

**INOVASI MEDIA ICT DETEKSI WAJAH UNTUK PRESENSI DIGITAL
MAHASISWA: ANALISIS AKURASI DAN PERFORMA****Wawan Sismadi^{1*}, Abdul Manan², Estuti Fitri Hartini³**Program Studi Informatika, Universitas IPWIJA^{1,2,3}e-mail: wawansismadi@ipwija.ac.id^{1*}, abdmanan080@gmail.com²,estufitrihartini@gmail.com³e-mail koresponden: wawansismadi@ipwija.ac.id^{1*}

Diterima: 09/07/2026; Direvisi: 13/08/2026; Diterbitkan: 09/09/2026

ABSTRAK

Transformasi digital pendidikan tinggi menuntut inovasi media ICT yang mendukung tata kelola pembelajaran secara lebih akurat, efisien, dan nirkontak, khususnya dalam validasi kehadiran mahasiswa. Penelitian ini menganalisis akurasi dan performa dua pustaka deteksi wajah berbasis web, yaitu *OpenCV.js* dan *face-api.js*, sebagai media ICT presensi digital mahasiswa yang diproses sepenuhnya di sisi klien (*client-side*) tanpa transmisi data biometrik, sejalan dengan prinsip *privacy-by-design*. Metode yang digunakan adalah eksperimen laboratorium terkendali dengan desain faktorial 36 skenario: tiga tingkat pencahayaan (100, 300, 500 lux) × empat sudut pose (0°, 15°, 30°, 45°) × tiga kelas perangkat (*low-end*, *mid-range*, *high-end*), dengan 50 mahasiswa berusia 18–19 tahun sebagai partisipan. Hasil pengujian pada 180 observasi per pustaka menunjukkan bahwa *face-api.js* unggul dalam akurasi deteksi (*recall* 97,8%; *F1-Score* 98,9%) dibandingkan *OpenCV.js* (*recall* 81,7%; *F1-Score* 89,9%), sementara *OpenCV.js* unggul dalam kecepatan (latensi 163,5 ms; FPS-eq 6,1) dibandingkan *face-api.js* (269,5 ms; FPS-eq 3,7). Kedua pustaka mencatat *precision* 100% dalam skenario *closed-set* dan latensi jauh di bawah 500 ms, memvalidasi kelayakan teknis sebagai *Proof of Concept* TKT Level 3. Kedua jalur *client-side* juga 1,65–2,51 kali lebih cepat daripada baseline *server-side* (410,1 ms). Temuan ini berkontribusi pada pengembangan media ICT pembelajaran yang ringan, hemat biaya, dan berorientasi privasi bagi presensi digital di perguruan tinggi.

Kata Kunci: Media ICT Pembelajaran, Deteksi Wajah, Presensi Digital Mahasiswa**ABSTRACT**

The digital transformation of higher education demands ICT-based learning media innovations that support accurate, efficient, and contactless classroom governance, particularly in student attendance validation. This study analyzes the accuracy and performance of two web-based face detection libraries, *OpenCV.js* and *face-api.js*, as ICT media for student digital attendance processed entirely on the client side without biometric data transmission, in line with the *privacy-by-design* principle. The method employed is a controlled laboratory experiment with a 36-scenario factorial design: three lighting levels (100, 300, 500 lux) × four pose angles (0°, 15°, 30°, 45°) × three device tiers (*low-end*, *mid-range*, *high-end*), involving 50 student participants aged 18–19 years. Results across 180 observations per library show that *face-api.js* excels in detection accuracy (*recall* 97.8%; *F1-Score* 98.9%) compared to *OpenCV.js* (*recall* 81.7%; *F1-Score* 89.9%), while *OpenCV.js* excels in speed (latency 163.5 ms; FPS-eq 6.1) compared to *face-api.js* (269.5 ms; FPS-eq 3.7). Both libraries achieved 100% precision in a closed-set scenario with latency well below 500 ms, validating technical feasibility as a *Proof of Concept* at Technology Readiness Level 3. Both client-side pathways were 1.65–2.51 times

faster than the server-side baseline (410.1 ms). These findings contribute to lightweight, cost-effective, and privacy-oriented ICT learning media for digital attendance in higher education.

Keywords: *ICT learning media, face detection, student digital attendance*

PENDAHULUAN

Transformasi digital pada institusi pendidikan tinggi mendorong kebutuhan mendesak akan media ICT pembelajaran yang mampu mendukung tata kelola kelas secara akurat, nirkontak, dan tetap dapat beroperasi pada kondisi infrastruktur terbatas, termasuk dalam administrasi kehadiran mahasiswa yang menjadi bagian dari penilaian partisipasi dan manajemen pembelajaran di kelas. Metode presensi konvensional, baik berbasis kertas maupun sidik jari, mengandung dua kelemahan mendasar, yaitu rentan terhadap kecurangan akademik dalam bentuk presensi titip (*proxy attendance*) dan kurang higienis karena mengandalkan kontak fisik bersama. Teknologi pengenalan wajah (*face recognition*) berbasis *computer vision* hadir sebagai alternatif media ICT yang menjanjikan, namun mayoritas implementasi saat ini masih bersifat *server-side* yang membutuhkan infrastruktur mahal, transmisi data berkapasitas tinggi, dan berpotensi menimbulkan risiko privasi data biometrik mahasiswa (Kamil et al., 2023).

Kemajuan pustaka JavaScript seperti *OpenCV.js* (OpenCV.org, 2024) dan Justad (2023) membuka peluang pemrosesan biometrik dilakukan langsung di sisi klien (*client-side*), sehingga berpotensi menjadi media ICT pembelajaran yang lebih ringan dan hemat biaya bagi institusi dengan keterbatasan anggaran. Pendekatan ini masih menghadapi tantangan berupa penurunan akurasi pada pencahayaan rendah, sensitivitas terhadap variasi pose wajah, dan keterbatasan komputasi pada perangkat mahasiswa yang heterogen (Fang & Zhou, 2023; Sholi et al., 2024). Kesenjangan antara potensi teknis *client-side* dan bukti empiris kelayakannya pada kondisi nyata kampus Indonesia belum banyak dijawab oleh penelitian terdahulu (Sholi et al., 2024; Sismadi et al., 2025). Tinjauan literatur sistematis terhadap sistem presensi mahasiswa berbasis pengenalan wajah turut mengonfirmasi bahwa pemilihan algoritma deteksi (CNN dibandingkan LBPH) masih menjadi pertanyaan terbuka pada konteks kampus (Budiman et al., 2023), memperkuat urgensi pengujian komparatif seperti yang dilakukan pada penelitian ini.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan menganalisis akurasi dan performa pustaka *OpenCV.js* dan *face-api.js* berbasis *client-side* sebagai media ICT presensi digital mahasiswa pada berbagai kondisi pencahayaan, pose wajah, dan kelas perangkat. Secara spesifik, penelitian ini menjawab tiga pertanyaan berbasis kerangka PICOT yang diadaptasi dari konteks klinis ke konteks rekayasa perangkat lunak: (1) apakah algoritma *client-side* mampu mencapai *precision* minimal 90% pada perangkat *low-end* dalam pencahayaan kelas nyata (100–500 lux) dan variasi sudut wajah (0°–45°); (2) sejauh mana latensi *client-side* dapat dipertahankan di bawah 500 milidetik, ambang yang ditetapkan peneliti berdasarkan standar interaksi *real-time* antarmuka web (Nielsen, 1993; Goh et al., 2022), dibandingkan baseline *server-side*; dan (3) apakah sistem ini dapat dibuktikan konsepnya (*Proof of Concept*) hingga memenuhi standar Tingkat Kesiapterapan Teknologi (TKT) Level 3 (Kementerian Riset dan Teknologi/Badan Riset dan Inovasi Nasional, 2022). Perlu dicatat bahwa komponen "T" (Time) pada kerangka PICOT dalam penelitian ini tidak merujuk pada durasi *follow-up* klinis, melainkan pada ambang batas waktu respons sistem sebagai indikator kelayakan teknis. Seluruh tahapan pengujian ditargetkan terselesaikan dalam periode 12 bulan pelaksanaan hibah Penelitian Dosen Pemula (PDP) Tahun Anggaran 2026 (Mei 2026–April 2027).

Penelitian *face recognition* telah mencapai kematangan tinggi pada arsitektur seperti *FaceNet* dan *VGGFace* dengan akurasi melampaui 98% (K. Huang et al., 2024; Zhao et al.,

2024). Riset pada perangkat mobile native mulai berkembang untuk presensi (Ennajar & Bouarifi, 2024) maupun pengenalan ekspresi wajah *real-time* (Ulhaq et al., 2023), namun tetap membutuhkan instalasi aplikasi khusus. Pendekatan *privacy-by-design* melalui pemrosesan lokal di sisi pengguna juga mendapat perhatian (Y. Huang et al., 2024). Arsitektur pengenalan wajah yang dirancang khusus untuk perangkat edge, seperti *EdgeFace* (George et al., 2024), semakin menegaskan tren efisiensi komputasi ini, yang memperkuat relevansi arsitektur *client-side* bagi institusi pendidikan dengan keterbatasan anggaran infrastruktur.

Kebaruan penelitian ini terletak pada tiga hal: (1) kerangka *benchmarking* komparatif antara dua pustaka *client-side* serta terhadap baseline *server-side* yang belum tersedia dalam literatur media ICT pembelajaran lokal; (2) eksperimen multi-variabel terkendali (36 skenario) yang merepresentasikan kondisi kelas nyata Indonesia; dan (3) penerapan prinsip *privacy-by-design* sebagai arsitektur inti. Secara ringkas, tujuan penelitian ini adalah menganalisis akurasi dan performa *OpenCV.js* dan *face-api.js* sebagai media ICT presensi digital mahasiswa berbasis *client-side* tanpa transmisi data biometrik, sejalan dengan prinsip *privacy-by-design*. Penelitian ini merupakan kelanjutan rekam jejak penulis dalam analisis performa komparatif arsitektur web untuk sistem pembelajaran (Sismadi et al., 2022, 2025).

METODE PENELITIAN

Penelitian ini menggunakan metode eksperimen laboratorium terkendali dengan pendekatan kuantitatif berbasis kerangka *Design Science Research* dan model PICOT. Prototipe aplikasi web dikembangkan dan diuji menggunakan desain faktorial penuh: tiga tingkat pencahayaan (100, 300, 500 lux) × empat sudut pose wajah (0°, 15°, 30°, 45°) × tiga kelas perangkat (*low-end*, *mid-range*, *high-end*) = 36 skenario, masing-masing lima trial (180 observasi per pustaka).

Data primer dikumpulkan dari 50 mahasiswa Program Studi Informatika Universitas IPWIJA berusia 18–19 tahun dengan *informed consent*, menggunakan *purposive sampling*. Setiap partisipan menjalani *enrollment* 10 frame pada kondisi standar (500 lux, 0°). Pengujian menggunakan skenario *closed-set* (seluruh wajah yang diuji merupakan identitas terdaftar). Spesifikasi perangkat tercantum pada Tabel 1.

Tabel 1. Spesifikasi Kelas Perangkat Keras yang Digunakan dalam Pengujian

Kelas	Prosesor	RAM	Kamera Depan	OS
<i>Low-end</i>	MediaTek Helio G35	3 GB	5 MP (720p)	Android 12
<i>Mid-range</i>	Snapdragon 695 / i3 Gen-11	6 GB	8 MP (1080p)	Android 13 / Win 11
<i>High-end</i>	i7 Gen-13 / Apple M2	16 GB	12 MP (1080p+)	Win 11 / macOS

Dua pustaka diuji: (1) *OpenCV.js* dengan *Haar Cascade* + LBP + *Cosine Similarity* (*threshold* $\geq 0,85$); (2) *face-api.js* dengan *TinyFaceDetector* + 68 *landmarks* + *descriptor* 128-D + *Euclidean Distance* (*threshold* $\leq 0,6$). Keduanya menggunakan akselerasi WebGL. *Ground truth* ditetapkan berdasarkan identitas partisipan yang dijadwalkan per skenario.

Analisis statistik meliputi: uji *Shapiro-Wilk* untuk normalitas; uji independent samples t-test dengan konfirmasi *Mann-Whitney U*; *Three-way ANOVA* (pencahayaan × pose × perangkat); $\alpha = 0,05$; ukuran efek η^2 dan Cohen's d; CI 95%. Desain observasi berulang menyebabkan asumsi independensi tidak sepenuhnya terpenuhi, sehingga hasil diinterpretasikan sebagai indikasi arah efek (*directional evidence*). Selain itu, jumlah lima trial

per skenario ditetapkan berdasarkan pertimbangan kelayakan pelaksanaan eksperimen di laboratorium, bukan hasil *power analysis* formal, sehingga ukuran sampel per sel tergolong terbatas untuk mendeteksi efek berukuran kecil.

HASIL DAN PEMBAHASAN

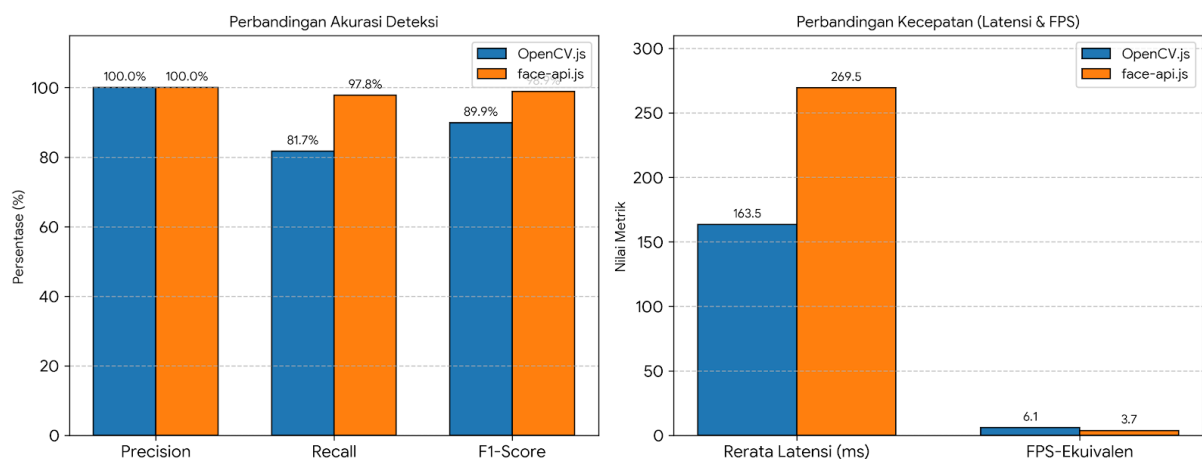
Hasil

Hasil pengujian menunjukkan adanya perbedaan karakteristik performa antara *OpenCV.js* dan *face-api.js* dalam mendukung deteksi wajah berbasis *client-side*. Perbandingan dilakukan pada 180 observasi untuk masing-masing pustaka dalam kondisi *closed-set*, dengan mempertimbangkan indikator akurasi dan kecepatan pemrosesan. Tabel 2 menyajikan perbandingan performa kedua pustaka berdasarkan indikator akurasi dan kecepatan pemrosesan. Perbedaan tersebut juga terlihat pada Gambar 1, yang menunjukkan bahwa *face-api.js* memiliki kinerja akurasi deteksi lebih tinggi, sedangkan *OpenCV.js* menghasilkan pemrosesan yang lebih cepat.

Tabel 2. Perbandingan Performa *OpenCV.js* dan *face-api.js* (n = 180; Closed-Set)

Metrik	OpenCV.js	face-api.js
<i>Precision</i>	1,000 (100%)	1,000 (100%)
<i>Recall</i>	0,817 (81,7%)	0,978 (97,8%)
<i>F1-Score</i>	0,899 (89,9%)	0,989 (98,9%)
Rerata Latensi (ms)	163,5	269,5
FPS-Ekuivalen	6,1	3,7

Tabel 2. Perbandingan Performa OpenCV.js dan face-api.js (n = 180; Closed-Set)



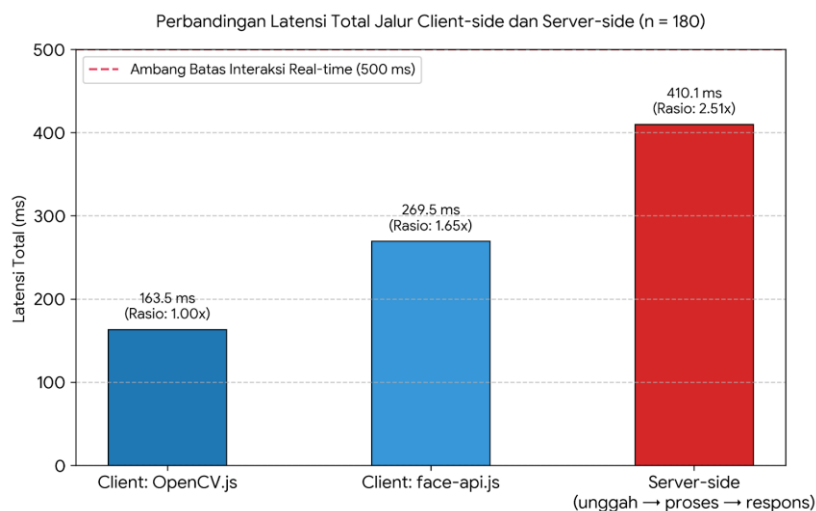
Gambar 1. Perbandingan Performa *OpenCV.js* dan *face-api.js*

Hasil pengujian menunjukkan adanya perbedaan karakteristik performa antara *OpenCV.js* dan *face-api.js*. Seperti ditunjukkan pada Tabel 2 dan Gambar 1, kedua pustaka memiliki tingkat *precision* yang sama, tetapi menunjukkan perbedaan pada kelengkapan deteksi dan kecepatan pemrosesan. *face-api.js* menghasilkan *recall* sebesar 97,8% dan *F1-Score*

sebesar 98,9%, lebih tinggi dibandingkan *OpenCV.js* yang masing-masing mencapai 81,7% dan 89,9%. Sebaliknya, *OpenCV.js* memiliki rerata latensi lebih rendah, yaitu 163,5 ms dibandingkan 269,5 ms pada *face-api.js*, serta FPS-ekuivalen yang lebih tinggi, yaitu 6,1 dibandingkan 3,7. Dengan demikian, *face-api.js* lebih unggul dalam kelengkapan dan konsistensi deteksi, sedangkan *OpenCV.js* menunjukkan keunggulan pada aspek kecepatan pemrosesan. Kedua pustaka juga mencatat *precision* sebesar 100%, yang menunjukkan tidak ditemukannya *false positive* dalam skenario *closed-set*. Selanjutnya, untuk melihat posisi kinerja kedua pendekatan tersebut dibandingkan dengan pemrosesan berbasis *server-side*, perbandingan latensi total disajikan pada Tabel 3 dan divisualisasikan pada Gambar 2.

Tabel 3. Perbandingan Latensi Total Jalur *Client-side* dan *Server-side* (n = 180)

Jalur	Latensi (ms)	Selisih (ms)	Rasio
<i>Client OpenCV.js</i>	163,5	0,0	1,00
<i>Client face-api.js</i>	269,5	105,9	1,65
<i>Server</i> (unggah → proses → respons)	410,1	246,6	2,51



Gambar 2. Perbandingan Latensi Total Jalur *Client-side* dan *Server-side*

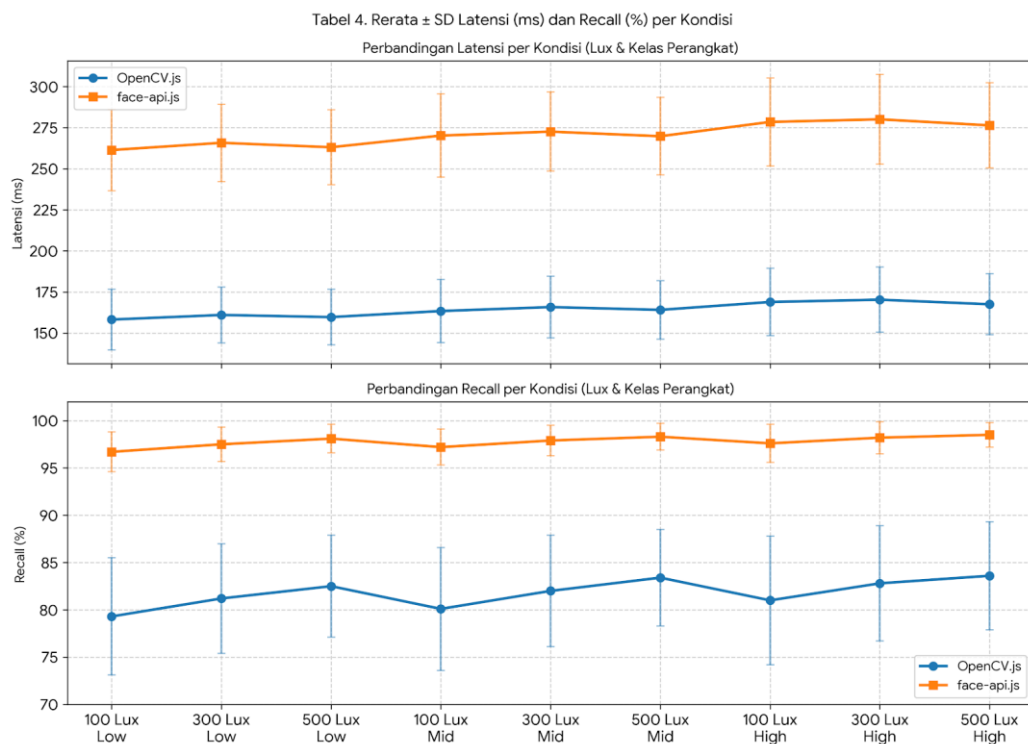
Berdasarkan Tabel 3 dan Gambar 2, uji t-test menunjukkan perbedaan signifikan pada latensi ($t(358) = -14,72$; $p < 0,001$; *Cohen's d* = 1,55; *Mann-Whitney U* = 4.231) dan *recall* ($t(358) = -9,84$; $p < 0,001$; *Cohen's d* = 1,04; *U* = 3.876). CI 95% selisih latensi: [91,8; 120,2] ms; CI 95% selisih *recall*: [0,101; 0,221]. Uji *Shapiro-Wilk* menunjukkan distribusi tidak sepenuhnya normal ($W = 0,94$ dan $0,93$; $p < 0,01$), sehingga *Mann-Whitney U* digunakan sebagai konfirmasi, kedua uji menghasilkan kesimpulan konsisten.

Three-way ANOVA menunjukkan: pada *OpenCV.js*, sudut pose berpengaruh signifikan terhadap *recall* ($F(3, 168) = 12,41$; $p < 0,001$; $\eta^2 = 0,18$), sementara pencahayaan ($p = 0,157$) dan kelas perangkat ($p = 0,393$) tidak signifikan. Pada *face-api.js*, tidak ada faktor yang signifikan terhadap *recall* (seluruh $p > 0,05$). Untuk latensi, kelas perangkat tidak signifikan pada kedua pustaka; sudut pose berpengaruh kecil pada *face-api.js* ($F(3, 168) = 3,21$; $p = 0,024$;

$\eta^2 = 0,05$). Pola perbedaan nilai *recall* dan latensi pada setiap kondisi pengujian dapat dilihat pada Gambar 3, yang memvisualisasikan data yang dirangkum dalam Tabel 4. Tabel 4 menyajikan rincian per kondisi ($n = 20$ per sel).

Tabel 4. Rerata $\pm$ SD Latensi (ms) dan Recall (%) per Kondisi

Lux	Perangkat	Latensi OpenCV.js	Latensi face-api.js	Recall OpenCV.js	Recall face-api.js
100	Low-end	158,2 $\pm$ 18,4	261,4 $\pm$ 24,7	79,3 $\pm$ 6,2	96,7 $\pm$ 2,1
300	Low-end	161,0 $\pm$ 17,1	265,8 $\pm$ 23,5	81,2 $\pm$ 5,8	97,5 $\pm$ 1,8
500	Low-end	159,7 $\pm$ 16,9	263,1 $\pm$ 22,8	82,5 $\pm$ 5,4	98,1 $\pm$ 1,5
100	Mid-range	163,4 $\pm$ 19,2	270,2 $\pm$ 25,3	80,1 $\pm$ 6,5	97,2 $\pm$ 1,9
300	Mid-range	165,8 $\pm$ 18,7	272,6 $\pm$ 24,1	82,0 $\pm$ 5,9	97,9 $\pm$ 1,6
500	Mid-range	164,1 $\pm$ 17,8	269,8 $\pm$ 23,6	83,4 $\pm$ 5,1	98,3 $\pm$ 1,4
100	High-end	168,9 $\pm$ 20,5	278,5 $\pm$ 26,8	81,0 $\pm$ 6,8	97,6 $\pm$ 2,0
300	High-end	170,3 $\pm$ 19,8	280,1 $\pm$ 27,2	82,8 $\pm$ 6,1	98,2 $\pm$ 1,7
500	High-end	167,5 $\pm$ 18,6	276,4 $\pm$ 25,9	83,6 $\pm$ 5,7	98,5 $\pm$ 1,3



Gambar 3. Rerata $\pm$ SD Latensi (ms) dan Recall (%) per Kondisi

Sesuai dengan Tabel 4 dan Gambar 3, pola performa konsisten lintas skenario. Seluruh latensi jauh di bawah 500 ms. *Recall* OpenCV.js (79,3–83,6%) lebih rendah dan lebih bervariasi dibandingkan face-api.js (96,7–98,5%). Kasus *false negative* pada OpenCV.js terkonsentrasi di pose 30°–45° pada 100 lux, di mana Haar Cascade gagal mendeteksi wajah secara konsisten. Pola ini mengindikasikan bahwa peningkatan kelas perangkat tidak serta-merta memperbaiki *recall* OpenCV.js pada sudut pose ekstrem, sehingga perbaikan pada sisi algoritma deteksi lebih relevan dibandingkan sekadar peningkatan spesifikasi perangkat keras.

Pembahasan

Hasil pengujian menunjukkan bahwa kedua pustaka *client-side* layak sebagai media ICT presensi digital mahasiswa, namun dengan profil keunggulan komplementer. *Face-api.js* direkomendasikan untuk konteks yang mengutamakan akurasi (*recall* 97,8%), sementara *OpenCV.js* sesuai untuk konteks yang mengutamakan kecepatan pada perangkat sangat terbatas (163,5 ms; FPS-eq 6,1). Pemilihan pustaka yang tepat dengan demikian perlu disesuaikan dengan prioritas operasional program studi, apakah menitikberatkan pada minimnya kesalahan deteksi atau pada kecepatan respons di kelas dengan perangkat mahasiswa yang heterogen. Perbedaan kedua profil kinerja ini juga terbukti signifikan secara statistik sebagaimana diuraikan pada bagian Hasil, sehingga bukan sekadar variasi acak antar pustaka.

Dari perspektif inovasi media pembelajaran berbasis ICT, temuan ini memberikan dua implikasi praktis. Pertama, penghapusan transmisi data ke server menghilangkan biaya infrastruktur dan bandwidth berulang, menurunkan hambatan adopsi bagi program studi dengan keterbatasan anggaran. Kedua, pemrosesan biometrik eksklusif di perangkat pengguna sejalan dengan prinsip *privacy-by-design* (Y. Huang et al., 2024; Song et al., 2025), relevan dengan etika perlindungan data pribadi mahasiswa.

Perbedaan arsitektur algoritmik menjadi penjelasan utama divergensi performa. OpenCV.js mengandalkan *Haar Cascade* (*handcrafted feature*) yang ringan namun sensitif terhadap variasi pose dan pencahayaan. Face-api.js menggunakan *TinyFaceDetector* berbasis *deep learning* yang lebih *robust*, namun membutuhkan komputasi inferensi lebih berat. Temuan ini sejalan dengan karakteristik umum *handcrafted* versus *learned features* (Zhao et al., 2024).

Mengenai latensi *high-end* yang sedikit lebih tinggi, kemungkinan penyebabnya adalah resolusi kamera bawaan yang dimuat browser (1080p+ pada *high-end* vs 720p pada *low-end*, sesuai Tabel 1), sehingga matriks piksel yang diproses lebih besar. Namun, interpretasi ini belum dapat dikonfirmasi definitif karena GPU load dan waktu pra-pemrosesan tidak diukur terpisah. Variasi performa inferensi lintas perangkat semacam ini konsisten dengan temuan pengukuran skala besar terhadap *inferensi deep learning di browser*, yang melaporkan disparitas performa signifikan antar perangkat akibat perbedaan kapasitas hardware dan karakteristik pustaka Web (Wang et al., 2025).

Precision 100% perlu dikontekstualisasikan: karena pengujian menggunakan skenario *closed-set*, tidak adanya *false positive* merupakan temuan yang diharapkan secara desain. Temuan ini belum dapat digeneralisasikan sebagai bukti ketahanan terhadap penyusup. Pengujian *open-set* dengan sampel non-terdaftar diperlukan untuk mengevaluasi *False Acceptance Rate* (FAR) secara sesungguhnya.

Dibandingkan penelitian sebelumnya yang membandingkan framework MVC pada sistem kehadiran (Sismadi et al., 2025) serta framework *CodeIgniter*, *Laravel*, dan *Ktupad* pada aplikasi ujian daring (Sismadi et al., 2022), penelitian ini memperluas metodologi *benchmarking* dari arsitektur aplikasi web menuju performa inferensi AI pada media pembelajaran berbasis ICT.

Beberapa keterbatasan perlu dicatat. Pertama, sampel homogen (50 mahasiswa, satu prodi, satu institusi, usia 18–19 tahun) sehingga generalisasi terbatas. Kedua, asumsi independensi tidak sepenuhnya terpenuhi karena observasi berulang; *mixed-effects model* direkomendasikan untuk lanjutan. Ketiga, jumlah trial berdasarkan pertimbangan praktis, bukan *power analysis*. Keempat, belum ada pengujian *spoofing/liveness detection*. Kelima, belum diuji pada kondisi *concurrent access* (Oroceo et al., 2022). Keenam, skenario *closed-set* saja.

KESIMPULAN

Berdasarkan hasil pengujian dan analisis statistik, penelitian ini menjawab tiga pertanyaan penelitian yang diajukan. Pertama, terkait akurasi, *face-api.js* terbukti melampaui ambang *recall* 90% yang ditargetkan (*recall* 97,8%; *F1-Score* 98,9%), sedangkan *OpenCV.js* berada di bawah ambang tersebut (*recall* 81,7%) akibat sensitivitas algoritma terhadap variasi pose wajah. Kedua, terkait latensi, kedua jalur *client-side* beroperasi secara *real-time* dengan latensi masing-masing 269,5 ms (*face-api.js*) dan 163,5 ms (*OpenCV.js*), jauh di bawah ambang 500 ms, serta 1,65–2,51 kali lebih cepat dibandingkan baseline *server-side* (410,1 ms); perbedaan latensi dan *recall* antar pustaka signifikan secara statistik ($p < 0,001$; Cohen's $d > 1,0$). Ketiga, terkait kelayakan konsep, sistem ini terbukti secara teknis (*Proof of Concept*) dan memenuhi indikator TKT Level 3 untuk skenario *closed-set*, sehingga pemrosesan biometrik di sisi klien layak dipertimbangkan sebagai media ICT presensi digital mahasiswa dengan profil keunggulan yang saling melengkapi antara akurasi (*face-api.js*) dan kecepatan pada perangkat terbatas (*OpenCV.js*).

Implikasi praktis dari temuan ini menunjukkan bahwa adopsi media ICT berbasis *client-side* berpotensi menurunkan kebutuhan infrastruktur dan biaya *bandwidth* bagi perguruan tinggi, sekaligus menjaga privasi data biometrik mahasiswa melalui prinsip *privacy-by-design*. Penelitian lanjutan diarahkan pada: (1) *liveness detection*; (2) pengujian *open-set*; (3) optimalisasi *recall OpenCV.js*; (4) perluasan sampel lintas institusi; dan (5) uji coba kelas nyata skala besar.

DAFTAR PUSTAKA

- Budiman, A., Fabian, Yaputera, R. A., Achmad, S., & Kurniawan, A. (2023). Student attendance with face recognition (LBPH or CNN): Systematic literature review. *Procedia Computer Science*, 216, 31–38. <https://doi.org/10.1016/j.procs.2022.12.108>
- Ennajar, S., & Bouarifi, W. (2024). Deep transfer learning approach for student attendance system during the COVID-19 pandemic. *Journal of Computer Science*, 20(3), 229–238. <https://doi.org/10.3844/jcssp.2024.229.238>
- Fang, Z., & Zhou, Z. (2023). Studies advanced in robust face recognition under complex light intensity. Dalam *Proceedings of the 2023 International Conference on Image, Algorithms and Artificial Intelligence (ICIAAI)* (pp. 1005–1012). https://doi.org/10.2991/978-94-6463-300-9_101
- George, A., Ecabert, C., Otroshi Shahreza, H., Kotwal, K., & Marcel, S. (2024). EdgeFace: Efficient face recognition model for edge devices. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 6(2), 158–168. <https://doi.org/10.1109/TBIOM.2024.3352164>
- Goh, H.-A., Ho, C.-K., & Abas, F. S. (2022). Front-end deep learning web apps development and deployment: A review. *Applied Intelligence*, 53(12), 15923–15945. <https://doi.org/10.1007/s10489-022-04278-6>



- Huang, K., Teng, Y., Chen, Y., & Wang, Y. (2024). From pixels to principles: A decade of progress and landscape in trustworthy computer vision. *Science and Engineering Ethics*, 30, 26. <https://doi.org/10.1007/s11948-024-00480-6>
- Huang, Y., Wu, Z., Chen, J., & Xiang, H. (2024). Privacy-preserving face recognition method based on randomization and local feature learning. *Journal of Imaging*, 10(3), 59. <https://doi.org/10.3390/jimaging10030059>
- Justad, V. M. (2023). face-api.js: A JavaScript face recognition API for the browser [Perangkat lunak komputer]. GitHub. <https://github.com/justadudewhohacks/face-api.js>
- Kamil, M. H. M., Zaini, N., Mazalan, L., & Ahamad, A. H. (2023). Online attendance system based on facial recognition with face mask detection. *Multimedia Tools and Applications*, 82(22), 34437–34457. <https://doi.org/10.1007/s11042-023-14842-y>
- Kementerian Riset dan Teknologi/Badan Riset dan Inovasi Nasional. (2022). Pedoman penyusunan Tingkat Kesiapterapan Teknologi (TKT). Kemenristek/BRIN.
- Nielsen, J. (1993). *Usability engineering*. Morgan Kaufmann.
- OpenCV.org. (2024). OpenCV.js: Computer vision library for the web. https://docs.opencv.org/4.x/d5/d10/tutorial_js_root.html
- Oroceo, P. P., Kim, J.-I., Caliwag, E. M. F., Kim, S.-H., & Lim, W. (2022). Optimizing face recognition inference with a collaborative edge–cloud network. *Sensors*, 22(21), 8371. <https://doi.org/10.3390/s22218371>
- Sholi, R. T., Sarker, M. F. H., Sohel, M. S., Islam, M. K., Tamal, M. A., Bhuiyan, T., Shakil, S. M. K. A., & Ahmed, M. F. (2024). Application of computer vision and mobile systems in education: A systematic review. *International Journal of Interactive Mobile Technologies*, 18(1), 168–187. <https://doi.org/10.3991/ijim.v18i01.46483>
- Sismadi, W., Indra, Martono, B. A., & Widyastuti, T. (2022). Comparative analysis of Codeigniter, Laravel and Ktupad frameworks: Case study online exam applications. *Indonesian Journal of Applied Research*, 3(3), 207–219. <https://iojs.unida.info/index.php/IJAR/article/view/236>
- Sismadi, W., Martono, B. A., Susanto, Y., & Muzaeni, A. (2025). Perbandingan performa framework MVC dalam sistem kehadiran berbasis pengenalan wajah. *EDUTECH: Jurnal Inovasi Pendidikan Berbantuan Teknologi*, 5(1), 1–7. <https://doi.org/10.51878/edutech.v5i1.4405>
- Song, Z., Wang, G., Yang, W., Li, Y., Yu, Y., Wang, Z., Zheng, X., & Yang, Y. (2025). Privacy-preserving method for face recognition based on homomorphic encryption. *PLOS ONE*, 20(2), e0314656. <https://doi.org/10.1371/journal.pone.0314656>
- Ulhaq, M. R. D., Zaidan, M. A., & Firdaus, D. (2023). Pengenalan ekspresi wajah secara real-time menggunakan metode SSD Mobilenet berbasis Android. *Journal of Technology Informatics (JoTI)*, 5(1), 48–52. <https://doi.org/10.37802/joti.v5i1.387>
- Wang, Q., Jiang, S., Chen, Z., Cao, X., Li, Y., Li, A., Ma, Y., Cao, T., & Liu, X. (2025). Anatomizing deep learning inference in Web browsers. *ACM Transactions on Software Engineering and Methodology*, 34(2), Article 47, 1–43. <https://doi.org/10.1145/3688843>
- Zhao, X., Wang, L., Zhang, Y., Han, X., Deveci, M., & Parmar, M. (2024). A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*, 57, 99. <https://doi.org/10.1007/s10462-024-10721-6>