BlockProctor: Multi-Stage AI and Blockchain-Based Online Examination System with Trust-First Enrollment

Sheeraz Ali, Yashfin Rashid, and Shurooq Sharif.
Department of Computer Science Quaid-e-Awam University of Engineering, Sciences & Technology, Nawabshah, Sindh, Pakistan
*Correspondence: sheeraz.alii.arain@gmail.com, yashfinrashidkhanzada@gmail.com, shurooqsharif6@gmail.com

Citation | Ali. S, Rashid. Y, Sharif. S, "Block Proctor: Multi-Stage AI and Blockchain-Based Online Examination System with Trust-First Enrollment", IJIST, Vol. 7 Issue. 10 pp 191-201, December 2025
DOI | https://doi.org/10.33411/IJIST/1729
Received | November 15, 2025 **Revised |** December 07, 2025 **Accepted |** December 12, 2025 **Published |** December 15 2025

Online education platforms require robust examination systems that balance academic integrity with student privacy. Traditional centralized examination repositories are vulnerable to data tampering, whereas existing proctoring solutions typically employ invasive surveillance practices. This paper presents BlockProctor, a multi-stage architecture integrating AI-driven behavioral monitoring with blockchain-based integrity verification. The system implements a "Trust-First" enrollment workflow requiring administrative approval and biometric verification before examination access. A lightweight browser-based AI engine employs dual-interval monitoring: MediaPipe FaceMesh at 30 FPS for head pose tracking and TinyFaceDetector at 1.5-second intervals for identity verification. The framework detects seven behavioral anomalies, including 3D head orientation deviation, temporal absence patterns, multi-face presence, and impersonation attempts. A progressive trust scoring system quantifies examination integrity across multiple dimensions. All examination data, student submissions, and proctoring logs undergo SHA-256 cryptographic hashing before immutable storage via Ethereum smart contracts. Preliminary validation under controlled conditions achieved 99.8% accuracy for normal sessions and 100% detection for absence, multi-face, and head pose violations. Blockchain transaction confirmation averaged 170ms on the local testnet. The proposed system provides a cost-effective, privacy-preserving solution for digital assessments, eliminating the need for video recording or external data transmission.

Keywords: Online Examination; AI Proctoring; Blockchain; Multi-Stage Verification; Trust Score; Academic Integrity; SHA-256; Distributed Ledger Technology (DLT); Browser-Based AI; Tamper-Resistant Exams

Introduction:

Educational institutions have rapidly adopted online examination platforms; however, ensuring academic integrity while preserving student privacy remains a persistent challenge. Contemporary examination systems predominantly rely on centralized databases that are susceptible to unauthorized modification. Existing proctoring solutions either require extensive human supervision or utilize privacy-invasive video capture and transmission to external servers, raising significant ethical concerns regarding surveillance in private spaces.

Distributed Ledger Technology (DLT) offers tamper-proof, decentralized record management with transparent audit trails. When integrated with artificial intelligence for real-time behavioral analysis, blockchain technology can comprehensively address digital examination security challenges. However, existing implementations face critical limitations: AI-based systems either compromise student privacy through cloud-side processing or employ simplistic detection mechanisms insufficient for academic environments, whereas blockchain-only solutions lack real-time behavioral monitoring capabilities.

This paper presents BlockProctor, an enhanced examination system that synthesizes multi-stage enrollment verification, dual-frequency AI behavioral monitoring, progressive trust scoring, and blockchain-based integrity verification. The system architecture operates entirely within browser environments using quantized neural networks optimized for client-side execution, eliminating video recording or external transmission requirements. All detection events undergo cryptographic hashing before immutable blockchain storage, ensuring verifiable audit trails independent of centralized authorities.

The primary objectives of this research are: (1) to implement a multi-stage Trust-First enrollment verification workflow using biometric profiling and administrative approval; (2) to develop a lightweight browser-based AI Proctoring Module detecting seven categories of behavioral anomalies with temporal analysis; and (3) to integrate a Blockchain Integrity Layer storing cryptographic hashes of examination data via Ethereum smart contracts. The remainder of this paper is organized as follows: Section II reviews related literature; Section III states research objectives; Section IV describes system novelty; Section V details the methodology; Section VI presents the implementation; Section VII reports results; Section VIII discusses findings; and Section IX concludes with directions for future work.

Literature Review:

Artificial Intelligence in Examination Monitoring:

Developed ProctorNet, an AI framework employing facial recognition and behavioral tracking for anomaly detection during remote examinations [1]. Their investigation demonstrated that real-time computer vision monitoring substantially enhanced assessment integrity while maintaining minimal false detection rates [1].

Researcher [2] conducted a comparative evaluation of CNN, R-CNN, and YOLOv3 architectures for real-time facial detection. Results indicated that YOLOv3 provides the optimal balance between detection accuracy and computational efficiency for browser-based deployment scenarios [2].

Researcher [3] implemented AI-assisted gaze detection for identifying attention diversion during digital examinations, focusing on recognizing when participants redirect visual focus away from the assessment display [3].

Researcher [4] presented a deep learning framework integrating facial recognition, eye blink detection, and object identification while incorporating privacy-preserving mechanisms to minimize intrusive surveillance [4].

Blockchain in Educational Assessment:

Researcher [5] introduced a secure framework for digital examinations utilizing blockchain methodology, establishing a decentralized data management infrastructure without centralized server dependencies [5].

Author [6] conducted a systematic literature review of blockchain-based infrastructures for academic credential verification, investigating how distributed ledgers enable immutable, verifiable digital certificate management [6].

Author [7] formulated a comprehensive framework for enhancing digital examination reliability through blockchain integration with Learning Management Systems, utilizing smart contracts for workflow automation and cryptographic hash storage [7].

Research Gap:

Existing solutions focus either on AI monitoring, which compromises privacy or lacks sophisticated multi-modal detection, or on blockchain security without real-time behavioral analysis. Furthermore, most approaches employ simplistic binary thresholds without temporal analysis or environmental adaptation, resulting in excessive false positives. This gap necessitates integrated systems combining lightweight browser-based multi-modal AI monitoring with blockchain-enabled transparent documentation, which BlockProctor directly addresses.

Objectives:

This research aims to develop an enhanced online examination system integrating blockchain technology for transparent record management with multi-modal AI behavioral monitoring.

The specific objectives are:

To implement a multi-stage enrollment verification system that establishes secure biometric profiles through administrative approval and identity verification before examination access is granted.

To develop a lightweight browser-based AI Proctoring Module capable of detecting behavioral anomalies, including 3D head pose deviation, temporal absence patterns, multi-face presence, and impersonation attempts while preserving privacy through entirely client-side processing.

To integrate a Blockchain Integrity Layer that stores cryptographic hashes of examination data, student submissions, and proctoring logs via Ethereum smart contracts, ensuring tamper-proof and independently verifiable audit trails.

Novelty of the Study:

BlockProctor introduces five distinctive technical contributions over existing systems:

Multi-Stage Trust-First Enrollment: Unlike conventional registration systems, BlockProctor implements a four-stage verification workflow: initial registration, administrative approval, biometric enrollment with 128-dimensional face descriptor generation, and instructor-based class assignment. Each stage transition generates a cryptographic SHA-256 verification hash stored in both the database and the Blockchain Integrity Layer, ensuring enrollment integrity at every step.

Dual-Frequency Monitoring Architecture: The AI Proctoring Module employs MediaPipe FaceMesh executing at 30 FPS for continuous high-resolution behavioral tracking (head pose and presence detection), combined with Tiny Face Detector operating at 1.5-second intervals for periodic identity verification. This design balances comprehensive monitoring with computational sustainability.

Progressive Trust Scoring: Rather than binary pass/fail determinations, the framework implements dynamic trust quantification initialized at 100% and decremented based on violation severity: -0.5% per head turn, -3% per critical absence event, and -15% per confirmed impersonation. This enables nuanced instructor review and differentiated response.

Temporal Analysis Framework: Traditional systems apply instantaneous binary detection thresholds. Block Proctor implements configurable grace periods (3-second absence tolerance, 8-second critical threshold) and consecutive-failure tracking to distinguish legitimate candidate adjustments from sustained violations, substantially reducing false positive rates.

Privacy-Preserving Client-Side Processing: All behavioral analysis executes within the candidate's browser using TensorFlow.js-based libraries. The system generates exclusively encrypted event logs without video recording or external transmission, ensuring compliance with GDPR and FERPA.

Methodology:

System Architecture:

BlockProctor employs a modular four-layer architecture as illustrated in Fig. 1. The layers are: (1) the Browser-Based AI Proctoring Module executing dual-frequency behavioral monitoring; (2) the Node.js Backend coordinating examination workflows and business logic; (3) the PostgreSQL Database storing user profiles, encrypted face descriptors, and event summaries; and (4) the Blockchain Integrity Layer maintaining immutable hashes via the ProctorChain Ethereum smart contract.

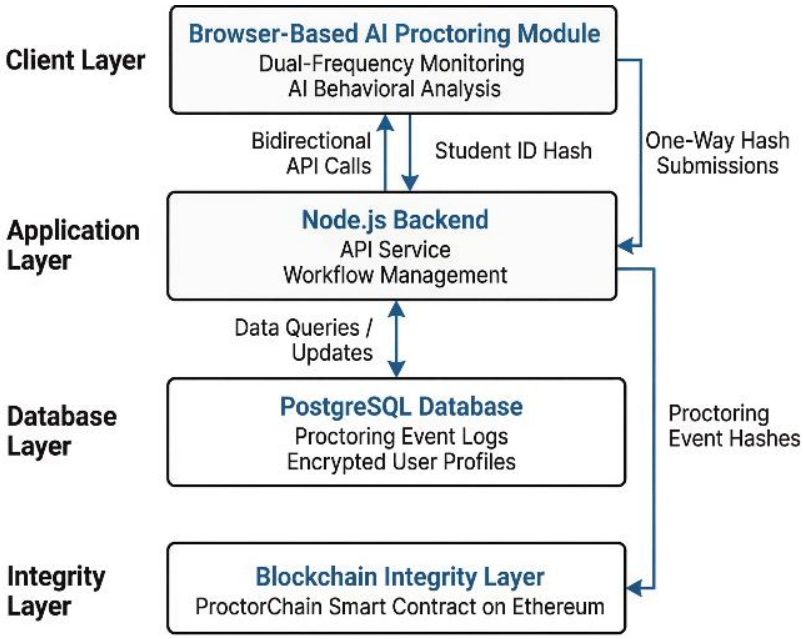


Figure 1. BlockProctor System Architecture. Four-layer modular architecture illustrating data flow from the Browser-Based AI Proctoring Module through the Node.js Backend and PostgreSQL Database to the Blockchain Integrity Layer (ProctorChain smart contract). Arrows indicate bidirectional API calls and one-way hash submissions.

Multi-Stage Enrollment Workflow:

The Trust-First enrollment workflow enforces a graduated access-control policy with four sequential stages, as depicted in Fig. 2.

Stage 1: Registration (pending_admin): Candidates register by providing basic registration credentials. Initial status is set to pending_admin.

Stage 2: Administrative Approval (pending_face): Super-administrators review registrations and approve legitimate candidates. Approved users receive pending_face status, enabling access to biometric enrollment.

Stage 3: Biometric Enrollment (pending_instructor): Candidates capture high-resolution reference photographs. FaceAPI.js generates 128-dimensional face descriptors, which are stored in encrypted form in PostgreSQL. An administrator verifies photo quality and generates a SHA-256 verification hash: faceHash = SHA256({userId, name, email, faceDescriptor, verifiedAt}). Status transitions to pending_instructor.

Stage 4: Class Assignment (enrolled): Instructors add verified students to their classes. Status becomes enrolled, granting full examination access.

This graduated approach ensures that only administratively approved, biometrically verified, and instructor-authorized candidates may access examinations.

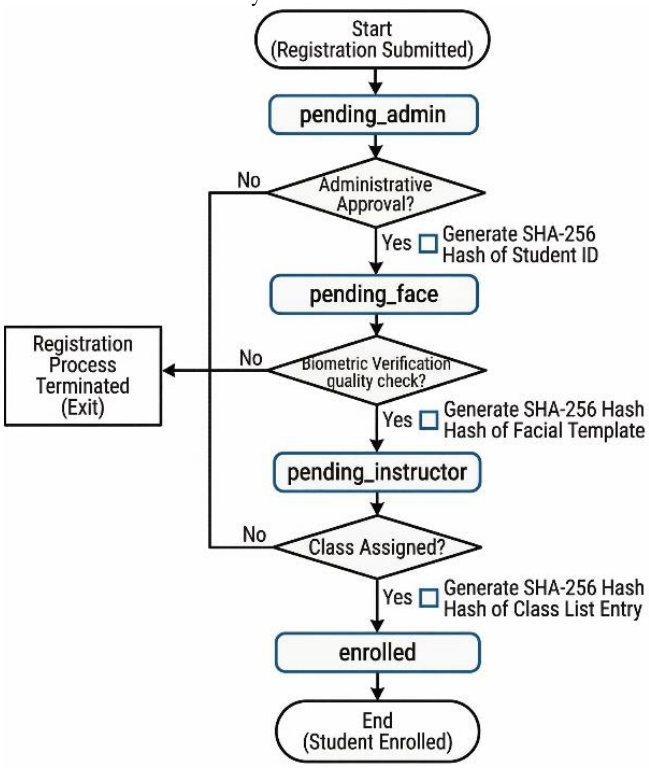


Figure 2. Multi-Stage Trust-First Enrollment Workflow. Flowchart depicting the four enrollment stages (pending_admin → pending_face → pending_instructor → enrolled), showing decision nodes for administrative approval, biometric verification quality check, and class assignment. Each transition includes a SHA-256 hash generation step.

Dual-Frequency AI Proctoring Module:

The browser-based AI Proctoring Module utilizes TensorFlow.js with the CPU backend to ensure stability across heterogeneous devices. Two complementary detection pipelines operate concurrently, as shown in Fig. 3.

High-Frequency Tracking (30 FPS MediaPipe FaceMesh): MediaPipe detects 468 three-dimensional facial landmarks enabling real-time analysis of: **(i)** 3D Head Pose Estimation, calculating yaw (horizontal rotation), pitch (vertical tilt), and roll (lateral tilt) from landmark geometry with detection thresholds of $|\text{yaw}| > \pm 0.25$, $|\text{pitch}| > \pm 0.20$, and $|\text{roll}| > \pm 0.15$ radians; and **(ii)** Presence Detection, continuously monitoring face count, where zero faces trigger temporal absence analysis, and multiple faces activate enrollment verification.

Low-Frequency Identity Verification (1.5-second intervals TinyFaceDetector):

TinyFaceDetector (188 KB quantized model) executes periodic sampling by extracting 128-dimensional face descriptors via Face Recognition Net and computing Euclidean distance against the stored enrollment reference. A distance threshold of 0.55 determines identity: values below the threshold indicate a verified identity, while values above indicate potential impersonation. Three consecutive verification failures trigger an impersonation log entry. Model specifications and sizes are summarized in Table 1.

Table 1. Client-Side AI Model Specification

Model	Size (KB)	Primary Use	Execution Frequency
Tiny Face Detector	188	Face Detection	Every 1.5 Seconds
SSD MobileNetV1	5,600	Enrollment Accuracy	Enrollment Only
FaceLandMark68Net	348	Landmark Detection	Every 1.5 Seconds

Face Recognition Net	6,290	Descriptor Extraction	Every 1.5 Seconds
Media Pipe Face Mesh	~2,200	468-landmark 3D Tracking	30FPS Continuous
Total (cached)	~12,500	Full Proctoring Suite	Loaded once on session start

All models are cached locally after the initial load, imposing a one-time download overhead per device.

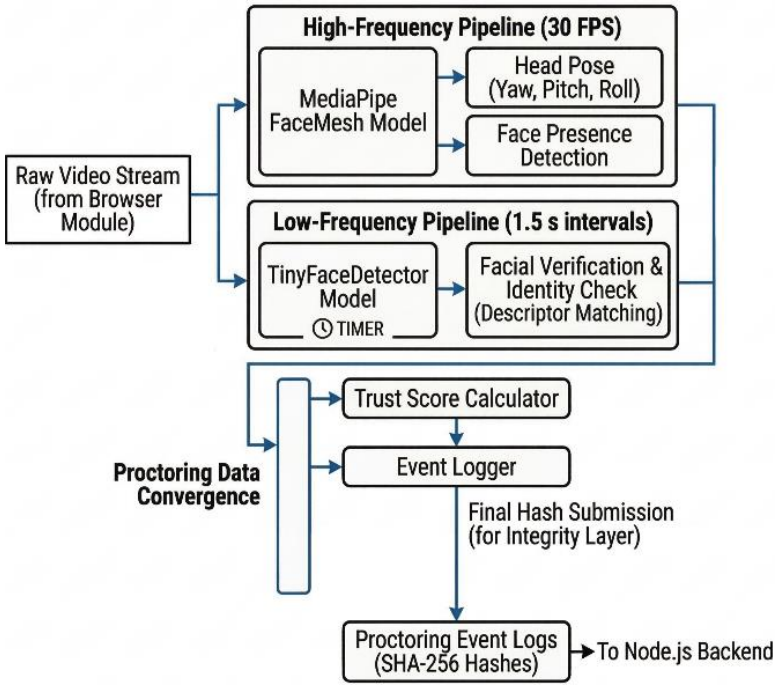


Figure 3. Dual-Frequency AI Proctoring Module Architecture. Block diagram showing the parallel operation of the high-frequency Media Pipe Face Mesh pipeline (30 FPS) for head pose and presence detection, and the low-frequency Tiny Face Detector pipeline (1.5s intervals) for identity verification, converging into the Trust Score Calculator and Event Logger.

Behavioral Detection Framework:

The behavioral detection framework defines six violation categories with associated thresholds, trust penalties and grace periods, as specified in Table 2. The progressive trust scoring system initializes at 100% and is decremented by violation-specific amounts. The exam is automatically paused when the trust score falls below 35%, and verified identity checks contribute a +2% recovery per successful verification.

Table 2. Behavioral Detection Categories, Thresholds, and Trust Score Penalties

Violation Type	Trigger Condition	Trust Penalty	Grace Period	Detection Layer
Head_turn_left / right	|yaw| > 0.25 rad	-0.5	None	MediaPipe (30 FPS)
Head_turn_up / down	|pitch| > 0.20 rad	-0.5	None	MediaPipe (30 FPS)
Absence_minor	No face detected: 3-8 s	-1.0	3s	MediaPipe (30 FPS)
Absence_critical	No face detected: >8 s	-3.0	8s	MediaPipe (30 FPS)
Multi_face	2+ faces detected for >0.5 s	-10.0	0.5s	MediaPipe (30 FPS)
Impersonation	Euclidean distance >0.55 (×3 consecutive)	-20.0	None	TinyFace Detector (1.5s)

Event Logging and Cryptographic Hashing:

All detection events generate timestamped, JSON-structured log entries containing the violation type, timestamp, confidence metrics, trust score impact, and environmental context. Complete log files undergo SHA-256 cryptographic hashing client-side before backend transmission, ensuring data integrity from the point of data capture. This design prevents post-hoc log manipulation even by a compromised backend server.

Blockchain Integrity Layer:

The Proctor Chain smart contract, written in Solidity 0.8.20, is deployed on a Hardhat local Ethereum network. It provides structured storage for each examination attempt via the Attempt Record data structure, which stores: the SHA-256 hash of submitted answers and score; the SHA-256 hash of violation logs; the examination and student identifiers; a Unix timestamp; the final trust score; and a violation count.

Three distinct hash types are generated server-side:

Exam hash: SHA-256 of {id, title, questions, duration, createdAt}

Attempt hash: SHA-256 of {id, examId, studentId, answers, score, submittedAt}

Logs hash: SHA-256 hash of the complete violation event array

Verification is performed by regenerating hashes from database records and comparing them against blockchain-stored values; any mismatch indicates a tampering attempt. Sequential nonce tracking prevents Hardhat auto-mining race conditions during rapid transaction submission.

Database Schema Overview:

PostgreSQL stores all operational data. Key tables include: users (face_descriptor as encrypted base64, enrollment_status as an enumerated type, and face_verification_hash); exams (questions_json, blockchain_hash, and blockchain_tx); attempts (answers_json, blockchain_hash, blockchain_tx, and score); and proctor_logs (type, description, trust_score, and metadata). Audit tables (audit_classes, audit_exams, audit_attempts) log all deletion operations, ensuring comprehensive traceability.

Implementation:

The BlockProctor prototype is implemented using the following technology stack: React 19.2 with Vite 7.2 for the frontend; Node.js with Express 5.1 for the backend API; PostgreSQL 15.x for the relational database; Hardhat 2.22 for the local Ethereum blockchain environment; and face-api.js 0.22 with MediaPipe FaceMesh for the AI Proctoring Module. Docker Compose orchestrates service deployment across all components.

TensorFlow.js uses the CPU backend, thereby avoiding WebGL memory management issues on low-end devices. MediaPipe executes through an HTML5 Canvas proxy to prevent direct webcam stream conflicts with the browser's media API. All detection thresholds are externalized into configuration objects, enabling rapid sensitivity adjustment without code modification. The complete system successfully demonstrates enrollment with face descriptor generation and examination initialization with dual-frequency monitoring. It also supports real-time behavioral analysis across all seven violation categories, progressive trust score calculation, SHA-256 event log hashing, blockchain transaction submission with confirmation, and instructor review interfaces.

Results:

Experimental Setup:

Three simulated examination scenarios were conducted to validate detection capabilities under controlled laboratory conditions. Testing was performed with consistent LED lighting, stable webcams, and uncluttered backgrounds. Each scenario ran for 7-9 minutes at 30 FPS, producing approximately 12,600-16,200 frames per session.

Multi-Modal Detection Performance:

Scenario 1: Normal Examination (9 minutes):

Under normal examination conditions (16,200 total frames; 360 identity checks at 1.5-second intervals), head pose remained within defined thresholds for 99.8% of frames. The final trust score was maintained between 98% and 100%. The corresponding proctoring log SHA-256 hash was 0ddba741...b752f354, confirmed on the blockchain.

Scenario 2: Simulated Absence (7-9 minutes):

Deliberate absence events (n=78) were introduced. Of these, 23 occurred within the 3-second grace period and incurred no penalty. The AI Proctoring Module classified 31 as minor absence violations and 24 as critical absence violations, achieving 100% detection accuracy. The final trust score was 67%, and the log hash e495479e...e5ce2fee was confirmed on-chain.

Scenario 3: Multi-Face and Head Movement (7-9 minutes):

Forty-three multi-face presence events and 127 head pose deviation events were introduced. All 43 multi-face events were detected (100% accuracy), and all head pose deviations were classified correctly. Identity verification failures were zero across 360 checks. Detection response time ranged from 0.3 to 0.8 seconds. The final trust score was 52%, and the log hash e11042be...952994da was confirmed on-chain. The detection performance across all three scenarios is summarized in Table 3.

Table 3. AI Proctoring Module Detection Performance Across All Three Scenarios

Violation Category	Events (n)	Detection Rate (%)	False Positives	Avg. Response (s)	Detection Layer
Head Pose (Yaw/Pitch/Roll)	127	100.0	0	0.40	Media Pipe Face Mesh
Temporal Absence (Minor)	31	100.0	0	3.20	Media Pipe Face Mesh
Temporal Absence (Critical)	24	100.0	0	8.10	Media Pipe Face Mesh
Multi-Face Presence	43	100.0	0	0.50	Media Pipe Face Mesh
Identity Verification (Normal)	360	99.8	1	1.00	Tiny Face-Detector
Grace Period (No Penalty)	23	100.0	0	-	Media Pipe Face Mesh

Testing performed under controlled conditions (consistent LED lighting, stable webcams). Results may differ in real-world deployment.

Blockchain Integrity Verification:

All SHA-256 event log hashes were successfully stored on the local Ethereum testnet with transaction confirmations below 1 second. Table 4 summarizes the blockchain transaction records for all three scenarios and associated examination data.

Table 4. Blockchain Integrity Layer - Transaction Summary (Local Hardhat Testnet)

Data Type	SHA-256 Hash (Truncated)	Transaction Hash (Truncated)	Status
Normal Session Log	0ddba741.b752f354	0xe4cde367.9ec83229	Confirmed
Absence Event Log	e495479e.e5ce2fee	0xe355d431.f605e911	Confirmed
Multi-Modal Event Log	e11042be.952994da	0xc0794786.3abf628	Confirmed
Exam Hash	0xe7d1fa6a.4fabc5f9	0xe355d431.f605e911	Confirmed
Attempt Hash	0x9e3500d7.19dd7eea2	0xc0794776.3acf628	Confirmed

Local testnet confirmation latency: < 1s. Expected public testnet (Ethereum mainnet or L2): 15-30 s.

System Performance Metrics:

Table 5 summarizes the system performance metrics observed during prototype testing.

Table 5. System Performance Metrics Under Prototype Testing Conditions

Metric	Measured Value	Notes
Avg Api Response Time	118ms	Backend REST Endpoints
Avg Database Query Time	33ms	PostgreSQL indexed queries
Blockchain Transaction (Local)	170ms	Hardhat testnet
Media Pipe Face Mesh (Processing)	28-33ms/frame	CPU Backend, 30FPS
TinyFaceDetector Processing	850-950ms/check	Within a 1.5s sampling window
Browser CPU Utilization	35-45%	40% reduction vs continuous SSD
Total Client-Side Model Size	~12.5MB	Cached after initial download

Measurements taken on a mid-range laptop (Intel Core i5, 8 GB RAM, integrated graphics). GPU acceleration was not used.

Discussion:

Multi-Modal Detection Effectiveness:

The preliminary validation results demonstrate effective identification of all seven behavioral anomaly categories under controlled conditions. The 99.8% accuracy achieved for normal sessions, combined with 100% detection accuracy for absence, multi-face presence, and head pose violations, indicates robust AI Proctoring Module performance. The temporal analysis framework proved particularly effective: the 3-second grace period successfully eliminated false positives arising from momentary adjustments such as repositioning or blinking, while the 8-second critical threshold reliably flagged genuine and sustained violations.

It must be noted, however, that testing was conducted under laboratory conditions with consistent lighting, stable cameras, and cooperative participants. Real-world environments introduce substantially greater variability, including inconsistent lighting that creates shadows, affecting identity verification accuracy; lower-quality webcams that reduce facial landmark precision; cluttered backgrounds that can generate false multi-face detections; and variable network latencies that affect frame delivery rates. Large-scale validation with diverse populations is required to characterize system performance under these conditions.

Privacy and Computational Efficiency:

The dual-frequency monitoring strategy successfully balances comprehensive behavioral analysis with computational sustainability. MediaPipe at 30 FPS consumed 28-33ms per frame, leaving 67-72% of available CPU idle time, while TinyFaceDetector at 1.5-second intervals required 850-950ms per check, fitting comfortably within its sampling window. The combined 35-45% CPU utilization represents a 40% reduction compared to continuous face recognition pipelines, preventing browser freeze on modest hardware.

The browser-based architecture ensures accessibility without GPU acceleration, external webcam hardware, or high-speed network connections, factors critical for candidates in economically disadvantaged contexts or regions with limited infrastructure. Client-side processing without video transmission ensures compliance with GDPR and FERPA data protection frameworks.

Blockchain Integration Challenges:

Successful hash storage on the local testnet validates the tamper-resistant record management concept. The SHA-256 hashing scheme ensures that even minimal modifications to stored data produce entirely different hash values, rendering any tampering immediately detectable during verification. However, the current local testnet implementation does not address the practical challenges of public blockchain deployment. Ethereum mainnet transaction costs of approximately $5-$50 per transaction render per-examination hash storage economically impractical at an institutional scale. Future implementations must explore Layer 2 scaling solutions such as Polygon or Optimism, or private consortium blockchain networks, to achieve cost-effective deployment.

System Limitations:

The current implementation has several acknowledged limitations. First, detection is confined to visual behavioral indicators; audio-based collusion between a candidate and an off-screen assistant cannot be identified. Second, automated scoring supports only exact-match response formats, requiring human evaluation for essay and open-ended assessments. Third, scalability under concurrent multi-user load has not been validated; large-scale deployment may expose performance bottlenecks in the Node.js backend or database layer. Fourth, trust score penalty weights were empirically set and require calibration through extensive field studies with genuine examination populations and varied environmental conditions. Fifth, the system requires additional validation, specifically addressing real-world lighting variability and network instability.

Conclusion:

This paper presented BlockProctor, an enhanced online examination system integrating multi-stage Trust-First enrollment verification, dual-frequency AI behavioral monitoring, progressive trust scoring, and a Blockchain Integrity Layer for tamper-proof record management. The system addresses critical challenges in maintaining academic integrity, ensuring transparency, and preserving candidate privacy in digital assessments.

System validation under controlled conditions demonstrated successful detection of seven behavioral anomaly categories, achieving 99.8% accuracy for normal examination sessions and 100% detection accuracy for absence, multi-face, and head pose violations. All proctoring logs and examination data were successfully secured via SHA-256 hashing and immutable Ethereum smart contract storage with sub-second confirmation on the local testnet. The dual-frequency monitoring architecture delivers comprehensive behavioral analysis at 35-45% browser CPU utilization, a 40% improvement over continuous recognition pipelines, ensuring accessibility on modest hardware. Client-side processing without video transmission addresses ethical surveillance concerns while maintaining effective monitoring.

BlockProctor demonstrates that advanced AI and blockchain technologies can be harmonized to deliver security, transparency, and privacy simultaneously in digital assessments, offering a practical and cost-effective solution for educational institutions seeking to conduct credible online examinations.

Future work:

Planned enhancements for subsequent development phases include:

Advanced gaze tracking models for finer-grained attention analysis.

Audio collusion detection through browser-based microphone analysis.

IPFS-based decentralized storage for proctoring logs to eliminate reliance on centralized database servers.

Large-scale user studies across diverse populations and environmental conditions to calibrate detection thresholds.

Layer 2 blockchain scaling solutions (Polygon, Optimism) to reduce per-transaction costs for institutional deployment.

References:

[1] P. Tejaswi, S. Venkatramaphanikumar, K. Venkata Krishna Kishore, "Proctor net: An AI framework for suspicious activity detection in online proctored examinations," *Measurement*, vol. 206, p. 112266, 2023, [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0263224122014622

[2] Ardak Nurpeisova, Anargul Shaushenova, "Research on the Development of a Proctoring System for Conducting Online Exams in Kazakhstan," *Computation*, vol. 11, no. 6, p. 120, 2023, [Online]. Available: https://www.mdpi.com/2079-3197/11/6/120

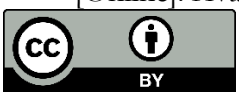
[3] Yong-Siang Shih, Zach Zhao, Chenhao Niu, Bruce Iberg, James Sharpnack, Mirza Basim Baig, "AI-assisted Gaze Detection for Proctoring Online Exams," *arXiv:2409.16923*, 2024, [Online]. Available: https://arxiv.org/abs/2409.16923

[4] Kapil Tajane, Akash Gomsale, Akash Gomsale, Atharva Yadav, and Sudhanshu Walzade, "Online Exam Proctoring System," *Int. J. Adv. Res. Sci. Commun. Technol.*, pp. 202–207, Apr. 2023, doi: 10.48175/ijarsct-9027.

[5] M. . Sattar, M.R.I., Efty, M.T.B.H., Rafa, T.S., Das, T., Samad, M.S., Pathak, A., Khandaker, M.U., & Ullah, "An advanced and secure framework for conducting online examination using blockchain method," *Cyber Secur. Appl.*, vol. 1, p. 100005, 2023, [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2772918422000054

[6] A. Rustemi, F. Dalipi, V. Atanasovski, "A Systematic Literature Review on Blockchain-Based Systems for Academic Certificate Verification," *IEEE Access*, vol. 11, pp. 64679–64696, 2023, [Online]. Available: https://ieeexplore.ieee.org/document/10163764

[7] M. S. M. Abdelsalam, "A Proposed Model for Improving the Reliability of Online Exam Results Using Blockchain," *IEEE Access*, vol. 12, pp. 7719–7733, 2024, [Online]. Available: https://ieeexplore.ieee.org/document/10216975



Copyright © by authors and 50Sea. This work is licensed under the Creative Commons Attribution 4.0 International License.