

ANALISIS KINERJA YOLOV11 VS YOLOV8 PADA DATASET DEEPPFASHION2: PENERAPAN MODEL TERBARU UNTUK DETEKSI KEPATUHAN PAKAIAN PADA APLIKASI WEB

Reksa Prayoga Syahputra ¹, Aubrey Nedwin Mantiri ²,
Aditya Rieyza Munif ³, Gema Parasti Mindara ⁴, Endang Purnama Giri ⁵

¹⁻³ Teknologi Rekayasa Perangkat Lunak, IPB University

⁴ Teknologi Rekayasa Komputer, IPB University

⁵ Ilmu Komputer, IPB University

Jalan Kumbang No.14, Kota Bogor, Indonesia

mantiriaubrey@apps.ipb.ac.id

ABSTRAK

Pemantauan kepatuhan aturan busana (*dress code*) di lingkungan institusi sering kali terkendala oleh proses verifikasi manual yang subjektif dan tidak efisien. Penelitian ini bertujuan untuk merancang sistem deteksi otomatis berbasis web serta melakukan analisis komparatif antara model YOLOv8 dan model terbaru YOLOv11 dalam mengenali atribut panjang lengan pakaian. Sistem dikembangkan menggunakan arsitektur *frontend* React dan *backend* Flask, dengan model deteksi yang dilatih pada subset dataset DeepFashion2 yang terdiri dari empat kelas yang merepresentasikan variasi lengan panjang dan pendek pada kemeja serta luaran. Hasil pengujian kuantitatif menunjukkan bahwa YOLOv11 mengungguli YOLOv8 pada deteksi *bounding box* dengan mAP@0.5 mencapai 0,57 dan peningkatan *Precision* yang signifikan hingga 0,809, meskipun YOLOv8 menunjukkan stabilitas yang sedikit lebih baik pada segmentasi mask. Kinerja terendah tercatat pada kelas *outwear* akibat ketidakseimbangan data yang ekstrem. Secara keseluruhan, sistem terbukti mampu menyajikan status kepatuhan secara optimal, menempatkan YOLOv11 sebagai opsi model yang lebih efisien dan akurat untuk skenario pemantauan *dress code* otomatis.

Kata kunci : DeepFashion2, Deteksi pakaian, Kepatuhan dress code, Segmentasi citra, YOLOv8, YOLOv11

1. PENDAHULUAN

Pesatnya perkembangan teknologi *computer vision* telah mendorong lahirnya berbagai aplikasi inovatif berbasis kamera dalam industri mode dan gaya hidup. Implementasi ini mencakup spektrum yang luas, mulai dari *smart mirror* untuk pengalaman ritel interaktif [1], teknologi *virtual try-on* yang memungkinkan pengguna mencoba pakaian secara digital [2], *personal styling assistant* [3], hingga sistem analitik pengunjung [4]. Tren ini menunjukkan adanya kebutuhan mendesak akan model visi komputer yang tidak hanya mampu mendeteksi keberadaan objek, tetapi juga memahami atribut visual pakaian secara andal dan *real-time*.

Di lingkungan institusi pendidikan maupun perkantoran, salah satu atribut visual yang relevan adalah panjang lengan pakaian. Aturan mengenai penggunaan lengan panjang atau lengan pendek sering kali menjadi parameter utama dalam kebijakan kode busana (*dress code*) yang membedakan aturan formal dan kasual pada hari-hari tertentu [5].

Namun, tantangan utama dalam penerapan aturan ini adalah ketiadaan sistem otomatis yang spesifik untuk memverifikasi kepatuhan panjang lengan secara instan. Sebagian besar penelitian terdahulu lebih berfokus pada klasifikasi gaya *fashion* secara umum atau rekomendasi pakaian, bukan pada skenario praktis pemeriksaan biner (patuh/tidak patuh) yang terhubung dengan aturan harian yang dinamis. Ketiadaan alat bantu ini menyulitkan pemantauan yang konsisten dan efisien.

Menanggapi permasalahan tersebut, penelitian ini bertujuan untuk mengembangkan sebuah prototipe sistem deteksi kepatuhan pakaian berbasis web yang stabil, akurat, dan mudah direplikasi. Fokus utama penelitian adalah menciptakan sistem yang mampu memisahkan kategori pakaian berdasarkan panjang lengan secara otomatis untuk keperluan validasi *dress code*. Penelitian ini juga bertujuan untuk mengkaji potensi peningkatan performa deteksi melalui analisis komparatif antara model YOLOv8 yang sudah mapan dengan model terbaru YOLOv11, guna memberikan rekomendasi teknis yang valid bagi pengembangan sistem serupa di masa depan.

Untuk mencapai tujuan tersebut, rencana penyelesaian masalah dilakukan melalui pengembangan *pipeline* deteksi objek menggunakan subset dataset DeepFashion2 yang dibagi menjadi empat kelas relevan. Penelitian ini menerapkan model YOLOv8 sebagai basis, serta melakukan analisis komparatif dengan model terbaru YOLOv11 untuk mengkaji potensi peningkatan performa. Solusi ini diintegrasikan ke dalam aplikasi web dengan *frontend* React dan *backend* Flask yang mampu memvisualisasikan hasil deteksi serta menyajikan ringkasan kepatuhan *dress code* secara otomatis.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Sejumlah penelitian telah mengeksplorasi penerapan visi komputer untuk analisis atribut pakaian dan pemantauan objek. Bashir et al. mengembangkan

sistem berbasis *Deep Learning* untuk mengklasifikasikan pelanggaran aturan busana di lingkungan pendidikan [6]. Penelitian tersebut berfokus pada klasifikasi pelanggaran multi-kelas secara luas. Namun, untuk pengenalan atribut yang lebih rinci (*fine-grained*), Xiang et al. mengusulkan kerangka kerja berbasis R-CNN dengan L-Softmax Loss untuk mengenali atribut kemeja wanita, termasuk variasi panjang lengan (*long, short, mid-sleeved*) [7]. Meskipun metode Xiang et al. mencapai tingkat keberhasilan pelabelan hingga 87,77%, pendekatan deteksi dua tahap (*two-stage*) yang digunakan memiliki kompleksitas komputasi yang lebih tinggi dibandingkan detektor satu tahap. Pendekatan yang lebih berorientasi pada efisiensi dilakukan oleh Li dan Wei, yang mengusulkan metode deteksi ringan (*lightweight*) untuk memantau penggunaan peralatan anti-statis pada pekerja [8]. Studi ini berhasil menekan latensi untuk penggunaan *real-time*, tetapi lingkup deteksinya terbatas pada pakaian pelindung standar industri dan bukan pada variasi pakaian kasual atau formal yang dinamis.

Dalam ranah aplikasi komersial, integrasi *computer vision* lebih banyak diarahkan pada interaksi pengguna. Uddin et al. mengimplementasikan algoritma pengenalan wajah pada *smart mirror* berbasis IoT untuk memberikan rekomendasi informasi yang dipersonalisasi [1]. Sementara itu, Bhogi et al. mengembangkan model *Deep Learning* untuk fitur *virtual try-on* yang memungkinkan pengguna melihat simulasi pakaian secara *real-time* [2]. Meskipun kedua studi ini memanfaatkan atribut visual pakaian, fokus utamanya adalah pada *User Experience* (UX) dan augmentasi visual, bukan pada validasi kepatuhan terhadap aturan yang bersifat biner dan administratif.

2.2. Model You Only Look Once (YOLO)

Model You Only Look Once (YOLO) merupakan arsitektur *Deep Learning* yang populer untuk deteksi objek karena keseimbangan efisien antara kecepatan inferensi dan akurasi. Dalam tinjauan komprehensif mengenai evolusi arsitektur ini, YOLO dikategorikan sebagai *one-stage detector*, di mana prediksi koordinat *bounding box* dan probabilitas kelas dilakukan dalam satu kali evaluasi jaringan saraf [9]. Karakteristik ini membedakan YOLO dari detektor dua tahap seperti R-CNN yang cenderung lebih lambat karena memisahkan proses proposal wilayah dan klasifikasi.

2.3. Komparasi Model YOLOv8 vs YOLOv11

Pemilihan arsitektur model memegang peranan krusial dalam keseimbangan antara akurasi dan kecepatan. Ma Muriyah et al. mengevaluasi evolusi dari YOLOv5 ke YOLOv8, menyimpulkan bahwa YOLOv8 menawarkan peningkatan signifikan berkat arsitektur *anchor-free* dan dukungan multi-task. YOLOv8 menyediakan skala model yang fleksibel,

mulai dari nano hingga x-large, yang memudahkannya diadopsi di berbagai perangkat keras.

Sebagai pengembangan terbaru, Ultralytics merilis YOLOv11 dengan klaim peningkatan efisiensi parameter. Studi oleh Sapkota et al. yang membandingkan YOLOv11 melawan pendahulunya (v8, v9, v10) pada lingkungan kompleks menunjukkan bahwa YOLOv11 memiliki kemampuan generalisasi yang lebih baik dengan jumlah parameter yang lebih hemat [10]. Temuan ini didukung oleh Vrsalovic et al. dan Fang et al. yang berhasil mengimplementasikan varian YOLOv11 untuk deteksi objek spesifik dengan presisi tinggi [11] [12].

2.4. Dataset DeepFashion

Dataset DeepFashion dan *DeepFashion2* menjadi rujukan utama untuk riset pengenalan pakaian karena menyediakan ratusan ribu citra pakaian dengan anotasi kategori, atribut, *landmark*, dan segmentasi [13]. *DeepFashion2* secara khusus mendukung tugas deteksi, pose, segmentasi, dan re-identifikasi pakaian, sehingga relevan untuk pengembangan sistem deteksi kepatuhan busana berbasis visi komputer di dunia nyata [14].

3. METODE PENELITIAN

3.1. Arsitektur Sistem

Arsitektur sistem yang dikembangkan terdiri dari tiga lapisan utama:

- Lapisan Data dan Model (*Training Pipeline*)**
Menggunakan *subset* *DeepFashion2* khusus pakaian atasan yang dianotasi di Roboflow dan diekspor ke format YOLOv8. *Dataset* ini dipakai untuk *fine-tuning* model YOLO di Google Colab (GPU T4) agar model terlatih sesuai kebutuhan deteksi pakaian.
- Lapisan *Backend* (Flask API + YOLO)**
Model YOLOv8 dimuat sekali saat layanan Flask dijalankan dan dilayani melalui *endpoint* POST */predict* untuk menerima citra dan mengembalikan hasil deteksi dalam format JSON. Tersedia juga *endpoint* GET */health* untuk mengecek status layanan, dan secara konsep YOLOv11 dapat menggantikan YOLOv8 tanpa mengubah kontrak API.
- Lapisan *Frontend* (React Web App)**
Aplikasi React memungkinkan pengguna mengunggah atau mengambil gambar, mengirimkannya ke *backend*, lalu menampilkan hasil deteksi berupa anotasi di atas citra. *Frontend* juga mengonversi hasil prediksi model menjadi status kepatuhan *dress code* berdasarkan hari (Senin–Kamis, Jumat, dan akhir pekan).

3.2. Dataset dan Skema Kelas

Dataset bersumber dari *DeepFashion2*, dipetakan ke format YOLO v8 (folder *images and labels*). Penelitian ini memfokuskan kelas pakaian yang relevan untuk kebijakan lengan panjang dan pendek. Dengan *split data: train* 3987, validasi

1212, uji 565, lalu dipetakan ke 4 kelas berikut: *long_sleeved_outwear*, *long_sleeved_shirt*, *short_sleeved_outwear*, dan *short_sleeved_shirt*. Data dibagi menjadi *train/val/test* dengan menjaga keragaman pencahayaan, pose, dan jarak kamera untuk mengurangi *domain shift*.

Tabel 1. Skema kelas dan jumlah citra per kelas

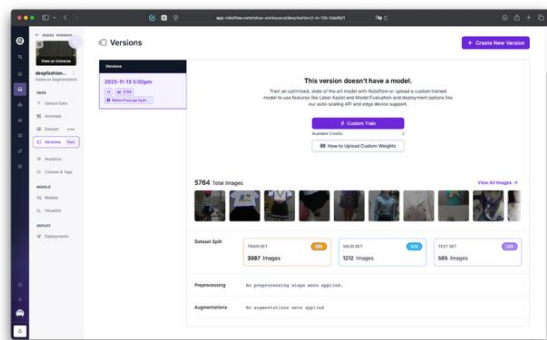
No	Kelas	Jumlah Citra
1	<i>long_sleeved_outwear</i>	602
2	<i>long_sleeved_shirt</i>	1898
3	<i>short_sleeved_outwear</i>	37
4	<i>short_sleeved_shirt</i>	3322

Terlihat bahwa terjadi ketidakseimbangan kelas yang signifikan terutama pada *short_sleeved_outwear* yang hanya memiliki 37 contoh, jauh di bawah kelas mayor seperti *short_sleeved_shirt*. Ketidakseimbangan ini berpotensi menurunkan AP pada kelas minor meskipun mAP agregat terlihat cukup baik.

3.3. Pembuatan Dataset pada Roboflow

Dataset yang digunakan dalam penelitian ini disusun menggunakan platform Roboflow untuk mempermudah proses manajemen citra, anotasi, dan ekspor ke format YOLOv8. Sumber citra berasal dari subset dataset DeepFashion2 yang berisi berbagai kategori pakaian, *dress code* kemudian dipilih subset yang relevan dengan kebutuhan analisis, khususnya kategori pakaian dengan variasi panjang lengan dan jenis *outwear*.

Proses pembuatan dataset diawali dengan mengunggah citra ke proyek Roboflow dan mendefinisikan skema kelas, misalnya: *long_sleeved_shirt*, *short_sleeved_shirt*, *long_sleeved_outwear*, *short_sleeved_outwear*. Setiap objek pakaian dianotasi dengan *bounding box* dan/atau *mask* (untuk segmentasi) sehingga model dapat mempelajari baik lokasi maupun bentuk area pakaian.



Gambar 1. Dataset DeepFashion2

Setelah proses anotasi selesai, dataset dibagi menjadi tiga subset, yaitu data pelatihan, validasi, dan pengujian. Proporsi mengikuti praktik umum dalam pelatihan model visi komputer, dengan pembagian 69% untuk pelatihan, 21% untuk validasi, dan 10% untuk pengujian. Pembagian ini bertujuan untuk

memastikan bahwa performa model yang dilaporkan mencerminkan kemampuan generalisasi pada citra yang tidak pernah dilihat sebelumnya. Roboflow kemudian digunakan untuk mengeksport dataset ke format YOLO, sehingga struktur direktori dan berkas anotasi sudah sesuai dengan ekspektasi framework Ultralytics, seperti yang terlihat pada Gambar 1.

3.4. Konfigurasi Pelatihan Model

Pelatihan awal difokuskan pada model YOLOv8-seg varian kecil (YOLOv8s-seg) untuk mendapatkan *trade-off* yang baik antara akurasi dan latensi. Hyperparameter utama yang digunakan dirangkum pada Tabel 2.

Tabel 2. Hiperparameter pelatihan

Parameter	Nilai	Catatan
epochs	40	Early stopping jika matrik validasi stabil
batch size	1	Sesuaikan VRAM
imgsz	640	Konsisten train/inferensi
optimizer	SGD / AdamW	Pilih sesuai convergence
lr0	0,01 (SGD)	Warmup 3–5 epochs
weight decay	0,0005–0,01	Cegah overfitting
mosaic/hs	aktif	Keragaman tampilan

Pelatihan dilakukan di Google Colab dengan GPU T4 dan replika eksperimen di MacBook berbasis Apple Silicon (M2) untuk menguji portabilitas model. Untuk YOLOv11, penelitian ini menyusun rancangan eksperimen dengan prinsip:

- Menggunakan konfigurasi dataset dan resolusi input yang sama (640×640).
- Memilih varian model yang setara skala parameternya dengan YOLOv8s-seg, yaitu YOLO11s-seg.
- Menggunakan jumlah epochs dan skema learning rate serupa untuk memastikan perbandingan yang adil.

Pada artikel ini, analisis perbandingan terhadap YOLOv11 masih bersifat konseptual dan mengacu pada hasil benchmark resmi serta literatur, sementara eksperimen penuh di subset DeepFashion2 disiapkan sebagai pekerjaan lanjutan.

3.5. Evaluasi dan Metrik

Evaluasi kuantitatif menggunakan metrik standar deteksi objek:

- Precision (P): proporsi prediksi positif yang benar.
- Recall (R): proporsi objek aktual yang berhasil terdeteksi.
- mAP@0,5: mean Average Precision pada ambang Intersection over Union (IoU) 0,5.
- mAP@0,5:0,95: rata-rata mAP pada IoU 0,5 hingga 0,95 dengan langkah 0,05.

Selain metrik agregat, AP per kelas dihitung untuk mengidentifikasi kelas yang lemah khususnya

kelas minor dengan jumlah contoh sangat sedikit. Kurva *precision recall* dan *training loss* (tidak ditampilkan di sini) digunakan sebagai alat bantu dalam menganalisis potensi *overfitting* atau *underfitting*.

3.6. Implementasi Backend Flask

Backend dikembangkan menggunakan Flask sebagai *microframework* web berbasis Python yang ringan dan mudah diintegrasikan dengan pustaka Ultralytics YOLO. Arsitektur *backend* mengikuti pola REST API dengan *endpoint* utama yaitu */predict* yang memiliki alur sebagai berikut:

- Klien (React) mengirim citra dalam bentuk *form data*.
- Flask membaca citra, melakukan pra-proses (*resize*, normalisasi) sesuai format input YOLOv8-seg.
- Model YOLOv8 / YOLOv11 yang sudah dimuat di memori menjalankan inferensi untuk menghasilkan daftar deteksi (*bbox/mask*, label, dan skor kepercayaan).
- Hasil dikemas ke format JSON yang mudah dikonsumsi oleh *frontend*.

Desain ini memungkinkan migrasi ke YOLOv11 dengan perubahan minimal, karena API Ultralytics untuk memuat dan menjalankan model tetap konsisten antarversi.

3.7. Implementasi Frontend React

Frontend dibangun menggunakan React dengan beberapa komponen kunci:

- Upload Component**
Mengizinkan pengguna memilih berkas citra dari sistem *file* atau menggunakan kamera, menampilkan pratinjau citra, dan menyediakan tombol untuk mengirim citra ke server.
- Detection Viewer**
Menerima respons JSON dari *backend* lalu menggambar *bounding box* atau *mask* di atas citra menggunakan *canvas HTML*, serta menampilkan label kelas dan skor kepercayaan di dekat hasil deteksi.
- Dress Code Summary Panel**
Mengagregasi kelas yang terdeteksi (*short_sleeved_** dan *long_sleeved_**) untuk menghitung status kepatuhan terhadap aturan harian (Senin–Kamis, Jumat, Sabtu–Minggu) dan menampilkannya sebagai teks deskriptif yang mudah dipahami.

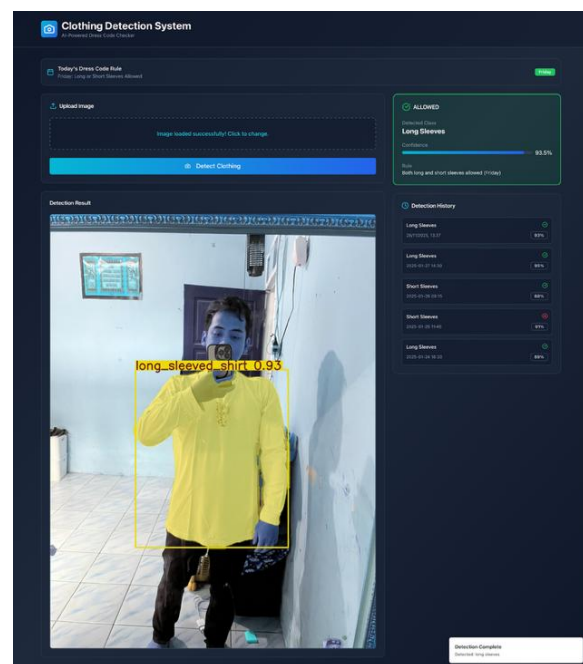
d. Riwayat Deteksi

Menyimpan beberapa hasil deteksi terakhir untuk keperluan demo, penelusuran ulang, dan debugging sederhana.

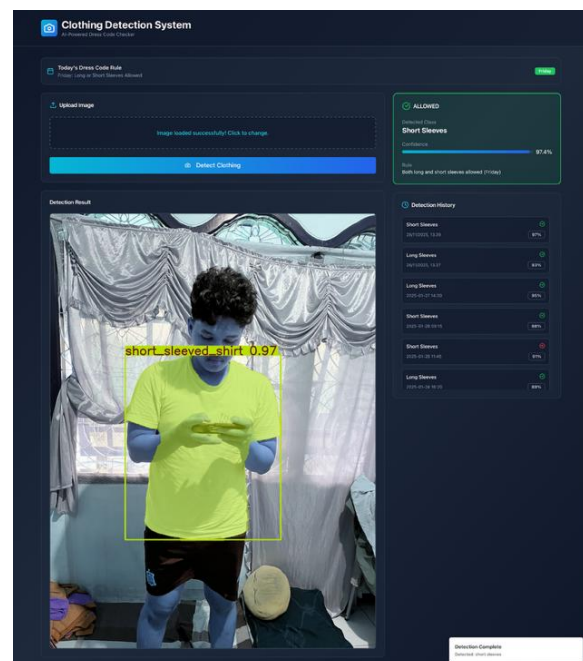
4. HASIL DAN PEMBAHASAN

4.1. Hasil Fungsional

Uji pada dua citra nyata (pemakai *long sleeves* dan *short sleeves*) menunjukkan sistem mengenali kategori dengan benar. *Backend* menghasilkan *annotated image* (*bbox/mask* + label & *confidence*) dan *frontend* menampilkan status kepatuhan sesuai aturan hari.



Gambar 2. Hasil deteksi *long_sleeved_shirt*



Gambar 3. Hasil deteksi *short_sleeved_shirt*

Hasil deteksi pada Gambar 2 menunjukkan bahwa sistem mampu mengenali *long_sleeved_shirt* pada dengan *confidence* 93,5%. *Bounding box* dan label yang ditampilkan di memperlihatkan lokasi dan kelas pakaian secara tepat, beserta status kepatuhan lengan panjang sesuai aturan hari.

Pada Gambar 3, sistem berhasil mendeteksi *short_sleeved_shirt* dengan *confidence* 97,4%. Citra hasil anotasi menampilkan *bounding box* dan label yang akurat, beserta status kepatuhan lengan pendek sehingga pengguna dapat langsung memahami hasil deteksi.

4.2. Hasil Kuantitatif YOLOv8

Pada model YOLOv8, kinerja deteksi *bounding box* berada pada level menengah.

Tabel 3. Ringkasan metrik evaluasi - *Bounding box* (B)

Metrik	Validasi	Uji
mAP@0,5	0.467	0.462
mAP@0,5:0,95	0.336	0.327
Precision	0.684	0.631
Recall	0.474	0.514

Dari Tabel 3 dapat dilihat bahwa nilai mAP@0,5 tercatat 0,467 pada data validasi dan 0,462 pada data uji, sedangkan mAP@0,5:0,95 berada di angka 0,336 dan 0,327. *Precision* yang cukup tinggi (0,684 → 0,631) menunjukkan bahwa prediksi positif model relatif “bersih”, sementara kenaikan *recall* dari 0,474 menjadi 0,514 mengindikasikan bahwa lebih banyak objek yang berhasil tertangkap pada data uji.

Tabel 4. Ringkasan metrik evaluasi - Mask/Seg (M)

Metrik	Validasi	Uji
mAP@0,5	0.467	0.462
mAP@0,5:0,95	0.340	0.331
Precision	0.685	0.629
Recall	0.470	0.530

Pola yang mirip terlihat pada tugas segmentasi mask dalam Tabel 4: mAP@0,5 (M) tetap di kisaran 0,467–0,462 dan mAP@0,5:0,95 (M) di kisaran 0,340–0,331, dengan *precision* (M) 0,685 → 0,629 dan *recall* (M) 0,470 → 0,530. Perbedaan yang tidak terlalu besar antara validasi dan uji memberi gambaran bahwa YOLOv8 masih cukup stabil saat diuji pada *subset* DeepFashion2 ini.

Tabel 5. AP per kelas (*Bounding Box*)

Kelas	AP@0,5:0,95 Val	AP@0,5:0,95 Uji
short_sleeved_shirt	0.613	0.605
long_sleeved_shirt	0.377	0.371
long_sleeved_outwear	0.334	0.308
short_sleeved_outwear	0.019	0.023

Tabel 6. AP per kelas (Mask/Seg)

Kelas	AP@0,5:0,95 Val	AP@0,5:0,95 Uji
short_sleeved_shirt	0.631	0.625
long_sleeved_shirt	0.385	0.379
long_sleeved_outwear	0.327	0.297
short_sleeved_outwear	0.018	0.025

Pada Tabel 5 dan Tabel 6, jika dilihat per kelas, YOLOv8 secara konsisten paling baik mengenali *short_sleeved_shirt*. Kelas ini mencapai AP@0,5:0,95 *box* sebesar 0,613 (val) dan 0,605 (uji), sedangkan AP *mask* sebesar 0,631 (val) dan 0,625 (uji). Kelas *long_sleeved_shirt* dan *long_sleeved_outwear* berada di tengah dengan nilai sekitar 0,30–0,38; sementara *short_sleeved_outwear* tertinggal jauh dengan AP yang sangat kecil (0,019 → 0,023 untuk *box* dan 0,018 → 0,025 untuk *mask*). Kondisi ini cukup sejalan dengan kenyataan pada *dataset*, di mana kelas ini jumlahnya sedikit, objeknya cenderung kecil, dan sering tertutup lapisan pakaian lain. Peringkat kelas yang tetap sama antara validasi dan uji menunjukkan bahwa penurunan kinerja bersifat merata, bukan karena satu kelas tertentu tiba-tiba gagal total.

4.3. Hasil Kuantitatif YOLOv11

Berbeda dengan YOLOv8, YOLOv11 memberikan lonjakan kinerja yang cukup terasa pada deteksi *bounding box*.

Tabel 7. Ringkasan metrik evaluasi - *Bounding box* (B)

Metrik	Validasi	Uji
mAP@0,5	0.568	0.573
mAP@0,5:0,95	0.420	0.444
Precision	0.493	0.809
Recall	0.649	0.548

Dari Tabel 7, dapat dilihat bahwa nilai mAP@0,5 pada data validasi mencapai 0,568 dan bahkan naik menjadi 0,573 pada data uji, dengan mAP@0,5:0,95 sebesar 0,420 dan 0,444. *Precision* di data uji meningkat tajam hingga 0,809 dengan *recall* 0,548, sehingga model tidak hanya lebih jarang salah deteksi, tetapi juga tetap mampu menemukan cukup banyak objek.

Tabel 8. Ringkasan metrik evaluasi - Mask/Seg (M)

Metrik	Validasi	Uji
mAP@0,5	0.559	0.564
mAP@0,5:0,95	0.256	0.263
Precision	0.486	0.803
Recall	0.639	0.541

Pada sisi segmentasi dari Tabel 8, mAP@0,5 (M) juga naik menjadi 0,559 (val) dan 0,564 (uji). Namun, mAP@0,5:0,95 (M) justru berada di angka 0,256 dan 0,263, sehingga kualitas bentuk *mask* pada IoU tinggi belum mengikuti peningkatan yang terjadi pada *bounding box*. Dengan kata lain, YOLOv11 lebih unggul dalam menemukan objek dan mengurungnya

dengan *box*, tetapi kontur *mask*-nya masih belum memiliki presisi yang memadai.

Tabel 9. AP per kelas (*Bounding Box*)

Kelas	AP@0,5:0,95 Val	AP@0,5:0,95 Uji
short_sleeved_shirt	0.695	0.679
long_sleeved_shirt	0.482	0.565
long_sleeved_outwear	0.433	0.522
short_sleeved_outwear	0.069	0.009

Tabel 10. AP per kelas (*Mask/Seg*)

Kelas	AP@0,5:0,95 Val	AP@0,5:0,95 Uji
short_sleeved_shirt	0.404	0.370
long_sleeved_shirt	0.317	0.359
long_sleeved_outwear	0.270	0.318
short_sleeved_outwear	0.033	0.004

Pada YOLOv11, *short_sleeved_shirt* tetap menjadi kelas yang paling mudah dikenali, dengan AP@0,5:0,95 *box* sebesar 0,695 di validasi dan 0,679 di uji. Menariknya, *long_sleeved_shirt* dan *long_sleeved_outwear* justru menunjukkan peningkatan yang cukup besar pada data uji (hingga 0,565 dan 0,522 untuk *box*), menandakan bahwa YOLOv11 lebih adaptif terhadap variasi pose dan *layering* pakaian lengan panjang, seperti yang ditunjukkan pada Tabel 9.

Untuk segmentasi *mask*, Tabel 10 menunjukkan nilai AP@0,5:0,95 di semua kelas berada pada kisaran yang lebih rendah (sekitar 0,27–0,40), dan *short_sleeved_outwear* kembali menjadi kelas paling lemah dengan nilai 0,033 → 0,004. Pola ini menguatkan kesan bahwa YOLOv11 sangat membantu untuk deteksi multikelas berbasis *box*, sementara detail tepi *mask* terutama pada objek kecil yang mengalami *occlusion*, masih cukup menantang.

4.4. Perbandingan YOLOv8 dan YOLOv11

Jika kedua model dibandingkan langsung, keunggulan YOLOv11 paling jelas terlihat pada metrik *bounding box*. Pada data uji, mAP@0,5 (BBox) meningkat dari 0,462 pada YOLOv8 menjadi 0,573 pada YOLOv11, dan mAP@0,5:0,95 naik dari 0,327 menjadi 0,444. Untuk segmentasi, mAP@0,5 (*Mask*) di data uji juga membaik dari 0,462 menjadi 0,564, meskipun mAP@0,5:0,95 (*Mask*) turun dari 0,331 menjadi 0,263. Secara praktis, hal ini menunjukkan bahwa YOLOv11 lebih bisa diandalkan untuk mendeteksi dan mengklasifikasikan pakaian secara cepat dan akurat, terutama jika fokus utama sistem adalah kepatuhan jenis lengan, bukan pengukuran bentuk area *mask* yang sangat presisi. Dalam konteks aplikasi *dress code*, peningkatan ini sudah cukup signifikan.

4.5. Ringkasan Kinerja Agregat

Dilihat secara keseluruhan, perbedaan antara hasil validasi dan uji pada kedua model masih dalam batas wajar. Pada YOLOv8, mAP@0,5 (Bbox) hanya

berubah sedikit dari 0,467 menjadi 0,462 dan mAP@0,5:0,95 dari 0,336 menjadi 0,327. *Precision* sedikit turun (0,684 → 0,631) tetapi *recall* justru naik (0,474 → 0,514), begitu juga pada *mask* yang tetap di sekitar 0,467–0,462 (mAP@0,5) dan 0,340–0,331 (mAP@0,5:0,95), dengan *recall* meningkat menjadi 0,530. Pada YOLOv11, pola pergeseran val–uji masih ada, namun level mAP-nya berada di atas YOLOv8. Kondisi ini menggambarkan adanya perbedaan distribusi antara data latih/validasi dan data uji (misalnya dari sisi pencahayaan, pose, atau jarak kamera), tetapi tidak sampai membuat model kehilangan kemampuan generalisasi.

4.6. Analisis per Kelas

Jika dilihat kinerja per kelas di kedua model, urutannya relatif konsisten: *short_sleeved_shirt* selalu menjadi kelas dengan skor tertinggi, diikuti *long_sleeved_shirt* dan *long_sleeved_outwear*, sementara *short_sleeved_outwear* hampir selalu berada di posisi terakhir. Pada YOLOv11, jarak antara tiga kelas teratas dan kelas terbawah sedikit menyempit untuk *bounding box*, tetapi masih cukup lebar pada *mask*. Hal ini sejalan dengan karakter data, di mana *short_sleeved_outwear* muncul lebih jarang, area objeknya lebih kecil, dan sering tertutup lapisan pakaian lain. Konsistensi peringkat ini dari validasi ke uji memperkuat dugaan bahwa penurunan kinerja lebih disebabkan oleh perubahan kondisi gambar secara umum daripada masalah label di satu kelas tertentu.

4.7. Sumber Error dan Mitigasi

Beberapa faktor teknis yang berkontribusi terhadap penurunan *recall* dan mAP@0,5:0,95 di data uji adalah (i) ketidakseimbangan kelas (terutama *short sleeved outwear*), (ii) *occlusion/layering* pada area lengan, (iii) variasi pencahayaan/pose, dan (iv) resolusi input yang membatasi detail objek kecil. Rekomendasi mitigasi dirangkum pada Tabel 11.

Tabel 11. Rekomendasi mitigasi *error*

Sumber Error	Dampak	Mitigasi
<i>Motion blur</i>	Menurunkan <i>confidence</i>	Tambah augmentasi <i>blur</i> , menaikkan <i>shutter</i>
Pencahayaan kontras	Salah segmentasi tepi	Normalisasi/HSV, <i>exposure lock</i>
Ketidakseimbangan kelas	AP kelas kecil rendah	<i>Reweighting</i> , <i>oversampling</i>

4.8. Analisis Komparatif dan Perspektif Arsitektur

Berdasarkan dokumentasi resmi Ultralytics dan sejumlah laporan eksperimen terbaru, YOLOv11 umumnya menawarkan akurasi (mAP) yang lebih baik sekaligus jumlah parameter yang lebih hemat dibanding YOLOv8 pada berbagai ukuran model dan tugas, khususnya pada *benchmark* COCO. Pada skala model menengah (m), YOLOv11 dilaporkan mampu

mencapai mAP COCO yang lebih tinggi dengan pengurangan parameter sekitar 20–30% dibanding YOLOv8m [11]. Secara arsitektural, YOLOv11 mengadopsi *backbone* dan *neck* yang telah diperbarui sehingga ekstraksi fitur menjadi lebih kaya tanpa harus memperbanyak jumlah parameter [12].

Dalam konteks deteksi kepatuhan pakaian pada subset DeepFashion2 yang banyak berisi objek berukuran sedang dengan detail tekstur lengan yang cukup penting, karakteristik YOLOv11 ini dapat diterjemahkan menjadi dua keuntungan utama:

- Potensi mAP yang lebih tinggi, terutama pada kelas dengan kontur lengan yang kompleks; dan
- Efisiensi parameter yang lebih baik, sehingga lebih mudah diintegrasikan ke perangkat *edge* seperti *mini PC* atau *embedded device* untuk implementasi *dress code compliance* secara *real-time*.

4.9. Implikasi pada Pipeline React-Flask

Karena Ultralytics menjaga konsistensi antarmuka pemanggilan model, migrasi dari YOLOv8 ke YOLOv11 pada *backend* Flask relatif sederhana:

- Mengganti *checkpoint* pelatihan dari YOLOv8s-seg.pt menjadi YOLO11s-seg.pt (hasil *fine-tuning* pada *dataset* yang sama).
- Menjaga format input (640×640, normalisasi) dan format output (bbox/mask, label, skor) tetap sama.
- Tidak ada perubahan pada kontrak API POST */predict* maupun logika *frontend* React.

Dengan demikian, *pipeline* yang telah dibangun dalam penelitian ini dapat dianggap “*model-agnostic*” selama model mengikuti standar keluaran Ultralytics. Hal ini memudahkan pengembangan selanjutnya di masa depan, misalnya menambahkan dukungan YOLOv11 sekaligus YOLOv8 dan membandingkan performa keduanya secara langsung pada log inferensi nyata (bukan hanya *dataset* uji).

4.10. Evaluasi Implementasi Sistem

Evaluasi implementasi sistem dilakukan dari perspektif kinerja praktis, kondisi fisik, privasi pengguna, dan potensi pengembangan fitur.

- Kinerja untuk Skenario Dress Code
Walaupun nilai mAP belum sangat tinggi, performa terbaik pada kelas *short_sleeved_shirt* dan *long_sleeved_shirt* udah memadai untuk skenario biner: membedakan lengan panjang vs lengan pendek. Kelas *outwear* yang masih sulit dideteksi dapat ditingkatkan bertahap melalui penyeimbangan *dataset* dan anotasi ulang.
- Pengaturan Lingkungan Fisik
Penataan posisi kamera, pencahayaan yang merata, serta instruksi sederhana kepada pengguna (misalnya berdiri tegak dan tangan di samping) dapat mengurangi *missed detection* tanpa perlu mengubah model. Hal ini krusial jika sistem digunakan sebagai alat bantu verifikasi *dress code* di pintu masuk gedung.

Atribut tambahan seperti warna, motif, atau jenis pakaian (formal vs kasual) dapat ditambahkan sehingga sistem tidak hanya mengecek panjang lengan, tetapi juga kesesuaian dengan standar kerapian dan profesionalitas.

5. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengembangkan prototipe sistem deteksi kepatuhan *dress code* berbasis web menggunakan arsitektur React-Flask yang terintegrasi dengan model YOLO. Hasil evaluasi komparatif menunjukkan bahwa YOLOv11 mengungguli YOLOv8 dalam tugas deteksi objek (*bounding box*) dengan peningkatan mAP@0.5 pada data uji mencapai 0,57 serta presisi yang signifikan sebesar 0,809. Meskipun YOLOv8 menunjukkan stabilitas yang sedikit lebih baik dalam pembentukan *mask* segmentasi pada IoU tinggi, YOLOv11 terbukti menjadi opsi yang lebih kompetitif untuk klasifikasi atribut pakaian secara cepat. Kendati demikian, sistem masih mengalami tantangan dalam mendeteksi kategori luaran (*outwear*) akibat ketidakseimbangan jumlah sampel data yang cukup ekstrem dibanding kelas kemeja.

Untuk pengembangan selanjutnya, disarankan agar penelitian berfokus pada penanganan ketidakseimbangan kelas minor tersebut melalui teknik *oversampling* atau strategi augmentasi data lanjutan guna meningkatkan akurasi deteksi *outwear*. Selain itu, pengujian performa sistem pada perangkat *edge* perlu dilakukan untuk memvalidasi efisiensi komputasi model dalam skenario operasional *real-time*. Integrasi fitur privasi, seperti penyensoran wajah otomatis (*face blurring*), juga direkomendasikan untuk menjamin keamanan identitas pengguna saat sistem diterapkan di lingkungan publik.

DAFTAR PUSTAKA

- [1] K. M. M. Uddin, S. K. Dey, G. U. Parvez, A. S. Mukta, and U. K. Acharjee, “MirrorME: implementation of an IoT based smart mirror through facial recognition and personalized information recommendation algorithm,” *International Journal of Information Technology*, vol. 13, no. 6, pp. 2313–2322, Dec. 2021, doi: 10.1007/s41870-021-00801-z.
- [2] R. Bhogi, R. Bade, H. K. Bhupathi, and U. T. Bantu, “Smart Mirror: Real-Time Try-On Experience Using Deep Learning Models,” *International Journal of Advanced Research in Science, Communication and Technology*, pp. 223–251, Nov. 2024, doi: 10.48175/IJARSCT-22440.
- [3] P. Manke, A. Bele, R. Rizvi, V. Naik, and H. Singh, “Virtual Personal Shopping Assistant (VSPA),” *INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, vol. 09, no. 03, pp. 1–9, Mar. 2025, doi: 10.55041/IJSREM42453.

- [4] M. Paolanti, R. Pietrini, A. Mancini, E. Frontoni, and P. Zingaretti, "Deep understanding of shopper behaviours and interactions using RGB-D vision," *Machine Vision and Applications*, vol. 31, no. 7–8, p. 66, Nov. 2020, doi: 10.1007/s00138-020-01118-w.
- [5] S. Aryanti, R. Arcanita, and K. Indrawari, "Etika Berpakaian Mahasiswa Dalam Perspektif Kontrak Perkuliahan dan Relevansinya Dengan Image Kampus Islami (Studi Kasus di IAIN Curup)," Bachelor Thesis, INSTITUT AGAMA ISLAM NEGERI (IAIN) CURUP, 2023. [Online]. Available: <http://e-theses.iaincurup.ac.id/id/eprint/4724>
- [6] A. S. Bashir, A. Ibrahim Mahmud, and M. Yahya, "Detection and Classification of Dress Code Violations in Educational Environments Using Deep Learning," *ABUAD Journal of Engineering Research and Development (AJERD)*, vol. 8, no. 2, pp. 88–95, June 2025, doi: 10.53982/ajerd.2025.0802.09-j.
- [7] J. Xiang, T. Dong, R. Pan, and W. Gao, "Clothing Attribute Recognition Based on RCNN Framework Using L-Softmax Loss," *IEEE Access*, vol. 8, pp. 48299–48313, 2020, doi: 10.1109/ACCESS.2020.2979164.
- [8] Q. Li and J. Wei, "Lightweight Real-time Detection Method for Dress Code of Anti-static Equipment," *Academic Journal of Computing & Information Science*, vol. 6, no. 10, 2023, doi: 10.25236/AJCIS.2023.061002.
- [9] J. Terven and D. Cordova-Esparza, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Machine Learning and Knowledge Extraction*, vol. 5, no. 4, pp. 1680–1716, Nov. 2023, doi: 10.3390/make5040083.
- [10] R. Sapkota, Z. Meng, M. Churuvija, X. Du, Z. Ma, and M. Karkee, "Comprehensive Performance Evaluation of YOLOv12, YOLO11, YOLOv10, YOLOv9 and YOLOv8 on Detecting and Counting Fruitlet in Complex Orchard Environments," 2024, *arXiv*. doi: 10.48550/ARXIV.2407.12040.
- [11] I. Vrsalovic, J. Lerga, and M. Ivacic-Kos, "A System for Real-Time Detection of Abandoned Luggage," *Sensors*, vol. 25, no. 9, p. 2872, May 2025, doi: 10.3390/s25092872.
- [12] S. Fang, Y. Shen, H. Zou, Y. Yin, W. Jin, and H. Zhou, "Birds-YOLO: A Bird Detection Model for Dongting Lake Based on Modified YOLOv11," *Biology*, vol. 14, no. 11, p. 1515, Oct. 2025, doi: 10.3390/biology14111515.
- [13] G. Hacheme and N. Sayouti, "Neural Fashion Image Captioning: Accounting for Data Diversity," July 01, 2021, *AfricArXiv*. doi: 10.31730/osf.io/hwtpq.
- [14] W. Jouanneau, A. Bugeau, M. Palyart, N. Papadakis, and L. Vezard, "Where are my clothes? A multi-level approach for evaluating deep instance segmentation architectures on fashion images," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Nashville, TN, USA: IEEE, June 2021, pp. 3946–3950. doi: 10.1109/CVPRW53098.2021.00443.