scientific reports

Check for updates

OPEN

Online exam cheating detection and blockchain trusted deposit based on YOLOv12

Haoliang Wang[1,2]✉, Zarina Shukur[1]✉, Khairul Akram Zainol Ariffin[1], Renhao Xiao[2] & Lili Wang[2]✉

In recent years, online examinations have been widely adopted because of their flexibility, but the covert and diverse nature of cheating behaviour poses a serious challenge to the fairness and integrity of examinations. Existing anti-cheating techniques are deficient in detecting diverse cheating behaviours in real-time and ensuring the credibility of evidence. To address this problem, this paper proposes an integrated solution for online exam cheating detection based on the lightweight YOLOv12 model and blockchain trusted depository. Firstly, we made targeted lightweight improvements to the benchmark YOLOv12n model by removing the computationally intensive Attention mechanism from the backbone network and simplifying the module structure (modifying the A2C2f module), as well as replacing the computationally heavy C3k2 module in the head network with the efficient C3Ghost module. These modifications aim to reduce the model's computational complexity and number of parameters, increasing inference speed, thus making it more suitable for real-time detection tasks. Secondly, to address the issue of credible evidence preservation concerning cheating, we constructed a evidence preservation system based on the Hyperledger Fabric consortium blockchain, combined with IPFS distributed storage technology. Key screenshots of suspected cheating behaviors are stored on IPFS, and their content identifier (CID) along with detection metadata (such as timestamp, detection type, confidence, etc.) is recorded on the blockchain through smart contracts, ensuring the originality, integrity, and immutability of the evidence. Experiments conducted on an online exam cheating dataset containing categories of 'person' and 'electronic devices' demonstrate that the proposed lightweight YOLOv12NoAttn model exhibits competitive detection performance (with slight improvements in mAP50 and Recall) while showing higher efficiency by significantly reducing parameters (approximately 28%) and GFLOPs (approximately 13%). Ablation experiments further verify the effectiveness of the lightweight improvements made to both the backbone and head networks. This research provides an efficient, accurate, and trustworthy solution for cheating detection and evidence management in online examinations, contributing to the maintenance of fairness and integrity in online assessments.

Background and challenges of online examination cheating

In recent years, with the rapid development of information technology and the transformation of global education models, online education and distance learning have become an important part of higher education and vocational training. Especially under the influence of global events (e.g., the COVID-19 pandemic), online examinations have been widely adopted as a flexible and efficient assessment method[1]. Online exams break the time and space constraints of traditional offline exams and provide great convenience for learners.

However, the popularity of online exams has also brought a series of new challenges, one of the most prominent and urgent problems is how to effectively prevent and detect cheating[2]. Compared with traditional offline exams, online exams are usually conducted in uncontrolled remote environments, and candidates may be more likely to cheat by using external resources, seeking assistance from others, or using unauthorised devices, which greatly increases the covertness and diversity of cheating behaviours[3]. Cheating not only undermines the fairness of examinations and the validity of assessment results, but also poses a serious threat to academic integrity and the credibility of the education system[4].

[1]Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), 43600 Bangi, Selangor, Malaysia. [2]School of Artificial Intelligence, Dongguan City University, Dongguan 523419, Guangdong, China. ✉email: p109025@siswa.ukm.edu.my; zarinashukur@ukm.edu.my; wangll@dgcu.edu.cn

The critical role of trust in online examination systems

Trust plays a pivotal role in online examination cheating detection systems, serving as the cornerstone for maintaining system integrity and ensuring fair assessment practices. The importance of trust in this context can be understood through several key dimensions that fundamentally impact the effectiveness and acceptance of online examination systems.

Trust is essential for maintaining the credibility of online examination systems to ensure fairness and academic integrity[5]. As educational institutions increasingly rely on digital platforms for assessment, the establishment of trustworthy systems becomes paramount to preserve the fundamental principles of fair evaluation. Without robust trust mechanisms, the entire foundation of online education and assessment could be undermined, leading to a crisis of confidence in digital learning platforms.

A critical challenge in this domain is the phenomenon of behavior change, where new forms of cheating behaviors often exceed the ability of traditional models to identify established cheating patterns[6]. As Azzedin and Ridha demonstrated in their seminal work on peer-to-peer systems, behavioral changes can significantly impact the performance of existing honesty checking mechanisms. This principle directly applies to online examination contexts, where evolving cheating strategies continuously challenge the detection capabilities of conventional systems. The dynamic nature of cheating behaviors necessitates adaptive trust models that can accommodate and respond to these changes effectively.

By embedding behavioral trust concepts such as reputation assessment into examination systems, institutions can significantly enhance system reliability and promote ethical examination practices[7,8]. These trust-based approaches can involve tracking students' honest behavioral patterns over time or implementing reputation-based scoring mechanisms. Such systems create a comprehensive profile of student behavior that extends beyond individual examination sessions, enabling more accurate and contextual assessment of potential cheating incidents[5]. The integration of reputation systems, as extensively studied in peer-to-peer environments, provides valuable insights for developing trust-aware examination platforms that can distinguish between genuine behavioral variations and deliberate cheating attempts.

The implementation of multi-layered trust frameworks serves a dual purpose: not only does it help deter cheating behaviors, but it also provides institutions and students with confidence that the assessment process is transparent and fair[8]. This comprehensive approach to trust ultimately enhances confidence in digital education platforms by creating an environment where all stakeholders can rely on the integrity and reliability of the assessment system. When students trust that the system fairly evaluates all participants and institutions trust that the results accurately reflect student capabilities, the entire educational ecosystem benefits from increased credibility and effectiveness.

Ensuring the integrity of online examinations faces multiple challenges. The uncontrolled nature of the environment makes it difficult for invigilators to fully monitor the behaviour of candidates and their surroundings, and candidates may cheat using multiple screens or using devices such as mobile phones. The diversity and intelligence of cheating methods continue to evolve, making it difficult for traditional rule-based or simple behavioural analysis methods to cope with the proliferation of new cheating methods. Real-time detection presents another significant challenge, as abnormal behaviours need to be detected and warned instantly during the examination process, which requires the detection system to have efficient and accurate real-time processing capabilities. Obtaining and credibly documenting evidence of cheating remains a key challenge. Even if suspected cheating is detected, how to obtain clear, objective and untamperable evidence for subsequent identification and processing is an important part of safeguarding the fairness of the examination and maintaining the reputation of the institution.

To address these challenges, academics and industry have proposed a variety of technical and management measures. Common technological tools include identity verification (e.g., biometrics, knowledge quizzes), browser locking (restricting candidates' access to computer resources), and remote invigilation based on video surveillance[9]. However, each of these methods has its limitations. Identity verification mainly solves the problem of 'who is taking the test', but cannot effectively monitor cheating behaviour during the test[10]. Locking the browser can only restrict the internal operation of the computer, but not the use of external devices or seek help from others to cheat. Manual remote invigilation, while providing some degree of monitoring, faces problems such as high cost, poor scalability, invigilator fatigue, and subjective judgement. Automatic detection methods based on early computer vision technologies may suffer from low accuracy, high false alarm and omission rates, difficulty in identifying complex behaviours, etc., and how to ensure the authenticity and undeniability of the evidence of detected cheating is still an under-solved problem[11].

Therefore, there is an urgent need for a more advanced and robust technological solution that can effectively detect diverse cheating behaviours in online examinations in real time and provide a trustworthy mechanism to securely store and manage the cheating evidence, in order to cope with the severe challenges posed by current online examination cheating.

Limitations of existing anti-cheating technologies

A variety of technological and managerial measures have been proposed by academia and industry to address the increasingly prominent problem of cheating in online examinations. These approaches mitigate the integrity risk of online exams to a certain extent, but each has obvious limitations and fails to provide a comprehensive, efficient and trustworthy solution. Existing mainstream anti-cheating technologies mainly include identity verification, locked browsers, rule-based behavioural analysis, and manual remote invigilation.

First, identity verification-based technologies (e.g., biometrics, knowledge quizzes) are mainly used to confirm that it is the registered student himself/herself who is taking the exam[12]. However, the core of these technologies is to solve the problem of 'who is taking the test', and cannot effectively monitor the behaviour of candidates during the test. Once authenticated, candidates may still use various means to cheat, such as accessing

unauthorised materials, using assistive devices or seeking help from others. As some scholars have pointed out, authentication does not address the more complex issue of 'verifying that a candidate does not have the help of others or the support of resources not permitted by the instructor'[13]. Therefore, relying on authentication alone cannot guarantee the integrity of the examination process.

Second, Secure Browsers technology is designed to restrict candidates from accessing other applications, websites, or copy-and-paste content on the computer during an exam[14]. This method is useful in preventing examinees from switching windows on the exam computer to search for answers or use local files. However, its limitation is that it only controls the computer environment that the examinee is using. The lockdown browser cannot do anything about cheating by using external devices (e.g., smartphones, tablets, smartwatches) to communicate or access information. It is also possible for a candidate to be assisted by another computer or by someone else in the room, and these behaviours are also beyond the scope of the Lockdown Browser's monitoring.

Behavioural analysis methods based on early computer vision or simple rules attempted to detect anomalous behaviours by analysing the video stream of the examinee, such as prolonged periods of time when the eyes stray away from the screen, a large head turn, or the detection of a second person in the room[15]. Despite the potential efficiency benefits of automated detection, these early methods often suffered from the following problems:

Insufficient accuracy and robustness Simple rules or models are difficult to cope with complex and changing examination environments (e.g., light changes, background interference) and subtle behavioural changes of examinees, which can easily lead to high false alarm rates (misclassifying normal behaviours as cheating) and underreporting (failing to detect cheating behaviours that actually occur)[16].

Limited detection scope Most methods focus on head posture or face detection, making it difficult to effectively identify specific cheating tools (e.g., mobile phones, headphones) or more subtle means of cheating (e.g., screen-sharing, micro-cameras) used by candidates.

Difficulty in adapting to new forms of cheating With the development of technology, cheating methods continue to evolve, such as the use of AI tools to assist in answering questions and the use of invisible headphones, etc. These new types of cheating behaviour often exceed the ability of traditional rules or models to identify.

Manual remote invigilation is a widely used method of real-time monitoring by human invigilators remotely viewing video streams of candidates[17]. This method can provide more flexible and comprehensive monitoring to some extent, but its drawbacks are also prominent:

High costs and poor scalability A single invigilator can usually only monitor a limited number of candidates at the same time, and large-scale examinations require a large number of invigilators, leading to a sharp increase in costs and difficulty in coping with unexpected large-scale examination demands.

Invigilator fatigue and subjectivity Prolonged, high-intensity monitoring can easily lead to invigilator fatigue and reduced concentration, which can lead to missed cheating behaviour. At the same time, the subjective judgements of different invigilators may lead to inconsistent detection standards.

privacy worry Constant video surveillance may trigger privacy concerns and discomfort for candidates.

Technology dependency and blind spots Dependent on a stable network connection and high-quality video streaming, any technical glitch may affect the effectiveness of invigilation. In addition, there are blind spots in the camera's view, which may be exploited by candidates to cheat.

Finally, a pervasive and often overlooked limitation is the issue of credible deposit of evidence of cheating. Even if suspected cheating is detected and videos or logs are recorded through one of the methods described above, it is an important challenge to ensure the originality, integrity and inerrancy of such evidence for subsequent impartial investigation and processing. The traditional way of storing evidence is susceptible to tampering or forgery and lacks sufficient credibility, which may make it difficult for cheating to be effectively recognised and punished, weakening the deterrent effect of anti-cheating measures.

Existing anti-cheating technologies for online exams each have obvious limitations. Whether it is identity verification, locked browsers, early automated detection or manual invigilation, none of them can individually or jointly provide a comprehensive, accurate, efficient and trustworthy solution to deal with the increasingly complex and covert online exam cheating behaviours. In particular, the significant shortcomings of existing technologies in detecting diverse cheating behaviours in real-time and ensuring the credibility of cheating evidence provide the need and research space for this study.

Contribution of this article

In this study, we propose an integrated system for online exam cheating detection based on the lightweight YOLOv12 model and blockchain trusted depository to address the current challenges of online exam cheating detection and the lack of credibility of cheating evidence. The main contributions of this paper are summarised as follows:

Proposed YOLOv12 Cheating Detection Model Based on Lightweight Improvement In this paper, based on an in-depth analysis of the characteristics of online exam cheating behaviour, we make structural lightweight improvements to the state-of-the-art target detection model YOLOv12n. Specifically, we optimise the A2C2f module in the backbone network (by removing the Attention mechanism, simplifying the structure, and reducing the number of stacking) and the C3k2 module in the head network (by replacing it with the more efficient C3Ghost module). With these improvements, a detection model with higher computational efficiency and significantly reduced number of parameters is successfully constructed. The experimental results show that compared with the baseline YOLOv12n, the improved model maintains good performance on the cheat detection task, especially in the mAP50 and Recall metrics, which verifies the effectiveness of the lightweight strategy and its adaptability to the real-time and high-efficiency detection needs of online exams.

Constructed a blockchain-based credible deposit mechanism for cheating evidence Creatively applying blockchain technology to the deposit of cheating evidence in online exams. In this paper, an evidence deposit

system based on Hyperledger Fabric federation chain is designed. When suspected cheating is detected, the relevant key screenshots are stored in the IPFS distributed file system, and the IPFS content identifier (CID) of the screenshot is recorded on the blockchain along with the detection metadata (e.g., timestamps, detection type, confidence level, etc.). This mechanism ensures the originality, integrity and non-tamperability of the evidence of cheating, provides credible technical support for the subsequent identification and processing of cheating, and effectively solves the problem that traditional evidence storage methods are easy to be tampered with and lack credibility.

Realised an integrated system of cheating detection and credible deposit This paper seamlessly integrates the real-time cheating detection module based on the lightweight YOLOv12 model with the blockchain-based credible deposit mechanism to construct an end-to-end online examination integrity guarantee system. The system is capable of real-time automated detection of cheating behaviours and automatically completes the secure and reliable recording of evidence to form a complete evidence chain. This integrated solution provides a comprehensive, efficient and highly credible anti-cheating solution for online exams, improving the efficiency and credibility of online exam supervision.

The core contribution of this research lies in the targeted lightweight improvement of the YOLOv12 model to adapt to the task of online exam cheating detection, and the creative introduction of blockchain technology to solve the problem of credible evidence, which provides a new idea and technical support for constructing a more secure, fair, and credible online exam environment.

Overview of relevant technologies

Target detection techniques and YOLO models in behaviour recognition

Computer vision is a central branch of the field of artificial intelligence, one of whose goals is to give machines the ability to understand and interpret image and video content. Among the many computer vision tasks, Object Detection (OD) occupies an important place, aiming at identifying specific classes of targets in an image or a video frame and determining their precise location and size (usually represented by a bounding box). This technique is not only fundamental to image understanding, but also an indispensable preprocessing step for many advanced visual tasks such as target tracking, scene understanding and behaviour recognition. Meanwhile, action recognition, as another important research direction, is dedicated to analysing and determining the movements or activities of people or objects in video sequences, which has a wide range of application needs in the fields of intelligent surveillance, human-computer interaction, and healthcare. Figure 1 is an example diagram of target detection.

Early target detection methods mostly relied on hand-designed features and machine learning classifiers, such as detectors based on Haar features and Adaboost[18] or pedestrian detectors based on HOG features and SVM[19]. With the rapid development of deep learning technology, breakthroughs have been made in the field of target detection, and a large number of excellent models based on convolutional neural networks (CNNs) have emerged. These models can be broadly classified into two categories: two-stage (Two-stage) detectors and one-stage (One-stage) detectors. Two-stage methods, such as the R-CNN family (including Faster R-CNN[20]), first generate candidate regions, and then perform classification and bounding-box regression on these regions, while one-stage methods, such as the SSD[21] and the YOLO family, directly predict the classes and locations of targets in an image. Compared to two-stage methods, single-stage detectors typically have faster inference speeds, making them more suitable for application scenarios that require high real-time performance. Figure 2 illustrates a comparison of the two-stage and single-stage target detection method flow.

Among single-stage target detectors, the YOLO (You Only Look Once) model has attracted much attention due to its unique design concept and excellent real-time performance.The core idea of YOLO is to treat the target detection task as a regression problem by predicting the bounding box and category probabilities directly from the whole image through a single neural network[22]. This end-to-end (End-to-End) detection approach significantly improves the detection speed. Since YOLOv1 was proposed, the YOLO family of models has

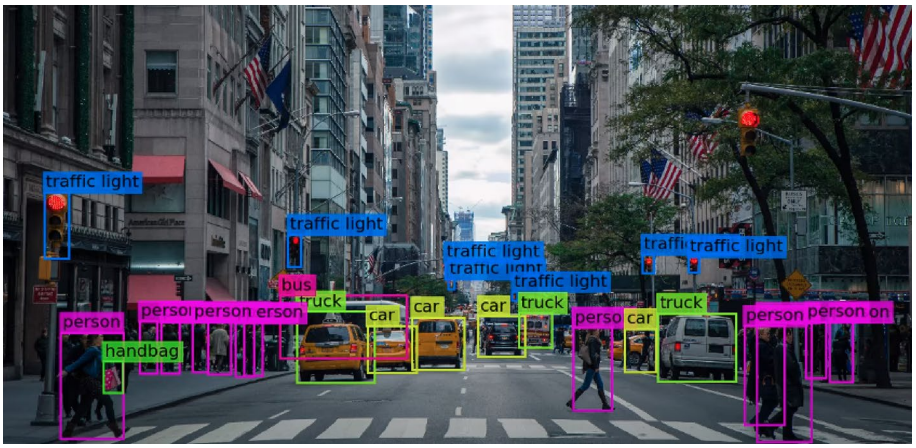

Fig. 1. Target detection example diagram.

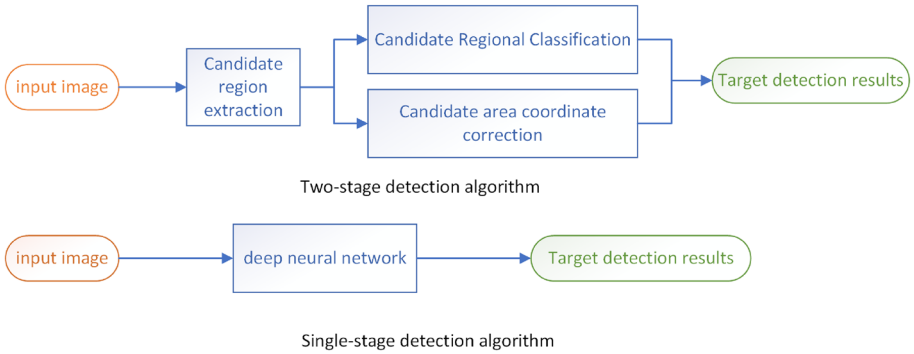

Fig. 2. Two-stage versus single-stage target detection method flow comparison.

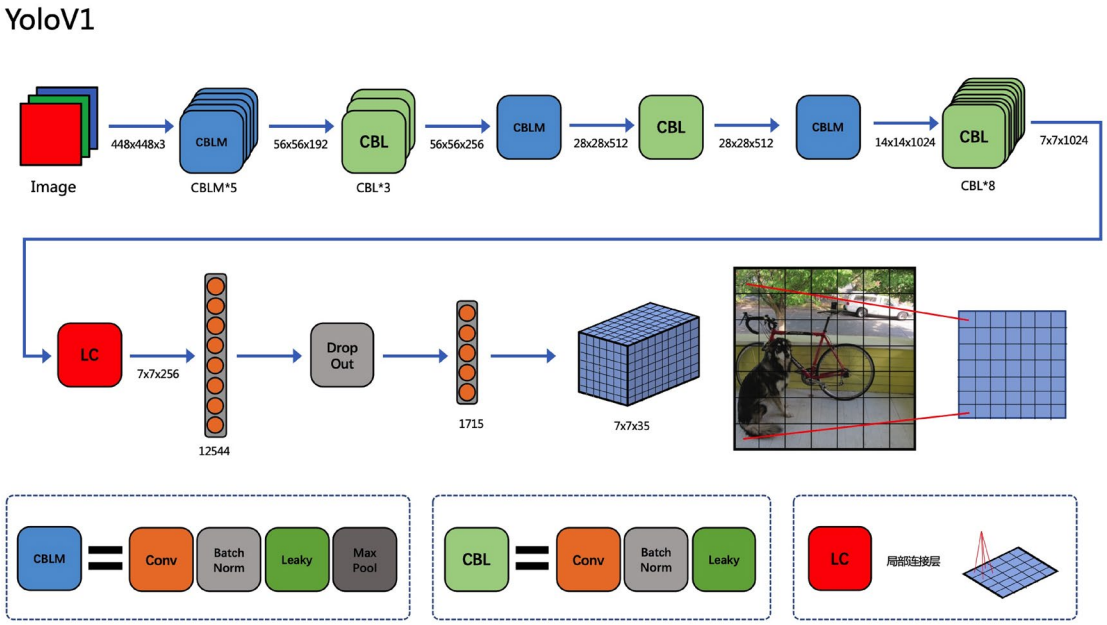

Fig. 3. Network structure diagram of YOLOv1.

undergone continuous iterations and optimisations, including the introduction of Anchor Boxes[23], multi-scale prediction[24], and the integration of various state-of-the-art training techniques and network structure improvements[25,26], to continuously achieve a better balance between detection accuracy and speed. This study is based on the YOLOv12 model, a version that provides good detection performance while maintaining high efficiency[27]. Figure 3 shows the network structure of YOLOv1.

Target detection techniques play a fundamental role in behaviour recognition. By accurately detecting key targets in video frames, such as characters, hands, heads, and behaviourally relevant objects (e.g., mobile phones, books, computers, etc.), we can obtain precise spatial information about the subjects in the scene and the objects they interact with[28]. These detection results can be used as inputs for subsequent behavioural analysis modules, e.g., to analyse the posture and movement trajectory of a character by tracking its bounding box and keypoints[29]; to determine the presence of a specific behaviour (e.g., looking down at a mobile phone, flipping through a book) by detecting a specific object and its relative position to the character[30]; or to determine the presence of multi-person collaborations by detecting the number of characters appearing in the frame, etc. situations[31]. Therefore, high-quality target detection is a key prerequisite for accurate behaviour recognition, especially for the detection of complex or abnormal behaviours.

Target detection technology is one of the core technologies in the field of computer vision[32], and the YOLO series of models has become a representative of single-stage target detection by virtue of its excellent real-time performance and continuously improving detection accuracy. Applying YOLO models to behaviour recognition tasks can effectively extract key target information from video streams and provide a reliable spatial basis for subsequent behaviour analysis[33]. Especially in scenarios that require real-time monitoring and abnormal behaviour detection (e.g., online exam proctoring), YOLO-based target detection technology can quickly and

accurately locate potential violating objects and behavioural subjects, which provides a solid technical support for the construction of an efficient and reliable behavioural recognition system[34].

Application of blockchain technology to data depositories

Blockchain technology, as a decentralised, distributed ledger technology, was initially well known for its use in cryptocurrencies[35]. However, its core characteristics - including Immutability, Transparency, Traceability, and security through cryptography and consensus mechanisms - -allow it to go beyond the financial domain and show great potential in scenarios that require a high degree of trust and data integrity[36]. Data Notarisation, i.e., proving the existence and integrity of data at a specific point in time, is a key requirement in many application domains (e.g., digital copyright protection, e-contracts, judicial evidence, supply chain traceability, etc.). Traditional depository methods often rely on centralised third-party institutions, with problems of inefficiency, high costs and trust risks. The emergence of blockchain technology provides a new mindset and technological foundation to address these challenges. Figure 4 illustrates the architectural diagram of the blockchain and IPFS data storage system.

The core of the application of blockchain in the data deposit lies in the use of its tamper-proof distributed ledger to record the 'fingerprint' of the data, rather than the data itself. Specifically, the user first hashes the original data to be deposited, generating a fixed-length, unique digital digest (Hash Value)[37]. Even if the original data undergoes minor alterations, its hash value will change significantly. Subsequently, this hash value is recorded on the blockchain as part of the transaction information. Since blockchain transactions are extremely difficult to tamper with or delete once they have been packed into a block and confirmed by a consensus mechanism[38], the hash value recorded on the chain becomes a strong proof that the data existed at a particular point in time and has not been tampered with. By querying the transaction records on the blockchain, it is possible to verify whether a hash value has been recorded at a specific time, and thus verify the integrity of the corresponding data.

The use of blockchain for data deposit brings multiple advantages. First, it significantly enhances the trustworthiness of data. Due to the decentralised nature of blockchain, the depository record does not depend on any single institution, avoiding the single point of failure and the trust risk that may be associated with centralised institutions[39]. Secondly, the tamperability provided by blockchain ensures that the depository records cannot be maliciously modified once generated, providing a technical guarantee for the integrity of the data. In addition, transactions on the blockchain are often accurately timestamped, providing verifiable proof of time for the data. This technology-based rather than institutional trust mechanism makes the data deposit process more efficient, transparent, and relatively low-cost, which is particularly suitable for large-scale and high-frequency deposit requirements[40].

Blockchain technology provides an innovative and reliable solution for data deposit by virtue of its inherent decentralised, tamper-proof and traceable characteristics. By recording the hash value of data on a distributed ledger, blockchain can provide strong proof of existence and integrity for digital assets and information, effectively addressing the challenges of traditional deposit methods. This application not only improves data credibility and security, but also lays a solid foundation for the authentication, circulation and verification of all

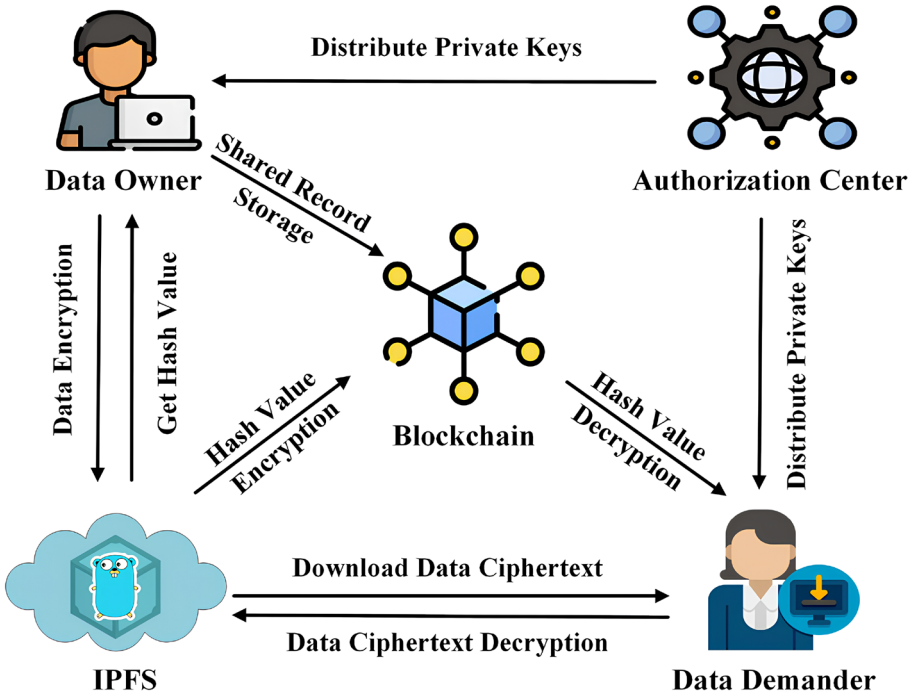

Fig. 4. Architecture diagram of data depository system for blockchain and IPFS.

kinds of data in the digital economy era, showing broad application prospects in a variety of fields, such as digital copyright, e-government, financial services, etc.[41].

Overview of IPFS distributed storage technology

Traditional Internet data access relies heavily on Location-addressed Hypertext Transfer Protocol (HTTP), where data is fetched by specifying the domain name or IP address of a server. This model suffers from the risk of a single point of failure, inefficiency (especially for popular content), and vulnerability to censorship. To address these challenges, the InterPlanetary File System (IPFS) has been proposed, which is a Peer-to-Peer (P2P) distributed file system that aims to connect all computing devices via Content-addressed, building a more robust , persistent, and open network[42].The core idea of IPFS is to make the data itself the key to addressing, rather than the location where the data is stored.

The key to enabling content addressing in IPFS is cryptographic hashing of the data. When a file is added to an IPFS network, it is first split into several data blocks (Blocks). Then, a unique cryptographic hash is calculated for each block. These hashes are called Content Identifier (CID). The root CID of a file is jointly determined by the CIDs of all its data blocks and the linking relationships between them[43]. Thus, the CID not only identifies the content of the data, but also implicitly contains the integrity checking information of the data. Any minor changes to the content of the data will result in a change in its CID. When a user requests a CID, the IPFS network looks up and fetches the corresponding block of data based on this CID in nodes around the world, without needing to know exactly which server the data is stored on.

The network architecture of IPFS is based on a decentralised P2P network. Each node in the network can store, request and provide data blocks. Relationships between data blocks are maintained through a data structure called the Merkle Directed Acyclic Graph (Merkle DAG)[44].The Merkle DAG ensures data integrity and tamper-proofness, as well as supports data de-duplication (since blocks with the same content have the same CID) and version control. When a node owns a block of data corresponding to a certain CID, it can participate in data sharing by announcing to other nodes that it owns the data. Retrieval of data can be performed from any node that owns that data block, and preference is usually given to the closest or most responsive node on the network topology, which significantly improves the efficiency of data access and resistance to single points of failure[45]. Figure 5 illustrates the data processing flowchart for blockchain and IPFS.

IPFS provides a distributed storage solution different from traditional HTTP by introducing content addressing and building a decentralised P2P network. Its hash-based CID ensures data integrity and tamperability, while the P2P network and Merkle DAG structure enhance data availability, transmission efficiency, and censorship resistance[46].IPFS technology shows broad application prospects, providing important technical support for building a more robust and open Internet infrastructure.

In order to more clearly illustrate the limitations of existing technologies and the targeted contributions of this study, we summarize the various technical gaps discussed above and explain how the integrated solution proposed in this paper addresses these gaps. The specific details are shown in Table 1.

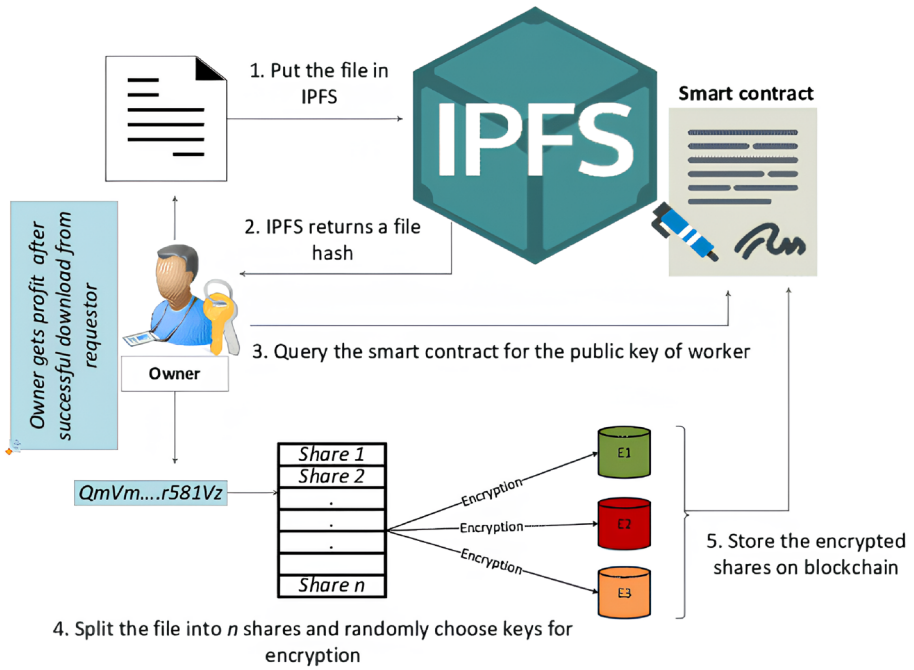

Fig. 5. Data processing flowchart for blockchain and IPFS.

Existing technology	Identified gaps / limitations	How this paper addresses the gaps
Identity verification	Limited to pre-exam identity confirmation; fails to monitor in-exam cheating behaviors.	The proposed system employs a lightweight YOLOv12 model for real-time detection, continuously monitoring the video feed to automatically identify suspicious objects and behaviors.
Secure browsers	Only restricts operations on the exam computer; cannot prevent the use of external devices (e.g., smartphones) or receiving assistance from others.	Our YOLOv12 model is specifically trained to detect 'electronic devices' and multiple 'persons' in the frame, directly addressing cheating via external aids or assistance.
Early automated detection	Insufficient accuracy and robustness, leading to high false positives and negatives. Limited detection scope (e.g., only head pose) and difficulty adapting to new cheating methods.	We use an advanced YOLOv12 model with targeted lightweight improvements, enhancing real-time performance while maintaining high detection accuracy. The model offers a broader and more reliable detection scope.
Manual remote invigilation	High cost and poor scalability. Prone to proctor fatigue and subjective judgment biases, leading to missed cheating behaviors.	The automated system provides continuous, objective, and scalable monitoring, significantly reducing reliance on human proctors and eliminating issues of subjectivity and fatigue.
Traditional evidence storage	Evidence stored on centralized servers is vulnerable to tampering or deletion, lacking sufficient credibility and legal validity.	We constructed an evidence depository system using Blockchain (Hyperledger Fabric) and IPFS. The hash (CID) and metadata of cheating evidence are recorded on-chain, ensuring it is immutable, traceable, and trustworthy.

Table 1. Summary of gaps in existing anti-cheating technologies and contributions of this paper.

Privacy-preserving technologies for online examination systems

Privacy protection has become a critical concern in online examination systems, where continuous video monitoring raises significant privacy implications for students. Recent advances in privacy-preserving technologies offer promising solutions to address these challenges while maintaining system effectiveness.

On-device Computing and Edge AI: On-device recommendation systems and processing have demonstrated significant potential in preserving user privacy by keeping sensitive data locally[47]. In the context of online examinations, deploying lightweight detection models directly on students' devices can minimize the transmission of raw video data to central servers, thereby reducing privacy risks while maintaining real-time detection capabilities.

Federated Learning for Privacy-Preserving Model Training: Federated learning enables the training of machine learning models without centralizing sensitive data. Privacy-preserving data contribution methods, such as those proposed in federated recommender systems[48], can be adapted for online examination scenarios. This approach allows institutions to collaboratively improve cheating detection models while ensuring that individual student data remains on local devices.

Differential Privacy: Differential privacy techniques can be integrated into the inference process to protect individual identities while maintaining system utility. By adding calibrated noise to detection results, the system can provide privacy guarantees without significantly compromising detection accuracy.

These privacy-preserving approaches represent essential directions for developing more ethical and compliant online examination systems that balance security needs with fundamental privacy rights.

System design

YOLOv12n model lightweight improved design

To address the characteristics of the cheating behaviour detection task, we have made targeted improvements to the structure of the benchmark YOLOv12n model. Through preliminary experiments, we found that the visual features of cheating behaviours are relatively simple and do not require much extreme feature extraction and fusion capability of the model, and the accuracy of the existing model can already meet the demand. Therefore, the core of our optimisation focuses on reducing the computational complexity and number of parameters of the model to achieve a lightweight model, which can significantly improve the inference speed and make it more suitable for actual deployment scenarios. The improvement mainly focuses on the backbone network and the header network.

Backbone Lightweighting

The original YOLOv12n model uses the A2C2f module in the deeper part of the backbone network, i.e., in the processing of the P4/16 (Layer 6) and P5/32 (Layer 8) feature maps.The A2C2f module is designed to enhance feature representation by combining region-based Attention blocks (ABlock) and convolutional blocks (C3k).

Considering the relative simplicity of cheating features, we believe that over-reliance on Attention mechanisms imposes an unnecessary computational burden. In order to reduce the computational effort and the number of parameters, we modify the internal structure of the two-layer A2C2f module:

Remove Attention mechanism: Remove the region-based Attention block (ABlock), which is computationally expensive.
Simplify the structure: We replace the function of the original Attention block with the convolution-based C3k block, so that the modified module contains only C3k blocks, and the Attention mechanism is no longer introduced in the backbone network.
Reducing the number of stacking: we reduce the number of C3k blocks stacked inside the two-layer module from the original 4–2 times.

It is evident that these modifications result in a substantial reduction in the computational burden of layers 6 and 8 of the backbone network, while ensuring the retention of adequate feature extraction capability. The configuration of the specific modified A2C2f module is illustrated in Fig. 6.

Head network lightweighting

In the head network of the model, layer 20, which is responsible for processing the maximum perceptual field feature map (P5/32), the original design adopts the computationally intensive C3k2 module, and is configured as c3k=True. According to the source code of the module, C3k2 is inherited from the C2f structure, and when c3k=True, its internal processing sequence consists of n C3k modules. According to the module source code,

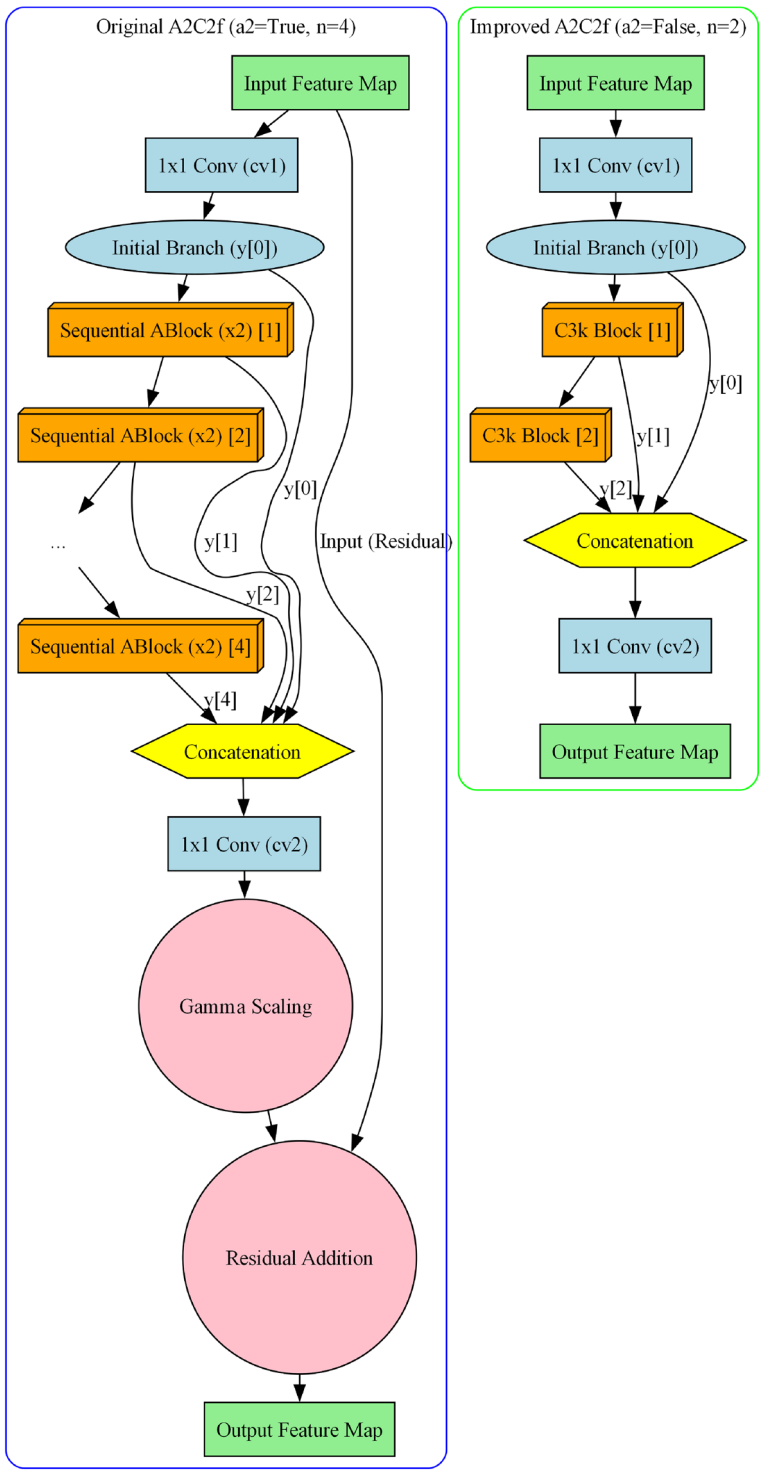

Fig. 6. Structure of the original A2C2f module and the modified module.

C3k2 inherits from C2f structure, when c3k=True, its internal processing sequence consists of n C3k modules. This structure based on the C2f framework with multiple layers of internally nested C3k (containing 2 Bottleneck) leads to a higher computational cost and number of parameters.

To further optimise the efficiency of the header network, we replaced the C3k2 (c3k=True) module at layer 20 with the lightweight C3Ghost module.

According to the source code of the module, C3Ghost inherits from the C3 structure, and its core lies in the internal use of the GhostBottleneck module, which is an efficient convolutional module that generates 'ghost' feature maps to significantly reduce the number of features required for traditional convolutional operations while maintaining feature diversity. GhostBottleneck is an efficient convolution module that significantly reduces the amount of computation and number of parameters required by traditional convolution operations while ensuring feature diversity. In this study, we use the C3Ghost module which contains one GhostBottleneck module[49].The structure of the C3k2 module and the C3Ghost module is shown in Fig. 7.

This substitution replaces the C3k2(c3k=True) module, which is based on the C3 structure and uses the efficient GhostBottleneck, with the C3k2(c3k=True) module, which is based on the C2f structure and internally nested with the standard Bottleneck, resulting in a significant reduction of the computational burden on the header network, and a further increase in the efficiency of the model inference and training.

The improved YOLOv12 network structure is shown in Fig. 8. These improvement strategies are not blindly deleting specific modules, but making targeted adjustments based on an in-depth analysis of the characteristics of the cheat detection task. By replacing the computationally expensive Attention module with a more efficient convolutional structure and employing the lightweight GhostBottleneck (C3Ghost), we successfully construct a computationally more efficient model. The experimental results show that for a task like cheating detection, which has relatively low feature complexity, the over-enhanced feature extraction and fusion module suffers from over-performance, whereas through the structure lightweighting approach proposed in this paper, the inference speed of the model can be effectively improved while maintaining sufficient detection accuracy, making it more suitable for practical deployment scenarios.

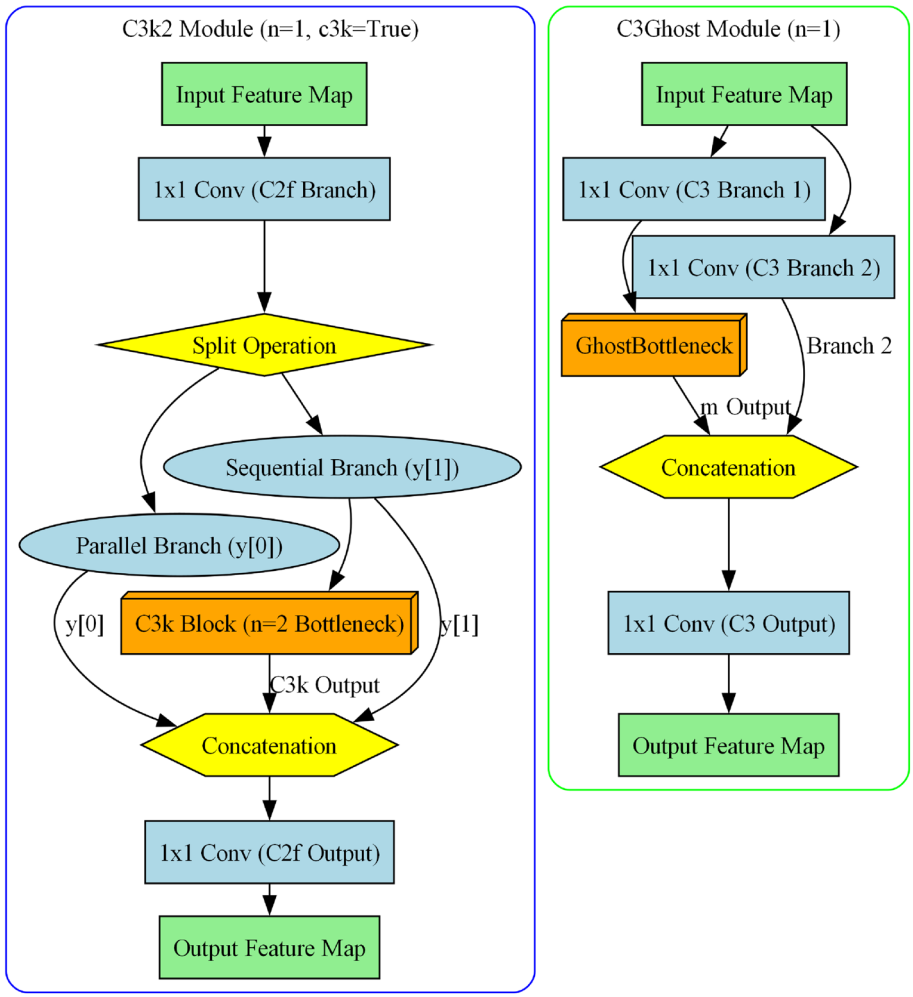

Fig. 7. C3k2(c3k=True) module with C3Ghost module.

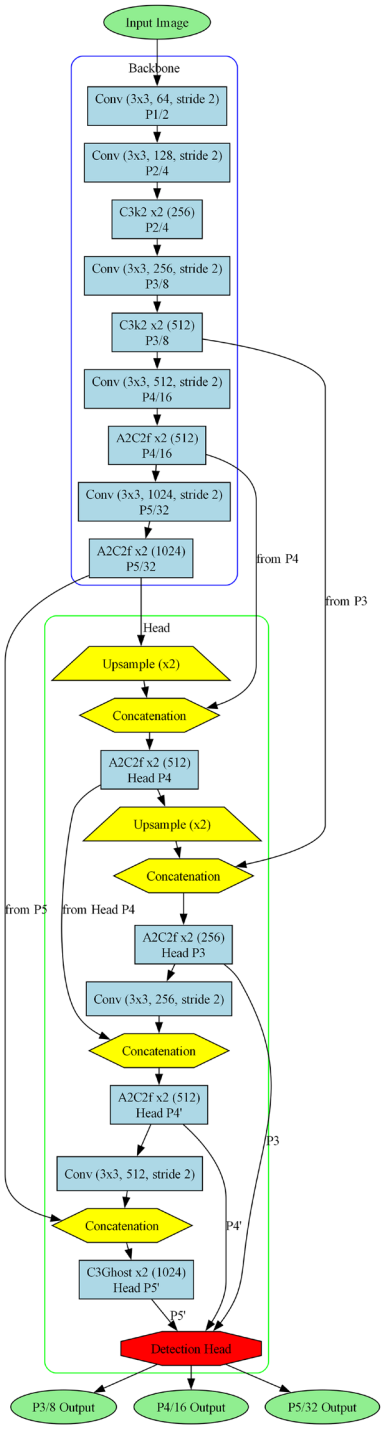

Fig. 8. Lightweighted YOLOv12 model structure diagram.

IPFS storage and hash generation for detection result data

During the online exam cheating detection process, when the YOLO model identifies a suspected cheating behaviour, it immediately captures the current screenshot as visual evidence. These screenshot image data are usually large and unsuitable for direct storage on the blockchain. To address the limitations of blockchain storage capacity and efficiency, and to ensure the integrity and traceability of the evidence images, this paper adopts the InterPlanetary File System (IPFS, InterPlanetary File System) as the distributed storage scheme for these screenshot images.

IPFS is a distributed file system based on content addressing. Its working principle is that any file uploaded to IPFS will have a unique hash value calculated based on its content, i.e., Content Identifier (CID). This CID is not only the address of the file on the IPFS network, but also a digital fingerprint of the file's content. Any

modification to the content of the file will result in a change of its CID, which fundamentally ensures the integrity and tamper-proofness of the stored data.

Specific screenshot image storage and hash generation process is as follows:

- When the YOLO model detection triggers a 'suspected cheating' inference, the system captures a screenshot of the current exam.
- This uploads the captured screenshot image file to the IPFS network.
- The image file is received and processed by the IPFS network to calculate its unique CID.
- This generated CID (hash value) represents the unique identity and content checksum of that particular screenshot image.

By storing suspected cheating screenshot images on IPFS, we can effectively strip a large amount of image data from the blockchain while leveraging the content addressing capabilities of IPFS to ensure the authenticity and integrity of this off-chain stored image evidence. Subsequently, this lightweight IPFS CID will be stored on the blockchain, through which the original screenshot images can be retrieved and verified from the IPFS network when needed.An overview of the IPFS storage and hash generation process is shown in Fig. 9.

Hyperledger fabric chain code design and testing results on the chain process

Hyperledger Fabric's federation chain architecture and its provision of privilege management, smart contracts (Chaincode), and efficient consensus mechanisms make it well suited for applications in scenarios such as online exams, which require trust and collaboration among multiple parties[50].

The core lies in the design and implementation of Chaincode. We design a special Chaincode for the deposit of cheating detection results, which defines the data structure of cheating event records and the logic of creating, querying and other operations on these records. The chain code will serve as a bridge between the cheat detection system and the blockchain ledger.

The data structure (or asset model) defined in the Chain Code shall contain the core metadata of the cheating event and links to the original evidence data in the IPFS.

The key information contained in the cheating event records defined in the chain code is shown in Table 2.

The process of uploading detection results is as follows:

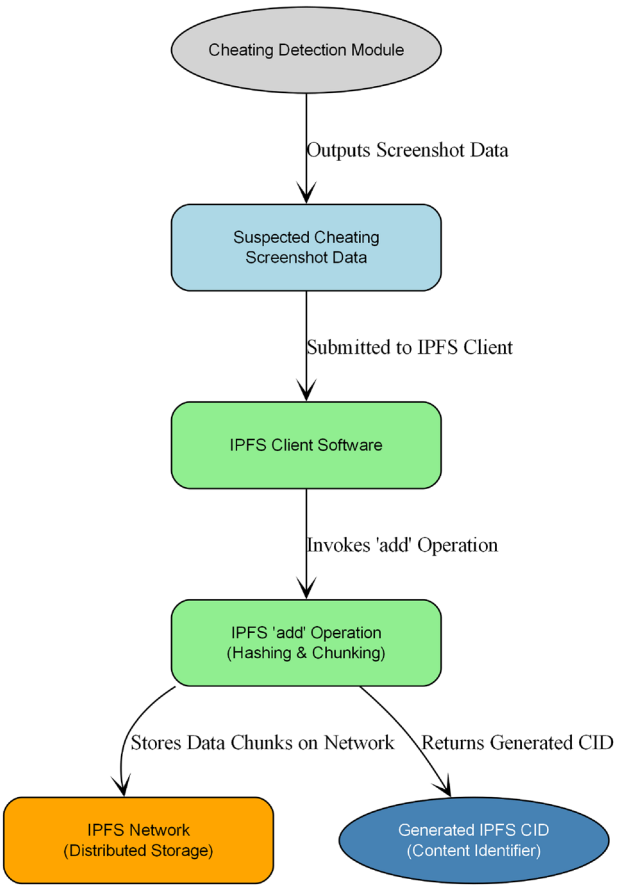

Fig. 9. IPFS storage and hash generation flowchart for suspected cheating screenshots.

examId	Unique identifier for the exam
studentId	Unique identifier of the candidate for whom cheating has been detected.
timestamp	Timestamp of the suspected cheating incident.
ipfsCID	Content identifier/hash on IPFS corresponding to the suspected cheating screenshot image for off-chain retrieval of original evidence.
yoloResults	The result of the YOLO model's inference on the screenshot, including details such as the bounding box coordinates and confidence level of the detected target (i.e., the 'suspected cheating' object).
cheatingType	Type of cheating, fixed to 'suspected cheating'.

Table 2. Form for logging cheating events defined in the chain code.

- Subsequent to the capture of a screenshot by the system designed for the detection of academic dishonesty and its subsequent upload to the IPFS system to obtain the ipfsCID (refer to Section "Hyperledger fabric chain code design and testing results on the chain process"), the system will consolidate all the relevant information pertaining to the detection event, including the examID, the studentID, the timestamp, the ipfsCID obtained, and the detailed inference results of the screenshot by the YOLO model yoloResults.
- The system's function as a client application entails the generation of a transaction proposal, which subsequently invokes the chain code. The proposal calls a function predefined in the chaincode for logging cheating events and passes the above integrated information as parameters.
- The transaction proposal is then transmitted to an endorsement node in the Hyperledger Fabric network for simulated execution. The chaincode operates on the endorsing node, constructs a cheating event record object based on the received parameters, and verifies the validity of the transaction.
- The endorsing node is responsible for the endorsement of the transaction proposal, and the return of a signed response to the client application.
- The client application is responsible for collecting sufficient endorsements and submitting the transaction to the Orderer.
- The system's ordering service functions by sorting the transactions from disparate clients and subsequently arranging them into blocks.
- The ordering service transmits the block to the Peer node, which performs the following functions: it validates the transactions contained within the block, it performs chaining (if required), and it writes valid transactions to the local distributed ledger.

As shown in the flowchart in Fig. 10, the key metadata of the suspected cheating event, the YOLO inference result, and the IPFS hash value pointing to the evidence of the original screenshot are permanently stored as a tamper-evident record on the distributed ledger of Hyperledger Fabric. Any authorised party can query the record on the chain to obtain the event details and IPFS CID, then retrieve the original screenshot from the IPFS network for review, and verify the authenticity of the screenshot by comparing the hash values, thus building a trustworthy chain of evidence of online exam cheating.

Experimental design

Introduction to the dataset

The dataset utilised in this study for the training and evaluation of online examination cheating detection models originates from the open-source project of Flying Paddle AI Studio. This dataset comprises surveillance photographs captured during online invigilation at Donghua University. The dataset was created in response to the demand for invigilation that has arisen due to the increasing number of online examination scenarios in the post-epidemic era. The aim of the creation of the dataset was to provide basic data support for research into cheating detection in related fields.

The original dataset contains 623 images. In the context of training deep learning target detection models, particularly in scenarios where multiple complex and potentially confounding cheating behaviours must be identified, this scale is considered to be inadequate. This limitation can result in the model overfitting during the training process, thereby compromising its capacity to generalise on actual, unseen data. In order to effectively expand the size of the dataset, improve the training effect of the model and enhance its robustness to changes in different scenarios, we performed data enhancement on the original dataset. A range of common data enhancement techniques was applied, including random rotation, scaling, horizontal flipping, brightness adjustment, etc., with the objective of expanding the dataset size tenfold to 6230 images. Figure 11 illustrates a partial sample image dataset.

The expanded dataset has been labelled in detail for key targets in the online examination scenario. Unlike the direct labelling of specific cheating behaviours, the labelling categories in this dataset are designed to be more basic and flexible, with two main categories: 'person' and 'electronic devices'. The design concept is to focus the target detection task on identifying the presence of a person or an electronic device in the picture, while the final judgement of cheating is adjusted in the inference stage based on the detection results and the specific test regulations. For example, for exams that allow the use of specific electronic devices, the presence of electronic devices can be controlled in the reasoning process not to be recognised as cheating; similarly, only one person is allowed to participate in the exam by default, and cheating is only recognised when multiple 'person' targets are detected. This flexible rule based on the basic target detection results allows the model to adapt to different exam formats and supervision requirements.

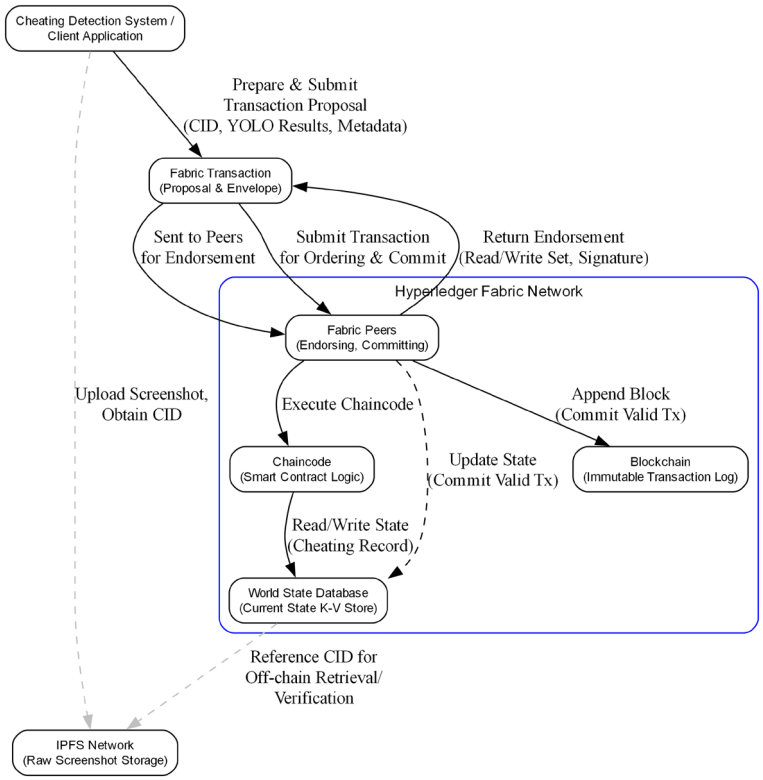

Fig. 10. Hyperledger Fabric chain code design and testing results on the chain flow diagram.

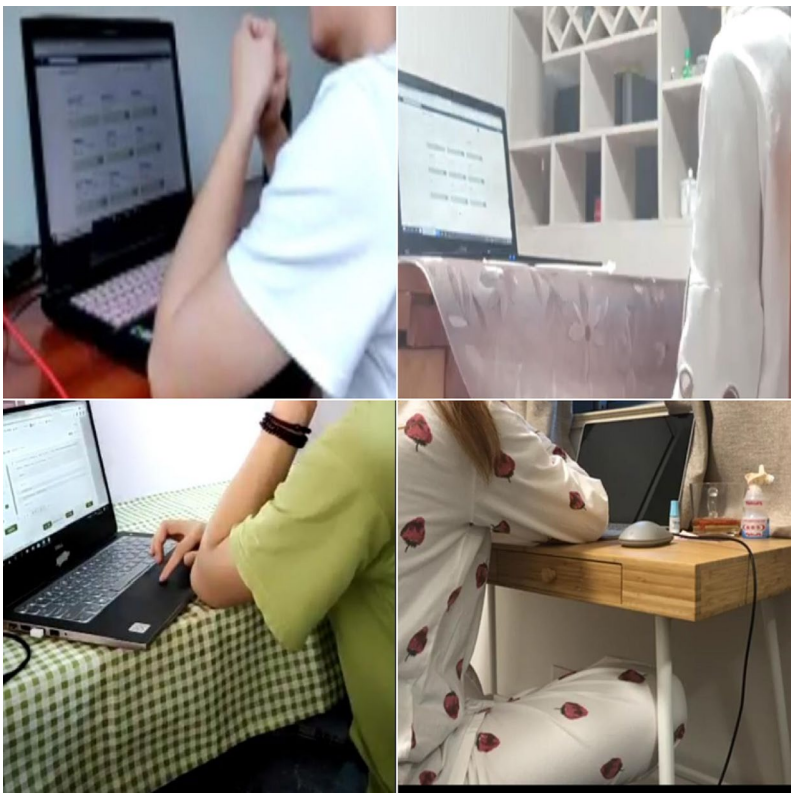

Fig. 11. Example of dataset image.

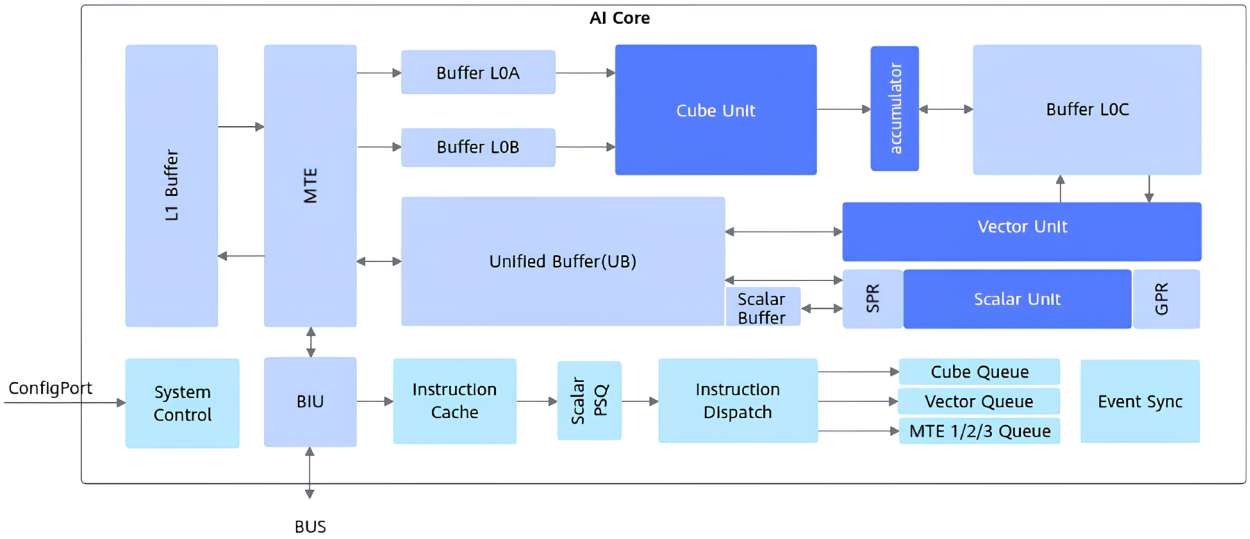

Fig. 12. AI core architecture diagram.

Environment	Specific
GPU	Huawei Ascend 910B 64GB
CPU	Huawei Kunpeng 920 CPU @ 2.60GHz 24-core processor
Memory	DDR4 220GB RAM
Operating System	Ubuntu 22.04
Python Version	Python 3.10
Deep learning framework	PyTorch 2.0.1 combined with CANN 8.0.0

Table 3. Software and hardware environment.

For effective model training, tuning and preliminary performance evaluation, we divide the expanded dataset into training and validation sets. According to the commonly used division strategy, we adopted a 9:1 ratio for division, i.e., the training set contains 5607 images and the validation set contains 623 images. The training set is used for parameter learning of the model, while the validation set is used for monitoring the training process, tuning the hyperparameters, and evaluating the performance of the model on unseen data during training to avoid overfitting.

Experimental environment and parameter settings

In order to ensure the fairness and reproducibility of the experimental results, this study standardised the training of all models based on a uniform hardware environment and consistent initial training parameters. The subsequent section provides a comprehensive account of the hardware and software environment configurations employed in the experimental setup. In addition, it delineates the specific parameter settings that were utilised during the training and evaluation of the model. This information is indispensable for comprehending the experimental conditions and conducting a thorough analysis of the results.

It is noteworthy that the computing platform for this experiment is the Huawei Rise Ascend 910B, and the training and validation of the models is accelerated and deployed through Huawei's CANN (Compute Architecture for Neural Networks) software stack, as opposed to the NVIDIA CUDA platform.

The core computing power of the Ascend 910B processor is mainly provided by its built-in AI Core. Unlike traditional CPUs and GPUs that support general-purpose computing, or ASICs (Application Specific Integrated Circuit) that are dedicated to a specific algorithm, the AI Core architecture is essentially designed to accommodate common applications and algorithms in a specific domain (i.e., AI computing), and is often referred to as a 'Domain Specific Architecture (DSA)'[51].

As shown in Fig. 12, the computational core of the AI Core consists of three main underlying computational resources: the Matrix Computing Unit (Cube Unit), the Vector Unit, and the Scalar Unit. Each of these three computing units has its own role, forming three independent execution pipelines, which cooperate with each other under the unified scheduling of the system software (i.e., CANN) to efficiently execute AI algorithmic tasks. The Cube Unit performs intensive matrix computations such as matrix multiplication; the Vector Unit performs parallel vector computations such as vector addition, subtraction, multiplication and division; and the Scalar Unit handles control flow and scalar computations.

The hardware and software environments are specified in Table 3.

The training configuration is as follows:

Number of training rounds: model training was set to 50 epochs to ensure that the model could fully learn the data features.
Batch size: each batch was automatically configured with 50% of the graphics memory (32G) of Huawei's Rise Ascend 910B, a setting that balances memory usage and training efficiency.
Input image size: All input images were uniformly resized to 640 pixels to fit the input requirements of the model.
Optimiser: The AdamW optimiser is used, which performs well in deep learning model training due to its adaptive learning rate adjustment strategy.
Learning rate: the initial learning rate is set to 0.01, which is adjusted by the learning rate scheduling strategy to promote model convergence.
Momentum: the momentum parameter is set to 0.937, which helps accelerate the convergence during the training process.
Weight decay: a weight decay of 0.0005 was applied to prevent model overfitting.
Warm-up rounds: the first 3 epochs are used as a warm-up phase to gradually increase the learning rate, which helps the model to be trained stably.
Data augmentation: Automatic data augmentation using RandAugment strategy combined with Mosaic data augmentation technique (Mosaic is turned off in the last 10 epochs).
NMS (Non-Maximum Suppression) Threshold: set to 0.7, used to filter out the optimal bounding box in the detection results and reduce overlapping detections.

The above hardware and software environments were carefully configured, along with meticulously set training parameters, in order to ensure that the improvements could be effectively validated in a controlled and consistent environment. These preparations aim to provide an efficient, stable and reproducible experimental platform for the training and evaluation of the YOLOv12 detection model, so that the impact of different algorithmic modules on the model performance can be reliably measured and compared.

We acknowledge the inherent stochasticity in the deep learning training process (e.g., from weight initialization and data augmentation). To ensure the reproducibility of our reported results, we fixed the random seeds across all experiments and configured PyTorch to use deterministic algorithms. Therefore, all performance metrics reported in this paper are stably reproducible under this controlled setup. While a single run does not provide statistical confidence intervals, we validate the robustness of our conclusions from multiple perspectives through the comprehensive comparative and ablation studies that follow.

Evaluation indicators

In evaluating the overall performance of the neural network model, we considered two key metrics, the size of the model and the detection accuracy.

The size of the model is measured by the number of parameters, which refers to the sum of parameters to be trained in the model. The smaller the number of parameters, the more suitable the model is for deployment on mobile devices, and also reflects the complexity and computational requirements of the model.

In evaluating the accuracy of target detection algorithms, we use metrics such as Precision (P), Recall (R), Mean Accuracy (mAP50) and Mean Average Precision (mAP50-95).

The precision rate P measures the proportion of all samples predicted to be in the positive category that are actually in the positive category, and is calculated as shown in Eq. 1.

$$P = \frac{TP}{TP + FP} \tag{1}$$

Recall R, on the other hand, is the ratio of samples correctly predicted to be positively classified to all actual positively classified samples and is calculated as shown in Eq. 2.

$$R = \frac{TP}{TP + FN} \tag{2}$$

Since there is a negative correlation between precision rate and recall rate, we usually plot the PR curve with recall rate as the horizontal axis and precision rate as the vertical axis, and the area under the PR curve is the AP value, as shown in Eq. 3.

$$AP = \int_0^1 p(r)\, dr \tag{3}$$

In this experiment, we will use metrics such as those presented in Table 4 to further refine the performance evaluation of the model.

Through the comprehensive analysis of these evaluation indexes, we are able to make a comprehensive assessment of the detection performance, complexity and applicability of the model, so as to provide a basis for the optimisation and selection of the model.

Indicator	Description
Precision	The Precision indicator measures the proportion of samples predicted by the model to be in the positive category that are actually in the positive category. It reflects the model's ability to avoid misclassifying non-positive samples as positive.
Recall	The Recall metric measures the proportion of samples correctly predicted by the model to be in the positive category as a proportion of all samples that are actually in the positive category. It measures the model's ability to identify all positively classified samples.
mAP50	The average precision calculated at an IoU (Intersection over Union) threshold of 0.5 is used to evaluate the detection performance of the model at moderate overlap. mAP50 is a commonly used evaluation metric in target detection that combines precision and recall.
mAP50-95	This metric is the average precision calculated over a range of IoU thresholds from 0.5 to 0.95. It provides a more comprehensive performance evaluation because different IoU thresholds require different precision for the detection frames, which enables a more nuanced evaluation of the model's detection capability.
Parameters	The number of parameters is the total number of all trainable parameters in the model. This metric reflects the complexity of the model, and usually the lower the number of parameters, the simpler the model and the lower the computational requirements, making it easier to deploy in resource-constrained environments.

Table 4. Introduction to performance indicators.

Comparative experiments

In order to comprehensively evaluate the performance of the detection module in the online exam cheating detection system based on YOLOv12 proposed in this paper and to compare it with other versions of YOLO target detection models, we designed and conducted a series of comparison experiments. The experiments are conducted on a specially constructed dataset of online exam cheating behaviours, and the key performance metrics of each model on the cheat detection task are documented in detail, including Precision (P), Recall (R), Average Precision mAP50, Average Precision mAP50-95 for IoU thresholds in the range of 0.5–0.95, as well as a measure of the efficiency of the model, the Parameters and GFLOPs.

The focus is on comparing the benchmark model YOLOv12n with the optimisation model YOLOv12NoAttn proposed in this paper, while YOLOv5n, YOLOv8n, YOLOv9t, and YOLOv10n, YOLOv11n are introduced as references. By systematically comparing these models, we aim to quantify the specific impact of the optimisation strategy in this paper on cheat detection performance and model efficiency, and explore the reasons for these performance differences. The specific experimental results are shown in Table 4.

As can be seen from the comparative experimental results in Table 4, the optimised model YOLOv12NoAttn proposed in this paper achieves competitive performance on the online exam cheating detection task and demonstrates significant advantages in terms of model efficiency.

Detection performance analysis: Compared with the benchmark model YOLOv12n, the optimised model YOLOv12NoAttn shows a small improvement in both mAP50 (0.98208 vs 0.98156) and Recall (0.95647 vs 0.94978). This suggests that the optimised model is able to detect cheating more effectively, reduce underreporting (i.e., increase recall), and has a higher average detection precision at an IoU threshold of 0.5. However, its precision rate Precision (0.93019 vs. 0.93823) and mAP50-95 (0.75436 vs. 0.75631) slightly decreases. In a cheating detection scenario, a high precision rate means fewer false positives (less interference with normal examinees), and a high recall rate means fewer missed positives (not missing cheating behaviours). The optimised model improves recall and mAP50 at the expense of precision and mAP50-95, which is a trade-off between detection coverage and detection accuracy. Considering the actual needs of online exams, a high recall rate is crucial for timely detection of cheating behaviours, while the slightly lower precision rate can be compensated by subsequent manual review and other means.

Model Efficiency Analysis: YOLOv12NoAttn performs well in terms of model complexity. The number of parameters is only 1,840,350, which is about 28% less than the benchmark YOLOv12n's 2,557,118. The number of floating-point operations GFLOPs is also reduced from 6.3 to 5.5, a reduction of about 13%. This significantly reduces the computation and storage overheads of the model, making the detection module more suitable for deployment on devices with limited computational resources or to support higher density concurrent detection on the server side, which is especially important for large-scale online exam scenarios.

Comparison with other models: Compared with other mainstream YOLO models, YOLOv12NoAttn has the lowest number of parameters and GFLOPs among all the models in the table, while maintaining similar or even better detection performance than YOLOv12n in some metrics. For example, compared to YOLOv8n, YOLOv12NoAttn is slightly lower on mAP50-95, but has a significant advantage on the number of parameters and GFLOPs. This highlights the effectiveness of the optimisation strategy in this paper in achieving model lightweighting, which provides a more efficient solution for online exam cheating detection.

In addition, by reporting both mAP50 and mAP50-95, we conduct a sensitivity analysis of the model's performance with respect to the evaluation criteria. mAP50 represents performance under a lenient IoU threshold, while mAP50-95 measures average performance across multiple, stricter IoU thresholds. Our model (YOLOv12NoAttn) maintains a highly competitive mAP50 score (0.98208), indicating that its core capability to 'detect the target' is reliable. The slight decrease in the more stringent mAP50-95 metric clearly illustrates the trade-off between efficiency and high-precision localization. This consistent performance across varying levels of evaluation stringency also indirectly corroborates the reliability of our results.

Experimental results show that the optimised model YOLOv12NoAttn proposed in this paper successfully strikes a good balance between detection performance (especially on mAP50 and Recall) and model efficiency in the online exam cheating detection task. By reducing the model complexity, it is made more suitable for real-world deployment requirements while maintaining a high cheating detection capability. This analysis not only validates the effectiveness of the optimisation strategy, but also provides a solid practical basis for further optimising the detection module for online exam cheating detection in the future.

Models	Precision	Recall	mAP50	mAP50-95	Parameters	GFLOPs
YOLOv5n	0.93855	0.95840	0.98088	0.75466	2,503,334	7.1
YOLOv8n	0.91304	0.97604	0.98513	0.79401	3,006,038	8.1
YOLOv9t	0.92883	0.94797	0.98123	0.73676	1,971,174	7.6
YOLOv10n	0.92774	0.94099	0.97988	0.77188	2,695,196	8.2
YOLOv11n	0.92852	0.96561	0.98199	0.76616	2,582,542	6.3
YOLOv12n	0.93823	0.94978	0.98156	0.75631	2,557,118	6.3
YOLOv12NoAttn	0.93019	0.95647	0.98208	0.75436	1,840,350	5.5

Table 5. Performance comparison of different YOLO models on online exam cheat detection tasks.

Configuration	Backbone (Layers 6 & 8)	Head (Layer 20)	P (%)	R (%)	mAP50 (%)	Parameters (M)	Description
Baseline YOLOv12n	Original A2C2f	Original C3k2	93.823	94.978	98.156	2,557,118	Original baseline model
Baseline + Backbone Light	Modified Module	Original C3k2	93.769	96.251	98.234	2,043,070	Lightweight improvements for backbone network applications only
Baseline + Head Light	Original A2C2f	C3Ghost	94.816	94.545	98.342	2,354,398	Lightweight improvements for head-only web applications
Full Lightweight Model	Modified Module	C3Ghost	93.019	95.647	98.208	1,840,350	Complete lightweight model (improvements applied to both backbone and header networks)

Table 6. Results of ablation experiments.

Ablation experiments

In order to systematically evaluate the impact of the lightweight improvement strategies proposed in this paper for the backbone and header networks on the performance and efficiency of the YOLOv12n model, a series of ablation experiments are conducted. By comparing the performance metrics under different model configurations, the ablation experiments aim to quantify the contribution of each improvement module and verify its effectiveness.

Based on the lightweight improvement proposed in Section "YOLOv12n model lightweight improved design", we designed the following four sets of experimental configurations for the study:

- Baseline Model: The original YOLOv12n-n architecture is used. This configuration uses the standard A2C2f module at layers 6 and 8 of the backbone network and the standard C3k2 (c3k=True) module at layer 20 of the header network. This configuration serves as a performance and parametric quantitative reference base for all subsequent experiments.
- Backbone Lightweight Only: Based on the baseline model, only the backbone lightweight improvements described in Section "Backbone lightweighting" are applied, i.e., the modification of the module structure at layers 6 and 8 (removal of Attention, replacement with C3k, reduction of the number of stacks). Layer 20 of the header network still uses the standard C3k2 module. This configuration is used to evaluate the independent effect of the backbone network lightweighting improvements.
- Head Lightweight Only: Based on the baseline model, only the Head Lightweight improvements described in Section "Head network lightweighting" are applied, i.e., the C3k2 (c3k=True) module is replaced by the C3Ghost module at layer 20 of the Head Network. Layers 6 and 8 of the backbone network still use the standard A2C2f module. This configuration is used to evaluate the independent effect of the header network lightweighting improvements.
- Full Lightweight Model: The lightweight improvements are applied to both the backbone and headend networks, i.e., the final model structure proposed in this paper. This configuration is used to evaluate the overall performance and efficiency of all the improved modules working together.

All experimental configurations were trained and evaluated under the same dataset, training hyperparameters (e.g., learning rate strategy, optimiser, batch size, etc.), and hardware environments to ensure comparable results. We recorded the Precision (P), Recall (R), mAP@0.5:0.95 (mAP), and Model Parameters (Parameters) for each configuration. The experimental results are summarised in Table 6.

By comparing the results of different configurations, we can analyse the respective impact of the backbone and head network lightweight improvements on the model performance and the number of parameters, and verify the effectiveness of the joint improvements.

Table 5 summarizes the results of the ablation experiment, clearly showing the impact of different lightweight strategies on model performance and parameter count. As a benchmark for performance and efficiency, the original YOLOv12n model (baseline model) achieved 93.823% precision (P), 94.978% recall (R), and 98.156% mAP@0.5, with approximately 2.56M parameters.

After only the backbone network was lightweighted (only the backbone network lightweight model), the number of model parameters was significantly reduced to about 2.04M, which is about 20% less than the baseline model. This shows that the optimization of the backbone network is one of the key contributors to the significant reduction in model parameters. In terms of performance, the recall rate (R) increased to 96.251%, while the

precision rate (P) decreased slightly to 93.769%, and mAP@0.5 remained at 98.234%, which is comparable to or slightly improved from the baseline model. This shows that the lightweight backbone network has little impact on the detection performance while greatly compressing the model, and even has a gain in recall rate, reflecting the efficiency of the improved module.

The experiment of only making lightweight improvements to the head network (only the lightweight model of the head network) shows that the number of parameters of the model is reduced to about 2.35M, which is a smaller reduction (about 8.7%) than the improvement of the backbone network. In terms of performance, the precision (P) is increased to 94.816%, the recall (R) is slightly reduced to 94.545%, and the mAP@0.5 is increased to 98.342%. This shows that the lightweighting of the head network has a positive impact on the model's precision and overall mAP while reducing the number of parameters to a certain extent, especially in improving the precision.

The complete lightweight model that simultaneously applies the backbone network and head network lightweight improvements achieves the maximum parameter compression, which is only about 1.84M. Compared with the baseline model, the number of parameters is reduced by about 28%. In terms of performance, the model achieved a precision (P) of 93.019%, a recall (R) of 95.647%, and a mAP@0.5 of 98.208%. Compared with the baseline model, the complete lightweight model has a substantially reduced number of parameters, and the mAP@0.5 is basically the same (or even slightly improved), the recall rate is improved, and the precision rate is slightly reduced. This shows that the backbone network and head network lightweight strategies proposed in this paper can work together, while significantly reducing the complexity of the model, effectively maintaining or even optimizing key detection performance indicators (especially mAP and R), and achieving a good balance between efficiency and performance.

More importantly, the ablation study (Table 5) provides strong support for the reliability of our conclusions. The experiment clearly demonstrates that the lightweight modifications to the backbone and head networks each contributed quantifiable and positive effects (e.g., the backbone modification significantly reduced parameters by 20% while improving recall; the head modification improved precision). The final balance of performance and efficiency achieved by the full lightweight model is a direct consequence of these systematic improvements working in concert, rather than a random artifact of a single training run. This systematic cause-and-effect relationship itself serves as evidence for the robustness of our findings, mitigating concerns about the stochasticity of a single experiment.

The results of the ablation experiments strongly demonstrate the effectiveness of the lightweight improvement strategies for the backbone and head networks proposed in this paper. The improvement of the backbone network is the key to achieve a significant compression of the number of model parameters, while the improvement of the head network has a positive effect on enhancing the accuracy and mAP of the model. The joint application of these improvements can significantly reduce the number of parameters and computational complexity of the model while guaranteeing high detection performance (especially mAP and R), making it more suitable for online examination environments with limited resources.

Limitations of the Proposed Solution

While the proposed integrated system demonstrates significant advancements in online exam cheating detection and evidence validation, it is crucial to acknowledge its inherent limitations, which also highlight avenues for future research.

The performance evaluation in this study is based on a single training and testing run, which limits the statistical robustness of our findings. To enhance the reliability of the assessment, future work should involve multiple runs with different random seeds to report averaged results and confidence intervals, thereby providing a more comprehensive evaluation of the model's performance stability and consistency.

The detection methodology is predominantly vision-based and relies on identifying specific objects (e.g., 'person', 'electronic devices') in conjunction with predetermined rules. This approach has limitations in identifying more subtle, complex, or non-visual cheating methods, such as the use of AI-powered assistance for answering questions or communication via micro-earpieces. The system currently does not incorporate complementary technologies like audio analysis or data from IoT sensors, which could create a more holistic and robust multi-modal detection framework.

Despite data augmentation, the dataset employed for model training is comparatively limited in size (6,230 images) and scope. It may not fully represent the vast diversity of real-world cheating scenarios, environmental conditions, and subtle behaviors. This constraint could potentially limit the model's generalization capability when deployed in more complex or previously unseen examination settings, affecting its effectiveness across different institutional contexts and examination formats.

The system's reliance on continuous video monitoring and the storage of student screenshots raises multifaceted privacy implications that extend beyond technical considerations. Drawing insights from privacy-preserving technologies in distributed systems[47], we recognize that centralized video processing inherently creates privacy vulnerabilities. While the current approach secures evidence integrity through blockchain technology, it does not address the fundamental privacy concern of sensitive biometric and behavioral data collection, which remains a critical limitation in the system's design.

The absence of advanced privacy-preserving techniques that have proven effective in related domains further compounds these concerns. Federated learning approaches, as demonstrated in privacy-preserving data contribution systems[48], could enable model training without centralizing sensitive student data, yet such mechanisms are not incorporated in the current system. Similarly, the lack of differential privacy mechanisms during inference means that individual student identities remain inadequately protected from potential reconstruction attacks, creating additional vulnerabilities in the privacy framework.

Future iterations must address these privacy challenges through a comprehensive approach that integrates on-device processing capabilities to minimize raw data transmission while implementing federated learning protocols for collaborative model improvement across institutions. The incorporation of differential privacy guarantees becomes essential to protect individual privacy while maintaining detection effectiveness. Beyond technical improvements, the system requires comprehensive compliance frameworks that address GDPR, FERPA, and other relevant data protection regulations to ensure ethical deployment in educational environments.

Conclusion and future work

This study presents a novel solution for enhancing online examination integrity by integrating a lightweight YOLOv12 model with blockchain technology, addressing the dual challenges of real-time detection and trusted evidence preservation. Our key achievement is the development of the YOLOv12NoAttn model, which, through targeted structural optimizations, strikes a thoughtful balance between high performance and computational efficiency. The model reduces parameters by approximately 28% and GFLOPs by 13% compared to its baseline, while delivering a strong mAP50 of 98.21% and an improved recall of 95.65%. Concurrently, our implementation of a Hyperledger Fabric and IPFS framework effectively addresses the long-standing challenge of creating tamper-proof evidentiary records, ensuring the originality and integrity of cheating evidence. Our work delivers a practical, efficient, and credible end-to-end system that significantly advances the state of online examination security.

Building on the foundation of this study, our future work will focus on several key extension directions to enhance the system's intelligence, scalability, and trustworthiness.

Advanced Cheating Behavior Recognition: A primary objective is to transition from the current object-based detection to a more sophisticated, end-to-end cheating behavior recognition model. While our system effectively identifies prohibited objects, it relies on rule-based logic to infer cheating. Future research will therefore explore spatio-temporal deep learning models (e.g., 3D-CNNs or Video Transformers) to directly learn and identify complex, subtle actions such as whispering or illicit human-computer interactions. This necessitates a significant effort in constructing a more comprehensive dataset annotated with fine-grained temporal action labels.

Edge Computing Architecture and Real-time Scalability: To address the practical challenges of real-time processing and scalability in large-scale deployments, we will investigate an architectural shift towards edge computing. Deploying the lightweight detection model directly on the candidate's device (the edge) can significantly reduce network latency, lower central server load, and enhance data privacy by minimizing raw video transmission. This direction logically extends our current work on model lightweighting, aiming for a truly distributed and efficient proctoring architecture.

Comprehensive Privacy-Preserving Architecture and Compliance Enhancement: A critical priority for future development is the implementation of comprehensive privacy-preserving mechanisms that address the fundamental ethical concerns raised by continuous video monitoring. Inspired by advances in on-device computing and federated systems[47], we will investigate the deployment of lightweight detection models directly on student devices to minimize sensitive data transmission while maintaining detection effectiveness. This approach aligns seamlessly with our current lightweighting efforts and represents a natural evolution toward privacy-by-design architecture.

Furthermore, we will explore federated learning protocols adapted from privacy-preserving data contribution frameworks[48] to enable collaborative model improvement across educational institutions without sharing raw student data. This federated approach will allow institutions to benefit from collective intelligence while ensuring that sensitive biometric and behavioral data remains locally protected. The integration of differential privacy techniques during model inference will provide mathematical guarantees for individual privacy protection, adding calibrated noise to detection results to prevent potential reconstruction attacks while maintaining system utility.

The blockchain evidence system will also be enhanced with privacy-preserving smart contracts that can process encrypted evidence metadata, ensuring that even the stored evidence maintains privacy protection throughout its lifecycle. Advanced cryptographic techniques such as zero-knowledge proofs may be integrated to enable evidence verification without revealing sensitive content details. Additionally, we will develop comprehensive compliance frameworks to ensure adherence to international data protection regulations including GDPR, FERPA, and regional privacy laws, incorporating automated compliance monitoring and reporting mechanisms.

Enhanced Blockchain Infrastructure and Decentralized Identity: The blockchain component itself presents significant opportunities for enhancement. Future iterations will involve designing more advanced smart contracts to automate the entire evidence lifecycle, including appeal mechanisms, dispute resolution tracking, and automated evidence expiration. We will also investigate integration with Decentralized Identity (DID) systems to create a more robust, privacy-preserving authentication process for all participants, eliminating the need for centralized identity management while maintaining security and accountability.

Multi-modal Detection and Experimental Rigor: To address current detection limitations, future work will explore multi-modal approaches integrating audio analysis, keystroke dynamics, and IoT sensor data to create a more comprehensive understanding of the examination environment. Simultaneously, we will implement more rigorous experimental protocols involving multiple independent runs with different random seeds to provide statistically robust performance assessments with confidence intervals, enhancing the reliability and reproducibility of our findings.

These integrated research directions will collectively advance toward a next-generation online examination integrity system that balances security, privacy, efficiency, and ethical considerations, providing a foundation for trustworthy digital assessment in the evolving landscape of online education.

Data Availability

The dataset used in this study is a publicly available open source dataset with the access link: https://aistudio.baidu.com/datasetdetail/128035. The dataset is released under the GPL-2.0 open source agreement with no ethical implications.

Received: 7 May 2025; Accepted: 1 September 2025
Published online: 26 September 2025

References

1. Selvaraj, A. et al. Effect of pandemic based online education on teaching and learning system. *Int. J. Educ. Dev.* **85**, 102444 (2021).
2. Garg, M. & Goel, A. A systematic literature review on online assessment security: Current challenges and integrity strategies. *Comput. Secur.* **113**, 102544 (2022).
3. Newton, P. M. & Essex, K. How common is cheating in online exams and did it increase during the covid-19 pandemic? A systematic review. *J. Acad. Eth.* **22**, 323–343 (2024).
4. Burgason, K. A., Sefiha, O. & Briggs, L. Cheating is in the eye of the beholder: An evolving understanding of academic misconduct. *Innov. High. Educ.* **44**, 203–218 (2019).
5. Schulz, A. Trust and Cheating: From Assessment Literacy to a Literacy of Practice for Digital Remote Assessments. In Hummel, S. & Donner, M.-T. (eds.) *Student Assessment in Digital and Hybrid Learning Environments*, 129–152, (Springer Fachmedien, Wiesbaden, 2023). https://doi.org/10.1007/978-3-658-42253-0_7.
6. Azzedin, F. & Ridha, A. The Effect of Behavior Change on Honesty Checking in Peer-to-Peer Systems. In *2008 Sixth Annual Conference on Privacy, Security and Trust*, 145–150,https://doi.org/10.1109/PST.2008.34 (2008).
7. Azzedin, F. Taxonomy of reputation assessment in peer-to-peer systems and analysis of their data retrieval. *Knowl. Eng. Rev.* **29**, 463–483. https://doi.org/10.1017/S0269888914000174 (2014).
8. Azzedin, F. A. *Trust modeling and its applications for peer-to-peer based systems*. phd, University of Manitoba, CAN (2004). AAINQ97011 ISBN-10: 0612970116.
9. Essahraui, S., El Mrabet, M. A., Bouami, M. F., El Makkaoui, K. & Faize, A. An intelligent anti-cheating model in education exams. In *2022 5th International conference on advanced communication technologies and networking (CommNet)*, 1–6 (IEEE, 2022).
10. Ahmed, F. R. A. et al. Analysis and challenges of robust e-exams performance under covid-19. *Res. Phys.* **23**, 103987 (2021).
11. Malhotra, M. & Chhabra, I. Student invigilation detection using deep learning and machine after covid-19: A review on taxonomy and future challenges. In *Future of Organizations and Work after the 4th Industrial Revolution: The Role of Artificial Intelligence, Big Data, Automation, and Robotics* 311–326 (2022).
12. Labayen, M., Vea, R., Flórez, J., Aginako, N. & Sierra, B. Online student authentication and proctoring system based on multimodal biometrics technology. *Ieee Access* **9**, 72398–72411 (2021).
13. Ramu, T. & Arivoli, T. A framework of secure biometric based online exam authentication: an alternative to traditional exam. *Int. J. Sci. Eng. Res.* **4**, 52–60 (2013).
14. Mistry, B., Parekh, H., Desai, K. & Shah, N. Online examination system with measures for prevention of cheating along with rapid assessment and automatic grading. In *2022 5th International Conference on Advances in Science and Technology (ICAST)*, 28–34 (IEEE, 2022).
15. Bernardes, R. C., Lima, M. A. P., Guedes, R. N. C., Silva, C. B. & Martins, G. F. Ethoflow: Computer vision and artificial intelligence-based software for automatic behavior analysis. *Sensors* **21**, 3237 (2021).
16. Drenkow, N., Sani, N., Shpitser, I. & Unberath, M. A systematic review of robustness in deep learning for computer vision: Mind the gap? arXiv preprint arXiv:2112.00639 (2021).
17. Xue, J., Wu, W. & Cheng, Q. Intelligent invigilator system based on target detection. *Multimed. Tools Appl.* **82**, 44673–44695 (2023).
18. Viola, P. & Jones, M. 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*, vol. 1, I–I (Ieee, 2001).
19. Dalal, N. & Triggs, B. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1, 886–893 (Ieee, 2005).
20. Ren, S., He, K., Girshick, R. & Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**, 1137–1149 (2016).
21. Liu, W. *et al.* Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, 21–37 (Springer, 2016).
22. Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788 (2016).
23. Redmon, J. & Farhadi, A. Yolo9000: Better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7263–7271 (2017).
24. Redmon, J. & Farhadi, A. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018).
25. Bochkovskiy, A., Wang, C.-Y. & Liao, H.-Y. M. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020).
26. Wang, C.-Y., Bochkovskiy, A. & Liao, H.-Y. M. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 7464–7475 (2023).
27. Tian, Y., Ye, Q. & Doermann, D. Yolov12: Attention-centric real-time object detectors. arXiv preprint arXiv:2502.12524 (2025).
28. Kong, Y. & Fu, Y. Human action recognition and prediction: A survey. *Int. J. Comput. Vision* **130**, 1366–1401 (2022).
29. Zhou, L. *et al.* Human pose-based estimation, tracking and action recognition with deep learning: a survey. arXiv preprint arXiv:2310.13039 (2023).
30. Gkioxari, G., Girshick, R., Dollár, P. & He, K. Detecting and recognizing human-object interactions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8359–8367 (2018).
31. Li, W., Chang, M.-C. & Lyu, S. Who did what at where and when: simultaneous multi-person tracking and activity recognition. arXiv preprint arXiv:1807.01253 (2018).
32. Nixon, M. & Aguado, A. *Feature extraction and image processing for computer vision* (Academic press, 2019).
33. Gowsikhaa, D., Abirami, S. & Baskaran, R. Automated human behavior analysis from surveillance videos: A survey. *Artif. Intell. Rev.* **42**, 747–765 (2014).
34. Ahmad, I., AlQurashi, F., Abozinadah, E. & Mehmood, R. A novel deep learning-based online proctoring system using face recognition, eye blinking, and object detection techniques. *Int. J. Adv. Comput. Sci. Appl.* **12**, 847 (2021).
35. Nakamoto, S. Bitcoin whitepaper. https://bitcoin.org/bitcoin.pdf-(: 17.07. 2019) **9**, 15 (2008).
36. Monrat, A. A., Schelén, O. & Andersson, K. A survey of blockchain from the perspectives of applications, challenges, and opportunities. *IEEE Access* **7**, 117134–117151 (2019).
37. Preneel, B. Cryptographic hash functions. *Eur. Trans. Telecommun.* **5**, 431–448 (1994).
38. Lashkari, B. & Musilek, P. A comprehensive review of blockchain consensus mechanisms. *IEEE access* **9**, 43620–43652 (2021).
39. Benisi, N. Z., Aminian, M. & Javadi, B. Blockchain-based decentralized storage networks: A survey. *J. Netw. Comput. Appl.* **162**, 102656 (2020).

40. Haga, S. & Omote, K. Blockchain-based autonomous notarization system using national eid card. *IEEE Access* **10**, 87477–87489 (2022).
41. Khanna, A. et al. Blockchain: Future of e-governance in smart cities. *Sustainability* **13**, 11840 (2021).
42. Benet, J. Ipfs-content addressed, versioned, p2p file system. arXiv preprint arXiv:1407.3561 (2014).
43. Shi, R., Cheng, R., Han, B., Cheng, Y. & Chen, S. A closer look into ipfs: Accessibility, content, and performance. *Proc. ACM Measure. Anal. Comput. Syst.* **8**, 1–31 (2024).
44. Trautwein, D. *et al.* Design and evaluation of ipfs: a storage layer for the decentralized web. In *Proceedings of the ACM SIGCOMM 2022 Conference*, 739–752 (2022).
45. Sangeeta, N. & Nam, S. Y. Blockchain and interplanetary file system (ipfs)-based data storage system for vehicular networks with keyword search capability. *Electronics* **12**, 1545 (2023).
46. Singh, H. J. & Bawa, S. Scalable metadata management techniques for ultra-large distributed storage systems-a systematic review. *ACM Comput. Surv. (CSUR)* **51**, 1–37 (2018).
47. Yin, H. *et al.* On-Device Recommender Systems: A Comprehensive Survey, https://doi.org/10.48550/arXiv.2401.11441 (2024). arXiv:2401.11441 [cs].
48. Yang, C., Yuan, W., Qu, L. & Nguyen, T. T. PDC-FRS: Privacy-Preserving Data Contribution forÂ Federated Recommender System. In Sheng, Q. Z. *et al.* (eds.) *Advanced Data Mining and Applications*, 65–79, https://doi.org/10.1007/978-981-96-0850-8_5 (Springer Nature, Singapore, 2025).
49. Han, K. *et al.* Ghostnet: More features from cheap operations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1580–1589 (2020).
50. Androulaki, E. *et al.* Hyperledger fabric: a distributed operating system for permissioned blockchains. In *Proceedings of the thirteenth EuroSys conference*, 1–15 (2018).
51. Liao, H. *et al.* Ascend: a scalable and unified architecture for ubiquitous deep neural network computing: Industry track paper. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, 789–801 (IEEE, 2021).

Acknowledgements

The work was supported by the Social Development Science and Technology Project of Dongguan City, China (20211800900282), Characteristic Innovation Projects of Regular Institution of High Learning in Guangdong Province, China (2024KTSCX215), Guangdong Provincial Education Science Planning Project (Higher Education Special Project), China (2024GXJK679), and the 2024 Guangdong Provincial Undergraduate Teaching Quality and Teaching Reform Project - Special Talent Development Program (Outstanding HarmonyOS Application-Oriented Talent Cultivation Program), China, under Grant Yue Jiao Gao Han [2024] No. 30.

Author contributions

Haoliang Wang: Conceptualization, Methodology, Writing, Proof Reading, Editing, Fund supporting; Zarina Shukur: Conceptualization, Methodology; Khairul Akram: Methodology, Proof Reading; Renhao Xiao: Experiments, Data Analysis, Data Curation; Lili Wang: Conceptualization, Proof Reading, Editing, Fund supporting.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to H.W., Z.S. or L.W.