*big data and cognitive computing*

MDPI

*Article*

# A Trusted Supervision Paradigm for Autonomous Driving Based on Multimodal Data Authentication

**Tianyi Shi [1,†], Ruixiao Wu [1,†], Chuantian Zhou [1], Siyang Zheng [1], Zhu Meng [1], Zhe Cui [1], Jin Huang [2,3], Changrui Ren [2,3] and Zhicheng Zhao [1,4,*]**

[1] School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Haidian District, Beijing 100876, China; sty0622@bupt.edu.cn (T.S.)
[2] Beijing Academy of Blockchain and Edge Computing, Haidian District, Beijing 100085, China
[3] Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing, Haidian District, Beijing 100085, China
[4] Beijing Key Laboratory of Network System and Network Culture, Haidian District, Beijing 100876, China
* Correspondence: zhaozc@bupt.edu.cn
† These authors contributed equally to this work.

**Abstract:** At the current stage of autonomous driving, monitoring the behavior of safety stewards (drivers) is crucial to establishing liability in the event of an accident. However, there is currently no method for the quantitative assessment of safety steward behavior that is trusted by multiple stakeholders. In recent years, deep-learning-based methods can automatically detect abnormal behaviors with surveillance video, and blockchain as a decentralized and tamper-resistant distributed ledger technology is very suitable as a tool for providing evidence when determining liability. In this paper, a trusted supervision paradigm for autonomous driving (TSPAD) based on multimodal data authentication is proposed. Specifically, this paradigm consists of a deep learning model for driving abnormal behavior detection based on key frames adaptive selection and a blockchain system for multimodal data on-chaining and certificate storage. First, the deep-learning-based detection model enables the quantification of abnormal driving behavior and the selection of key frames. Second, the key frame selection and image compression coding balance the trade-off between the amount of information and efficiency in multiparty data sharing. Third, the blockchain-based data encryption sharing strategy ensures supervision and mutual trust among the regulatory authority, the logistic platform, and the enterprise in the driving process.

**Keywords:** autonomous driving; blockchain; abnormal behavior detection; deep learning



**Citation:** Shi, T.; Wu, R.; Zhou, C.; Zheng, S.; Meng, Z.; Cui, Z.; Huang, J.; Ren, C.; Zhao, Z. A Trusted Supervision Paradigm for Autonomous Driving Based on Multimodal Data Authentication. *Big Data Cogn. Comput.* **2024**, *8*, 100. https://doi.org/10.3390/bdcc8090100

Academic Editor: Moulay A. Akhloufi

Received: 30 June 2024
Revised: 8 August 2024
Accepted: 24 August 2024
Published: 2 September 2024



## 1. Introduction

Traffic accidents are one of the major threats to human life. About 1.19 million people die in road traffic accidents every year according to the World Health Organization. In particular, with the development of artificial intelligence [1–6], remarkable progress has been achieved in areas such as image segmentation, classification, and detection. Similarly, self-driving cars are becoming even more popular [7,8], and safety issues [9,10] caused by autonomous driving have increasingly attracted the attention of society. Currently, most self-driving cars are equipped with a safety steward, whose responsibility is to take over the controlling system when unpredictable problems happen to self-driving systems. A notification regarding the pilot work on the admission and on-road operation of intelligent connected vehicles was issued by the Chinese government on 17 November 2023, and it explicitly emphasizes that autonomous driving systems need to be equipped with safety steward. The autonomous driving system should possess the capability of monitoring the safety steward's behaviors and also be able to identify whether the safety steward is capable of taking over dynamic driving tasks. Warning signals should be issued when the safety steward's capabilities are insufficient to meet the requirements. Therefore, there are

two important issues that need to be resolved: (1) the safety steward's behavior detection while the vehicle is driving, and (2) the liability determination when safety problems occur.

In most autonomous driving scenarios, the supervision of the safety steward relies on surveillance videos. In recent years, the methods of video analysis based on deep learning keep appearing. Some of them are used to detect abnormal behavior, which can promptly alert the safety steward when he (she) is detected to have abnormal behavior. At present, abnormal behavior detection methods based on deep learning are mainly divided into two categories: unsupervised [11–13] and supervised [14–17]. Unsupervised methods are usually used for reconstruction tasks, while supervisory information is used for model training and selecting samples with poor reconstruction quality as abnormal behavior samples in inference phase, and supervised methods treat abnormal behavior as one of the target categories for training.

In terms of the liability determination, electronic evidence preservation can verify the behavioral trajectory and environmental conditions of autonomous vehicles in the event of accidents or violations, and then accurately reconstruct the accident scenarios and analyze liability [18]. Traditional electronic evidence preservation methods often rely on centralized storage and management institutions, which leads to problems such as data tampering and high storage costs [19]. To address the problem of data tampering, in this paper, blockchain, as a decentralized and tamper-resistant distributed ledger technology, is used as a tool for data storage. Blockchain can record the driving data, sensor data, map data, and even videos of the safety steward when driving autonomous vehicles. These data can be verified automatically by smart contracts. For the high storage problem, key frames filtering of video data and image compression coding strategies are designed to solve this problem. In order to monitor the abnormal behavior of the safety steward, this paper trains a deep learning model for abnormal driving behavior detection.

The contributions of this paper can be summarized as follows:

- A trusted supervision paradigm for autonomous driving (TSPAD) based on multi-modal data authentication is proposed for the timely and intelligent quantification and recording of the safety steward's abnormal behavior in autonomous driving. TSPAD is an innovative framework designed to address the demands of real-world business scenarios, and its feasibility is thoroughly validated in this paper.
- A encrypted evidence storage method based on dual-chain blockchain architecture is designed to enhance the sharing efficiency and trustworthiness among multiple stakeholders. This method integrates a private key chain for secure identity authentication and a notary chain for reliable data verification, ensuring both data integrity and privacy protection in distributed environments.
- The abnormal behavior of safety drivers in autonomous driving is quantified and key frames are extracted through deep learning models. This paper designs a variety of optional key frames selection strategies to cope with different application scenarios. For scenarios with high speed requirements, the intermediate frames or random frames selection strategy is adopted. For scenarios with high accuracy requirements, the adaptive clustering key frames selection strategy can be utilized.
- An image compression algorithm is employed to alleviate the burden on distributed storage. In the context of autonomous driving, leveraging the computational power of hardware, an image compression algorithm accelerated by graphics processing units (GPUs) is utilized to reduce the burden on distributed storage, while preserving maximum image details.

## 2. Related Work

### 2.1. Blockchain for Traffic Incident and Accident Management

Block4Forensic (B4F) [20] is an innovative framework introduced for securing digital evidence in vehicle operation scenarios using blockchain technology. The B4F framework integrates blockchain with Vehicle Public Key Infrastructure (VPKI), fragmented ledgers, and forensic watchdog protocols to provide a comprehensive solution for forensic analysis.

Yao, Q. et al. [21] presented a model designed to address challenges such as forensic data preservation, privacy leaks, and legal accountability in the context of autonomous vehicle accidents.

As society's demand for automation increased, machine learning methods were progressively applied to evidence preservation systems for data analysis and extraction. The Secure Incident & Evidence Management Framework (SIEMF) [22] introduced a pioneering integration of deep learning and blockchain technologies for accident prediction and evidence management within vehicular networks. By employing convolutional neural networks (CNNs) [23] to analyze continuous data streams instead of merely relying on snapshot data, the framework enhanced the accuracy of accident risk prediction and the issuance of timely alerts. Tamper-Proof [24] integrated Internet of Things, artificial intelligence (AI), and smart contract technologies, enabling real-time communication and verification of accident-related videos among autonomous vehicles, and using blockchain to authenticate the origin and veracity of digital media.

Historically, blockchain was considered unsuitable for storing large media and evidence files [25]. Philip et al. [26] proposed a blockchain-based framework for post-accident investigations that integrated evidence from nearby vehicles, closed-circuit television, and other road users. Using the InterPlanetary File System (IPFS) for efficient evidence transmission, this framework aided in investigating the causes and determining responsibilities in vehicle accidents. Building on the SIEMF framework, the Improved Secure Incident and Evidence Management Framework (ISIEMF) [27] introduced an approach that employed long short-term memory (LSTM) [28] networks and Bayesian models to address incident reporting of internal conflicts within vehicle systems. Additionally, ISIEMF leveraged the IPFS as a distributed storage solution, enhancing storage efficiency and data integrity.

However, retaining raw data in local file systems increases the risks of tampering and loss, thus limiting the credibility of blockchain-based evidence preservation systems. This vulnerability underscores the need for improved security measures to ensure the integrity and reliability of such systems in handling critical data.

### 2.2. Abnormal Behavior Detection

At present, abnormal behavior detection methods based on deep learning are mainly divided into two categories: unsupervised [11–13] and supervised [14–17].

Unsupervised methods usually use other tasks such as reconstruction tasks as the target tasks to supervise the network to learn normal behavior features. In the inference phase, if there is a behavior that causes the network to respond abnormally, it is considered an abnormal behavior. Deepak et al. [12] proposed an unsupervised abnormal behavior detection framework which was trainable end-to-end. Its residual spatiotemporal autoencoder could extract the high-dimensional features of video cases, and the main task of its method was video reconstruction. In the training phase, the reconstruction loss was expected to be lower. In the inference phase, if the loss value was larger than a threshold, it would be judged to be an abnormal sample. Cho et al. [13] suggested that abnormal behavior was determined by appearance and motion, so they proposed an unsupervised method that could learn both appearance and motion features by two branches, which were static flow and dynamic flow. They inputted $T/\tau$ frames into the static encoder ($T$ was the total number of frames of the video case and $\tau = 4$ was the sampling rate) while they inputted $T$ frames into the dynamic encoder. Fewer frames allowed static flow to focus on learning the appearance features of objects in the video case. More frames allowed dynamic flow to learn the changes between two frames and obtain motion features. Similar to Deepak et al. [12], the main task was reconstruction.

Supervised methods usually treat both abnormal and normal behaviors as different classification categories. The key frame extractor of Tang et al. [16] was also a prefect abnormal behavior detector. The purpose of the paper was to train a key frame extractor unsupervised, but in terms of a video classification task, it could be considered as a supervised method. The key frame extractor was divided into two stages. In stage one, the video

frames were transformed into feature maps through CNN. Then, the model adopted an algorithm similar to k-means to extract key frames using those feature maps. The clustering algorithm redefined the calculation method, which could reflect the representative of each feature map in the video case it belonged to, rather than the traditional calculation of the sum of the distance. The representative scores were sorted from high to low, and the frames corresponding to top $k$ scores would be selected to the next stage, which was an extractor of the temporal feature. In stage two, the key frames were sorted by time order, and were fed into an LSTM network as a time series. The output of the LSTM was sent to a classifier for the video classification task. The work of Tan et al. [17] had the ability to deal with long videos with multiple scenarios and themes. Long videos were cropped into small video clips according to scenarios and themes by large model TransNetV2 [29]. These video clips extracted feature maps frame by frame with an image feature extractor. An adaptive cluster strategy was adopted to select the key frames from each video clip. The number of cluster centers, in other words, the number of key frames, was not a fixed value but could change according to different video samples. Obtaining the feature maps of key frames, the model learned the distributions of different categories of video clips supervised by classification labels.

Unsupervised methods exhibit strong adaptability across various scenarios. However, their accuracy tends to be limited. In contrast, supervised methods offer relatively high accuracy but lack adaptability to diverse scenarios. This paper proposes an abnormal behavior detection approach that is suitable for different scenarios. For scenarios demanding high accuracy, adaptive clustering of frames based on a large visual model is employed. Conversely, for scenarios prioritizing speed, random frames or center frames are selected.

### 2.3. Video Coding Standards

The development of video coding standards has spanned several important stages.

In 1988, the H.261 [30] standard was proposed by the International Telecommunication Union Telecommunication Standardization Sector (ITU-T), aiming to support Integrated Services Digital Network (ISDN) video telephony and video conferencing applications. This standard was the first to introduce block motion compensation and discrete cosine transform (DCT) technology to the video coding field. However, H.261 exhibited limitations in handling low bit rates and high-motion scenes, which spurred the development of subsequent standards.

In 1991, the Moving Picture Experts Group (MPEG) of International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) released the MPEG-1 [31] standard, which aimed at storing and playing video on CDs and digital cable/satellite televisions. Five years later, the video part of MPEG-2 [32] standard was proposed. MPEG-2 significantly improved upon MPEG-1 by supporting higher resolutions and bit rates and introducing variable bit rate (VBR) and layered coding technologies. Despite its widespread application in digital television and digital video disc (DVD) storage, MPEG-2's coding efficiency was insufficient for high-resolution video.

In 1998, the H.264 [33] standard was jointly released by the ISO/IEC Joint Technical Committee (JTC), which proposed Context-Adaptive Binary Arithmetic Coding (CABAC) and multireference frame technology. Compared with MPEG-2, H.264 achieved 50% higher compression efficiency at most, supporting more granular bit rates, resolutions, and frame rates. This versatility made it widely adopted in high-definition television (HDTV), Blu-ray discs, and online video streaming. However, the computational complexity and patent fees of H.264 prompted researchers to continue exploring more efficient coding standards. The H.265 [34] (also known as high-efficiency video coding, HEVC) standard was introduced in 2013, incorporating more complex prediction and transformation techniques, such as coding units (CUs), prediction units (PUs), and transform units (TUs). These advancements increased compression efficiency by approximately 50% and supported video transmission up to 8K resolution. These technological improvements made H.265 widely used in 4K and 8K ultra-high-definition television, video streaming services, and virtual reality (VR).

Nonetheless, the high computational complexity and patent fee issues of H.265 remain significant barriers to its widespread adoption.

## 3. Methods

The framework of TSPAD is illustrated in Figure 1. At the vehicle end, the message publisher, embedded with an abnormal behavior detection module, detects dangerous behaviors, such as eating and drinking, working with phone or laptop, reading or writing magazine or newspaper, watching video and smoking, by the drivers during vehicle operation in real time. Key frames of surveillance videos are compressed and recorded in the certification blockchain by smart contracts. The message subscribers, consisting of logistics platform, enterprises, and regulatory authority, receive real-time updates of newly generated transactions on the certification chain. The private key chain-based key distribution mechanism performs identity verification and permission control during data retrieval. The techniques used in TSPAD will be elaborated in the rest of this section.

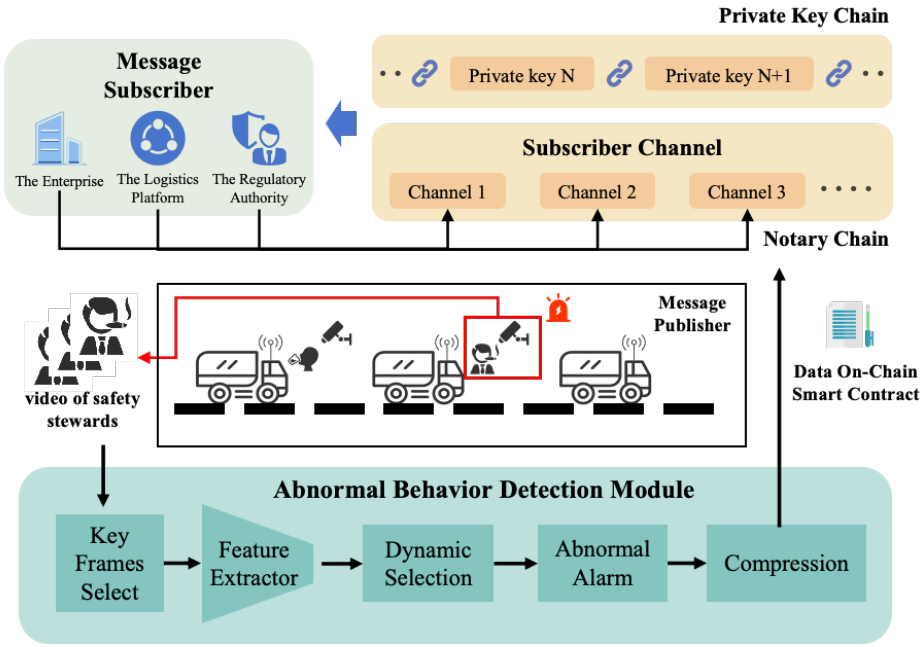


**Figure 1.** Framework of TSPAD based on multimodal data authentication, which illustrates an anomaly detection module integrated with a dual-chain architecture. TSPAD employs a compact anomaly detection model to capture key frames and, through feature extraction and a dynamic selection mechanism, identifies and records potentially abnormal driving behaviors in real time, such as fatigue driving and reckless driving. Upon detecting such behaviors, the system triggers an alert and processes the data using video compression techniques to accommodate real-time uploads to the notary chain. The entire dual-chain architecture comprises a private key chain and a notary chain. The private key chain is responsible for implementing identity authentication and management, ensuring data privacy and security, while the notary chain records real-time driving data to guarantee data integrity and reliability. Blockchain participants such as enterprises, logistics platforms, and regulatory agencies can access vehicle operation information on the notary chain through subscriber channels. This capability enables real-time monitoring and management of the transportation process, thereby enhancing system transparency and security.

As shown in Figure 2, in this section, Section 3.1 designs a blockchain system for trusted evidence preservation, providing a foundational overview of the entire framework. This system ensures data integrity and security through a dual-chain architecture and an evidence notarization process. Section 3.2 focuses on the detection of abnormal driving behaviors, utilizing a key frame adaptive clustering algorithm to analyze videos and identify potentially hazardous behaviors during driving. Section 3.3 addresses video

compression based on intraframe and interframe coding, effectively reducing data volume and enhancing storage and transmission efficiency through discrete cosine transform and motion compensation techniques. The abnormal behavior data obtained in Section 3.2 are compressed in Section 3.3 and stored within the blockchain architecture of Section 3.1, ensuring data security and integrity. Thus, Section 3.2 provides the raw data for processing, while Section 3.1 offers a trusted platform for data storage and management throughout the entire process.

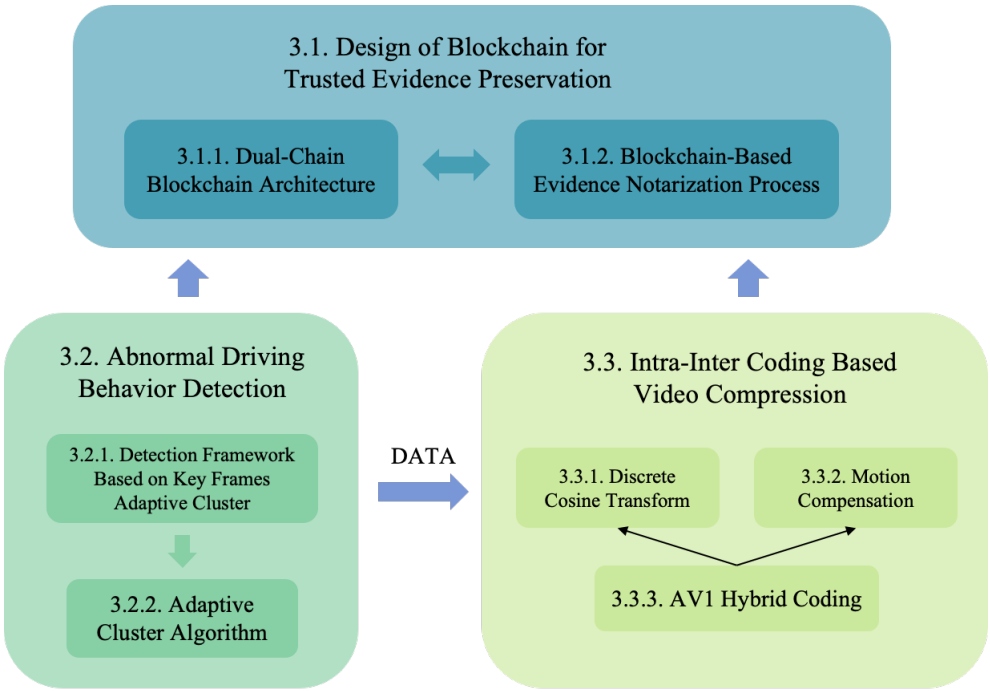


**Figure 2.** The logical structure of the "Methods" section in this paper.

### *3.1. Design of Blockchain for Trusted Evidence Preservation*

#### 3.1.1. Dual-Chain Blockchain Architecture

In the field of modern logistics and autonomous driving, it is crucial for the notarization of vehicle operation information and driver abnormal behavior detection to ensure the safe operation of vehicles and the delineation of responsibilities post-accident. These data are typically characterized by their vast volume, randomness, and dispersion, posing challenges to traditional notary systems [35]. To address these challenges, this paper introduces an innovative dual-chain architecture to achieve the reliability and openness of vehicle operation information while preserving the security and privacy of corporate cargo information.

The proposed dual-chain architecture, as shown in Figure 3, involves the independent deployment and governance of a private key chain and a notary chain. Each chain interacts with blockchain participants, with on-chain data being stored and managed separately, thus achieving a decoupling of identity management and data notarization functions. The private key chain is responsible for identity management and data privacy protection, while the notary chain records real-time electronic evidence generated by the vehicle, including text and video formats. This design ensures efficient data access and credible notarization by separating key management from the notarized information. As a result, enterprises involved in transactions can receive real-time updates on the transportation status of vehicles and cargo, ensuring data privacy while maintaining the security and reliability of logistics information. The system's functionalities are primarily implemented through the interface of smart contracts, ensuring the automation and transparency of operations. After the data are encrypted, they are broadcasted to the other nodes through a

peer-to-peer (P2P) network, and then stored on the chain through a consensus mechanism, thereby enhancing the security and immutability of the data.

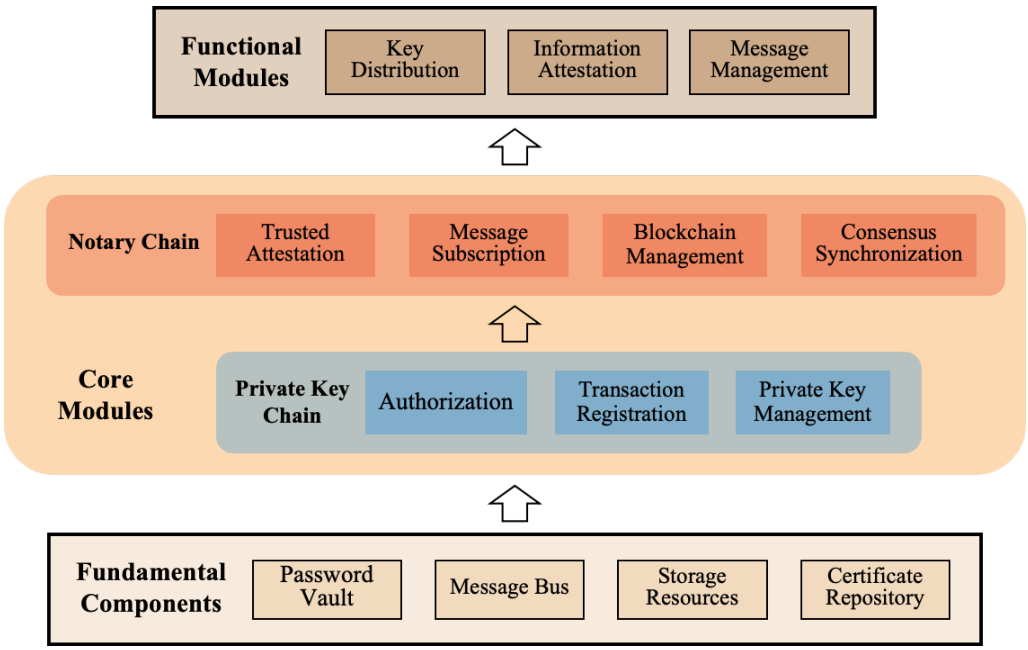


**Figure 3.** Trusted evidence dual-chain blockchain architecture which illustrates the core components of the dual-chain blockchain architecture, including foundational components, functional modules, and core modules. The foundational components at the bottom layer (password vault, message bus, storage resources, and certificate repository) provide supportive services for the core modules in the middle section. The private key chain is responsible for user identity authentication, transaction registration, and private key management, providing foundational support for system security and privacy protection. The notary chain handles trust proof, message subscription, blockchain management, and consensus synchronization, ensuring data authenticity and network synchronization. The top-level functional modules enable the extension of system functions and service integration through key distribution, information authentication, and message management. The overall architecture ensures the credibility of information and the traceability of operations through the combination of the notary chain and the private key chain.

The private key chain is primarily responsible for recording the details of order transactions between logistics companies and enterprises and storing the encrypted transaction private keys. Anchoring order transaction information on the blockchain ensures the immutability and high transparency of the information. The transaction private key encrypted by the enterprise's or the regulatory authority's public key is notarized and stored on the private key chain by a trusted logistics company. Consequently, the involved enterprise can retrieve the transaction private key from the chain, which enhances the transparency and reliability of the private key distribution process. The private key distribution enables user identity verification and authorization when enterprises retrieve transportation information. Only the transportation information encrypted by the corresponding transaction public key can be accessed and decrypted by the enterprise involved in the transaction, thereby improving the security of the system.

The notary chain is employed to document the vehicle's operational data, the condition of the cargo, and videos of abnormal behavior detection of drivers during transportation. The data uploaded to the chain include information such as timestamps, vehicle identification numbers, and the original data. Before uploading electronic evidence to the notary chain, the vehicle encrypts it using the transaction public key, enabling user identity authentication for enterprises when retrieving transportation information. This mechanism

ensures the authenticity, integrity, and traceability of transportation information, while also securing the privacy of corporate cargo information.

This paper adopts a dual-chain architecture based on the following considerations: (1) The design of the dual-chain architecture enables specialization of chain functions, facilitating the management and expansion of subsequent notary information services, as well as the adjustment and modification of user authentication methods. (2) Encrypted private key information is stored separately in the private key chain, effectively isolating it from the data on the notary chain, thereby providing higher privacy and security for the transaction content. (3) The dual-chain architecture enables each chain to select its own trusted users to join, reducing the opportunities for malicious attacks. (4) The dual-chain architecture is more applicable to the key distribution mechanism.

#### 3.1.2. Blockchain-Based Evidence Notarization Process

As shown in Figure 4, the dual-chain architecture proposed in this paper consists of five main processes: (A) transaction initiation, (B) transaction public–private key distribution, (C) transportation information on-chain, (D) enterprise's retrieval of transportation information, and (E) regulatory intervention.

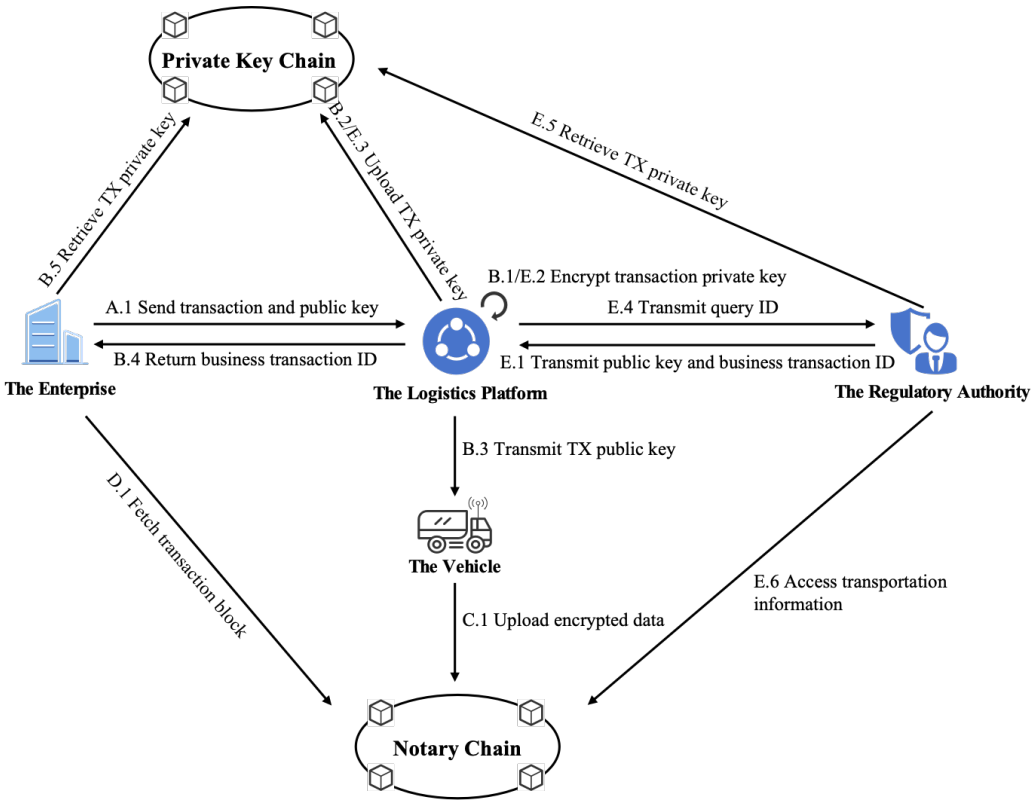


**Figure 4.** Evidence service model for evidence upload and retrieval.

**Transaction initiation**. After reaching a transaction agreement off-chain, the enterprise generates an asymmetric private key $SK_e$ and initiates a new transaction with the logistics platform, transmitting the public key $PK_e$ for data encryption to the logistics platform.

**Transaction public–private key distribution**. Upon receiving this transaction request, the logistics platform generates a corresponding set of transaction asymmetric keys and a business transaction ID. The transaction content and the private key $SK_{tx}$ of the asymmetric keys are encrypted using the public key $PK_e$ provided by the enterprise. The encrypted data are serialized and packaged into a blockchain transaction, which is then uploaded to the private key chain with the business transaction ID as the blockchain transaction index. The generated public key $PK_{tx}$ is distributed to the vehicle executing the task to

encrypt the data recorded during transportation. After $SK_{tx}$ is uploaded to the chain, the logistics platform returns the business transaction ID to the enterprise, which retrieves the corresponding block on the private key chain using the business transaction ID and decrypts the $SK_{tx}$ with its own private key $SK_e$.

**Transportation information on-chain**. The vehicle computes a hash for the recorded vehicle driving data or the compressed video of driver's abnormal behavior. This hash value can be used to verify whether the data have been tampered with during subsequent evidence collection. Using a hybrid encryption method, the vehicle generates a symmetric key to encrypt the data and encrypts the symmetric key with the transaction public key $PK_{tx}$. The encrypted data and key are then chained onto the notary chain with the business transaction ID as the blockchain transaction index. The purpose of this approach is to leverage the security provided by asymmetric encryption algorithms to safely transmit the symmetric key and then use the efficiency of symmetric encryption algorithms to encrypt large amounts of data.

**Retrieval of transportation information**. The enterprise creates a subscriber channel within the notary chain through the blockchain's message subscription mechanism. When a new transaction is generated, the blockchain sends a notification through the subscriber channel. Enterprises can retrieve blocks associated with their transactions using the business transaction ID and decrypt the symmetric key with the transaction private key $SK_{tx}$. Through this approach, enterprises can access real-time information on vehicle operations and cargo transportation.

**Regulatory intervention**. The regulatory authority possesses a private asymmetric key pair. When the regulatory authority needs to access the transportation information of a particular transaction, it initiates a request to the logistics platform, transmitting its public key $PK_r$ and the business transaction ID to be queried. Upon verifying the digital identity of the regulatory authority, the logistics platform uses the transmitted public key $PK_r$ to encrypt the transaction private key $SK_{tx}$. Subsequently, the logistics platform generates a query ID to serve as the index for the blockchain transaction and uploads the encrypted transaction private key to the private key chain. Finally, the logistics platform returns the query ID to the regulatory authority, enabling it to use this ID and its private key $SK_r$ to retrieve the transaction private key from the private key chain, and thereby access the desired transportation information from the notary chain.

### *3.2. Abnormal Driving Behavior Detection*

#### 3.2.1. Detection Framework Based on Key Frames Adaptive Cluster

The abnormal driving behavior detection framework is shown in Figure 5. Given a video shot $V = \{v_1, v_2, \ldots, v_n\}$, $v_i$ denotes its $i$-th frame, and $n$ is the total number of its all frames. The number of frames of video shot is normalized to $f$ using uniform sampling ($n > f$) or interpolation ($n < f$) methods before training, so the video shot becomes $V = \{v_1, v_2, \ldots, v_f\}$. Feeding $V$ into the key frames selection module, key frames $V_k$ are selected from $V$. There are three selection strategies that can be used: (1) select center frame $V_k = \{v_{f/2}\}$, (2) randomly select one frame $V_k = \{v_i\}, v_i \in V$, and (3) select adaptive cluster frames $V_k = \{v_1, v_2, \ldots, v_c\}$. In adaptive cluster frames module, the image encoder of the Contrastive Language-Image Pre-training [36] (CLIP) model is used to extract the high-dimensional features of video frames, and each feature has 512 dimensions. The adaptive cluster algorithm takes these high-dimensional features as the input and outputs the index of key frames that can represent the semantic of video shot. The adaptive cluster algorithm has the best performance in all of three strategies, and $c$ frames are selected from $V$ to form $V_k$. Feeding $V_k$ to the feature extractor module to obtain the high-dimensional representations, each representation has 2048 dimensions. Then, the dynamic selection module selects one frame tensor $f_i$ from $c$ representations, which can use random selection strategy or extract one frame at fixed location. The classifier module is a multilayer perceptron (MLP) which accepts $f_i$ as the input and outputs $X \in \mathbb{R}^{n \times K}$,

where $n$ is the number of samples in one batch and $K$ is the quantity of categories in this classification task. The loss function is as follows:

$$Loss = -\frac{1}{n}\sum_{i=1}^{n}(w_i \times \sum_{j=1}^{K}(\log(\frac{e^{x_{ij}}}{\sum_{i=1}^{K} e^{x_{ij}}})y_{ij})) \quad (1)$$

where $x_{ij} \in X$ and $y_{ij} \in \{0, 1\}$ is the $j$-th category one-hot label of the $i$-th sample. The $w_i = \frac{N_j}{N}$ represents the weight of the category, where $N$ is the total number of samples in the dataset and $N_j$ is the number of samples of the $j$-th category.

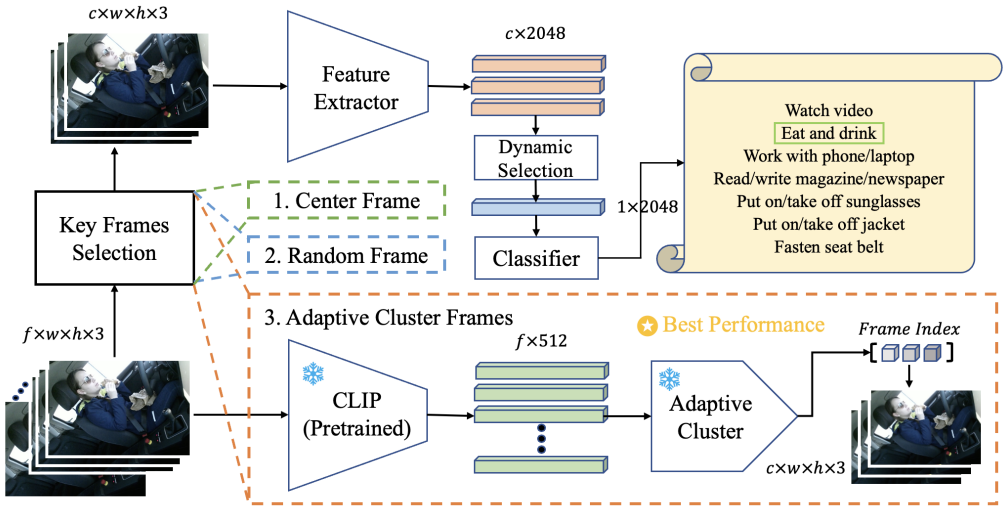


**Figure 5.** Framework of abnormal driving behavior detection. The dotted box represents 3 different key frame selection strategies that can be selected in practical applications. The adaptive cluster strategy has the best performance. The colored cuboids represent the feature tensors of video frames, and the gray cubes are indexes of key frames. The colorless modules with solid lines are functional modules of the network, and those with snowflake icons indicate that no training is required. The original images are samples of the public dataset Drive&Act [37].

### 3.2.2. Adaptive Cluster Algorithm

Inspired by Tan et al. [17], the adaptive cluster algorithm is adopted to select key frames from video shot, which is similar to K-means. The algorithm is illustrated in Algorithm 1. The initial cluster centers are determined by minimizing the center initialization error ($CIE$):

$$CIE(M) = \sum_{k=1}^{n}\sum_{j=1}^{|M|}(f_{kj} - C_j)^2, \quad (2)$$

where $f_{kj}$ represents the feature vector of the $k$-th frame belonging to the $j$-th cluster, $|M|$ is the number of cluster centers ($M$ is the cluster center set), $C_j$ represents the cluster center of the $j$-th cluster, and $n$ is the number of frame features of video shot $V$.

Before adaptive clustering, video frames need to be uniformly clustered into $k_{max} = \lceil\sqrt{n}\rceil$ categories ($\lceil\bullet\rceil$ denotes round up), which needs $k_{max}$ steps. In each step, the $CIE$ values corresponding to their feature vectors are calculated and then one feature vector with the minimum $CIE$ is chosen to be a new cluster center and is added to $M$ ($M$ is empty at first). When the number of cluster centers reaches $k_{max}$, the above steps are terminated.

The $k_{max}$ cluster centers selected at this time may be redundant, which can be determined as whether the location and number of cluster centers are optimal in the following ways, using the $CCE$ (center correction error):

$$CCE = \frac{1}{n}\sum_{i=1}^{n}\frac{o(i) - s(i)}{\max\{s(i), o(i)\}}, \quad (3)$$

where $s(i)$ denotes the average Euclid distance of the frame feature $f_i$ to the other features within the same cluster and $o(i)$ marks the minimum Euclid distance of $f_i$ to the other cluster centers.

The specific method is to merge the two nearest cluster centers each time and calculate the corresponding $CCE$ value. The merged new cluster center is the mean of all samples belonging to the original two cluster centers. All $CCE$ values and their corresponding cluster centers will be recorded, where $\Omega$ is responsible for recording the $CCE$ values, and $\Phi$ is responsible for recording the cluster centers. When only one cluster center remains, the above operation is terminated. Finally, we select the cluster center corresponding to the largest $CCE$ value (optimal clustering strategy). However, these clustering centers may not coincide with the real frame features, so the frames closest to the clustering centers are selected as the actual key frames, and their index is $A_i$.

**Algorithm 1:** Adaptive Cluster Algorithm.

**Data:** Frames' features $F = \{f_1, f_2, \dots, f_n\}$, $n$ is the frame number of the shot.
**Result:** Index of key frames $I = \{i_1, i_2, \dots, i_m\}$, $m \leq k_{max}$, $k_{max}$ is a hyper-parameter, which indicates the maximum number of cluster centers.

$M = \{\}$
**while** $|M| \leq k_{max}$ **do**
  $c = \arg\min_{f_i \in F, f_i \notin M} CIE(M \cup \{f_i\}), \quad M = M \cup \{c\}$
**end**
**while** $|C| \geq 2$ **do**
  $f_i, j = \arg\min_{j \in [1, k_{max}],\ f_i \in F,\ f_i \notin M} distance(M[j],\ f_i)$;
  $C_j = C_j \cup \{f_i\}, \quad C = \{C_1, C_2, \dots, C_{max}\}$;
  $\Omega = \{\}, \quad \Phi = \{\}$;
  Calculate $CCE, \quad \Omega = \Omega \cup CCE, \quad \Phi = \Phi \cup C$;
  $j, k = \arg\min_{j \in [1, |M|]} distance(M[j],\ M[k])$;
  $C = \{C_1, C_2, \dots, C_j \cup C_k, \dots\}$;
  Update $M$;
**end**
$j = \arg\max_{j \in [1, k_{max}]} \Omega[j], \quad A_i = \Phi[j]$;
**Return** $Index(F,\ A_i)$

### *3.3. Intra–Intercoding-Based Video Compression*

In the CAP (consistency availability partition) tolerance theorem [38], blockchain as a distributed system sacrifices consistency to meet availability and partition tolerance. Through consensus algorithms, nodes partially alleviate the difficulty of cluster synchronization in an asynchronously consistent manner. However, challenges persist in low throughput and high latency in blockchain systems. In the context of automated driving evidence storage, stringent requirements are imposed on the timeliness of data for blockchain upload to ensure the credibility of evidence. Compressing videos before uploading to the blockchain while retaining semantic integrity is one feasible solution to address these issues. This paper adopts the AOMedia Video 1 (AV1) video coder, which combines DCT and motion compensation techniques, offering an efficient and viable method to reduce upload time to the blockchain, as illustrated in Figure 6 for the overall architecture.

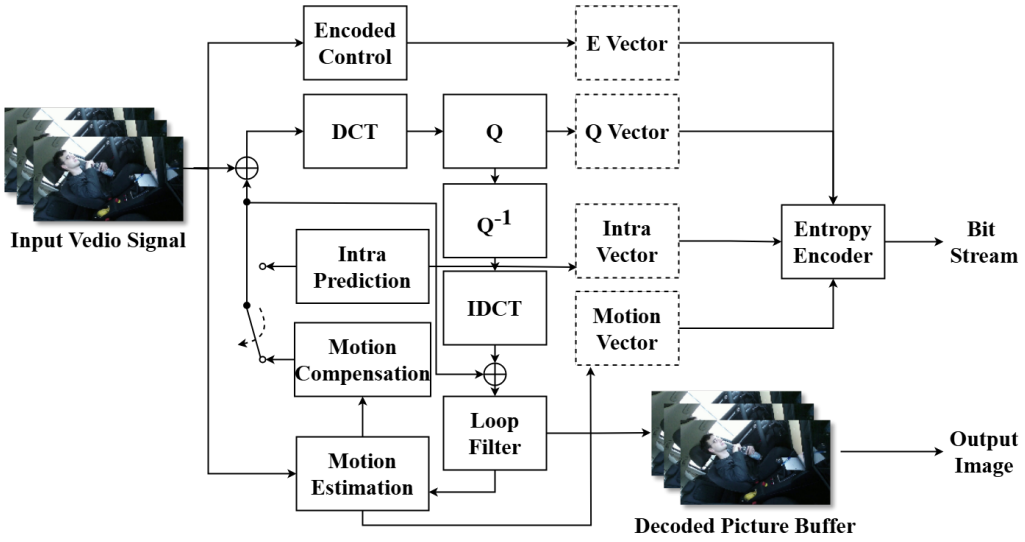


**Figure 6.** Hybrid Coding Architecture. The Hybrid Coding Architecture involves several key components and processes. The Encoded Control determines the strategy for encoding and quantization. Its parameters are output in the form of a Control Vector. The DCT converts the input video signal from the temporal domain into continuous frequency domain signals. Quantization (Q) maps the frequency domain signals into a stepped finite field. Its parameters are output as a Q Vector. Intra Prediction removes spatial redundancy within a single frame. The parameters set for Intra Prediction can be formulated as an Intra Vector. Motion Estimation and Motion Compensation, which eliminate temporal redundancy between adjacent frames, output their parameters as Motion Vector. The Loop Filter reduces block artifacts after quantization. The Decoded Picture Buffer (DPB) stores the decoded image frames for use in interframe prediction. The Entropy Encoder performs lossless encoding of the aforementioned parameters, combined with essential reference frames, to produce the compressed bitstream. The original images are samples of the public dataset Drive&Act [37].

#### 3.3.1. Discrete Cosine Transform

Semantic information in real-world images is usually concentrated in the low-frequency domain. Human perception of chrominance information exhibits a nonlinear relationship [39], leading to the advantage of selectively reducing high-frequency components for more efficient encoding with fewer bits by the encoder. DCT [40] serves as the central module for video compression, responsible for transforming spatial domain image signals into the frequency domain, which works in conjunction with subsequent quantization and entropy coding to facilitate lossy compression. The specific formulas for DCT forward and inverse transforms are as follows:

- Two-dimensional DCT. Let there be an image of size $W \times H$, where $f(i,j)$ represents the value of the pixel at row $i$ and column $j$. The image is transformed from the pixel matrix $f(i,j)$ to the frequency domain matrix $F(u,v)$ after DCT, where u and v represent the indexes of the horizontal and vertical frequency components, respectively. The definition of $F(u,v)$ is as follows:

$$F(u,v) = \frac{2G(u)G(v)}{\sqrt{WH}} \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} \cos\left[\frac{(2i+1)u\pi}{2W}\right] \cos\left[\frac{(2j+1)v\pi}{2H}\right] f(i,j), \quad (4)$$

$$G(x) = \begin{cases} \frac{1}{\sqrt{2}} & , x = 0 \\ 1 & , x \neq 0 \end{cases}, \quad (5)$$

  where $u = \{0,1,2,\cdots,W\}$, $v = \{0,1,2,\cdots,H\}$.
- Two-dimensional inverse discrete cosine transform (IDCT). The inverse transform $\tilde{f}(i,j)$ is given by:

$$\tilde{f}(i,j) = \sum_{u=0}^{W-1} \sum_{v=0}^{H-1} \frac{2G(u)G(v)}{\sqrt{WH}} \cos\left[\frac{(2i+1)u\pi}{2W}\right] \cos\left[\frac{(2j+1)v\pi}{2H}\right] F(u,v), \quad (6)$$

- Matrix formulation of DCT. The DCT formula is characterized by cyclic summation, which does not effectively harness the efficient matrix computation capabilities of GPUs. Therefore, the summation formula is often converted into matrix form to enhance computational efficiency. The specific formula is as follows:

$$F = M(i,j) \cdot f(i,j) \cdot M^T(i,j), \quad (7)$$

$$M(i,j) = \begin{cases} \frac{1}{\sqrt{L}} & , i = 0 \\ \sqrt{\frac{2}{L}} \cdot \cos\frac{(2j+1)\cdot i\pi}{2L} & , i > 0 \end{cases}, \quad (8)$$

where $M^T(i,j)$ is transposed matrix of $M(i,j)$, $i,j = \{0,1,2,\cdots,L\}$.

3.3.2. Motion Compensation

Moving objects between consecutive frames can be considered as displacements of fixed macroblocks. Motion compensation replaces the original pixel information in the target frame with the displacement vectors of reference frame macroblocks and their corresponding temporal differences [41]. The process involves three steps:

1. Compare pixel differences between adjacent frames to estimate the direction and size of macroblock motion;
2. Use the results of motion estimation for inter-frame prediction;
3. Generate the residual output by computing the difference between actual values and predicted values.

The residual calculation method is as follows:

$$MAD(i,j) = \frac{1}{W_{mac}H_{mac}} \sum_{k=0}^{W_{mac}-1} \sum_{l=0}^{H_{mac}-1} |T(x+k,y+l) - R(x+i+k,y+j+l)|, \quad (9)$$

where $i$ and $j$ represent the motion vectors, $W_{mac}$ and $H_{mac}$ are the width and length of the macroblock, $k$ and $l$ represent the row and column indexes of the macroblock, $x$ and $y$ are the coordinates of the top-left corner of the macroblock, $T$ denotes the pixel points of the target macroblock, and $R$ denotes the pixel points of the reference macroblock after displacement.

3.3.3. AV1 Hybrid Coding

AV1 adopts a hybrid coding architecture consisting of intraframe coding and interframe coding. The intraframe coding predicts and encodes the feature information of a single frame image under built-in prediction modes, reducing spatial redundancy by recording residual information between predicted macroblocks and reference macroblocks. The specific process is as follows:

1. Segmentation of the single frame image into macroblocks of different sizes;
2. Flexibly selecting from 56 prediction modes based on image texture to obtain intraframe prediction signals;
3. Calculating the residual signal by subtracting the predicted macroblocks from the original macroblocks;
4. Applying cosine Fourier transform and quantization to the residual signal to remove high-frequency information;
5. Completing lossy compression by inverse processing the processed frequency domain residual signal;
6. Adding the residual signal to the intraframe prediction signal and using loop filtering to mitigate block artifacts;

7. Outputting the result through entropy coding.

The interframe coding reduces temporal redundancy in video frames by recording the displacements of macroblock vectors and the numerical differences between adjacent frames. Unlike intraframe coding, it uses interframe motion compensation instead of fixed prediction modes in the second step.

The AV1 format has introduced several advancements and innovations across its various submodules, significantly enhancing its performance and efficiency [42]. Notably, in the area of macroblock partitioning, the macroblock size has been increased to 128 × 128, while the chroma interframe prediction resolution has been reduced to 2 × 2. In the realm of intraframe prediction, the granularity of angle prediction modes has been refined to 3 degrees, and specialized encoding tools have been developed for artificial content. For interframe prediction, the number of reference frames per frame has been increased to 7, and the scope of motion vector searches in both temporal and spatial domains has been extended through the use of temporal motion field estimation. Collectively, these improvements enable the AV1 encoder to achieve an average bit rate reduction exceeding 30% compared to its predecessor, the VP9 encoder, under equivalent conditions.

## 4. Experiment Result

### 4.1. Dataset

The publicly available dataset Drive&Act [37] was reorganized for the validation of the method proposed in this paper. The Drive&Act dataset is a multimodal benchmark for action recognition in automated vehicles. It has 12 h of video data in 29 long sequences, consisting of 3 modals: near infrared reflectance (NIR), depth, and color data. The dataset is calibrated camera system with 5 views: center mirror, driver, co-driver, ceiling and steering wheel. In this paper, the co-driver view color data with task-level labels are adopted. Task-level label consists of 12 categories of driving behaviors, such as 'watch video', 'eat drink', 'work', 'read and write newspapers', 'read and write magazine', 'put on jacket', 'fasten seat belt', 'final task', 'take off jacket', 'put on sunglasses', 'take off sunglasses', and 'hand over'.

However, there are a lot of mislabeled samples with the task-level labels. As shown in Figure 7, panels (a) and (c) are the frames of video cases with error (dirty) labels. The driver in (a) is obviously watching a video, but is labeled 'eat drink'. Similarly, the person in (c) is driving normally, while the label is 'read and write magazine'. The CNN learning of the true distribution of the dataset will be disturbed by these dirty samples. Thus, this paper relabels the Drive&Act dataset with task-level labels. A straightforward and efficient method is employed, which involves reviewing video cases individually, deleting incorrectly labeled samples, and retaining correctly labeled ones. The video cases like Figure 7b,d are mostly remained, while (b) is labeled as 'eat drink' and (d) is labeled as 'read write magazine'.

In addition, the categories of 'put on jacket' and 'take off jacket' are combined into one category, 'off on jacket'. The 'put on sunglasses' and 'take off sunglasses' are also combined into 'off on sunglasses'. The new category 'read write magazine newspaper' comes from 'read write magazine' and 'read write newspaper'. This paper mainly focuses on abnormal behaviors during driving, so the 'final task' is considered useless and is removed from the task. After relabeling, the Drive&Act dataset contains seven categories. To clarify, histograms of the original and relabeled sample categories are shown in Figure 8, with panel (a) representing the data before relabeling and panel (b) representing the data after relabeling.

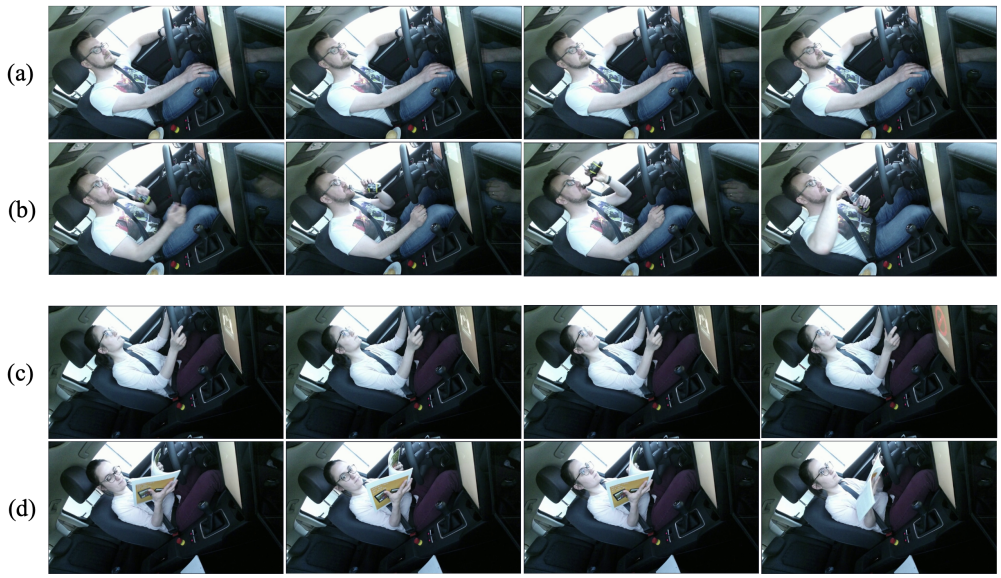


**Figure 7.** Comparison between dirty labels and clean labels. In the illustration, panels (**a**,**c**) represent the frames of video cases with dirty labels, while (**b**,**d**) are frames of video cases with clean labels. The original images are samples of the public dataset Drive&Act [37].

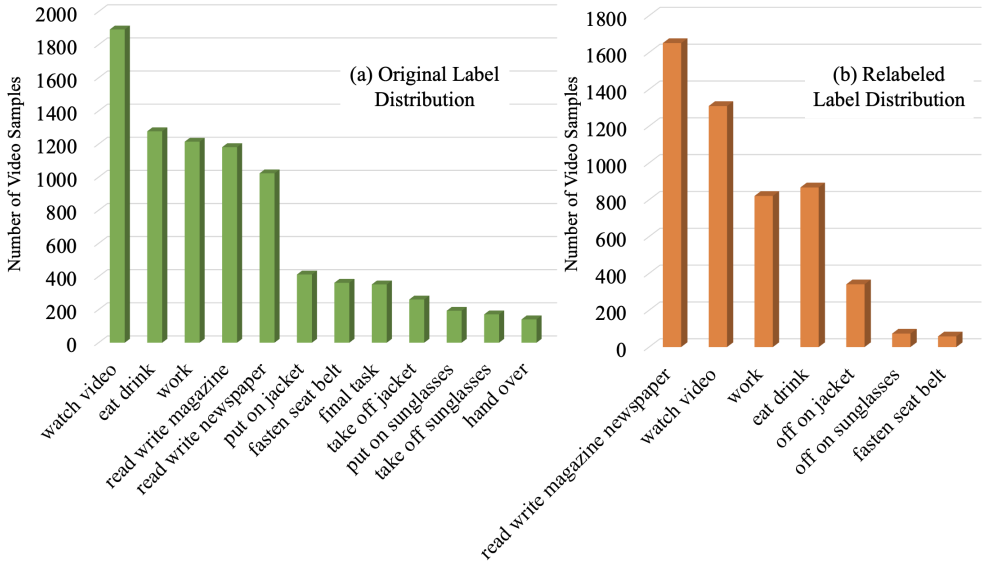


**Figure 8.** Histogram of original labels and relabeled labels of Drive&Act [37] dataset. Panel (**a**) represents the histogram of original labels which consists of 12 categories, and panel (**b**) shows the relabeled labels histogram, which consists of 7 categories.

### *4.2. Driving Abnormal Behavior Detection*

The samples of Drive&Act dataset belong to 16 different drivers; 3 of these are selected to be the test dataset, and are never used by the model during the training phase. For the feature extractor, this paper uses ResNet [1] series (ResNet-18, ResNet-34 and ResNet-50), lightweight MobileNetV3 [43] series (MobileNetV3-Small, MobileNetV3-Large), and CLIP [36] as the backbone. The CLIP model is loaded with pretrained parameters before training, and it is frozen throughout the training process. The experiment results are shown in Table 1.

**Table 1.** Experiments of driving abnormal behavior detection using MobileNet and ResNet as feature extractor.

| Backbone | Parameters | $M_P$ | $M_R$ | $M_{F1}$ | $W_P$ | $W_R$ | $W_{F1}$ | *Top3* |
|---|---|---|---|---|---|---|---|---|
| CLIP(Frozen) [36] | - | 0.458 | 0.447 | 0.401 | 0.652 | 0.589 | 0.550 | 0.860 |
| MBNetV3-S [1] [43] | 5 M | 0.525 | 0.489 | 0.472 | 0.666 | 0.535 | 0.562 | 0.839 |
| MBNetV3-L [2] [43] | 14 M | 0.609 | 0.648 | 0.570 | 0.732 | 0.641 | 0.642 | 0.810 |
| ResNet-18 [1] | 43 M | 0.678 | 0.722 | 0.673 | 0.758 | 0.736 | 0.727 | 0.911 |
| ResNet-34 [1] | 82 M | 0.640 | 0.687 | 0.630 | 0.789 | 0.755 | 0.756 | 0.895 |
| ResNet-50 [1] | 100 M | **0.743** | **0.795** | **0.754** | **0.827** | **0.803** | **0.805** | **0.926** |

[1] MBNetV3-S represents MobileNetV3-Small. [2] MBNetV3-L represents MobileNetV3-Large. The bold numbers represent the optimal values of the indicators.

To explain other evaluation indexes, the meaning of the following parameters must be understood:

$$P_i = \frac{TP_i}{TP_i + FP_i}, \quad R_i = \frac{TP_i}{TP_i + FN_i}, \quad F1_i = 2 \times \frac{P_i \times R_i}{P_i + R_i} \tag{10}$$

where $TP_i$ represents the number of samples that belong to category $i$ that are classified correctly, $FP_i$ represents the number of samples that belong to category $i$ that are misclassified into other categories, while $FN_i$ represents the number of samples that do not belong to category $i$ but are classified into category $i$. Therefore, $P_i$ represents the accuracy rate of category $i$, $R_i$ is the recall rate of it, and $F1_i$ is a comprehensive evaluation index that combines $P_i$ and $R_i$.

In Table 1, $M_P$ is the average precision, $M_R$ is the average recall, and $M_{F1}$ is the average $F1$ score. All of them treat each category equally and their formulas are as follows:

$$M_P = \frac{1}{K}\sum_{i=1}^{K} P_i, \quad M_R = \frac{1}{K}\sum_{i=1}^{K} R_i, \quad M_{F1} = \frac{1}{K}\sum_{i=1}^{K} F1_i, \tag{11}$$

where $K$ is the number of all categories. Another strategy for evaluation is weighted by number of samples, which is illustrated by the following:

$$W_P = \frac{n_i}{N}\sum_{i=1}^{K} P_i, \quad W_R = \frac{n_i}{N}\sum_{i=1}^{K} R_i, \quad W_{F1} = \frac{n_i}{N}\sum_{i=1}^{K} F1_i, \tag{12}$$

where $N$ is the total number of all samples, $n_i$ is the number of samples belonging to category $i$, K is the number of all categories, and $Top3$ is defined as follows:

$$Top3 = \frac{C_{top3}}{N}, \tag{13}$$

where $N$ is the total number of all samples and $C_{top3}$ is the number of samples where the correct label is among the top three predicted labels. It can be seen from the experimental results that the model performance is positively correlated with the number of parameters.

The ablation experiment is shown in Table 2, where 'Random Frame' is the strategy of selecting a frame randomly from the video case, 'Center Frame' is selecting the center frame, and 'Cluster Frame' is selecting frames using the adaptive cluster strategy. All evaluation metrics are consistent with those in Table 1. It can be seen that the adaptive cluster strategy achieves the best results on most metrics.

Table 3 presents the precision, recall, and F1 score for all categories. The performance metrics of each category are influenced by the number of samples available. For instance, the performance of the category of 'OOS' is relatively poor due to its limited sample size.

**Table 2.** Ablation experiments with different key frame selection methods.

| Method | $M_P$ | $M_R$ | $M_{F1}$ | $W_P$ | $W_R$ | $W_{F1}$ | *Top3* |
|---|---|---|---|---|---|---|---|
| Random Frame | 0.693 | 0.682 | 0.678 | 0.778 | 0.792 | 0.780 | **0.937** |
| Center Frame | 0.740 | 0.776 | 0.718 | **0.834** | 0.797 | 0.795 | 0.925 |
| Cluster Frame | **0.743** | **0.795** | **0.754** | 0.827 | **0.803** | **0.805** | 0.926 |

The bold numbers represent the optimal values of the indicators.

**Table 3.** Performance in different categories by ResNet-50 with adaptive clustering strategy.

| Indicators | WV | ED | Work | RWMN | OOJ | FSB | OOS |
|---|---|---|---|---|---|---|---|
| Precision | 0.784 | 0.668 | 0.982 | 0.950 | 0.661 | 0.812 | 0.324 |
| Recall | 0.909 | 0.593 | 0.668 | 0.915 | 0.933 | 0.928 | 0.571 |
| F1 | 0.842 | 0.628 | 0.795 | 0.932 | 0.774 | 0.866 | 0.413 |

WV: watch video. ED: eat drink. RWMN: read write magazine newspaper. OOJ: off on jacket. FSB: fasten seat belt. OOS: off on sunglasses.

The experimental results of different backbones in Table 1 show the universality for feature extractors of the proposed method in this paper. The ablation experiment in Table 2 shows that the adaptive clustering strategy has the highest accuracy. From the results of different abnormal action categories shown in Table 3, it can be seen that the performance of the few-sample categories needs to be improved.

### *4.3. Data Compression*

This experiment compares the performance of mainstream video encoders in terms of compression speed, compression ratio, and post-compression quality, aiming to select a suitable video encoder for automated driving evidence storage scenarios. The experiment uses a 22-second high-resolution raw video recorded by a static camera, with a resolution of 7680 × 4320 and a file size of 762 MB. To maintain experimental consistency and fairness, the constant rate factor (CRF) parameter for all encoders is standardized at 30. The results are presented in Table 4.

**Table 4.** Video encoder performance comparison.

| Coder | Size (MB) | Compress Rate | Bit Rate (KPS) | Coding Time (s) | Code Rate (MB/s) | PSNR (dB) | SSIM |
|---|---|---|---|---|---|---|---|
| ORIGION | 762 | 1.00 | 278,171 | / | / | / | / |
| MPEG2 [32] | 89.8 | 8.49 | 32,750 | **70.69** | **10.78** | 33.601 | 0.9573 |
| MPEG4 [33] | 78 | 9.77 | 28,470 | 71.34 | 10.68 | 32.9286 | 0.9441 |
| H.264_MF [44] | 35 | 21.77 | 12,789 | 73.98 | 10.30 | 26.1753 | 0.8605 |
| LIBX265 [34] | 6.99 | 109.01 | 2415 | 968.19 | 0.79 | 45.3921 | 0.9812 |
| AV1_NV | **4.22** | **180.57** | **1533** | 82.90 | 9.19 | 43.3955 | 0.9769 |
| LIBSVTAV1 | 7.27 | 104.81 | 2645 | 121.79 | 6.26 | **45.9398** | **0.9825** |
| H.266 [45] | 67.19 | 11.34 | 100,965 | 1933 | 0.63 | 45.2085 | 0.9801 |

The bold numbers represent the optimal values of the indicators.

Peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) are commonly used to evaluate compressed video quality. The PSNR is defined as the ratio of the maximum value of the signal to the noise power within the reconstructed image or video:

$$MSE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} (X(i,j) - Y(i,j))^2, \quad (14)$$

$$PSNR = 10\log_{10}(\frac{(2^{\alpha}-1)^2}{MSE}), \tag{15}$$

where $H$ and $W$ are the height and width of the images, respectively. $i$ and $j$ represent the row and column of the image, respectively. $X(i,j)$ and $Y(i,j)$ denote the pixel values at the $i$-th row and $j$-th column of the original and compressed images, respectively. $\alpha$ indicates the number of color sampling points of the original image. A PSNR value exceeding 40 dB indicates that the compressed image is nearly indistinguishable from the original.

The SSIM quantitatively evaluates the human visual system's sensitivity to image structure from three aspects: luminance, contrast, and structure.

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + A_1)(2\sigma_{xy} + A_2)}{(\mu_x^2 + \mu_y^2 + A_1)(\sigma_x^2 + \sigma_y^2 + A_2)}, \tag{16}$$

where $x$ and $y$ are the windows of the original image and the compressed image, respectively. $\mu_x$ and $\mu_y$ are the mean values of the images $x$ and $y$, respectively. $\sigma_x^2$ and $\sigma_y^2$ are the variances of the images $x$ and $y$, respectively. $\sigma_{xy}$ is the covariance of the images $x$ and $y$. $A_1$ and $A_2$ are small constants added to avoid division by zero. $SSIM \in [-1,1]$. Once the index approaches 1, the loss of visual fidelity becomes imperceptible to the human eye.

The PSNR index [46] indicates that the distortion range of all encoding algorithms for compressed videos falls within an acceptable level (greater than 30 dB). A similar result can be demonstrated through the SSIM index. MPEG2 exhibits the fastest encoding speed, but its compression quality is not satisfactory. The H.266 standard announced in 2020 has not yet reached its expected performance. This discrepancy can be attributed to the nascent stage of the standard and the lack of comprehensive supporting software toolsets. The ecosystem surrounding H.266, including encoders, decoders, and optimization tools, is still under development. AV1_NV requires high-performance GPUs for computational support; it is well-suited for the high-computational-power hardware environment for autonomous driving vehicles. In combination with the high-compression-ratio requirements of this scenario and the patent royalty incurred in encoder usage, this paper selects AV1_NV as an efficient and economical method for video compression that can achieve a minimum of 99% space savings for the distributed storage system. As shown in Figure 9, the AV1_NV restores the detailed texture of the image with better performance.


**Figure 9.** Visual comparison of various video compression algorithms. The original image is a sample from the public dataset Drive&Act [37].

### *4.4. Blockchain Performance Analysis*

Experiments are based on deploying a consortium chain on the Chainmaker [47] blockchain platform. Smart contracts are developed using the Golang language and deployed in the form of chain codes within the consortium chain. The experimental setup includes the following components: Ubuntu 20.04, Chainmaker 2.3.3, Docker 26.1.0, Go 1.20.1, running on a computer equipped with a 4.20 GHz Intel Core i7-7700 K CPU and 46 GB RAM. The Chainmaker network operates through Docker containers, and the consortium chain is composed of four organizational entities, each consisting of one consensus node and one sync node, using MySql as the database for storing block transactions.

Due to the evidence chain in the dual-chain blockchain architecture handling the majority of data interaction tasks in the system, it is necessary to simulate information evidence and retrieval experiments in a vehicle transportation scenario. This experiment focuses on performance testing for two interfaces: data uploading for evidence and querying data on the chain. The system's high availability and stability are evaluated based on request–response speeds. Within the range of data sizes typically encountered in real-world scenarios, experiments are conducted to initiate data upload and query requests, testing and comparing the efficiency of uploading and querying based on the IPFS data storage method.

Storage and retrieval experiments create three sets of test datasets, ensuring a 100% success rate for writes, called by nodes invoking smart contracts. Each experiment is repeated 10 times to average the results, avoiding the occasional nature of experimental results.

As shown in Figure 10, for data sizes of 0.25 MB, 0.5 MB, 1 MB, 2 MB, and 4 MB, respectively, data are stored in both IPFS and the blockchain, comparing the efficiency of on-chain and off-chain storage. The results indicate that both blockchain and IPFS storage times increase with data size, but IPFS storage efficiency is higher than blockchain storage efficiency, averaging 1.9% higher. Files of sizes 0.25 MB, 0.5 MB, 1 MB, 2 MB, and 4 MB are queried from both IPFS and blockchain storage. Retrieving data from the blockchain and IPFS is positively correlated with data size, but IPFS querying efficiency is higher than blockchain, averaging 46.8% higher.

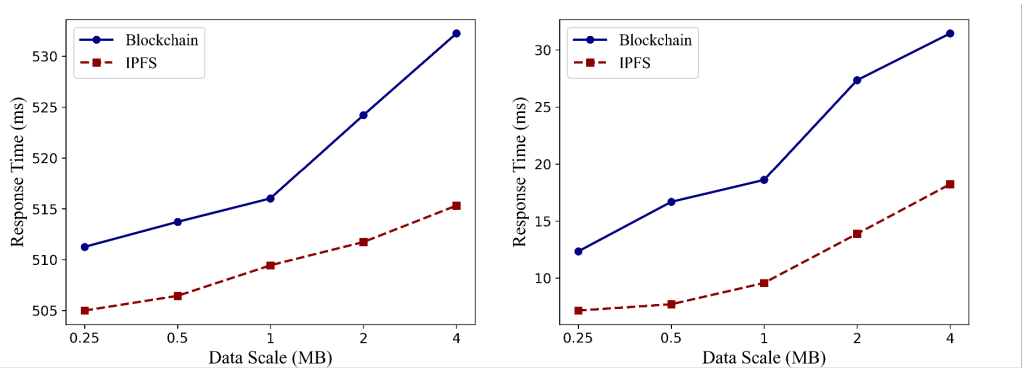


**Figure 10.** Comparison of storage (**left**) and querying (**right**) efficiency at different data scales.

From the experimental results, it is evident that storing off-chain data based on IPFS reduces data transmission overhead, effectively improving model scalability and storage efficiency. However, the proposed method of compressing data before on-chain storage increases decentralization, effectively preventing data tampering. In practical applications, since the size of the compressed data transmitted is limited and does not exceed the maximum data size used in the experiment, the resulting increase in time is within an acceptable range for the entire framework. Thus, compared to directly uploading data to the chain, the trade-off between increased trustworthiness in the evidence system and an acceptable loss in storage efficiency is reasonable.

### *4.5. Limitations and Weaknesses*

The above experimental results prove the feasibility of TSPAD in practical application scenarios. However, the paradigm proposed in this paper still has some limitations and shortcomings. The following part analyzes the three important technologies of TSPAD, respectively, including on-chain evidence storage, abnormal behavior detection, and image compression.

For the on-chain evidence storage module, although the experiment compared the storage and querying efficiency of IPFS and blockchain, it may not have fully considered the variations under different network conditions or in large-scale application scenarios. While the dual-chain architecture (notary chain and private key chain) improves security and privacy, it also introduces complexity, increasing the design and maintenance costs of the system. This complexity limits the generalizability of the approach in practical applications.

In the abnormal behavior detection module, the strategy of adaptive clustering to select key frames meets the requirements for high precision, but the inference speed of the clustering algorithm is slow. Especially for high-frame-rate video streams, the speed drops significantly. In subsequent research, the focus will be on improving the processing speed of the clustering algorithm without losing precision.

The image compression experiments compare between the performance of seven video encoders under controlled variables. For a more comprehensive evaluation of the compression performance of video compression algorithms across different scenarios, future studies could compare the compression ratios and image quality of encoders at varying resolutions and bit rates.

## 5. Conclusions

Currently, there is a lack of a mature and reliable information security mechanism for evidence preservation regarding the liability division of autonomous driving accidents. This deficiency not only compromises the timely recording of data but also undermines the security, transparency, and privacy of information preservation. Even if information is recorded, it is susceptible to unauthorized access and tampering. This paper proposes a TSPAD framework for managing evidence of driver abnormal behavior. In this framework, multimodal data undergo compression algorithms to significantly reduce storage space, while compressed data are certified on the blockchain without relying on local storage. The design of a dual blockchain ensures efficient data access and trustworthy certification, separating key management from certification information storage. The backend can receive vehicle and cargo transport information while ensuring data privacy, balancing information security and reliability.

**Author Contributions:** Conceptualization, T.S., R.W., C.Z., S.Z., Z.M., Z.C., J.H., C.R. and Z.Z.; methodology, T.S., R.W., C.Z., S.Z. and Z.M.; software, T.S., R.W. and C.Z.; validation, T.S., R.W., C.Z. and S.Z.; formal analysis, T.S., R.W. and C.Z.; investigation, T.S., R.W., C.Z., S.Z., Z.M., Z.C. and Z.Z.; resources, J.H., C.R. and Z.Z.; data curation, T.S., R.W., C.Z. and S.Z.; writing—original draft preparation, T.S., R.W., C.Z. and S.Z.; writing—review and editing, T.S., R.W., C.Z., S.Z., Z.M., Z.C., J.H., C.R. and Z.Z.; visualization, T.S., R.W. and C.Z.; supervision, J.H., C.R. and Z.Z.; project administration, Z.M. and Z.Z.; funding acquisition, T.S. and Z.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is supported by Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing (GJJ-24-002) and BUPT Innovation and Entrepreneurship Support Program (2024-YC-A127).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available in the public dataset Drive& Act at https://driveandact.com/#contact (accessed on 20 June 2024).

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References


1. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
2. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
3. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; Volume 30.
4. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
5. Xiao, L.; Luo, C.; Yu, T.; Luo, Y.; Wang, M.; Yu, F.; Li, Y.; Tian, C.; Qiao, J. DeepACEv2: Automated Chromosome Enumeration in Metaphase Cell Images using Deep Convolutional Neural Networks. *IEEE Trans. Med. Imaging* **2020**, *39*, 3920–3932. [CrossRef]
6. Meng, Z.; Fan, Z.; Zhao, Z.; Su, F. ENS-Unet: End-to-End Noise Suppression U-Net for Brain Tumor Segmentation. In Proceedings of the 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Honolulu, HI, USA, 18–21 July 2018; pp. 5886–5889.
7. Caesar, H.; Bankiti, V.; Lang, A.H.; Vora, S.; Liong, V.E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; Beijbom, O. Nuscenes: A Multimodal Dataset for Autonomous Driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 11621–11631.
8. Han, C.; Zhao, Q.; Zhang, S.; Chen, Y.; Zhang, Z.; Yuan, J. Yolopv2: Better, Faster, Stronger for Panoptic Driving Perception. *arXiv* **2022**, arXiv:2208.11434.
9. Campbell, M.; Egerstedt, M.; How, J.P.; Murray, R.M. Autonomous driving in Urban Environments: Approaches, Lessons and Challenges. *Philos. Trans. R. Soc. A* **2010**, *368*, 4649–4672. [CrossRef]
10. Muhammad, K.; Ullah, A.; Lloret, J.; Del Ser, J.; de Albuquerque, V.H.C. Deep Learning for Safe Autonomous Driving: Current Challenges and Future Directions. *IEEE Trans. Intell. Transp.* **2020**, *22*, 4316–4336. [CrossRef]
11. Ionescu, R.T.; Smeureanu, S.; Popescu, M.; Alexe, B. Detecting Abnormal Events in Video Using Narrowed Normality clusters. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 1951–1960.
12. Deepak, K.; Chandrakala, S.; Mohan, C.K. Residual Spatiotemporal Autoencoder for Unsupervised Video Anomaly Detection. *Signal Image Video Process.* **2021**, *15*, 215–222. [CrossRef]
13. Cho, M.; Kim, T.; Kim, W.J.; Cho, S.; Lee, S. Unsupervised Video Anomaly Detection Via Normalizing Flows with Implicit Latent Features. *Pattern Recogn.* **2022**, *129*, 108703. [CrossRef]
14. Cui, X.; Liu, Q.; Gao, M.; Metaxas, D.N. Abnormal Detection Using Interaction Energy Potentials. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 3161–3167.
15. Vu, H. Deep Abnormality Detection in Video Data. In Proceedings of the IJCAI, Melbourne, Australia, 19–25 August 2017; pp. 5217–5218.
16. Tang, H.; Ding, L.; Wu, S.; Ren, B.; Sebe, N.; Rota, P. Deep Unsupervised Key Frame Extraction for Efficient Video Classification. *ACM Trans. Multim. Comput.* **2023**, *19*, 1–17. [CrossRef]
17. Tan, K.; Zhou, Y.; Xia, Q.; Liu, R.; Chen, Y. Large Model Based Sequential Keyframe Extraction for Video Summarization. *arXiv* **2024**, arXiv:2401.04962.
18. Tian, Z.; Li, M.; Qiu, M.; Sun, Y.; Su, S. Block-DEF: A Secure Digital Evidence Framework Using Blockchain. *ISCI* **2019**, *491*, 151–165. [CrossRef]
19. Agrawal, T.K.; Kumar, V.; Pal, R.; Wang, L.; Chen, Y. Blockchain-Based Framework for Supply Chain Traceability: A Case Example of Textile and Clothing Industry. *Comput. Ind. Eng.* **2021**, *154*, 107130. [CrossRef]
20. Cebe, M.; Erdin, E.; Akkaya, K.; Aksu, H.; Uluagac, S. Block4forensic: An Integrated Lightweight Blockchain Framework for Forensics Applications of Connected Vehicles. *IEEE Commun. Mag.* **2018**, *56*, 50–57. [CrossRef]
21. Yao, Q.; Li, T.; Yan, C.; Deng, Z. Accident Responsibility Identification Model for Internet of Vehicles Based on Lightweight Blockchain. *Comput. Intell.* **2023**, *39*, 58–81. [CrossRef]
22. Philip, A.O.; Saravanaguru, R.K. Secure Incident & Evidence Management Framework (SIEMF) for Internet of Vehicles Using Deep Learning and Blockchain. *Open Comput. Sci.* **2020**, *10*, 408–421.
23. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based Learning Applied to Document Recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
24. Bhowmik, D.; Feng, T. The Multimedia Blockchain: A Distributed and Tamper-Proof Media Transaction Framework. In Proceedings of the 2017 22nd International Conference on Digital Signal Processing (DSP), London, UK, 23–25 August 2017; pp. 1–5.
25. Du, W.; Liu, H.; Luo, G.; Zhang, J.; Xu, W. A Consortium Blockchain-Enabled Evidence Sharing System for Public Interest Litigation. *J. Glob. Inf. Manag. (JGIM)* **2023**, *31*, 1–19. [CrossRef]
26. Philip, A.O.; Saravanaguru, R.K. Smart Contract Based Digital Evidence Management Framework over Blockchain for Vehicle Accident Investigation in IoV era. *J. King Saud-Univ.-Comput. Inf. Sci.* **2022**, *34*, 4031–4046. [CrossRef]

27. Philip, A.O.; Saravanaguru, R.K. Multisource Traffic Incident Reporting and Evidence Management in Internet of Vehicles using Machine Learning and Blockchain. *Eng. Appl. Artif. Intel.* **2023**, *117*, 105630. [CrossRef]
28. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
29. Souček, T.; Lokoč, J. Transnet v2: An Effective Deep Network Architecture for Fast Shot Transition Detection. *arXiv* **2020**, arXiv:2008.04838.
30. Turletti, T.H. 261 Software Codec for Videoconferencing over the Internet. Ph.D. Thesis, INRIA, Paris, France, 1993.
31. Aramvith, S.; Sun, M.T. MPEG-1 and MPEG-2 Video Standards. In *Handbook of Image and Video Processing*; 2000; pp. 597–610. Available online: https://preetikale.wordpress.com/wp-content/uploads/2018/07/handbook-of-image-and-video-processing-al-bovik1.pdf (accessed on 20 June 2024).
32. Akiyama, T.; Aono, H.; Aoki, K.; Ler, K.; Wilson, B.; Araki, T.; Morishige, T.; Takeno, H.; Sato, A.; Nakatani, S.; et al. MPEG2 Video Codec using Image Compression DSP. *IEEE Trans. Consum. Electron.* **1994**, *40*, 466–472. [CrossRef]
33. Schwarz, H.; Marpe, D.; Wiegand, T. Overview of The Scalable H. 264/MPEG4-AVC Extension. In Proceedings of the 2006 International Conference on Image Processing, Atlanta, GA, USA, 8–11 October 2006; pp. 161–164.
34. Pastuszak, G.; Abramowski, A. Algorithm and Architecture Design of the H. 265/HEVC Intra Encoder. *IEEE Trans. Circuits Syst. Video Technol.* **2015**, *26*, 210–222. [CrossRef]
35. Islam, A.; Morol, M.K.; Shin, S.Y. A Federated Learning-Based Blockchain-Assisted Anomaly Detection Scheme to Prevent Road Accidents in Internet of Vehicles. In Proceedings of the 2nd International Conference on Computing Advancements, Dhaka, Bangladesh, 10–12 March 2022; pp. 516–521.
36. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning Transferable Visual Models from Natural Language supervision. In Proceedings of the International Conference on Machine Learning. PMLR, Virtual Event, 18–24 July 2021; pp. 8748–8763.
37. Martin, M.; Roitberg, A.; Haurilet, M.; Horne, M.; Reiß, S.; Voit, M.; Stiefelhagen, R. Drive&act: A Multi-Modal Dataset for Fine-Grained Driver Behavior Recognition in Autonomous Vehicles. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 2801–2810.
38. Fox, A.; Brewer, E.A. Harvest, Yield, and Scalable Tolerant Systems. In Proceedings of the Seventh Workshop on Hot Topics in Operating Systems, Rio Rico, AZ, USA, 28–30 March 1999; pp. 174–178.
39. Witzel, C.; Gegenfurtner, K.R. Color perception: Objects, Constancy, and Categories. *Annu. Rev. Vis. Sci.* **2018**, *4*, 475–499. [CrossRef] [PubMed]
40. Khayam, S.A. *The Discrete Cosine Transform (DCT): Theory and Application*; Michigan State University: East Lansing, MI, USA, 2003; Volume 114, p. 31.
41. Karczewicz, M.; Niewegłowski, J.; Haavisto, P. Video Coding Using Motion Compensation with Polynomial Motion Vector Fields. *Signal Process-Image* **1997**, *10*, 63–91.
42. Chen, Y.; Murherjee, D.; Han, J.; Grange, A.; Xu, Y.; Liu, Z.; Parker, S.; Chen, C.; Su, H.; Joshi, U.; et al. An Overview of Core Coding Tools in the AV1 Video Codec. In Proceedings of the 2018 Picture Coding Symposium (PCS), San Francisco, CA, USA, 24–27 June 2018; pp. 41–45.
43. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for Mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.
44. Marpe, D.; Wiegand, T.; Sullivan, G.J. The H. 264/MPEG4 Advanced Video Coding Standard and Its Applications. *IEEE Commun. Mag.* **2006**, *44*, 134–143. [CrossRef]
45. Fu, T.; Zhang, H.; Mu, F.; Chen, H. Fast CU partitioning algorithm for H. 266/VVC intra-frame coding. In Proceedings of the 2019 IEEE International Conference on Multimedia and Expo (ICME), Shanghai, China, 8–12 July 2019; pp. 55–60.
46. Hore, A.; Ziou, D. Image Quality Metrics: PSNR vs. SSIM. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 2366–2369.
47. Chainmaker. Available online: https://chainmaker.org.cn/home (accessed on 28 June 2024).