antara akurasi dan kecepatan (Bochkovskiy et al., 2020).

YOLOv8 termasuk generasi *model* YOLO modern yang digunakan pada berbagai tugas deteksi objek Menurut (Terven et al., 2023) menempatkan YOLOv8 sebagai bagian dari perkembangan arsitektur YOLO kontemporer. (Yaseen, 2024) menjelaskan bahwa YOLOv8 memiliki pembaruan pada proses *feature extraction*, *feature fusion*, dan pendekatan *anchor-free*.

Model yang digunakan dalam penelitian ini adalah YOLOv8n. Varian nano dipilih karena lebih ringan dibandingkan varian YOLOv8 yang lebih besar, sehingga relevan untuk menguji kombinasi akurasi dan kecepatan inferensi. Pada *dataset* atribut seragam siswa, YOLOv8n digunakan untuk mendeteksi objek dengan variasi ukuran dan posisi dalam satu citra.

Deteksi atribut seragam memiliki kedekatan dengan persoalan objek kecil. Beberapa studi terbaru menunjukkan bahwa *small object detection* masih menjadi tantangan karena objek memiliki informasi spasial terbatas dan mudah dipengaruhi latar belakang. Kondisi tersebut sesuai dengan karakteristik *name tag*, epolet, dan sabuk pada *dataset* penelitian ini (Khalili & Smyth, 2024).

2.3 SSD MobileNet

SSD atau Single Shot MultiBox Detector merupakan *model* deteksi objek satu tahap yang memprediksi kelas dan *bounding box* secara langsung. Liu et al. (2016) menjelaskan bahwa SSD memanfaatkan *default box* dengan variasi skala dan rasio aspek pada beberapa *feature map*. Mekanisme ini memungkinkan *model* menangani objek dengan ukuran berbeda tanpa tahapan *proposal region* terpisah.

MobileNet dikembangkan sebagai arsitektur *convolutional neural network* yang menekankan efisiensi komputasi. MobileNetV2 menggunakan *inverted residual* dan *linear bottleneck* untuk menghasilkan representasi yang lebih ringan (Sandler et al., 2018). MobileNetV3 kemudian memperbaiki rancangan arsitektur agar lebih sesuai untuk perangkat dengan sumber daya terbatas (Howard et al., 2019).

Dalam eksperimen ini, SSD MobileNet diimplementasikan melalui SSDLite MobileNetV3 pada ekosistem PyTorch/Torchvision. Torchvision menyediakan model ssdlite320_mobilenet_v3_large dengan ukuran input 320 x 320 dan backbone MobileNetV3 Large (Torchvision, 2026). PyTorch digunakan karena mendukung pengembangan model deteksi objek secara fleksibel dalam penelitian (Paszke et al., 2019).

SSD MobileNet diposisikan sebagai *model* pembanding karena mewakili arsitektur deteksi ringan yang berorientasi pada efisiensi. Pada studi komparatif, *model* pembanding diperlukan untuk membaca sejauh mana perbedaan kinerja *model* utama terhadap pendekatan lain

yang relevan. Dalam penelitian ini, SSD MobileNet digunakan untuk menilai kemampuan *model* ringan pada *dataset* atribut seragam siswa.

2.4 Metrik Evaluasi Deteksi Objek

Metrik evaluasi yang digunakan terdiri atas *precision, recall, mAP@50, mAP@50-95, inference time, dan FPS*. *Precision* digunakan untuk menilai ketepatan prediksi positif model, sedangkan *recall* digunakan untuk membaca kemampuan model menemukan objek yang benar-benar terdapat pada *ground truth*. Dalam konteks atribut seragam siswa, *precision* penting untuk mengurangi kesalahan deteksi atribut, sedangkan *recall* penting untuk melihat apakah atribut yang seharusnya terdeteksi masih terlewat oleh model.

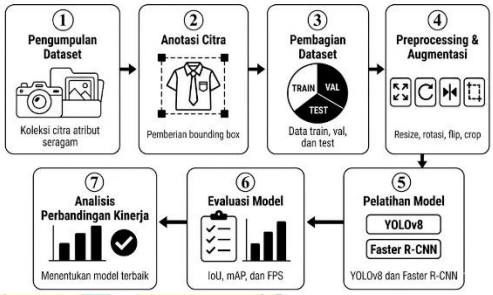
mAP@50 menilai performa deteksi pada ambang *IoU 0,50*, sementara *mAP@50-95* memberikan penilaian lebih ketat karena menggunakan beberapa ambang IoU. Selain akurasi, penelitian ini juga menilai efisiensi model melalui inference time dan FPS. Inference time menunjukkan waktu pemrosesan satu citra, sedangkan FPS menunjukkan banyaknya frame yang dapat diproses setiap detik. (Reis et al., 2023) menunjukkan bahwa mAP dan FPS sering digunakan bersama untuk membaca keseimbangan antara akurasi dan kecepatan pada aplikasi YOLOv8 real-time. Dalam penelitian ini, hasil utama diambil dari test set agar mencerminkan kemampuan model terhadap data uji.

3. METODOLOGI

3.1 Desain Penelitian

Penelitian ini menggunakan desain eksperimen komparatif dengan membandingkan YOLOv8n dan SSD MobileNet. Kedua *model* menggunakan *dataset* yang sama agar hasil pengujian dapat dibaca secara seimbang. YOLOv8n ditempatkan sebagai *model* utama, sedangkan SSD MobileNet digunakan sebagai pembanding dari keluarga detektor ringan.

Prosedur penelitian dilakukan melalui penyusunan *dataset* mandiri, pemberian anotasi *bounding box*, pembagian data menjadi *train*, *validation*, dan *test*, pelatihan kedua *model*, evaluasi menggunakan *test set*, serta penyusunan analisis komparatif. Alur tersebut disajikan pada Gambar 1 sebagai gambaran umum tahapan eksperimen.



Gambar 1. Alur penelitian

3.2 Dataset Penelitian

Dataset penelitian berupa citra atribut seragam siswa yang dikumpulkan dan disusun secara mandiri. Objek yang diamati terdiri atas lima kelas, yaitu epolet, *name_tag*, sabuk, *sepatu_hitam*, dan topi. Setelah augmentasi, *dataset* berjumlah 1.532 citra dengan 5.294 anotasi objek.

Penggunaan *dataset* mandiri memungkinkan penelitian menyesuaikan kelas objek dengan konteks pemeriksaan atribut seragam siswa. *Dataset* ini juga memuat karakteristik objek yang tidak selalu mudah dideteksi, seperti ukuran kecil, posisi yang bervariasi, dan kemungkinan tertutup sebagian. Tantangan tersebut sejalan dengan kajian *small object detection* yang menekankan keterbatasan informasi spasial pada objek kecil (Cheng et al., 2023); (Nikouei et al., 2025).

P-ISSN: 1411-3635
E-ISSN: 2962-5645

Dataset dibagi menjadi *train*, *validation*, dan *test*. Data *train* digunakan untuk pembelajaran *model*, data *validation* digunakan untuk memantau performa selama pelatihan, sedangkan data *test* digunakan sebagai dasar pengujian akhir. Komposisi data disajikan pada Tabel 1.

Tabel 1. Komposisi *dataset*

Jenis data	Jumlah gambar	Keterangan
Train	1.341	Digunakan untuk proses pelatihan *model*
Validation	128	Digunakan untuk validasi selama proses pelatihan
Test	63	Digunakan untuk pengujian akhir *model*
Total	1.532	Total citra setelah augmentasi

3.3 Anotasi dan Pembagian Data

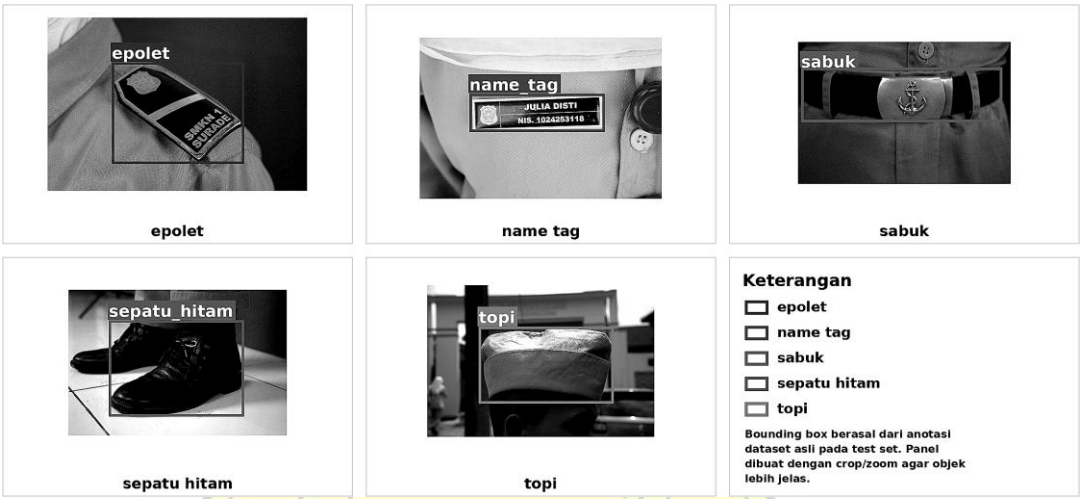
Anotasi objek disusun dalam format YOLO *bounding box*. Setiap baris anotasi memuat *class_id*, *x_center*, *y_center*, *width*, dan *height* yang dinormalisasi terhadap ukuran citra. Format ini dipilih karena sesuai untuk pelatihan YOLOv8n dan tetap dapat dibaca ulang untuk kebutuhan *model* pembanding.

Pada eksperimen SSD MobileNet berbasis PyTorch/Torchvision, label YOLO dikonversi saat proses pembacaan data menjadi format *xmin*, *ymin*, *xmax*, dan *ymax*. Konversi tersebut hanya dilakukan pada tahap *input model*, sehingga struktur *dataset* asli tetap dipertahankan. Dengan demikian, kedua *model* diuji menggunakan sumber data yang sama.

Daftar kelas objek ditampilkan pada Tabel 2. Contoh visual anotasi disajikan pada Gambar 2 untuk memperlihatkan bentuk *bounding box* pada setiap atribut seragam yang diamati.

Tabel 2. Kelas objek *dataset*

No	Kelas objek	Keterangan
1	epolet	Atribut pada bagian bahu seragam
2	*name_tag*	Tanda nama pada seragam siswa
3	sabuk	Sabuk sebagai atribut seragam
4	*sepatu_hitam*	Sepatu hitam yang digunakan siswa
5	topi	Topi sebagai atribut kelengkapan seragam



Gambar 2. Contoh anotasi *dataset*

3.4 Arsitektur Model

Model utama dalam penelitian ini adalah YOLOv8n. *Model* ini digunakan karena memiliki ukuran ringan dan dirancang untuk menghasilkan deteksi objek dengan waktu inferensi yang relatif cepat. Pada eksperimen ini, YOLOv8n dilatih untuk mengenali lima kelas atribut seragam siswa dan dievaluasi menggunakan *test set*.

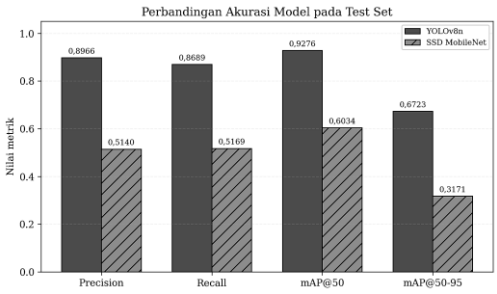
Model pembanding adalah SSD MobileNet berbasis SSDLite MobileNetV3. *Model* ini menggabungkan pendekatan SSD dengan *backbone* MobileNetV3 yang efisien. Pemilihan SSD MobileNet bertujuan memberikan pembanding terhadap YOLOv8n dari sisi akurasi deteksi dan efisiensi komputasi.

Kedua *model* dinilai menggunakan metrik yang sama, yaitu *precision*, *recall*, *mAP@50*, *mAP@50-95*, *inference time*, dan FPS. Kesamaan metrik diperlukan agar perbandingan kinerja tidak dipengaruhi oleh perbedaan prosedur evaluasi.

3.5 Tahapan Eksperimen

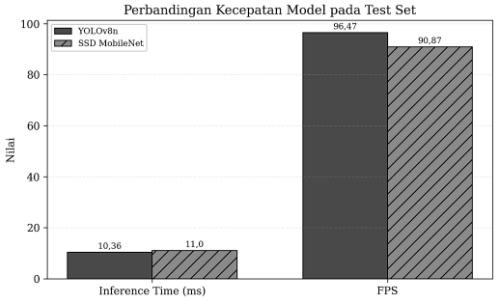
Eksperimen dimulai dengan menyiapkan citra atribut seragam siswa dan anotasi *bounding box*. Setelah *dataset* dibagi menjadi *train*, *validation*, dan *test*, YOLOv8n dan SSD MobileNet dilatih secara terpisah menggunakan sumber data yang sama. Hasil kedua *model* kemudian diuji pada *test set* dan dibandingkan melalui tabel serta grafik.

Gambar 3 menyajikan perbandingan *precision*, *recall*, *mAP@50*, dan *mAP@50-95* antara YOLOv8n dan SSD MobileNet. Sementara itu, Gambar 4 memperlihatkan perbandingan *inference time* dan FPS untuk melihat efisiensi kedua *model* dalam memproses citra.



Gambar 3. Grafik perbandingan akurasi *model*

P-ISSN: 1411-3635
E-ISSN: 2962-5645



Gambar 4. Grafik perbandingan kecepatan *model*

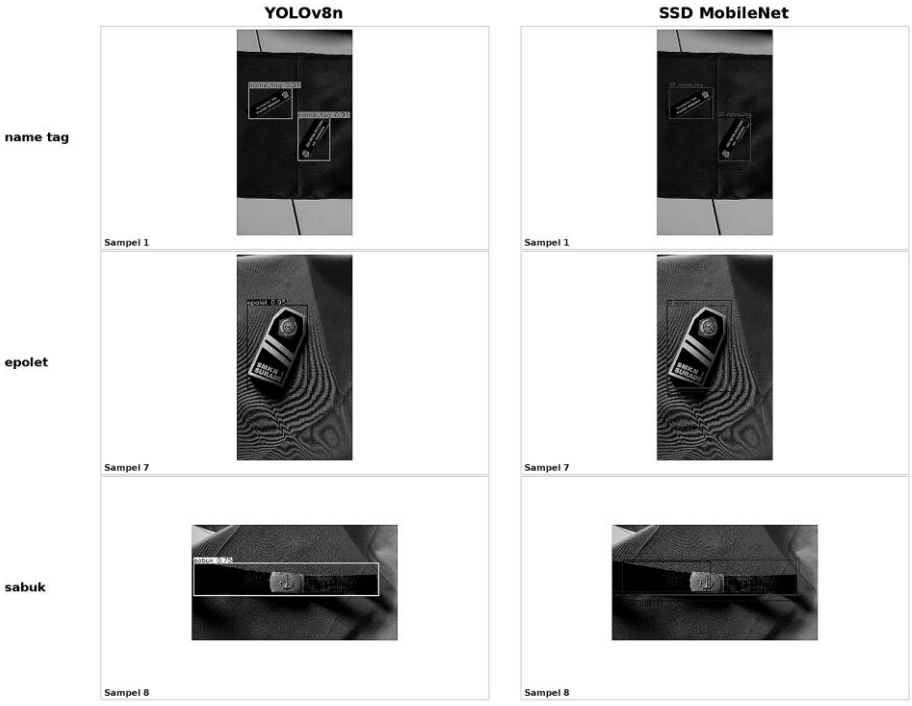
3.6 Metrik Evaluasi

Metrik evaluasi yang digunakan terdiri atas *precision*, *recall*, *mAP@50*, *mAP@50-95*, *inference time*, dan FPS. *Precision* digunakan untuk menilai ketepatan prediksi positif *model*, sedangkan *recall* digunakan untuk membaca kemampuan *model* menemukan objek yang benar. *mAP@50* menilai deteksi pada ambang IoU 0,50, sementara *mAP@50-95* memberikan penilaian lebih ketat karena menggunakan beberapa ambang IoU.

Inference time menunjukkan lama waktu pemrosesan satu citra, sedangkan FPS menunjukkan banyaknya *frame* yang dapat diproses setiap detik. (Reis et al., 2023) menunjukkan bahwa *mAP* dan FPS sering digunakan bersama untuk membaca keseimbangan antara akurasi dan kecepatan pada aplikasi YOLOv8 *real-time*. Dalam penelitian ini, hasil utama diambil dari *test set* agar mencerminkan kemampuan *model* terhadap data uji.

4. HASIL DAN PEMBAHASAN

Contoh hasil prediksi kedua *model* pada beberapa atribut seragam siswa ditampilkan pada Gambar 5. Visualisasi tersebut menunjukkan perbedaan keluaran deteksi antara YOLOv8n dan SSD MobileNet pada objek yang sama.



Gambar 5. Contoh hasil prediksi YOLOv8n dan SSD MobileNet

Tabel 3. Perbandingan hasil evaluasi *model* pada *test set*

Metrik	YOLOv8n	SSD MobileNet
Precision	0,8966	0,5140
Recall	0,8689	0,5169
mAP@50	0,9276	0,6034
mAP@50-95	0,6723	0,3171
Inference time (ms)	10,36	11,00
FPS	96,47	90,87

Berdasarkan evaluasi *test set*, YOLOv8n memperoleh *precision* 0,8966, *recall* 0,8689, *mAP@50* 0,9276, *mAP@50-95* 0,6723, *inference time* 10,36 ms, dan FPS 96,47. SSD MobileNet memperoleh *precision* 0,5140, *recall* 0,5169, *mAP@50* 0,6034, *mAP@50-95* 0,3171, *inference time* 11,00 ms, dan FPS 90,87. Selisih tersebut menunjukkan bahwa YOLOv8n lebih stabil dalam mengenali atribut seragam pada data uji.

Nilai *mAP@50* dan *mAP@50-95* menunjukkan bahwa YOLOv8n tidak hanya lebih baik dalam mengenali objek, tetapi juga lebih konsisten dalam menempatkan *bounding box*. Penurunan *mAP@50-95* pada kedua *model* mengindikasikan bahwa lokalisasi objek tetap menjadi tantangan, terutama pada *name tag* dan epolet yang berukuran kecil serta sabuk yang bentuknya memanjang dan dapat tertutup sebagian. Dari sisi kecepatan, YOLOv8n juga lebih efisien dengan *inference time* 10,36 ms dan FPS 96,47. Dengan demikian, pada *dataset* dan skenario pengujian ini, YOLOv8n menunjukkan keseimbangan akurasi dan kecepatan yang lebih baik dibandingkan SSD MobileNet.

5. KESIMPULAN

Penelitian ini membandingkan YOLOv8n dan SSD MobileNet untuk mendeteksi kelengkapan atribut seragam siswa pada *dataset* mandiri. *Dataset* terdiri atas 1.532 citra setelah augmentasi, lima kelas objek, dan 5.294 anotasi *bounding box*. Evaluasi akhir dilakukan menggunakan *test set* dengan metrik *precision*, *recall*, *mAP@50*, *mAP@50-95*, *inference time*, dan FPS.

Hasil pengujian memperlihatkan bahwa YOLOv8n menghasilkan *precision* 0,8966, *recall* 0,8689, *mAP@50* 0,9276, *mAP@50-95* 0,6723, *inference time* 10,36 ms, dan FPS 96,47. SSD MobileNet menghasilkan *precision* 0,5140, *recall* 0,5169, *mAP@50* 0,6034, *mAP@50-95* 0,3171, *inference time* 11,00 ms, dan FPS 90,87. Berdasarkan hasil tersebut, YOLOv8n menjadi *model* yang lebih sesuai untuk skenario deteksi atribut seragam siswa dalam penelitian ini.

DAFTAR PUSTAKA

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. In *arXiv preprint arXiv:2004.10934*. https://arxiv.org/abs/2004.10934

Cheng, G., Yuan, X., Yao, X., Yan, K., Zeng, Q., Xie, X., & Han, J. (2023). Towards large-scale small object detection: Survey and benchmarks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2023.3290594

Girshick, R. (2015). Fast R-CNN. *Proceedings of the IEEE International Conference on Computer*

P-ISSN: 1411-3635
E-ISSN: 2962-5645

Vision, 1440–1448. https://doi.org/10.1109/ICCV.2015.169

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. https://www.deeplearningbook.org/

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V, & Adam, H. (2019). Searching for MobileNetV3. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1314–1324. https://doi.org/10.1109/ICCV.2019.00140

Khalili, B., & Smyth, A. W. (2024). SOD-YOLOv8: Enhancing YOLOv8 for small object detection in traffic scenes. In *arXiv preprint arXiv:2408.04786*. https://arxiv.org/abs/2408.04786

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. *European Conference on Computer Vision*, 21–37. https://doi.org/10.1007/978-3-319-46448-0_2

Nikouei, M., Baroutian, B., Nabavi, S., Taraghi, F., Aghaei, A., Sajedi, A., & Ebrahimi Moghaddam, M. (2025). Small object detection: A comprehensive survey on challenges, techniques and real-world applications. In *arXiv preprint arXiv:2503.20516*. https://arxiv.org/abs/2503.20516

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Koepf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., … Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, *32*. https://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 779–788.

Reis, D., Kupec, J., Hong, J., & Daoudi, A. (2023). Real-time flying object detection with YOLOv8. In *arXiv preprint arXiv:2305.09972*. https://arxiv.org/abs/2305.09972

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4510–4520. https://doi.org/10.1109/CVPR.2018.00474

Szeliski, R. (2022). *Computer vision: Algorithms and applications*. Springer. https://doi.org/10.1007/978-3-030-34372-9

Terven, J., Córdova-Esparza, D.-M., & Romero-González, J.-A. (2023). A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Machine Learning and Knowledge Extraction*, *5*(4), 1680–1716.

Torchvision. (2026). ssdlite320_mobilenet_v3_large. In *PyTorch Documentation*. https://docs.pytorch.org/vision/stable/models/generated/torchvision.models.detection.ssdlite320_mobilenet_v3_large.html

Ultralytics. (2026). YOLOv8. In *Ultralytics Documentation*. https://docs.ultralytics.com/models/yolov8/

Yaseen, M. (2024). What is YOLOv8: An in-depth exploration of the internal features of the next-generation object detector. In *arXiv preprint arXiv:2408.15857*. https://arxiv.org/abs/2408.15857

Zhao, Z.-Q., Zheng, P., Xu, S.-T., & Wu, X. (2019). Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(11), 3212–3232. https://doi.org/10.1109/TNNLS.2018.2876865