AKAL IMITASI
Journal of Utilities, Contributions, and Impacts for AI-Driven Futures

https://journals.bim.ac.id/index.php/ai

CAT TKA–AI Proctoring: Real-Time Multimodal Integrity Monitoring for Computer-Assisted Academic Tests

Adi Jufriansah[1], Harun Joko Prayitno[2]*, Abdul Mu'ti[3], Toni Toharudin[4], Siwi Manganti[5], Fahmi Johan Syah[6], Carnaval Rego Rio Sidabutar[7], Rabiah Abdul Kadir[8]

[1]Faculty of Informatics and Applied Science, Bali International Muhammadiyah University, Indonesia
[2,6]Faculty of Teacher and Training Education, Universitas Muhammadiyah Surakarta, Indonesia
[3]Faculty of Tarbiyah, Universitas Islam Negeri Jakarta, Indonesia
[4]Faculty of Mathematics and Natural Science, Universitas Padjajaran, Indonesia
[5,7]Faculty of Business and Law, Bali International Muhammadiyah University, Indonesia
[8]Institute of Visual Informatics, University Kebangsaan Malaysia, Malaysia

DOI: xxxxx

History

Received:
March 23th, 2026

Revised:
April 3rd, 2026

Accepted:
April 10th, 2026

Available Online:
April 16th, 2026

Published Regularly:
April 2026

Abstract

Indonesia's Tes Kemampuan Akademik (TKA) and the school try-outs that prepare students for it are increasingly delivered as computer-assisted tests (CAT), yet browser-based delivery cannot, by itself, verify who is sitting the test or what happens outside the browser window. This study develops CAT TKA–AI Proctoring, an architecture that couples the CAT Tryout TKA platform of Universitas Bali Internasional Muhammadiyah (cat.bim.ac.id) with a real-time multimodal proctoring service. We adopt a design science perspective, linking the session layer (tokens, server-side timing, autosave, tab-switch logging, and item analysis) by attempt identifier to seven vision modules: iris-based gaze estimation, mouth-aspect-ratio speech cues, Perspective-n-Point head pose, ArcFace identity verification, YOLO11 object detection, four-cue anti-spoofing, and DPT monocular depth. Module outputs are fused into a confidence score with warm-up, persistence windows, penalty weighting, and recovery and reported to invigilators alongside CAT telemetry. On an Apple M4 Pro, the pipeline sustained over 10 processed frames per second with all modules active through staggered scheduling; deterministic simulation showed no flags under compliant behavior, bounded score dips under intermittent aversion, and a flag 8.2 s after the onset of sustained phone use. The architecture provides schools an explainable, human-in-the-loop path to integrity assurance in computer-assisted testing.

Keywords: AI proctoring; akal imitasi; artificial intelligence; computer-assisted test; multimodal deep learning; routine institutional infrastructure; Tes Kemampuan Akademik

This is an open access article under the CC-BY license.

✉***Corresponding Author:***
Harun Joko Prayitno, Faculty of Teacher and Training Education, Universitas Muhammadiyah Surakarta
Email: harun.prayitno@ums.ac.id

1. Introduction

The COVID-19 pandemic accelerated the digitalization of public services, work, and education to an extent that is now regarded as structural rather than temporary (Grinin et al., 2022; Amankwah-Amoah et al., 2021; Döhring et al., 2021). Assessment followed the same trajectory: computer-delivered and remotely supervised examinations moved from niche practice to routine institutional infrastructure, and with that move came a sharp rise in concern about academic dishonesty (Noorbehbahani et al., 2022; Kharbat and Abu Daabes, 2021). Empirical evidence suggests that unsupervised online assessment is associated with inflated scores relative to proctored conditions (Dendir and Maxwell, 2020), which places the credibility of computer-based results at risk precisely when institutions rely on them most.

In Indonesia, this problem has acquired a new policy dimension. The Ministry of Primary and Secondary Education introduced the Tes Kemampuan Akademik (TKA) through Ministerial Regulation No. 9 of 2025, establishing a standardized academic test for students at the end of primary, junior secondary, and senior secondary education, with Indonesian language and mathematics at every level and English plus an elective subject at the upper secondary level (Kemendikdasmen, 2025). Because TKA outcomes may inform further study selection, schools and universities increasingly run computer-assisted tryouts to familiarize students with the format. One such platform is the CAT Tryout TKA application of Universitas Bali Internasional Muhammadiyah (BIM), accessible at cat.bim.ac.id, which reproduces the TKA item formats (multiple choice, complex multiple choice, and true/false statement sets), displays the TKA indicator hierarchy for each item, and provides a complete administration workflow from item banking to item analysis (Universitas Bali Internasional Muhammadiyah, 2026). Such a platform controls the *session* rigorously, but it observes only what happens inside the browser: it can log that a student left the test tab, yet it cannot tell whether the person at the keyboard is the registered examinee, whether a second person is present, or whether a phone lies next to the keyboard.

The utilization of artificial intelligence (AI), or akal imitasi, as a technological tool across various aspects of human life has advanced rapidly in the era of global computing. Its applications are no longer confined to technological domains but have expanded extensively into the social sciences and humanities, law, the arts, and various other fields. Among these applications, AI-based proctoring has emerged as an established response to address the blind spots and limitations of conventional examination supervision, offering automated approaches to monitoring, analyzing, and detecting potentially suspicious behaviours during online examinations. Systematic reviews describe a common pattern of face detection and recognition, gaze and head-pose monitoring, and environmental analysis, typically combined by rules or learned classifiers (Nigam et al., 2021; Noorbehbahani et al., 2022). Early automated proctoring combined webcam and wearable-camera streams with hand-crafted features (Atoum et al., 2017), while more recent systems apply deep networks to detect suspicious behavior from video (Kaddoura and Gumaei, 2022; Satre et al., 2023). First-generation building blocks, such as Haar-cascade face detection (Viola and Jones, 2001), histogram-of-oriented-gradient

detectors (Dalal and Triggs, 2005), template-matching eye trackers, and coarse head-pose classifiers, suffered high false-positive rates and were easily defeated by printed photographs or screen replays. Contemporary models have largely removed these technical ceilings. MediaPipe provides a production-ready perception framework with a 478-point face mesh that includes iris landmarks (Lugaresi et al., 2019; Arabian et al., 2024); margin-based losses such as CosFace and ArcFace produce highly discriminative identity embeddings (Wang et al., 2018; Deng et al., 2019); YOLO11 offers a better speed–accuracy trade-off for real-time detection (Jocher and Qiu, 2024; Khanam and Hussain, 2024); and dense prediction transformers can recover plausible depth from a single RGB frame (Ranftl et al., 2021).

Three gaps nevertheless remain. First, most proctoring studies treat the examination platform as an abstract client and do not explain how visual evidence connects to the session-level controls schools actually use, such as tokens, timers, autosave, and tab-switch counters. Second, many systems rely on a single detector or monolithic classifier, making it difficult to explain to students, parents, and invigilators why a session was flagged (Coghlan et al., 2021). Third, ethical and practical critiques of proctoring emphasize surveillance burden, opaque automated decisions, and the need for human judgment (Coghlan et al., 2021; Lee and Fanguy, 2022; Selwyn et al., 2023), yet few technical designs build human-in-the-loop decision points into their architecture. To our knowledge, no prior work has addressed these gaps in the TKA context.

This study therefore asks how a TKA-oriented CAT platform and a multimodal AI proctoring pipeline can be integrated so that integrity evidence is real-time, interpretable, and actionable by invigilators on commodity hardware. The study supports and extends prior modular proctoring work (Nigam et al., 2021; Satre et al., 2023) rather than refuting it, and its novelty lies in the integration. The contributions fall into four areas: (i) an integration architecture that connects the CAT Tryout TKA session layer to a seven-module vision pipeline using a shared attempt identifier and a unified integrity record; (ii) an explainable confidence-fusion rule with warm-up and persistence windows, module-specific penalties, and recovery; (iii) a heterogeneous-acceleration scheduling strategy that keeps all seven models running in a browser-based, installation-free deployment; and (iv) an evaluation that combines runtime profiling, deterministic behavioral simulation, and a functional walkthrough of the integrated invigilation workflow.

2. Method

2.1 Research design

The study follows the design science research methodology (DSRM), which structures artifact-oriented research into problem identification, objective definition, design and development, demonstration, evaluation, and communication (Peffers et al., 2007). The problem and objectives were derived from the operational requirements documented in the CAT Tryout TKA user guide (version 1.0) and from the proctoring literature reviewed above. Design and development produced the integration architecture and the seven analysis modules described in Sections 2.2–2.6. Demonstration and evaluation comprised three complementary activities (Section 2.7): runtime profiling of every module on the target hardware, deterministic simulation of the confidence-fusion rule under scripted behavior scenarios, and a functional

walkthrough of the integrated examinee and invigilator workflows using the demonstration data bundled with the platform. The study is therefore exploratory-evaluative: it establishes technical feasibility and behavioral properties of the design rather than estimating detection accuracy on a population of examinees.

2.2 The CAT Tryout TKA platform

The CAT Tryout TKA platform separates an examinee area, reached through a "Member Area" login on the home page, from an administration panel at a dedicated /admin path that is never linked from the public page (Universitas Bali Internasional Muhammadiyah, 2026). Examinees authenticate with credentials printed on a participant card, confirm their identity data, and start an attempt only after entering a six-letter token announced by the room invigilator; the server-side timer starts at that moment and continues even if the browser is closed. Each answer is saved to the server immediately, a single account can be active on only one device, and the client records every occasion on which the examinee leaves the test page. Table 1 summarizes how these session-level controls map onto integrity threats and which threats remain outside their reach; the latter define the requirements for the proctoring service.

Table 1. Session-level controls of the CAT Tryout the TKA platform, and the residual threats addressed by AI proctoring

Threat	Native CAT control	Residual gap → proctoring module
Unauthorised start or early leakage	Six-letter token per package; package active only during the test; token regeneration	–
Account sharing	Single-device session; forced logout of the previous device	Impersonation by a different person → ArcFace verification
Time manipulation	Server-side timer; automatic submission and scoring at expiry	–
Browser-based lookup	Visibility-loss counter shown to invigilators	Off-screen materials → gaze, head pose, object detection
Collusion in the room	Randomised item and option order per examinee	Talking, extra persons → mouth tracker, person count
Presentation attack	–	Photo/screen replay → anti-spoofing, depth

Items are authored in a form editor or imported in bulk from CSV, and each item may carry TKA metadata (code, element, sub-element, competence, and indicator) that can be displayed to examinees. Complex multiple-choice and true/false sets can be scored dichotomously or with partial credit, and after the test the platform reports a ranking with score, duration, and tab-switch count, together with an item analysis that classifies each item by the proportion of correct responses (easy ≥ 70%, moderate 30–69%, difficult < 30%) and shows the distribution of selected options. A real-time monitoring page, refreshed every five seconds, lists every examinee's status, progress, remaining time, and tab-switch count and exposes the invigilator actions: +10 min, Finish, Reset, and Logout.

2.3 Integrated architecture

Figure 1 shows the integrated architecture. The examinee browser (layer A) hosts the CAT item renderer and visibility listener and, in parallel, captures webcam frames through the MediaDevices API. Frames are drawn to a canvas, JPEG-encoded at quality 0.7 at approximately 10 frames per

second (fps), Base64-encoded, and streamed as JSON messages over a WebSocket connection (Fette and Melnikov, 2011) to the proctoring service (layer C), which is built on FastAPI and the Uvicorn ASGI server (Lathkar, 2023). The CAT server (layer B) remains the authority for identity, time, and answers. The two services share only an attempt identifier, created when the examinee clicks Start Test, so the proctoring service never receives answer data, and the CAT server never receives video. A central orchestrator sends frames to seven analysis modules (layer D), combines their outputs into a confidence score, and generates integrity events. Both streams come together on the invigilator dashboard (layer E), where native CAT telemetry (progress, remaining time, tab switches) and the proctoring record (confidence trajectory and timestamped alerts) are shown for each attempt. Any consequential action remains the invigilator's explicit decision.

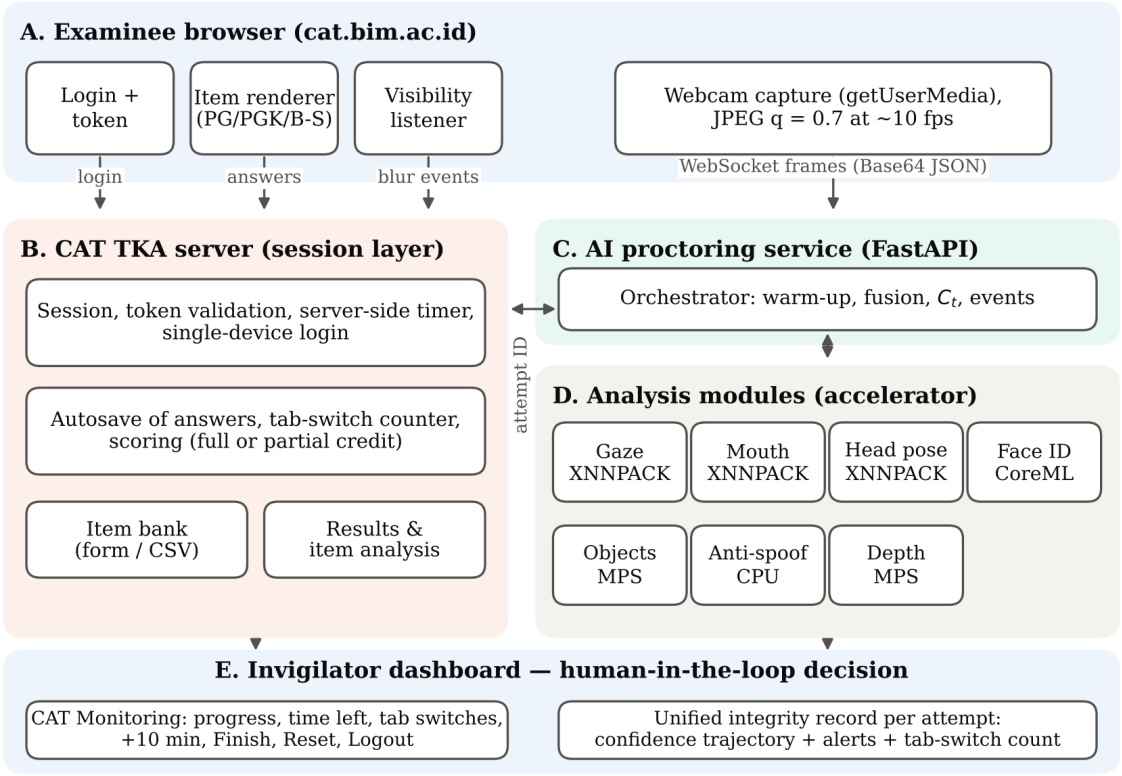


Figure 1. *Architecture of CAT TKA–AI Proctoring. The CAT server (B) governs the session; the proctoring service (C) analyses webcam frames with seven modules (D); both feed a unified integrity record reviewed by the invigilator (E).*

Three modules (gaze, mouth, and head pose) use MediaPipe FaceLandmarker v2 through the Tasks API in VIDEO running mode with a single face, detection and tracking confidence of 0.5, and blendshape output enabled (Lugaresi et al., 2019; Martinelli et al., 2023). Each module owns its landmarker instance so that internal tracking states remain independent. Because the Tasks API requires strictly increasing timestamps, each instance keeps its own offset and derives the timestamp from the wall clock as in Eq. (1).

$$t_{\text{ms}} = \lfloor 1000(t_{\text{now}} - t_{\text{offset}}) \rfloor \quad (1)$$

2.4 Behavioral cues from facial landmarks

Gaze. Gaze direction is estimated geometrically from the position of the iris

within the eye opening, an interpretable alternative to appearance-based regression networks (Zhang et al., 2015; Jain et al., 2024; Vidhya and Resende Faria, 2025). The left eye is bounded by landmarks {362, 385, 387, 263, 373, 380} with iris landmarks {474–477}, and the right eye by {33, 160, 158, 133, 153, 144} with iris landmarks {469–472}. For each eye, the iris center is the mean of its four iris landmarks, and the eye bounding box is taken from the six contour landmarks; the horizontal and vertical ratios follow Eq. (2).

$$h_{\text{ratio}} = \frac{x_{\text{iris}} - x_{\min}}{x_{\max} - x_{\min}}, \; v_{\text{ratio}} = \frac{y_{\text{iris}} - y_{\min}}{y_{\max} - y_{\min}} \quad (2)$$

The binocular means of the two ratios are compared with a tolerance $\tau_g = 0.25$ around the neutral value 0.5. Gaze is labelled left or right when the horizontal mean falls below $0.5 - \tau_g$ or above $0.5 + \tau_g$, up or down when the vertical mean does so, and centre otherwise. A violation is raised only after off-centre gaze persists for $T_{\text{gaze}} = 3.0$ s.

Mouth. Possible speech is detected from the mouth aspect ratio (MAR), the ratio of the inner-lip opening between landmarks 13 and 14 to the mouth width between landmarks 78 and 308 (Eq. (3)), where $d(\cdot,\cdot)$ is the Euclidean distance in pixels. The mouth is considered open when MAR > $\tau_m = 0.5$, and a violation is raised after $T_{\text{mouth}} = 2.0$ s of sustained opening.

$$\text{MAR} = \frac{d(L_{13}, L_{14})}{d(L_{78}, L_{308})} \quad (3)$$

Head pose. Head orientation is recovered with the Perspective-n-Point formulation (Lepetit et al., 2009), which estimates the camera extrinsics from correspondences between six image landmarks and a generic three-dimensional face model (Table 2). Landmark-based PnP reuses the face mesh already computed for gaze and mouth analysis and therefore avoids the extra inference cost of dedicated deep head-pose regressors such as WHENet or 6DRepNet (Zhou and Gregson, 2020; Hempel et al., 2022).

Table 2. Generic 3D face model points (mm) and corresponding MediaPipe landmark indices used for PnP

Point	3D coordinates (x, y, z)	Landmark index
Nose tip	(0, 0, 0)	1
Chin	(0, −330, −65)	152
Left eye outer corner	(−225, 170, −135)	33
Right eye outer corner	(225, 170, −135)	263
Left mouth corner	(−150, −150, −125)	61
Right mouth corner	(150, −150, −125)	291

The camera is approximated by the pinhole intrinsic matrix of Eq. (4), where the focal length is approximated by the frame width, $f = w$, the principal point is the image center, $(c_x, c_y) = (w/2, h/2)$, h is the frame height, and lens distortion is neglected. The rotation vector returned by the iterative PnP solver is converted to a rotation matrix by the Rodrigues formula and decomposed with RQ decomposition into the factorisation of Eq. (5) (Boyko et al., 2018), giving pitch α, yaw β, and roll γ.

$$K = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$R = R_z(\gamma) \cdot R_y(\beta) \cdot R_x(\alpha) \quad (5)$$

Because the model's z-axis points away from the camera, the raw pitch is wrapped into a symmetric range by Eq. (6). Head direction is classified as left/right when $|\beta| > 30°$ and up/down when $|\alpha'| > 25°$, and a violation is

raised after $T_{\text{head}} = 2.0$ s of sustained deviation.

$$\alpha' = \begin{cases} \alpha - 180°, & \text{if } \alpha > 90° \\ \alpha + 180°, & \text{if } \alpha < -90° \\ \alpha, & \text{otherwise} \end{cases} \quad (6)$$

2.5 Identity, environment, and liveness

Identity verification. Identity is verified with the InsightFace buffalo_l model suite. Faces are detected with the SCRFD-based det_10g detector at 640 × 640 input (Guo et al., 2022), aligned with a five-point similarity transform, and embedded by the w600k_r50 ResNet-50 network trained with the additive angular margin (ArcFace) loss into 512-dimensional, L2-normalized vectors (Deng et al., 2019). ArcFace improved on earlier embedding approaches such as FaceNet and MTCNN–FaceNet pipelines (Schroff et al., 2015; Qi et al., 2022; Alimuin et al., 2020) and reaches 99.83% verification accuracy on LFW. A reference embedding $\boldsymbol{e}_{\text{ref}}$ is enrolled once, ideally immediately after the CAT data-confirmation step, and each verification embedding is compared with it by cosine similarity (Eq. (7)). A match requires sim > τ_{f} = 0.45; verification runs every 30 frames through ONNX Runtime with the CoreML execution provider and a CPU fallback.

$$\text{sim}(\boldsymbol{e}_1, \boldsymbol{e}_2) = \frac{\boldsymbol{e}_1 \cdot \boldsymbol{e}_2}{\|\boldsymbol{e}_1\| \|\boldsymbol{e}_2\|} \quad (7)$$

Environment. Prohibited items are detected with YOLO11n, the nano variant of the YOLO11 family, which replaces the C2f blocks of YOLOv8 with C3k2 blocks and adds a C2PSA spatial-attention block; it reaches 39.5% $\text{mAP}_{50–95}$ on COCO val2017 with about 2.6 M parameters (Jocher and Qiu, 2024; Khanam and Hussain, 2024), continuing the single-stage lineage begun by Redmon et al. (2016). Only three COCO categories are relevant (Table 3). Detections are accepted at confidence τ_{d} = 0.5, and the module runs every 10 frames.

Table 3. COCO categories monitored by the object-detection module

Class ID	Category	Violation condition
0	Person	Count ≠ 1 (examinee absent or additional person present)
67	Cell phone	Any detection
73	Book	Any detection

Anti-spoofing. Deep face anti-spoofing networks such as CDCN achieve strong benchmark results but require attack-specific training data and generalize unevenly across attack types (Yu et al., 2020; Yu et al., 2023; Yang et al., 2014). The proposed module instead fuses four complementary, training-free cues computed on the face crop I_{face}: local binary pattern (LBP) texture variance, which separates skin micro-texture from print or display reproductions (Ojala et al., 2002) (Eq. (8)); a frequency cue that penalizes the high-frequency energy characteristic of moiré patterns from screen replays (Amidror, 2000) (Eq. (9)), where E_{hf} is the spectral energy outside the central 30% of the shifted Fourier spectrum; an HSV colour-spread cue, since printed photographs show narrower saturation and value distributions (Eq. (10)); and a Laplacian-variance cue for the depth flatness of planar attacks (Eq. (11)).

$$S_{\text{tex}} = \min\left(1, \frac{\text{Var}(\text{LBP}(I_{\text{face}}))}{3000}\right) \quad (8)$$

$$S_{\text{moiré}} = 1 - \min\left(1, 2\frac{E_{\text{hf}}}{E_{\text{total}}}\right) \quad (9)$$

$$S_{\text{col}} = \frac{1}{2}\left(\frac{\sigma_S}{50} + \frac{\sigma_V}{60}\right) \quad (10)$$

$$S_{\text{dep}} = \min\left(1, \frac{\text{Var}(\nabla^2 I_{\text{face}})}{200}\right) \quad (11)$$

The cues are fused with fixed weights (Eq. (12)) and smoothed with a causal moving average over at most 15 evaluations (Eq. (13)); the face is classified as live when the smoothed score is at least 0.5.

$$C_{\text{sp}} = 0.30S_{\text{tex}} + 0.25S_{\text{moiré}} + 0.20S_{\text{col}} + 0.25S_{\text{dep}} \quad (12)$$

$$\hat{C}_{\text{sp}}(t) = \frac{1}{\min(t+1,15)} \sum_{k=\max(0,t-14)}^{t} C_{\text{sp}}(k) \quad (13)$$

Monocular depth. A DPT-Hybrid model (Intel/dpt-hybrid-midas) produces a dense relative depth map from a single RGB frame (Ranftl et al., 2021; Ranftl et al., 2022; Zuo et al., 2022). The output is bicubically resized to the frame resolution and min-max normalized to [0, 1]. A live, three-dimensional face typically yields depth variance above 0.01 in the central face crop, whereas a photograph or screen held to the camera produces a nearly uniform depth field. With roughly 100 ms per inference on the Metal Performance Shaders (MPS) backend, the module runs every 60 frames and supplies depth evidence to the liveness decision.

2.6 Confidence fusion and human-in-the-loop policy

The orchestrator maintains a session confidence score $C_t \in [0, 1]$, initialized to $C_0 = 1$. No penalties are applied during a warm-up of 150 processed frames, which lets detectors stabilize, camera auto-exposure settle, and the examinee adopt a stable posture. Afterward, at each processed frame *t*, let V_t denote the set of modules that were evaluated at *t* and reported a violation that satisfied its persistence window. The score is updated by Eq. (14), with recovery rate $r = 0.005$ and the module penalties p_i listed in Table 4.

$$C_{t+1} = \begin{cases} \min(1, C_t + r), & \text{if } V_t = \emptyset \\ \max\left(0, C_t - \sum_{i \in V_t} p_i\right), & \text{otherwise} \end{cases} \quad (14)$$

Penalties are proportional to the evidential weight of each cue: brief gaze or mouth events are common in honest test-taking and receive the smallest penalties, whereas identity mismatch and presentation attacks indicate deliberate circumvention. In the stand-alone configuration, a session is terminated when $C_t \leq 0.30$. In the integrated CAT configuration, the same threshold raises a *flag* on the invigilator dashboard instead of ending the attempt automatically; the invigilator inspects the alert timeline and can then use the native CAT actions (for example, Finish or Logout). This choice reflects the recommendation that automated proctoring should support rather than replace human judgment (Coghlan et al., 2021; Selwyn et al., 2023).

Table 4. Module schedule, hardware backend, measured latency, and penalty weight

Module	Backend	Period (frames)	Rate at 30 fps	Latency (ms)	Penalty p_i
Eye tracker	XNNPACK (CPU) + GPU delegate	1	30 Hz	~8	0.005
Mouth tracker	XNNPACK (CPU) + GPU delegate	1	30 Hz	~8	0.004
Head pose	XNNPACK + OpenCV	1	30 Hz	~10	0.006
Object detection	PyTorch MPS	10	3 Hz	~25	0.008
Face recognition	ONNX Runtime CoreML	30	1 Hz	~45	0.020
Anti-spoofing	CPU (OpenCV, SciPy)	30	1 Hz	~15	0.050
Depth estimation	PyTorch MPS	60	0.5 Hz	~100	– (feeds liveness)

2.7 Implementation and evaluation protocol

The proctoring service was built in Python and runs on FastAPI, using Jinja2 templates and static assets. To prevent WebSocket encoding errors, NumPy scalar types are converted to native Python types before JSON serialization. In addition to the examinee

WebSocket, the service provides an administrator WebSocket (/ws/admin) and REST endpoints for checking health, enrolling faces, and listing or viewing session details. The CAT dashboard accesses these endpoints by attempt identifier. Experiments used the hardware and software listed in Table 5.

Table 5. Experimental hardware and software environment

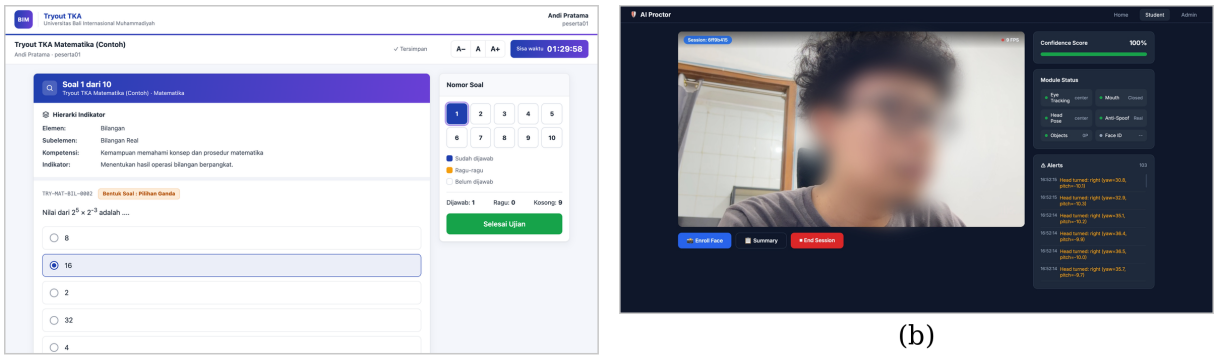
Component	Specification / version	Role
Processor	Apple M4 Pro, 12-core CPU, 16-core GPU, 48 GB unified memory	Host
Operating system	macOS Sequoia	Host
Camera	iPhone Continuity Camera (AVFoundation), 1920 × 1080 at 30 fps	Video input
Python	3.14.2	Runtime
MediaPipe	0.10.32	Face landmarks (Tasks API)
InsightFace / ONNX Runtime	0.7.3 / 1.24.3	ArcFace inference (CoreML)
PyTorch / Transformers	2.10.0 / 5.3.0	DPT depth on MPS
Ultralytics	8.4.21	YOLO11 detection
OpenCV	4.13.0	Image processing, PnP
FastAPI / Uvicorn	0.135.1 / 0.41.0	Web and WebSocket server

Evaluation addressed three questions. **(Q1) Runtime feasibility**: per-module inference latency, end-to-end throughput, and initialization time were measured with all seven modules loaded. **(Q2) Behavior of the fusion rule**: because the rule is deterministic, its response was characterized by simulating three scripted 180-s sessions at the observed 10 processed frames per second, applying Eq. (14) with the persistence windows, schedules, and penalties of Table 4: a *compliant* session with brief head adjustments (1.5 s every 40 s); an *intermittent-aversion* session with 4.5-s off-screen glances every 20 s and 3-s head turns every 30 s; and a *sustained phone-use* session in which gaze aversion, downward head pose, and a visible phone begin at $t = 90$ s. **(Q3) Workflow integration**: the examinee and invigilator workflows were walked through end to end with the platform's demonstration package (ten mathematics items, nine demonstration accounts) to verify that session telemetry and proctoring evidence are available together at each decision point.

3. Result and Discussion

3.1 Integrated examinee and invigilator workflow

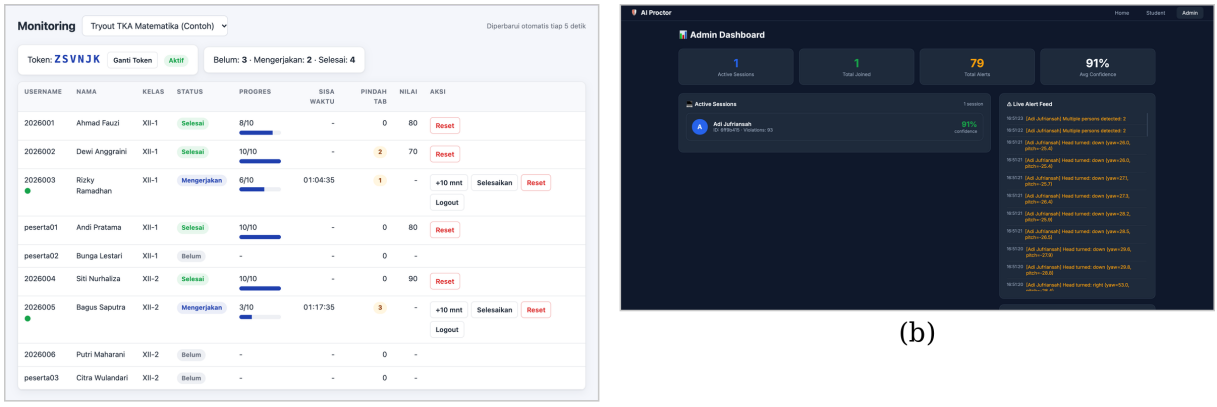
Figure 2 contrasts the two examinee-facing views of the integrated system. The CAT screen (Figure 2a) presents the item with its TKA indicator hierarchy, the item code and format label, a server-driven countdown, an autosave indicator, font-size controls, and a navigation panel that color-codes answered, doubtful, and unanswered items. The proctoring view (Figure 2b) shows the live camera stream with the face blurred for publication, the current confidence score, the status of each module, and a timestamped alert list; in the session shown, consecutive alerts report a rightward head turn with yaw between 30.8° and 36.5°, just above the 30° threshold. Because both views are bound to the same attempt, the token step that starts the CAT timer is also the natural point for face enrolment, which closes the impersonation gap identified in Table 1.



(a) (b)

Figure 2. *Examinee-facing views: (a) CAT Tryout TKA item screen with indicator hierarchy, timer, and item navigator; (b) AI proctoring view with confidence score, module status, and alert stream (face blurred).*

On the invigilator side (Figure 3), CAT Monitoring (Figure 3a) lists every account with status, progress, remaining time, tab-switch count, and score, and offers per-attempt actions. In the demonstration package, three of the nine accounts had not yet started, two were in progress, and four had finished. Of the examinees who were in progress, two had one and three tab switches, respectively, while one examinee who had finished had two. The proctoring dashboard (Figure 3b) brings together active sessions, total alerts, and mean confidence and provides alerts such as multiple people being detected or a sustained downward head pose. These signals complement one another: the tab-switch counter can reveal lookups on the same device, but it cannot detect phone use or help from another person, while the visual modules can detect those events. Showing both for each attempt brings these two partial views together in one evidential record—the key practical benefit of the integration.



(a) (b)

Figure 3. *Invigilator-facing views: (a) CAT Monitoring with progress, remaining time, tab switches, and actions (demonstration data); (b) AI proctoring admin dashboard with session confidence and live alert feed.*

3.2 Runtime performance

With staggered scheduling, the pipeline maintained more than 10 processed fps over the WebSocket channel, matching the browser's capture rate, while keeping all seven modules active (Figure 4). Lightweight landmark modules use most of the per-frame budget (8–10 ms each at 30 Hz), while the heavier models run less frequently to balance the workload: object detection at 3 Hz (25 ms), face recognition and anti-spoofing at 1

Hz (45 ms and 15 ms), and depth estimation at 0.5 Hz (100 ms). Even in the busiest case, when every module runs on the same frame, the total latency (about 211 ms) affects just that frame without interrupting the stream. Loading all modules took about 12 s, mainly for the DPT weights (about 6 s) and the InsightFace models on CoreML (about 4 s). This cost is incurred only once at server start-up, rather than for each examinee. Mapping each model to the accelerator that suits it, namely XNNPACK for MediaPipe graphs, CoreML for ONNX face models, and MPS for PyTorch networks, is what allows a single consumer workstation to host the full pipeline.

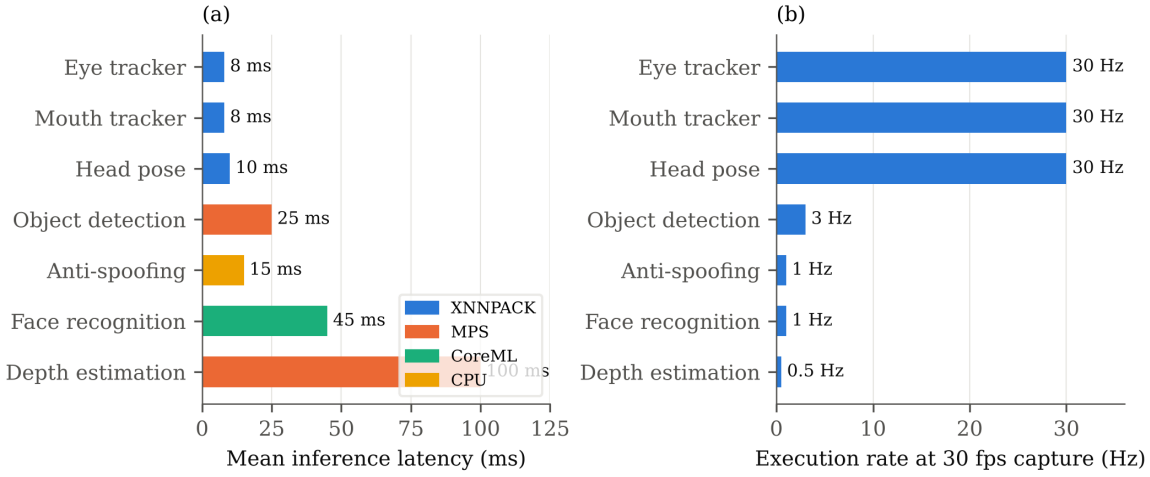


Figure 4. *Module performance on an Apple M4 Pro: (a) mean inference latency by module, colored by hardware backend; (b) execution rate under the staggered schedule at 30 fps capture.*

3.3 Behaviour of the confidence score

Figure 5 shows the penalty weights and the simulated confidence trajectories. In the compliant session, no cue outlasted its persistence window, so C_t remained at 1.00 throughout. In the intermittent-aversion session, each glance produced a short saw-tooth dip once the 3-s gaze window had elapsed, and the score recovered at +0.005 per frame between episodes; the minimum was 0.85 and the mean 0.99, far above the 0.30 threshold. During the sustained phone-use session, penalties for head pose (after 2 s), gaze (after 3 s), and object detection (every 10 frames) accumulated without recovery. The score crossed the threshold at t = 98.2 s, just 8.2 s after the behavior began. These persistence windows help filter out natural micro-movements, while the additive penalties bring sustained misconduct involving multiple cues to light within seconds. The simulation also highlights how warm-up length depends on processing rate: 150 frames equal 5 s at a 30 fps capture rate, but 15 s at the 10 fps WebSocket rate. Future releases should express this parameter in seconds.

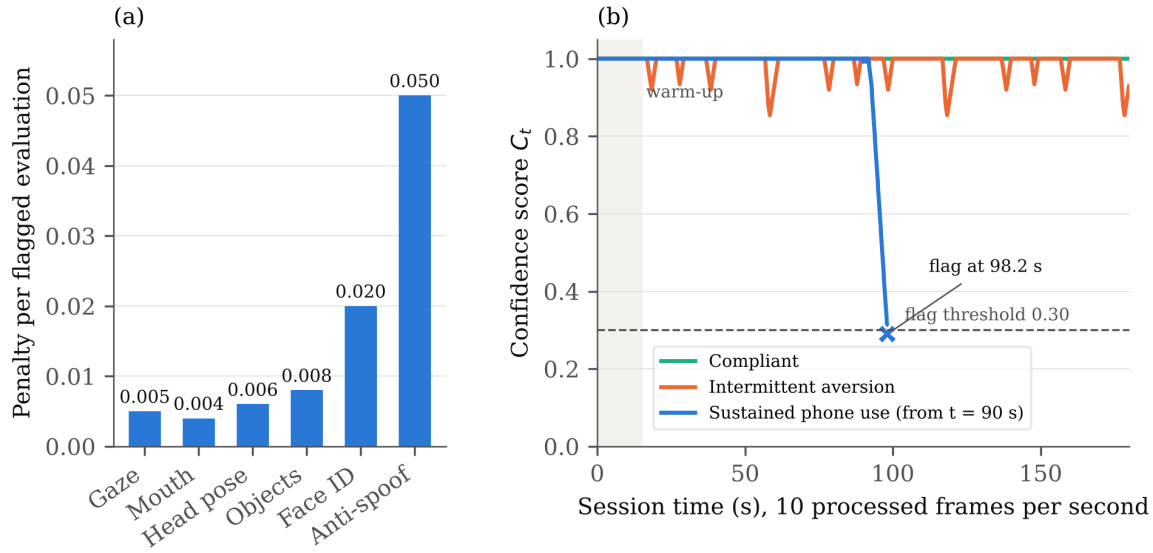
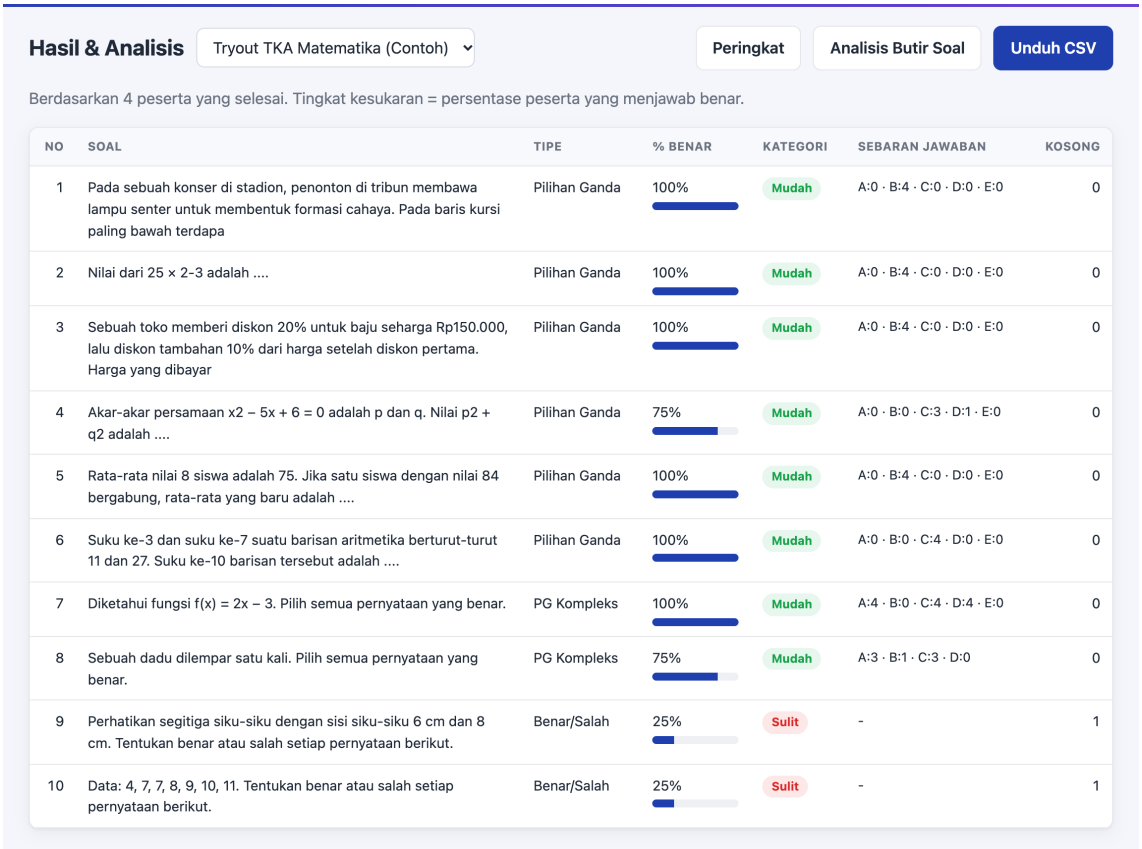


Figure 5. *(a) Penalty applied per flagged evaluation for each module; (b) simulated confidence trajectories for compliant, intermittent-aversion, and sustained phone-use sessions (10 processed fps; dashed line: flag threshold).*

3.4 Item-level evidence and comparison with prior approaches

Proctoring evidence becomes even more informative when considered alongside psychometric evidence. In the CAT item analysis of the demonstration package (Figure 6), eight of the ten items were classified as easy. By contrast, only 25% of finishers answered each of the two true/false items correctly, so both were classified as difficult. The platform's guidance is to verify the answer key of such items first; in an integrated record, an invigilator or test developer can additionally check whether an anomalous item coincides with clustered integrity alerts or tab switches, separating defective items from potential collusion. Table 6 summarizes how the integrated design differs from typical earlier proctoring systems.



Hasil & Analisis Tryout TKA Matematika (Contoh) Peringkat Analisis Butir Soal Unduh CSV

Berdasarkan 4 peserta yang selesai. Tingkat kesukaran = persentase peserta yang menjawab benar.

NO	SOAL	TIPE	% BENAR	KATEGORI	SEBARAN JAWABAN	KOSONG
1	Pada sebuah konser di stadion, penonton di tribun membawa lampu senter untuk membentuk formasi cahaya. Pada baris kursi paling bawah terdapa	Pilihan Ganda	100%	Mudah	A:0 · B:4 · C:0 · D:0 · E:0	0
2	Nilai dari 25 × 2-3 adalah	Pilihan Ganda	100%	Mudah	A:0 · B:4 · C:0 · D:0 · E:0	0
3	Sebuah toko memberi diskon 20% untuk baju seharga Rp150.000, lalu diskon tambahan 10% dari harga setelah diskon pertama. Harga yang dibayar	Pilihan Ganda	100%	Mudah	A:0 · B:4 · C:0 · D:0 · E:0	0
4	Akar-akar persamaan x2 – 5x + 6 = 0 adalah p dan q. Nilai p2 + q2 adalah	Pilihan Ganda	75%	Mudah	A:0 · B:0 · C:3 · D:1 · E:0	0
5	Rata-rata nilai 8 siswa adalah 75. Jika satu siswa dengan nilai 84 bergabung, rata-rata yang baru adalah	Pilihan Ganda	100%	Mudah	A:0 · B:4 · C:0 · D:0 · E:0	0
6	Suku ke-3 dan suku ke-7 suatu barisan aritmetika berturut-turut 11 dan 27. Suku ke-10 barisan tersebut adalah	Pilihan Ganda	100%	Mudah	A:0 · B:0 · C:4 · D:0 · E:0	0
7	Diketahui fungsi f(x) = 2x – 3. Pilih semua pernyataan yang benar.	PG Kompleks	100%	Mudah	A:4 · B:0 · C:4 · D:4 · E:0	0
8	Sebuah dadu dilempar satu kali. Pilih semua pernyataan yang benar.	PG Kompleks	75%	Mudah	A:3 · B:1 · C:3 · D:0	0
9	Perhatikan segitiga siku-siku dengan sisi siku-siku 6 cm dan 8 cm. Tentukan benar atau salah setiap pernyataan berikut.	Benar/Salah	25%	Sulit	-	1
10	Data: 4, 7, 7, 8, 9, 10, 11. Tentukan benar atau salah setiap pernyataan berikut.	Benar/Salah	25%	Sulit	-	1

Figure 6. *CAT item analysis of the demonstration package: proportion correct, difficulty category, option distribution, and omissions per item.*

Table 6. Comparison of the proposed design with typical earlier proctoring approaches

Aspect	Typical earlier approach	CAT TKA–AI Proctoring
Face landmarks	dlib 68-point model	MediaPipe 478-point mesh with iris
Identity	FaceNet / OpenFace embeddings	ArcFace (InsightFace buffalo_l)
Gaze	Template matching	Iris-to-eyelid ratio with persistence window
Head pose	Coarse CNN classes	PnP with RQ decomposition (continuous angles)
Environment	Not monitored or manual	YOLO11n (person, phone, book)
Liveness	None or single cue	Four-cue fusion plus monocular depth
Session control	Separate from proctoring	Token, timer, autosave, tab log bound by attempt ID
Decision	Automatic termination	Threshold flag reviewed by invigilator
Deployment	Desktop client, CUDA-only	Browser, no install; MPS, CoreML, XNNPACK

Relative to systematic reviews of AI proctoring (Nigam et al., 2021; Noorbehbahani et al., 2022), the proposed design occupies the multimodal, rule-fused end of the spectrum. Compared with the face-recognition and behavior-monitoring framework of Satre et al. (2023) and the video-classification approach of Kaddoura and Gumaei (2022), its distinctive features are modular orchestration with explicit, per-cue evidence; training-free anti-spoofing strengthened by transformer-based depth (Ranftl et al., 2021); and binding to an operational CAT session layer. The modular design echoes the shift from classical detectors to deep multi-face detection reported by Saputra et al. (2025) and the use of MediaPipe landmarks for lightweight real-time analysis (Sánchez-Brizuela et al., 2023). Its explainability addresses a central objection in the ethics literature, that opaque automated judgments undermine fairness and trust

(Coghlan et al., 2021), because every flag can be traced to named cues, durations, and frames.

Several limitations qualify these findings. First, detection accuracy (precision and recall of each cue and of the fused flag) was not estimated on a labeled corpus of real TKA try-out sessions, and the thresholds and penalties were set by engineering judgment; a field study with annotated sessions, receiver-operating analysis of τ values, and fairness analysis across skin tones, eyewear, and lighting conditions is required before high-stakes use. Second, we measured throughput on a single high-end workstation. Serving an entire computer lab will likely require scaling out across multiple systems. or edge inference on examinee devices. Third, webcam proctoring cannot observe everything: off-camera devices, earpieces, and audio cues remain outside its scope, so invigilators remain indispensable, a point also emphasized by the platform guide. Finally, when collecting facial video from minors, it's essential to obtain informed consent from students and their guardians, keep retention to a minimum, and comply with Indonesia's personal data protection framework. These governance measures are just as important as the technical design. (Selwyn et al., 2023; Lee and Fanguy, 2022).

4. Conclusion

This study designed and evaluated CAT TKA–AI Proctoring, an architecture that binds the session controls of a TKA-oriented computer-assisted test platform to a seven-module, real-time vision pipeline and presents both forms of evidence to invigilators in a single integrity record. The session layer secures who may start, for how long, and with which answers, while the vision layer adds evidence on identity, attention, environment, and liveness that the browser alone cannot observe. Staggered scheduling across heterogeneous accelerators kept all modules running at more than 10 processed frames per second on a consumer workstation, with no client installation required. The confidence-fusion rule was designed to disregard natural micro-movements while flagging sustained misconduct supported by multiple cues within about eight seconds. Replacing automatic termination with an invigilator-reviewed flag also makes the system more transparent and accountable in a school setting. Overall, this design offers Indonesian schools and universities a practical way to support more credible TKA preparation and, in time, secure computer-assisted testing more broadly. Future work will assess detection accuracy and fairness through field trials with annotated sessions, learn cue weights from data rather than fixing them in advance, add privacy-preserving audio analysis, explore WebRTC or on-device inference to reduce server load, and support integration through standard learning-tool interfaces such as LTI and xAPI.

Declaration of competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors thank Universitas Muhammadiyah Surakarta for funding this research and Universitas Bali Internasional Muhammadiyah for providing access to the CAT Tryout TKA platform (cat.bim.ac.id).

Ethical statement

The webcam images shown in this article were recorded from consenting members of the research team, and their faces are blurred. CAT

screenshots contain only fictitious demonstration accounts. This study was approved by the Health Research Ethics Committee, Faculty of Medicine, Universitas Muhammadiyah Surakarta, which reviewed the study involving human participants and granted an exemption from full ethical review on the grounds that the study qualified as minimal-risk research, under Exemption/Protocol.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used AI-assisted tools for language editing and proofreading. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Data availability

The source code configuration, initialisation logs, and simulation script are available from the corresponding author on reasonable request.

5. References

Alimuin, R., Dadios, E., Dayao, J., Arenas, S., 2020. Deep hypersphere embedding for real-time face recognition. *TELKOMNIKA (Telecommunication Computing Electronics and Control)* 18, 1671. https://doi.org/10.12928/telkomnika.v18i3.14787

Amankwah-Amoah, J., Khan, Z., Wood, G., Knight, G., 2021. COVID-19 and digitalization: The great acceleration. *Journal of Business Research* 136, 602–611. https://doi.org/10.1016/j.jbusres.2021.08.011

Amidror, I., 2000. Moiré minimization, in: *The Theory of the Moiré Phenomenon*. Springer, Dordrecht, pp. 59–80. https://doi.org/10.1007/978-94-011-4205-2_3

Arabian, H., Abdulbaki Alshirbaji, T., Chase, J.G., Moeller, K., 2024. Emotion recognition beyond pixels: Leveraging facial point landmark meshes. *Applied Sciences* 14, 3358. https://doi.org/10.3390/app14083358

Atoum, Y., Chen, L., Liu, A.X., Hsu, S.D.H., Liu, X., 2017. Automated online exam proctoring. *IEEE Transactions on Multimedia* 19, 1609–1624. https://doi.org/10.1109/TMM.2017.2656064

Boyko, N., Basystiuk, O., Shakhovska, N., 2018. Performance evaluation and comparison of software for face recognition, based on dlib and OpenCV library, in: *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*. IEEE, Lviv, pp. 478–482. https://doi.org/10.1109/DSMP.2018.8478556

Coghlan, S., Miller, T., Paterson, J., 2021. Good proctor or "Big Brother"? Ethics of online exam supervision technologies. *Philosophy & Technology* 34, 1581–1606. https://doi.org/10.1007/s13347-021-00476-1

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection, in: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. IEEE, San Diego, pp. 886–893. https://doi.org/10.1109/CVPR.2005.177

Dendir, S., Maxwell, R.S., 2020. Cheating in online courses: Evidence from online proctoring. *Computers in Human Behavior Reports* 2, 100033. https://doi.org/10.1016/j.chbr.2020.100033

Deng, J., Guo, J., Xue, N., Zafeiriou, S., 2019. ArcFace: Additive angular margin loss for deep face recognition, in: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Long Beach, pp. 4685–4694. https://doi.org/10.1109/CVPR.2019.00482

Döhring, B., Hristov, A., Maier, C., Roeger, W., Thum-Thysen, A., 2021. COVID-19 acceleration in digitalisation, aggregate productivity growth and the functional income distribution. *International Economics and Economic Policy* 18, 571–604. https://doi.org/10.1007/s10368-021-00511-8

Fette, I., Melnikov, A., 2011. The WebSocket Protocol. RFC 6455. Internet Engineering Task Force. https://doi.org/10.17487/RFC6455

Grinin, L., Grinin, A., Korotayev, A., 2022. COVID-19 pandemic as a trigger for the acceleration of the cybernetic revolution, transition from e-government to e-state, and change in social relations. *Technological Forecasting and Social Change* 175, 121348. https://doi.org/10.1016/j.techfore.2021.121348

Guo, J., Deng, J., Lattas, A., Zafeiriou, S., 2022. Sample and computation redistribution for efficient face detection, in: *International Conference on Learning Representations (ICLR 2022)*. https://doi.org/10.48550/arXiv.2105.04714

Hempel, T., Abdelrahman, A.A., Al-Hamadi, A., 2022. 6D rotation representation for unconstrained head pose estimation, in: *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, Bordeaux, pp. 2496–2500. https://doi.org/10.1109/ICIP46576.2022.9897219

Jain, T., Bhatia, S., Sarkar, C., Jain, P., Jain, N.K., 2024. Real-time webcam-based eye tracking for gaze estimation: Applications and innovations, in: *2024 15th International Conference on*

Computing Communication and Networking Technologies (ICCCNT). IEEE, pp. 1–7. https://doi.org/10.1109/ICCCNT61001.2024.10724037

Jocher, G., Qiu, J., 2024. Ultralytics YOLO11 (version 11.0.0) [Software]. https://github.com/ultralytics/ultralytics (accessed 26 October 2026).

Kaddoura, S., Gumaei, A., 2022. Towards effective and efficient online exam systems using deep learning-based cheating detection approach. *Intelligent Systems with Applications* 16, 200153. https://doi.org/10.1016/j.iswa.2022.200153

Kemendikdasmen [Ministry of Primary and Secondary Education of the Republic of Indonesia], 2025. Peraturan Menteri Pendidikan Dasar dan Menengah Nomor 9 Tahun 2025 tentang Tes Kemampuan Akademik [Ministerial Regulation No. 9 of 2025 on the Academic Ability Test]. Jakarta. https://peraturan.bpk.go.id/Details/321171 (accessed 26 October 2026).

Khanam, R., Hussain, M., 2024. YOLOv11: An overview of the key architectural enhancements. arXiv preprint. https://doi.org/10.48550/arXiv.2410.17725

Kharbat, F.F., Abu Daabes, A.S., 2021. E-proctored exams during the COVID-19 pandemic: A close understanding. *Education and Information Technologies* 26, 6589–6605. https://doi.org/10.1007/s10639-021-10458-7

Lathkar, M., 2023. Getting started with FastAPI, in: *High-Performance Web Apps with FastAPI*. Apress, Berkeley, CA, pp. 29–64. https://doi.org/10.1007/978-1-4842-9178-8_2

Lee, K., Fanguy, M., 2022. Online exam proctoring technologies: Educational innovation or deterioration? *British Journal of Educational Technology* 53, 475–490. https://doi.org/10.1111/bjet.13182

Lepetit, V., Moreno-Noguer, F., Fua, P., 2009. EPnP: An accurate O(n) solution to the PnP problem. *International Journal of Computer Vision* 81, 155–166. https://doi.org/10.1007/s11263-008-0152-6

Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.-L., Yong, M.G., Lee, J., Chang, W.-T., Hua, W., Georg, M., Grundmann, M., 2019. MediaPipe: A framework for building perception pipelines. arXiv preprint. https://doi.org/10.48550/arXiv.1906.08172

Martinelli, D., Cerbaro, J., Teixeira, M.A.S., Fabro, J.A., Schneider de Oliveira, A., 2023. A tutorial to use the MediaPipe framework with ROS2, in: Koubaa, A. (Ed.), *Robot Operating System (ROS): The Complete Reference (Volume 7)*. Springer, Cham, pp. 57–79. https://doi.org/10.1007/978-3-031-09062-2_1

Nigam, A., Pasricha, R., Singh, T., Churi, P., 2021. A systematic review on AI-based proctoring systems: Past, present and future. *Education and Information Technologies* 26, 6421–6445. https://doi.org/10.1007/s10639-021-10597-x

Noorbehbahani, F., Mohammadi, A., Aminazadeh, M., 2022. A systematic review of research on cheating in online exams from 2010 to 2021. *Education and Information Technologies* 27, 8413–8460. https://doi.org/10.1007/s10639-022-10927-7

Ojala, T., Pietikäinen, M., Mäenpää, T., 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 971–987. https://doi.org/10.1109/TPAMI.2002.1017623

Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S., 2007. A design science research methodology for information systems research. *Journal of Management Information Systems* 24, 45–77. https://doi.org/10.2753/MIS0742-1222240302

Qi, S., Zuo, X., Feng, W., Naveen, I.G., 2022. Face recognition model based on MTCNN and FaceNet, in: *2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC)*. IEEE, pp. 1–5. https://doi.org/10.1109/ICMNWC56175.2022.10031806

Ranftl, R., Bochkovskiy, A., Koltun, V., 2021. Vision transformers for dense prediction, in: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, pp. 12179–12188. https://doi.org/10.1109/ICCV48922.2021.01196

Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., Koltun, V., 2022. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 1623–1637. https://doi.org/10.1109/TPAMI.2020.3019967

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection, in: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Las Vegas, pp. 779–788. https://doi.org/10.1109/CVPR.2016.91

Sánchez-Brizuela, G., Cisnal, A., de la Fuente-López, E., Fraile, J.-C., Pérez-Turiel, J., 2023. Lightweight real-time hand segmentation leveraging MediaPipe landmark detection. *Virtual Reality* 27, 3125–3132. https://doi.org/10.1007/s10055-023-00858-0

Saputra, S., Jufriansah, A., Mu'min, M.A., Akbar, M., Jayanto, D.L., Hernita, A., 2025. Comparison of multi-face detection performance on images using Haarcascade, Dlib, and RetinaFace. *Journal of Novel Engineering Science and Technology* 4, 134–146. https://doi.org/10.56741/jnest.v4i03.1134

Satre, S., Patil, S., Mane, T., Molawade, V., Gawand, T., Mishra, A., 2023. Online exam proctoring system based on artificial intelligence, in: *2023 International Conference on Signal Processing, Computation, Electronics, Power and Telecommunication (IConSCEPT)*. IEEE, pp. 1–6. https://doi.org/10.1109/IConSCEPT57958.2023.10170577

Schroff, F., Kalenichenko, D., Philbin, J., 2015. FaceNet: A unified embedding for face recognition and clustering, in: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Boston, pp. 815–823. https://doi.org/10.1109/CVPR.2015.7298682

Selwyn, N., O'Neill, C., Smith, G., Andrejevic, M., Gu, X., 2023. A necessary evil? The rise of online exam proctoring in Australian universities. *Media International Australia* 186, 149–164. https://doi.org/10.1177/1329878X211005862

Universitas Bali Internasional Muhammadiyah, 2026. Panduan Penggunaan CAT Tryout TKA [User Guide of the CAT Tryout TKA Application], version 1.0. Denpasar. https://cat.bim.ac.id (accessed 26 October 2026).

Vidhya, V., Resende Faria, D., 2025. Real-time gaze estimation using webcam-based CNN models for human–computer interactions. *Computers* 14, 57. https://doi.org/10.3390/computers14020057

Viola, P., Jones, M., 2001. Rapid object detection using a boosted cascade of simple features, in: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*. IEEE, Kauai, pp. I-511–I-518. https://doi.org/10.1109/CVPR.2001.990517

Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W., 2018. CosFace: Large margin cosine loss for deep face recognition, in: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Salt Lake City, pp. 5265–5274. https://doi.org/10.1109/CVPR.2018.00552

Yang, J., Lei, Z., Li, S.Z., 2014. Learn convolutional neural network for face anti-spoofing. arXiv preprint. https://doi.org/10.48550/arXiv.1408.5601

Yu, Z., Qin, Y., Li, X., Zhao, C., Lei, Z., Zhao, G., 2023. Deep learning for face anti-spoofing: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 5609–5631. https://doi.org/10.1109/TPAMI.2022.3215850

Yu, Z., Zhao, C., Wang, Z., Qin, Y., Su, Z., Li, X., Zhou, F., Zhao, G., 2020. Searching central difference convolutional networks for face anti-spoofing, in: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Seattle, pp. 5295–5305. https://doi.org/10.1109/CVPR42600.2020.00534

Zhang, X., Sugano, Y., Fritz, M., Bulling, A., 2015. Appearance-based gaze estimation in the wild, in: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Boston, pp. 4511–4520. https://doi.org/10.1109/CVPR.2015.7299081

Zhou, Y., Gregson, J., 2020. WHENet: Real-time fine-grained estimation for wide range head pose, in: *Proceedings of the 31st British Machine Vision Conference (BMVC)*. https://doi.org/10.48550/arXiv.2005.10353

Zuo, S., Xiao, Y., Chang, X., Wang, X., 2022. Vision transformers for dense prediction: A survey. *Knowledge-Based Systems* 253, 109552. https://doi.org/10.1016/j.knosys.2022.109552