

Sistem Verifikasi Otomatis Atribut Foto Yudisium Menggunakan YOLO Dan OpenCV di UNESA

Febry Sanpratama¹, Yuni Yamasari²

^{1,2} Program Studi S1 Teknik Informatika, Universitas Negeri Surabaya

¹febrypratama@unesa.ac.id

³yuniyamasari@unesa.ac.id

Abstrak—Verifikasi foto yudisium di Universitas Negeri Surabaya (UNESA) saat ini masih dilakukan secara manual oleh staf akademik, yang menghadapi tantangan berupa volume berkas yang besar (mencapai 1.600 berkas per periode), waktu pemrosesan yang lama, serta potensi subjektivitas dan ketidakkonsistenan penilaian. Tujuan dari penelitian ini adalah untuk mengembangkan sebuah sistem verifikasi otomatis yang menggunakan teknologi computer vision dan deep learning untuk meningkatkan efisiensi, akurasi, dan kredibilitas validasi atribut foto resmi siswa.

Sistem yang dikembangkan mengintegrasikan algoritma YOLO (You Only Look Once) untuk deteksi objek dan OpenCV untuk pengolahan citra. YOLO digunakan untuk mengidentifikasi keberadaan serta posisi logo almamater dan penggunaan dasi, sementara OpenCV dimanfaatkan untuk memvalidasi kesesuaian warna latar belakang biru melalui analisis ruang warna HSV. Sistem ini dibangun menggunakan bahasa pemrograman Python dengan framework Flask sebagai layanan backend untuk memproses unggahan gambar. Penelitian ini menggunakan dataset sebanyak 2.000 gambar yang dibagi menjadi data training (80%), validation (10%), dan testing (10%).

Hasil pengujian menunjukkan bahwa model YOLO mencapai performa yang sangat tinggi dengan nilai rata-rata mAP50 sebesar 0,987 dan precision sebesar 0,978 pada data uji. Secara spesifik, kelas dasi mencapai nilai recall sempurna sebesar 1,000, yang berarti seluruh objek dasi dalam data uji berhasil dideteksi oleh model. Implementasi sistem ini terbukti mampu memberikan keputusan verifikasi "Sesuai" atau "Tidak Sesuai" secara otomatis dan konsisten, sehingga dapat menjadi solusi efektif dalam mempercepat proses administrasi akademik di lingkungan perguruan tinggi.

Kata Kunci—YOLO, OpenCV, Computer Vision, Foto Yudisium, Verifikasi Otomatis.

I. PENDAHULUAN

Yudisium merupakan tahapan penting dalam proses akademik yang menandai bahwa mahasiswa telah memenuhi persyaratan akademik dan administratif untuk dinyatakan lulus [1], [2]. Pada tahap ini, mahasiswa harus melengkapi beberapa dokumen administratif, salah satunya adalah foto resmi. Foto yudisium digunakan sebagai identitas mahasiswa dalam arsip akademik, dokumen kelulusan, dan kebutuhan administrasi lainnya. Oleh karena itu, foto yudisium harus memenuhi standar tertentu, seperti menggunakan latar belakang berwarna biru, mengenakan almamater resmi, memakai atribut yang diwajibkan, memiliki kualitas pencahayaan yang baik, serta tidak berada dalam kondisi cermin atau mirror.

Di Universitas Negeri Surabaya, proses verifikasi foto yudisium masih dilakukan secara manual oleh petugas

akademik. Petugas harus memeriksa setiap foto satu per satu untuk memastikan kesesuaian foto dengan ketentuan yang berlaku. Karena jumlah foto yang perlu diperiksa dalam satu waktu dapat mencapai sekitar 1.500 hingga 1.600 berkas, ini membutuhkan waktu yang cukup lama. Selain itu, pemeriksaan manual juga berpotensi menimbulkan subjektivitas karena hasil verifikasi bergantung pada ketelitian dan persepsi masing-masing petugas. Faktor kelelahan, perbedaan interpretasi, dan tingginya volume pekerjaan dapat menyebabkan hasil penilaian menjadi kurang konsisten.

Perkembangan teknologi kecerdasan buatan memberikan peluang untuk mengotomatisasi proses pemeriksaan foto. Salah satu bidang yang relevan adalah computer vision, yaitu bidang yang memungkinkan komputer menganalisis dan mengekstraksi informasi dari citra digital. Computer vision banyak digunakan untuk berbagai tugas, seperti klasifikasi citra, deteksi objek, segmentasi, dan pengenalan pola visual. Dalam konteks verifikasi foto yudisium, computer vision dapat dimanfaatkan untuk mengenali atribut visual pada foto mahasiswa secara otomatis.

Kemajuan computer vision didukung oleh perkembangan machine learning dan deep learning. Machine learning memungkinkan sistem mempelajari pola dari data dan menghasilkan prediksi berdasarkan data yang telah dipelajari sebelumnya [3], [4]. Deep learning merupakan bagian dari machine learning yang menggunakan jaringan saraf tiruan dengan banyak lapisan untuk mempelajari representasi data secara hierarkis. Pendekatan ini telah banyak digunakan dalam pengolahan citra karena mampu mengekstraksi fitur visual secara otomatis [5].

Convolutional Neural Network (CNN) merupakan salah satu arsitektur deep learning yang paling masif digunakan dalam pemrosesan citra. Arsitektur ini dirancang khusus untuk mengolah data bergambar, dengan kemampuan mempelajari representasi fitur visual secara hierarkis atau bertingkat. Berbagai penelitian telah membuktikan bahwa CNN mampu mendeteksi dan mengklasifikasikan objek visual dengan performa yang sangat baik. Putra dan Saputra menerapkan CNN untuk mendeteksi penggunaan masker pada gambar [6], sedangkan Maulana, Khairunisa, dan Mufidah menggunakan CNN untuk mendeteksi bentuk wajah [7]. Namun, pada kasus verifikasi foto yudisium, sistem tidak cukup hanya melakukan klasifikasi citra, tetapi juga perlu mengetahui keberadaan dan posisi atribut tertentu pada foto.

Object detection lebih sesuai digunakan dalam kasus ini karena mampu mendeteksi objek sekaligus menentukan lokasinya pada gambar. Di antara beragam teknik deteksi objek

yang tersedia, You Only Look Once (YOLO) menonjol sebagai salah satu metode yang paling banyak diadopsi [8]. Secara fundamental, YOLO merupakan pendekatan deteksi satu tahap (single-stage) yang mengintegrasikan proses identifikasi kelas (klasifikasi) dan penentuan lokasi spasial (lokalisasi) objek secara simultan dalam satu kali proses inferensi [9]. YOLO memiliki kemampuan untuk mendeteksi objek dengan cepat dan akurat. Penelitian Yanto, Aziz, dan Irmawati menunjukkan bahwa YOLOv8 dapat digunakan untuk mendeteksi penggunaan masker wajah dengan performa tinggi [10]. Selain itu, YOLO juga telah digunakan pada berbagai konteks deteksi objek lain, seperti deteksi kerusakan jalan [11].

Selain deteksi atribut berbasis objek, verifikasi foto yudisium juga membutuhkan validasi warna latar belakang. Warna latar belakang tidak berbentuk objek tertentu, melainkan area warna yang perlu dianalisis berdasarkan karakteristik piksel. Oleh karena itu, OpenCV digunakan untuk melakukan pengolahan citra berbasis warna. OpenCV mendukung berbagai proses pengolahan citra, termasuk konversi ruang warna, thresholding, dan analisis piksel. Penelitian Akbar, Suwondo, Ramadhan, dan Hasugian menunjukkan bahwa metode thresholding HSV dengan OpenCV dapat digunakan untuk mendeteksi warna dasar pada citra [12]. Ruang warna HSV juga banyak digunakan karena mampu merepresentasikan warna secara lebih intuitif melalui komponen hue, saturation, dan value [13]. Dan juga, OpenCV juga telah banyak diaplikasikan dalam deteksi atribut visual seperti masker dalam berbagai kondisi pencahayaan [14].

Berdasarkan permasalahan tersebut, penelitian ini mengembangkan sistem verifikasi otomatis atribut foto yudisium menggunakan kombinasi YOLO dan OpenCV. YOLO digunakan untuk mendeteksi atribut visual berupa logo almamater dan dasi, sedangkan OpenCV digunakan untuk memvalidasi kesesuaian warna latar belakang biru. Sistem dikembangkan menggunakan Python dan Flask sebagai backend agar dapat menerima unggahan foto, memproses gambar, dan mengembalikan hasil verifikasi kepada pengguna. Flask dipilih karena ringan dan dapat digunakan untuk membangun layanan API yang terintegrasi dengan model machine learning [15], [16].

Pengembangan sistem verifikasi otomatis yang memanfaatkan deteksi objek dan validasi warna untuk membantu proses administrasi akademik adalah kontribusi utama dari penelitian ini. Sistem ini diharapkan dapat mempercepat proses pemeriksaan foto, mengurangi beban kerja petugas, meningkatkan objektivitas penilaian, serta memberikan umpan balik yang lebih jelas kepada mahasiswa.

II. METODOLOGI PENELITIAN

Metodologi yang diterapkan dalam penelitian ini berbasis pada penggabungan paradigma deep learning dan visi komputer. Rangkaian eksperimen dan pengembangan dilakukan melalui tahapan yang meliputi: perumusan masalah dan telaah literatur, analisis kebutuhan serta pengumpulan dan anotasi dataset, proses pelatihan model YOLO, penerapan fungsi validasi latar menggunakan OpenCV, konstruksi

backend dengan Flask, dan evaluasi akhir terhadap sistem yang telah dibangun.

A. Identifikasi Masalah

Tahap identifikasi masalah dilakukan untuk memahami kendala pada proses verifikasi foto yudisium manual, seperti banyaknya jumlah foto, lamanya waktu pemeriksaan, dan potensi subjektivitas penilaian. Setelah itu, dilakukan studi literatur untuk mempelajari metode yang relevan, terutama deep learning, object detection, YOLO, OpenCV, dan pengolahan citra berbasis warna.

B. Studi Literatur

Untuk memastikan bahwa penelitian ini berjalan dengan benar dan mencapai tujuan yang telah ditetapkan, kajian teori ini dikumpulkan dari berbagai jurnal ilmiah.

C. Analisis Kebutuhan

Penelitian ini dilakukan dengan dukungan *environment* perangkat keras seperti prosesor Intel Core i5 atau AMD Ryzen 5, lalu dibantu dengan GPU Nvidia RTX 3050, kapasitas RAM 16 GB, dan penyimpanan sebesar 512 GB. GPU digunakan untuk mempercepat proses pelatihan model karena pelatihan deep learning membutuhkan kemampuan komputasi yang cukup tinggi.

Sistem operasi yang digunakan adalah Windows 11 atau Linux Ubuntu 22.04, bahasa pemrograman Python, framework Flask, pustaka OpenCV, serta framework pendukung implementasi YOLO. Python digunakan karena memiliki ekosistem yang luas untuk pengembangan machine learning dan computer vision. Flask digunakan sebagai backend untuk menerima unggahan foto, menjalankan proses inferensi, dan mengembalikan hasil verifikasi kepada pengguna.

D. Perancangan Sistem

1. Pengumpulan Dataset

Penelitian ini menggunakan foto yudisium mahasiswa Universitas Negeri Surabaya. Dataset yang digunakan berjumlah 2.000 gambar dan dibagi menjadi tiga bagian: pelatihan (*train*), validasi (*val*), dan pengujian (*testing*). Data training berjumlah 1.600 gambar atau 80% dari total dataset. Data validation berjumlah 200 gambar atau 10% dari total dataset. Data testing berjumlah 200 gambar atau 10% dari total dataset.

2. Pelatihan Model

Dataset yang digunakan dalam penelitian ini berupa foto yudisium mahasiswa Universitas Negeri Surabaya. Jumlah dataset yang digunakan adalah 2.000 citra. Dataset tersebut dibagi menjadi tiga bagian, yaitu data training, validation, dan testing. Data training berjumlah 1.600 gambar atau 80% dari total dataset. Data validation berjumlah 200 gambar atau 10% dari total dataset. Data testing berjumlah 200 gambar atau 10% dari total dataset.

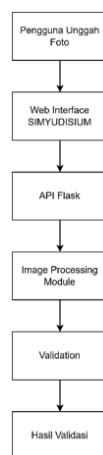
Setiap gambar diberi anotasi menggunakan *bounding box* untuk menandai lokasi objek. Anotasi ini

dilakukan terhadap kelas logo dan tali, dan label disimpan dalam format YOLO dengan informasi kelas objek dan koordinat *bounding box*. Struktur dataset disusun ke dalam folder images dan labels, masing-masing terdiri atas subfolder train, validation, dan test. Selain itu, file konfigurasi data.yaml digunakan untuk menentukan lokasi dataset dan nama kelas objek.

3. Perancangan Backend

Sistem dirancang sebagai layanan backend berbasis Flask. Alur sistem dimulai ketika pengguna mengunggah foto yudisium melalui aplikasi. Foto yang diunggah kemudian dikirimkan ke server untuk diproses. Server membaca gambar, menjalankan validasi latar belakang, melakukan inferensi menggunakan model YOLO, dan menentukan keputusan akhir berdasarkan aturan verifikasi.

4. Arsitektur Sistem



Gbr. 1 Flowchart Arsitektur Sistem

Arsitektur sistem pada penelitian digambarkan pada Gambar 1, ini dirancang untuk mengotomatisasi proses verifikasi atribut foto yudisium mahasiswa dengan memanfaatkan teknologi computer vision. Sistem terdiri dari sejumlah komponen utama yang saling berintegrasi, seperti modul untuk mengunggah gambar dan modul untuk mendeteksi objek dengan YOLO, modul analisis warna latar belakang menggunakan OpenCV, serta modul validasi hasil deteksi.

E. Pelatihan Model Yolo

Evaluasi sistem dilakukan pada dua aspek, yaitu evaluasi performa model dan evaluasi aplikasi. Evaluasi performa model dilakukan menggunakan metrik precision, recall, mAP50, dan mAP50-95. Precision digunakan untuk mengukur ketepatan model dalam mendeteksi objek. Recall digunakan untuk mengukur kemampuan model dalam menemukan objek yang sebenarnya ada. mAP50 digunakan untuk menilai performa model pada ambang IoU 0,50, sedangkan mAP50-95 digunakan untuk menilai performa model pada rentang IoU yang lebih ketat.

III. HASIL DAN PEMBAHASAN

Bab ini menguraikan hasil penerapan dan pengujian akurasi sistem verifikasi otomatis atribut foto yudisium.

A. Proses Persiapan Dataset



Gbr. 2 Pelabelan Objek pada Gambar

Persiapan dataset pada Gbr 2 adalah tahap pertama yang dilakukan sebelum proses pelatihan model. Untuk tujuan penelitian ini, gambar foto yudisium mahasiswa telah dikumpulkan dan dipilih sesuai dengan tujuan penelitian. Selanjutnya, foto – foto dilabeli berdasarkan elemen visual yang akan dideteksi, seperti logo almamater dan dasi.

Setelah dataset disiapkan, tahap berikutnya adalah pelatihan model YOLO. Pelatihan model dilakukan untuk membuat sistem mampu mengenali atribut visual yang terdapat pada foto yudisium. Proses pelatihan dilakukan dengan menggunakan dataset yang telah diberi label. Model YOLO akan mempelajari pola visual dari setiap kelas objek berdasarkan gambar dan label yang tersedia. Selama proses pelatihan, model akan melakukan pembaruan bobot secara bertahap agar hasil prediksi semakin mendekati label sebenarnya

B. Proses Pelatihan Model YOLO

Dalam penelitian ini, arsitektur YOLOv8 Nano (yolov8n.pt) digunakan sebagai dasar untuk pelatihan model deteksi objek. Pendekatan transfer learning digunakan untuk mengidentifikasi fitur logo almamater dan penggunaan dasi. Dataset yang digunakan berjumlah 2.000 citra, yang didistribusikan secara acak menjadi 80% data pelatihan (1.600 gambar), 10% data validasi (200 gambar), dan 10% data pengujian (200 gambar) dengan resolusi input citra sebesar 640 x 640 piksel.

Proses pelatihan dilakukan melalui empat skenario percobaan dengan variasi jumlah epoch (50 hingga 100) dan ukuran mini-batch (4 dan 8) untuk mengevaluasi performa deteksi serta efisiensi waktu komputasi. Untuk mengoptimalkan konvergensi gradien, digunakan Cosine Learning Rate Scheduler yang menyesuaikan nilai learning rate secara bertahap mengikuti kurva cosine hingga akhir pelatihan. Dan juga, mekanisme *Early Stopping* dengan nilai patience sebesar 30 epoch digunakan untuk mencegah overfitting saat pelatihan model.

Eksperimen dijalankan menggunakan akselerasi perangkat keras GPU dengan bantuan fitur Automatic Mixed Precision (AMP) untuk efisiensi memori, serta menonaktifkan augmentasi mosaik pada 15 epoch terakhir (close_mosaic=15) demi meningkatkan stabilitas deteksi akhir. Berdasarkan hasil perbandingan, percobaan ke – 4 (100 interval, ukuran batch 8) dipilih karena memiliki hasil terbaik, dengan capaian Precision

sebesar 0,95678 dan mAP50-95 sebesar 0,78902 dalam durasi pelatihan yang efisien, yaitu selama 2.344,92 detik.

C. Implementasi Deteksi Atribut Foto

Tahap deteksi atribut foto dilakukan dengan memanfaatkan model YOLOv8 yang telah melalui proses pelatihan pada tahap sebelumnya untuk mengidentifikasi atribut visual spesifik pada foto yudisium, yaitu logo almamater dan penggunaan dasi. Proses ini merupakan inti dari sistem verifikasi otomatis, di mana model melakukan inferensi terhadap citra yang diunggah untuk menghasilkan tiga informasi utama: lokasi objek dalam bentuk *bounding box*, nama kelas objek (logo atau tie), serta nilai *confidence score*.

Nilai *confidence score* menjadi parameter kritis dalam implementasi ini karena merepresentasikan tingkat keyakinan model terhadap objek yang terdeteksi. Dalam sistem yang dibangun, nilai ini digunakan sebagai ambang batas (*threshold*) minimal sebesar 0,7 untuk memastikan hasil deteksi valid dan akurat. Secara visual, hasil deteksi ini memungkinkan sistem untuk memverifikasi pemenuhan standar institusi secara objektif; misalnya, keberhasilan deteksi logo almamater pada posisi yang tepat menandakan kriteria atribut almamater telah terpenuhi.



Gbr. 3 Hasil Deteksi dari Model

Sebagaimana ditunjukkan pada Gambar 3, model mampu mengenali atribut logo dan dasi dengan tingkat keyakinan yang tinggi meskipun terdapat variasi dalam posisi atau ukuran objek pada foto uji. Output dari tahap deteksi ini kemudian diteruskan ke modul validasi aturan untuk digabungkan dengan hasil analisis latar belakang guna menghasilkan keputusan akhir verifikasi.

D. Implementasi Validasi Latar Belakang dengan OpenCV

Selain deteksi atribut menggunakan YOLO, sistem melakukan validasi warna latar belakang foto secara terpisah menggunakan pustaka OpenCV. Pendekatan pengolahan citra berbasis warna dipilih karena latar belakang bukan merupakan objek dengan bentuk geometri tertentu, melainkan area warna pada citra. Proses validasi ini dimulai dengan melakukan konversi ruang warna dari BGR ke HSV (Hue, Saturation, Value). Penggunaan ruang warna HSV memberikan stabilitas lebih tinggi dibandingkan RGB/BGR dalam menghadapi variasi intensitas cahaya, bayangan, maupun perbedaan kualitas sensor kamera, karena komponen jenis warna (Hue) dipisahkan dari intensitas kecerahan (Value).

Dalam implementasinya, sistem menetapkan Region of Interest (ROI) pada area 20% bagian atas citra sebagai representasi area latar belakang guna menghindari gangguan dari objek utama atau pakaian mahasiswa. Segmentasi warna biru dilakukan melalui teknik rule-based thresholding dengan rentang nilai Hue antara 90 hingga 130, Saturation 50-255, dan Value 50-255. Proses ini menghasilkan masker biner di mana piksel yang masuk dalam rentang warna biru akan diberi nilai aktif.

Hasil akhir validasi ditentukan berdasarkan nilai *blue_ratio*, yaitu persentase jumlah piksel biru terhadap total piksel pada area ROI. Sistem menetapkan ambang batas (*threshold*) sebesar 0,3 atau 30% sebagai nilai toleransi terhadap kemungkinan adanya bayangan, kompresi gambar, atau kualitas latar belakang yang tidak ideal. Jika nilai *blue_ratio* melebihi ambang batas tersebut, latar belakang dinyatakan memenuhi standar universitas (Sesuai), sedangkan jika di bawah ambang batas, sistem akan mengidentifikasi kesalahan "Background tidak sesuai" sebagai bagian dari keputusan akhir verifikasi.

E. Hasil Uji Coba

1. Perbandingan Pelatihan Model YOLO

Skenario pertama dilakukan untuk membandingkan performa pelatihan model YOLOv8n berdasarkan empat konfigurasi yang berbeda guna menentukan model yang paling optimal dalam mendeteksi atribut foto yudisium. Perbandingan ini dievaluasi melalui metrik *precision*, *recall*, mAP50, mAP50-95, serta efisiensi waktu pelatihan. Hasil eksperimen menunjukkan bahwa peningkatan jumlah epoch dari 50 menjadi 100 secara konsisten meningkatkan nilai mAP50-95, sementara penggunaan batch size yang lebih besar (batch 8) secara signifikan meningkatkan efisiensi waktu komputasi.

a. Formulasi Precision

$$\text{Precision} = \frac{TP}{TP + FP} \quad (1)$$

Keterangan komponen :

- 1) TP (*True Positive*) : jumlah objek yang sukses dideteksi dengan benar.
- 2) FP (*False Positive*) : jumlah objek yang diprediksi ada, namun sebenarnya tidak ada.

b. Formulasi Recall

$$\text{Precision} = \frac{TP}{TP + FN} \quad (2)$$

Keterangan komponen :

- 1) TP (*True Positive*) : jumlah objek yang sukses dideteksi dengan benar.
- 2) FP (*False Negative*) : jumlah objek yang sebenarnya ada tetapi tidak berhasil dideteksi oleh model.

c. Mean Average Precision (mAP)

Dalam penelitian deteksi objek, Mean Average Precision (mAP) adalah metrik evaluasi yang sering digunakan untuk mengukur performa keseluruhan model. Metrik ini dibuat berdasarkan rata-rata nilai Average Precision (AP) dari setiap kelas objek yang dideteksi.

TABEL I
UJI COBA PELATIHAN MODEL YOLO

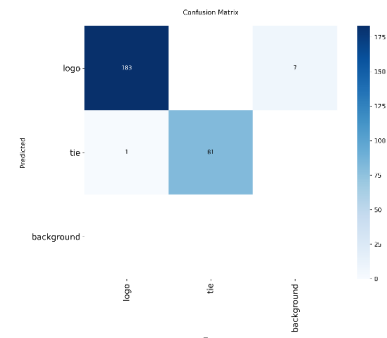
Percobaan	Waktu Training	Precision	Recall	mAP 50	mAP50-95	Keterangan
Percobaan 1 (Epoch 50, Batch 4)	2127,42 detik	0,94932	0,99248	0,97533	0,77998	Model dasar dengan pelatihan 50 epoch.
Percobaan 2 (Epoch 50, Batch 8)	1.236,04 detik	0,94909	0,99385	0,98009	0,77721	Paling efisien untuk 50 epoch dengan waktu tercepat.
Percobaan 3 (Epoch 100, Batch 4)	4599,17 detik	0,95453	0,98855	0,97841	0,78694	Durasi terlama (100 epoch) dengan peningkatan mAP50-95.
Percobaan 4 (Epoch 100, Batch 8)	2344,92 detik	0,95678	0,99107	0,97999	0,78902	Performa terbaik secara keseluruhan pada 100 epoch.

Berdasarkan data pada Tabel I, Percobaan 4 (100 epoch, batch 8) terpilih sebagai konfigurasi terbaik dengan nilai Precision tertinggi (0,95678) dan mAP50-95 (0,78902) dalam waktu pelatihan yang efisien sebesar 2.344,92 detik. Analisis visual melalui kurva loss pada Gambar 4.8 hingga 4.10 menunjukkan proses konvergensi yang stabil dengan adanya fenomena step drop di fase akhir pelatihan, yang menandakan optimasi parameter telah mencapai titik maksimal sebelum pelatihan berakhir. Secara keseluruhan, durasi 100 epoch memberikan ruang bagi model untuk mempelajari detail spasial objek secara lebih mendalam, sebagaimana terlihat pada grafik mAP50-95 yang mencapai fase plateau setelah epoch ke-80.

2. Pengujian Deteksi Atribut Foto

Skenario ini dilakukan untuk menguji generalisasi model dengan performa terbaik yang diperoleh dari Percobaan 4, fokus pada pendeteksian logo almamater dan dasi pada set data uji yang sama sekali belum dikenali oleh sistem selama fase pelatihan. Pengujian melibatkan 191 citra yang mencakup total 265 objek. Model menghasilkan nilai rerata Precision sebesar 0,978, rerata Recall sebesar 0,997, serta nilai mAP50 sebesar

0,987, yang mengindikasikan tingkat akurasi deteksi yang tergolong tinggi.



Gbr. 4 Confusion Matrix

$$\text{Precision Logo} = \frac{183}{183 + 7} = 0,963 \quad (3)$$

$$\text{Recall Logo} = \frac{183}{183 + 1} = 0,955 \quad (4)$$

$$\text{Precision Dasi} = \frac{81}{81 + 1} = 0,988 \quad (5)$$

$$\text{Recall Dasi} = \frac{81}{81 + 0} = 1 \quad (6)$$

Secara spesifik, kelas dasi mencapai performa sempurna dengan nilai Recall 1,000, yang berarti seluruh objek dasi dalam dataset pengujian berhasil dideteksi tanpa kesalahan. Pada kelas logo almamater, model menghasilkan Precision 0,963 dan Recall 0,995, menunjukkan tingkat ketepatan yang tinggi meski terdapat variasi ukuran dan posisi. Analisis Confusion Matrix pada Gambar 4 mengonfirmasi bahwa sebagian besar atribut diklasifikasikan dengan benar, membuktikan bahwa model memiliki keandalan yang sangat tinggi untuk digunakan sebagai komponen utama sistem verifikasi otomatis

3. Pengujian Aplikasi

```
{
  "success": false,
  "message": "Terdapat beberapa kesalahan pada foto.",
  "data": {
    "blue_ratio": 0.6841,
    "detected_objects": [],
    "logo_score": null,
    "tie_score": null,
    "total_objects": 0
  },
  "errors": [
    "Harap, mengenakan dasi yang sesuai.",
    "Tidak mengenakan almamater yang sesuai."
  ],
  "message": "Terdapat beberapa kesalahan pada foto.",
  "status": "error"
}
```

Pengujian aplikasi dilakukan untuk memastikan seluruh modul sistem, mulai dari unggah foto, deteksi YOLO, validasi latar belakang OpenCV, hingga pengambilan keputusan akhir, berjalan sesuai rancangan fungsi. Sistem secara otomatis mengintegrasikan hasil analisis spasial dan warna untuk memberikan keputusan "Sesuai" atau "Tidak Sesuai".

Gbr 5 menunjukkan bahwa aplikasi mampu memberikan umpan balik detail mengenai alasan penolakan foto. Sebagai contoh, melalui respons JSON pada sistem backend, jika model gagal mendeteksi atribut meskipun latar belakang telah sesuai (blue_ratio di atas ambang batas), sistem akan menghasilkan pesan kesalahan spesifik seperti "Harap mengenakan dasi yang sesuai" atau "Tidak mengenakan almamater yang sesuai". Implementasi ini membuktikan bahwa sistem dapat menggantikan peran pemeriksaan manual secara efektif, memberikan hasil verifikasi yang objektif, konsisten, dan transparan bagi pengguna.

F. Analisa Hasil

Analisa hasil bertujuan untuk mengevaluasi efektivitas sistem verifikasi foto yudisium dalam menjalankan fungsi otomatisasi yang telah dirancang, baik dari aspek performa model deteksi YOLO, akurasi validasi warna latar belakang melalui OpenCV, maupun integrasi sistem secara keseluruhan.

1. Analisa Hasil Pelatihan Model YOLO

Berdasarkan perbandingan empat skenario pelatihan, Percobaan 4 (100 epoch, batch 8) dianalisis sebagai konfigurasi yang paling optimal. Dengan Precision 0,95678, Recall 0,99107, dan mAP50-95 tertinggi 0,78902, model ini diambil karena memiliki performa terbaik. Nilai precision yang tinggi mengindikasikan bahwa model sangat akurat dalam memberikan prediksi positif dan meminimalkan false positives, sementara nilai recall yang mendekati angka 1 menunjukkan sensitivitas model yang luar biasa dalam menemukan objek target.

Analisa terhadap kurva loss menunjukkan bahwa model mengalami proses konvergensi yang stabil tanpa adanya indikasi divergence atau overfitting yang ekstrem. Fenomena step drop atau penurunan tajam nilai loss pada fase akhir pelatihan (sekitar epoch 85-90) dianalisis sebagai hasil dari optimasi parameter final yang efektif meningkatkan presisi spasial model sebelum pelatihan berakhir. Durasi 100 epoch terbukti memberikan ruang bagi model untuk mempelajari detail fitur yang lebih kompleks dibandingkan skenario 50 epoch, terutama terlihat pada peningkatan metrik mAP50-95.

2. Analisa Hasil Deteksi Atribut Foto

Pengujian pada data independen menunjukkan bahwa atribut Dasi merupakan kelas yang paling mudah dikenali dengan nilai Recall 1,000, yang berarti seluruh objek dasi berhasil ditemukan oleh model. Hal ini disebabkan oleh posisi dasi yang konsisten di tengah pakaian dan bentuk geometrisnya yang memanjang. Sementara itu, kelas Logo Almamater memiliki tantangan tersendiri dengan skor mAP50 sebesar 0,982.

Berdasarkan analisa confusion matrix, ditemukan kesalahan minor di mana terdapat area latar belakang yang terdeteksi sebagai logo (7 kasus) dan satu objek logo yang salah diklasifikasikan sebagai dasi. Kesalahan ini kemungkinan dipengaruhi oleh variasi pencahayaan,

kualitas gambar, atau ukuran logo yang relatif kecil pada citra masukan. Namun, secara keseluruhan, nilai F1-score maksimal sebesar 0,99 pada ambang batas kepercayaan 0,719 membuktikan bahwa model memiliki keandalan yang sangat tinggi untuk verifikasi atribut visual.

3. Analisa Hasil Validasi Latar Belakang

Analisa validasi latar belakang menunjukkan bahwa penggunaan ruang warna HSV melalui OpenCV jauh lebih stabil dibandingkan RGB dalam menghadapi variasi intensitas cahaya dan bayangan. Dengan memfokuskan analisis pada ROI 20% bagian atas citra, sistem berhasil meminimalisir gangguan dari objek utama (mahasiswa). Penetapan ambang batas blue_ratio sebesar 0,3 (30%) dianalisis sebagai nilai toleransi yang tepat untuk mengakomodasi degradasi kualitas warna akibat kompresi gambar atau pencahayaan yang tidak ideal, sehingga sistem tidak terlalu sensitif terhadap variasi minor namun tetap mampu menolak latar belakang yang salah.

4. Analisa Hasil Pengujian Aplikasi dan Kelebihan Sistem

Integrasi antara YOLO dan OpenCV memungkinkan sistem untuk memberikan keputusan verifikasi yang objektif dan konsisten dalam waktu singkat. Kelebihan utama sistem ini adalah kemampuannya memberikan umpan balik detail kepada pengguna, seperti mengidentifikasi apakah kesalahan terletak pada latar belakang, dasi, atau posisi logo (kondisi mirror). Melalui deteksi posisi tengah logo, sistem secara cerdas dapat mendeteksi foto yang terbalik (mirror) yang sering kali terlewatkan dalam pemeriksaan manual.

Secara keseluruhan, sistem ini mampu menggantikan subjektivitas penilaian manusia dengan aturan validasi yang baku, sehingga meningkatkan efisiensi administrasi akademik di lingkungan universitas. Meskipun memiliki keterbatasan pada ketergantungan terhadap kualitas input dan kelas objek yang telah dilatih, sistem ini telah memenuhi tujuan penelitian dalam mengotomatisasi verifikasi foto yudisium secara akurat.

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Hasil dari proses rancang bangun, implementasi, dan pengujian sistem verifikasi otomatis atribut foto yudisium dapat ditarik kesimpulan sebagai berikut:

1. Studi ini menghasilkan sistem verifikasi otomatis yang menggabungkan metode pembelajaran mendalam dengan algoritma YOLO untuk deteksi objek dan visi komputer dengan OpenCV untuk analisis warna. Sistem ini mampu memberikan keputusan verifikasi "Sesuai" atau "Tidak Sesuai" secara objektif berdasarkan standar Universitas Negeri Surabaya (UNESA).
2. Model YOLOv8 Nano menunjukkan performa yang sangat tinggi dalam mendeteksi atribut visual.

Konfigurasi terbaik diperoleh dari Percobaan 4 (100 epoch, batch 8) yang mencapai nilai rata-rata mAP50 sebesar 0,987, precision 0,978, dan recall 0,997 pada data pengujian. Secara khusus, atribut dasi mencapai nilai recall sempurna (1,000), menandakan seluruh objek dasi berhasil dideteksi tanpa ada yang terlewatkan.

3. Penggunaan pustaka OpenCV dengan ruang warna HSV terbukti efektif dalam memvalidasi warna latar belakang foto. Metode ini mampu memberikan hasil yang stabil meskipun terdapat variasi intensitas cahaya dengan menggunakan ambang batas blue ratio sebesar 0,3 pada area Region of Interest (ROI) bagian atas citra.
4. Implementasi aplikasi berbasis Flask ini terbukti dapat membantu mempercepat proses administrasi akademik dengan mengurangi subjektivitas dan beban kerja staf dalam memeriksa ribuan berkas foto yudisium setiap periodenya.

B. Saran

Beberapa ide yang dapat dipertimbangkan untuk penelitian lebih lanjut adalah:

1. Menambahkan jumlah serta variasi dataset foto/gambar untuk mencakup kondisi ekstrem seperti pencahayaan yang sangat rendah, kualitas gambar yang blur, atau variasi warna biru latar belakang yang lebih beragam agar model semakin tangguh dalam proses generalisasi.
2. Memperluas kelas deteksi untuk kriteria standar lainnya, seperti penggunaan kacamata, arah wajah frontal, serta penggunaan almamater secara utuh (tidak hanya logo) untuk meningkatkan cakupan validasi sistem.
3. Mengembangkan metode validasi latar belakang yang lebih adaptif, misalnya dengan memanfaatkan teknik segmentasi citra atau model klasifikasi warna berbasis deep learning untuk menggantikan pendekatan thresholding yang statis.
4. Menambahkan fitur deteksi wajah menggunakan MediaPipe untuk mencegah penggunaan manekin, serta mengintegrasikan identifikasi biometrik guna memastikan foto yang diunggah sesuai dengan identitas mahasiswa yang terdaftar di sistem akademik.
5. Melakukan integrasi langsung dengan sistem informasi yudisium universitas yang sudah berjalan agar proses verifikasi dapat dilakukan secara real-time saat mahasiswa melakukan unggahan dokumen.

UCAPAN TERIMA KASIH

Segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas limpahan rahmat, karunia, serta petunjuk-Nya, sehingga penelitian ini dapat terselesaikan dengan lancar. Penulis menyampaikan penghargaan dan terima kasih yang tulus kepada kedua orang tua atas doa, kasih sayang, dan dukungan moril yang senantiasa menyertai selama proses penyusunan penelitian. Ucapan terima kasih yang mendalam juga penulis sampaikan kepada dosen pembimbing, yang dengan penuh dedikasi telah memberikan bimbingan, arahan,

saran, serta motivasi di setiap tahap penelitian. Tak lupa, penulis mengucapkan terima kasih kepada rekan-rekan yang telah turut memberikan bantuan teknis, semangat, dan dukungan, sehingga terselesaikan dengan baik.

REFERENSI

- [1] Badan Pengembangan dan Pembinaan Bahasa, "Kamus Besar Bahasa Indonesia", Accessed: Dec. 16, 2025. [Online]. Available: <https://kbbi.web.id/yudisium>
- [2] R. Gupitha, "Perancangan Aplikasi Pengolahan Nilai Untuk Penentuan Yudisium Mahasiswa Berbasis Web," *Jurnal Global*, vol. VI, no. 1, 2019.
- [3] Firdaus Bagus Wicaksono and Y. Yamasari, "Pengembangan Model Pengawas Ujian Berbasis Kecerdasan Buatan untuk Ujian Online," *Journal of Informatics and Computer Science (JINACS)*, vol. 6, no. 03, 2025, doi: 10.26740/jinacs.v6n03.p882-890.
- [4] Aurélien Géron, *Hands-on Machine Learning with Scikit-Learn, Keras, and Hands-On Machine Learning TensorFlow*. 2022.
- [5] J. Heaton, "Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning," *Genet. Program. Evolvable Mach.*, vol. 19, no. 1–2, 2018, doi: 10.1007/s10710-017-9314-z.
- [6] D. R. R. Putra and R. A. Saputra, "IMPLEMENTASI CONVOLUTIONAL NEURAL NETWORK (CNN) UNTUK MENDETEKSI PENGGUNAAN MASKER PADA GAMBAR," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 11, no. 3, 2023, doi: 10.23960/jitet.v11i3.3286.
- [7] I. Maulana, N. Khairunisa, and R. Mufidah, "DETEKSI BENTUK WAJAH MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN)," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 6, 2024, doi: 10.36040/jati.v7i6.8171.
- [8] D. Nafis Alfarizi, R. Agung Pangestu, D. Aditya, M. Adi Setiawan, and P. Rosyani, "Penggunaan Metode YOLO Pada Deteksi Objek: Sebuah Tinjauan Literatur Sistematis," *Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan*, vol. 1, no. 1, 2023.
- [9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016. doi: 10.1109/CVPR.2016.91.
- [10] Y. Yanto, F. Aziz, and I. Irmawati, "YOLO-V8 PENINGKATAN ALGORITMA UNTUK DETEKSI PEMAKAIAN MASKER WAJAH," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 3, 2023, doi: 10.36040/jati.v7i3.7047.
- [11] L. Sakinah, E. Haryatmi, and T. A. Riyadi, "Implementasi Algoritma Yolo Untuk Mendeteksi Jalan Berlubang dan Retak," *Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 6, no. 3, 2025.
- [12] Zidanul Akbar, Asrul Suwondo, Rizky Ramadhan, and Abdul Halim Hasugian, "Deteksi Warna Dasar Menggunakan Metode Thresholding HSV dengan OpenCV," *Neptunus: Jurnal Ilmu Komputer Dan Teknologi Informasi*, vol. 3, no. 3, 2025, doi: 10.61132/neptunus.v3i3.1020.
- [13] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (4th ed)*. 2018.
- [14] A. Sopian, I. Kurniati, and E. N. Dewi, "FACE MASK DETECTION MENGGUNAKAN PYTHON DAN OPENCV UNTUK MENDETEKSI PELANGGARAN PROTOKOL KESEHATAN COVID-19," *JEIS : Jurnal Elektro dan Informatika Swadharma*, vol. 3, no. 1, 2023, doi: 10.56486/jeis.vol3no1.304.
- [15] P. N. Dacipta and R. E. Putra, "Sistem Klasifikasi Limbah Menggunakan Metode Convolutional Neural Network (CNN) Pada Webservice Berbasis Framework Flask," *Journal of Informatics and Computer Science (JINACS)*, vol. 3, no. 04, 2022, doi: 10.26740/jinacs.v3n04.p394-402.
- [16] C. Wijayanto and Y. A. Susetyo, "Implementasi Flask Framework Pada Pembangunan Aplikasi Sistem Informasi Helpdesk (SIH)," *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 7, no. 3, 2022, doi: 10.29100/jupi.v7i3.3161.