

Review

Deep Learning for Video Application in Cooperative Vehicle-Infrastructure System: A Comprehensive Survey

Beipo Su, Yongfeng Ju and Liang Dai *

School of Electronics and Control Engineering, Chang'an University, Xi'an 710054, China;
2019032001@chd.edu.cn (B.S.); yfju@chd.edu.cn (Y.J.)

* Correspondence: ldai@chd.edu.cn; Tel.: +86-136-0929-3915

Abstract: Video application is a research hotspot in cooperative vehicle-infrastructure systems (CVIS) which is greatly related to traffic safety and the quality of user experience. Dealing with large datasets of feedback from complex environments is a challenge when using traditional video application approaches. However, the in-depth structure of deep learning has the ability to deal with high-dimensional data sets, which shows better performance in video application problems. Therefore, the research value and significance of video applications over CVIS can be better reflected through deep learning. Firstly, the research status of traditional video application methods and deep learning methods over CVIS were introduced; the existing video application methods based on deep learning were classified according to generative and discriminative deep architecture. Then, we summarized the main methods of deep learning and deep reinforcement learning algorithms for video applications over CVIS, and made a comparative study of their performances. Finally, the challenges and development trends of deep learning in the field were explored and discussed.

Keywords: video application; CVIS; deep learning; generative and discriminative deep architecture; deep reinforcement learning



Citation: Su, B.; Ju, Y.; Dai, L. Deep Learning for Video Application in Cooperative Vehicle-Infrastructure System: A Comprehensive Survey. *Appl. Sci.* **2022**, *12*, 6283. <https://doi.org/10.3390/app12126283>

Academic Editor: Byung-Gyu Kim

Received: 12 May 2022

Accepted: 15 June 2022

Published: 20 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, with the rise of the social vehicle ownership rate, the needs of traffic safety and quality of user experience (QOE) are increasing. Intelligent transportation systems (ITS) are expected to relieve traffic pressure and prevent traffic jams through video target detection, video-assisted driving and wireless resource management. The proposal of CVIS can greatly alleviate the pressure of network overheads in ITS. Vehicles equipped with wireless communication units and sensing units are used as mobile nodes, and their communication modes include vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I) and vehicle-to-roadside (V2R) [1] modes. Video applications over CVIS is one of the important parts of ITS. These were divided into video transmission, video content distribution and video target detection, which can effectively grasp road conditions, improve driving safety and provide users with popular video streams to improve the QOE. However, this attention-worthy problem ensures the accuracy of target detection, the low delay of video transmission and the high quality of video streams in a CVIS with the characteristics of a dynamic network topology, small transmission radius and time-varying channel.

As one of the current popular research fields, deep learning was first proposed by Hinton in 2006 [2]. It shows excellent performance in dealing with large amounts of data and high computational complexity. Through the nonlinear transformation of a multi-layer network structure, shallow features are combined to extract deep abstract features and realize the distributed representation of data. Because of its shallow structure, the traditional algorithm has difficulty obtaining the essential features from high-dimensional datasets in complex traffic environments. Researchers have applied deep learning to solve this problem, because of the advantage of the deep network structure, which can extract

the essential features of high-dimensional datasets. This paper summarized the research of deep learning in the field of video applications over CVIS in recent years, compared and analyzed the existing research methods, and discussed the current challenges and prospects for the future of video applications over CVIS based on deep learning.

This paper summarized the main research methods in the field of video applications over CVIS, and classified them based on the characteristics of their research methods, as shown in Figure 1.

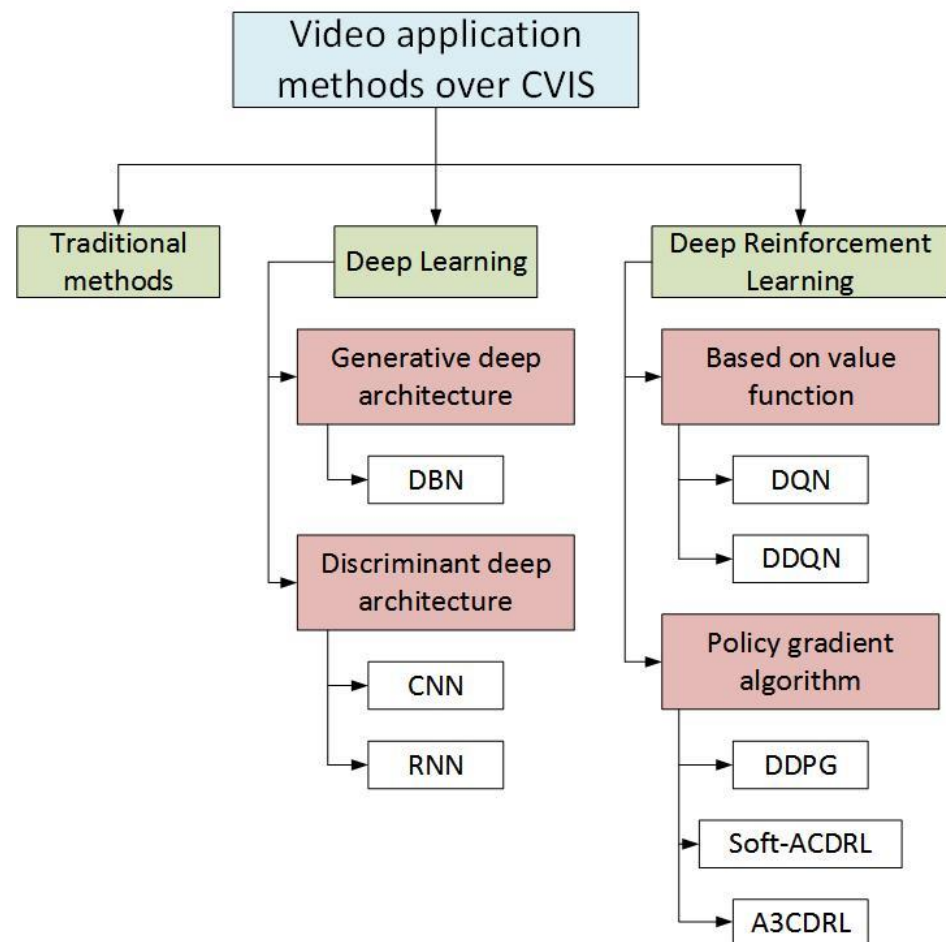


Figure 1. Video application methods over CVIS.

The rest of this paper is organized as follows. Section 2 analyzes the advantages and disadvantages of traditional methods for solving video application problems in vehicle–road coordination systems. Section 3 focuses on addressing the current challenges faced by video applications through deep learning. Section 4 expounds the solution of video application problems in complex CVIS through deep reinforcement learning. Section 5 presents the datasets and model performance evaluation metrics. Section 6 discusses potential future trends in the field.

2. Traditional Video Application Methods over CVIS

Video applications play a very important role in vehicle driving assistance, vehicle safety applications and entertainment applications. Due to the intelligence of the roadside infrastructure in CVIS, it is equipped with information storage units, computing units, etc., which can interact with vehicles in real time, and greatly promotes the development of video applications. In recent years, many researchers have conducted in-depth research on the use of video applications over CVIS and have proposed a considerable number of video transmission methods and video target detection methods.

Modern video coding standards can be divided into the following categories: H.264/AVC, H.264/SVC, H.265/HEVC, AVS2 and AV1 [3–5]. At present, the encoder optimization algorithm based on the H.26X standard is widely studied, and it can reduce the video freeze time and computational complexity under the condition of ensuring video quality in complex CVIS environments. The extensible modulation (s-mod) scheme proposed by Lin et al. [6] ensures the performance while significantly reducing the complexity of the encoder optimization algorithm. Belyaev et al. [7] reduced the computational complexity through unequal packet loss protection and rate control algorithms for scalable video coding. Zhang et al. [8] proposed a video layering scheme based on the H.265/SVC standard to improve the average offload traffic.

Forward error correction (FEC) is widely used in video coding, and can increase the credibility of data communication. Generally speaking, it causes a waste of channel resources when the receiver has no right to request retransmission in unidirectional communication after an error occurs in the transmission data. FEC is an error control mode which encodes according to an algorithm set in advance, before the data transmission, and adds redundant code information with its own characteristics. When an error occurs in the data transmission, the receiver decodes the received data stream according to the corresponding algorithm and determines the resulting error code to correct it. Ali et al. [9] used RaptorQ code as an FEC scheme to reduce data packet loss and improve transmission performance. Sofiane et al. [10] proposed an enhanced adaptive sub-packet forward error correction (EASP-FEC) video stream transmission scheme which aims to improve the quality of video transmissions over CVIS. To minimize video distortion, Zhu et al. [11] proposed an adaptive truncation hybrid automatic repeat request (ATHARQ) algorithm to find hierarchical video packet scheduling for interlayer forward error correction (IL-FEC) coding.

Resources can be divided into network resources and cache resources based on different standards. According to the different forms of resources, the means of resource allocation and scheduling will be different.

The cluster methods allocate video resources to vehicles with the same demand characteristics. Aiming to minimize video interruption probability, Jetendra et al. [12] showed a video resource decision scheme based on a cluster coverage and fast handoff mobile IP system (COMIP). Yaacoub et al. [13] divided mobile vehicles into cooperation clusters to improve the quality of service (QoS) and quality of experience (QOE), and proposed a V2V cooperative communication method.

The main joint DSRC/LTE communication method is that vehicles are allocated a corresponding communication channel according to different needs. Ben et al. [14] proposed a hybrid communication method, based on DSRC/LTE, which improves the overall reliability of communication. Xiaoman et al. [15] proposed a service-aware wireless access technology (RAT) selection algorithm to improve performance.

The multipath transmission method means allocating video sub-resources to different links for transmission. Xie et al. [16] proposed a multipath solution based on a disjoint algorithm to increase the transmission rate and reduce the delay. Aliyu et al. [17] proposed an interference-aware multipath video streaming (I-MVS) framework which aims to reduce packet error rate.

A method to reduce transmission cost, backhaul burden and transmission delay is named video cache transmission. It mainly caches the video to the roadside unit (RSU), and then the RSU transmits the video stream to the user. Guo et al. [18] proposed a dual-time-scale dynamic cache scheme to reduce the backhaul burden and improve video quality. Liu et al. [19] proposed a cache-based cooperative video transmission scheme in cellular networks which improves the QOE. Zhikai et al. [20] proposed an active video content caching (RCC) scheme to minimize transmission delay. Sun et al. [21] designed cache placement and short-term transfer strategies for long-term-aware transmission to improve the QOE.

Video target detection is mentioned to solve such problems as predicting a traffic situation, tracking a trajectory and identifying vehicle types, so as to improve traffic convenience

and driving safety. Ravi et al. [22] collected target parameters through video graphics detection, and predicted vehicle delay, which is defined as the extra time the vehicle spends at an intersection, by using support vector machine (SVM) and artificial neural network (ANN) algorithms. Comparing the two algorithms, it was concluded that ANN is the best-fitting model for estimating the delay of signalized intersections. Olayode et al. [23] obtained datasets from traffic data equipment, such as cameras, at seven road intersections of the most congested roads in South Africa, and developed an ANN model that divides these data sets into thirteen inputs and one output. The results showed that the accuracy of the ANN prediction method is effective.

The traditional video transmission method can show good performance in a simple CVIS environment. However, the algorithm of the traditional method converges slowly, and the model optimization is more difficult, due to more interference and the large data scale in complex environments.

3. Video Application Based on Deep Learning Algorithm

CVIS has the characteristics of a dynamic topology and a time-varying channel state. The traditional video application methods can not accurately analyze and extract the deep features of traffic data. The computational complexity of the traditional method is significantly improved, resulting in dimensional disaster [24] when the dimension of the dataset is high. By taking into account relatively few variables, the traditional methods can not reflect the real traffic characteristics of the CVIS environment. In addition, the traditional method's model is relatively simple, and it is easy to find the local optimal solution. However, deep learning can deal with these problems very well.

Deep learning is an important branch of machine learning, which originates from the study of artificial neural networks and is a further deepening of the basis of traditional neural networks. Shallow networks, such as most classification and regression algorithms, are limited to showing good performance in dealing with complex high-dimensional problems when the sample size is not large enough and the computing power is weak. On the other hand, deep learning mainly completes the complex function approximation through a deep nonlinear network structure, and can extract an abstract distributed representation of its essential features from a small number of samples [25,26]. Hinton et al. [2] proposed a fast learning algorithm based on a deep belief network (DBN) which is mainly divided into two steps: firstly, unsupervised learning is used to pre-train the constrained Boltzmann machine (RBM) network at each layer, and then supervised learning is used to fine-tune the whole DBN. This model reduces the difficulty of deep structure optimization, so the development and application of deep learning are paid increased attention by researchers. Because of the differences in the structure and training methods, deep learning can be divided into the following three categories:

1. Generative deep architecture: this architecture can be described as a model for generating data which belongs to a probability model. Through the joint probability distribution of the observed data and the corresponding categories, the feature set of the generated data contains the high-order correlation features of the input dataset, which is more like an unsupervised learning method;
2. Discriminant deep architecture: this architecture classifies patterns through its own discriminant ability, which estimates a posteriori probability through the conditional probability distribution of the observed data, similar to a kind of supervised learning;
3. Mixed deep architecture: this architecture combines the advantages of generative and discriminative deep architecture, and has excellent expression ability and discriminant ability.

Based on the above three architectures, the advantages of algorithms in the field of video application over CVIS were summarized, respectively. At present, most of the research methods in the field are based on the generative and discriminant models, and the research on the hybrid model is still lacking. In addition, deep reinforcement learning

also promotes the further development of deep learning, so this paper also summarized the research in this field.

3.1. Generative Deep Architecture

DBN is a probability generation model which is obtained by a multi-RBM stack and layer-by-layer greedy training. RBM is a shallow neural network model based on energy. Its structure, shown in Figure 2, consists of a visible layer and a hidden layer. The nodes between layers belong to two-way connections, while the nodes within layers do not produce connections. The visible layer, also known as the input layer, is used to describe data features. The hidden layer will not send and receive signals through the outside; its function is to extract the abstract dependency relationship between data features so that it can be better linearized.

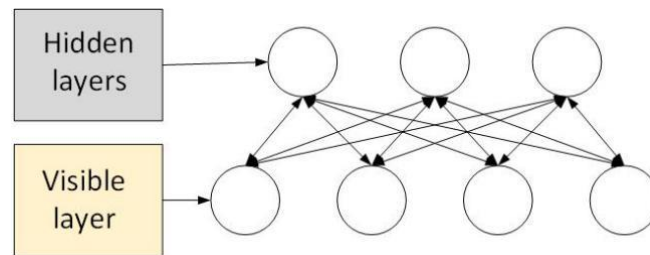


Figure 2. RBM structure schematic diagram.

Because of the structure of multi-layer neural networks, deep neural networks lead to some problems, such as a low learning efficiency, a high dataset sample size and a difficulty with parameter selection. However, DBN can deal with the above problems well by its novel training methods, so deep learning has been widely studied and applied.

The existing routing and caching strategies in vehicular networks cannot effectively cope with time-varying network conditions; thus, Zhang et al. [27] proposed an intelligent routing algorithm based on DBN (IRA) which aims to provide high-quality video streaming services in vehicular networks. Its model structure is shown in Figure 3. In order to discover the deep abstract relationship between the order of requesting video content and the time spent requesting video content, the DBN-based learning model [28] optimizes itself through back propagation regret after layer-by-layer pre-training (regret is defined as the difference between the predicted results and the actual results). Simulation results show that IRA achieves better network performance than other algorithms in terms of lookup data delay and cache hit rate.

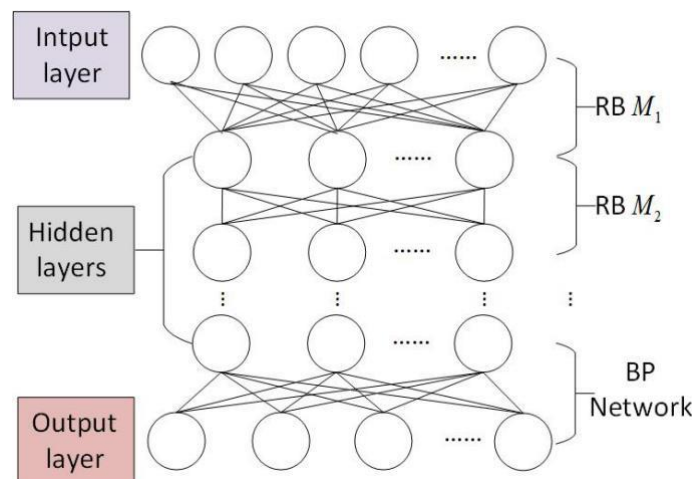


Figure 3. Structure of DBN.

3.2. Discriminant Deep Architecture

The models proposed by researchers based on the discriminant deep architecture include convolutional neural networks (CNNs) and cyclic neural networks (RNNs) in the field of video applications over CVIS.

3.2.1. CNN

The model output of a CNN [29,30] has no feedback connection with itself, so it is a kind of feed-forward deep network. It consists of an input layer, a convolution layer, a pooling layer, a full connection layer and an output layer.

The input layer is used mainly to preprocess the original data, to reduce the impact of differences. As the most important layer of a CNN, the convolutional layer is composed of multiple feature graphs, each of which is composed of multiple neural nodes. The convolution kernel is a weight matrix: the convolution effect of convolution kernels of different sizes will be different when the extraction target is the same. The convolutional layer extracts the specific features of all nodes on the feature graph through the sliding convolution kernel, and the sliding step is expressed as the length of a sliding of the convolution kernel. Each neural node in the convolution layer is locally connected with the feature graph of the corresponding input layer through a set of weights [26]; then, its local weighting sum is mapped by a nonlinear function to obtain the data output of the convolution layer. Importantly, its weights are shared when extracting feature data on the same feature graph. The pooling layer is the output layer of the convolution layer, which is also composed of multiple feature graphs, and each feature graph corresponds to the convolution layer one by one. Average pooling and maximum pooling are common methods in this condition which can reduce the resolution of the feature surface while ensuring that the feature data will not be affected by spatial changes [31], reduce the dimension of the feature data by removing redundant information and prevent over-fitting to a certain extent. The purpose of pooling is to reduce the computational complexity. The connection layer's purpose is to integrate local information with category distinctions, after alternating multiple convolutional and pooling layers [32].

It becomes very difficult to deal with the problem of image distortion when encountering bad weather such as heavy fog and rainy nights. Image quality assessment [33] is mainly used to predict the perceptual quality of digital images, and CNN models are found to be robust to distortion [34]. Varga, Domonkos [35] proposed a depth-based no-reference image-quality assessment architecture that incorporates multiple CNN models, which effectively evaluates image quality by considering multiple image quality scores from different CNN models. This architecture was confirmed in experimental tests.

Chen et al. [36] described a new method, based on deep learning, which removes multiple error sources in sensor signals such as cameras at the same time in the laboratory environment. By correctly identifying the classified signal, the CNN algorithm is developed, which essentially eliminates the sensor error source. In order to test the efficiency of the algorithm, the results are compared with the traditional six-bit static test, the rate test and other methods. An accuracy of 80% is achieved in correctly identifying accelerometer and gyroscope signals.

Kumar et al. [37] introduced a case study of a CNN-based solution using camera video data for real-time open-air off-street intelligent parking management. Experiments were carried out on real-time 24 h data from the input camera video source installed in the parking lot of IIT Hyderabad (IITH). The experimental results show that this scheme can improve parking performance.

Jeon et al. [38] proposed a scene representation method, that is, a multi-channel occupation grid graph (OGM) to describe the whole traffic scene. The deep learning architecture of the OGM is used to predict the future traffic scene. By using this 2D traffic scene representation, the future prediction can be modeled as a video processing problem in which the future time series image series needs to be predicted. In order to predict the future traffic scene according to the past traffic scene, a deep learning architecture using

a CNN and long-term short-term memory network is proposed. In the case of highly conflicting traffic, the future prediction accuracy can be as high as 90%, and the prediction range is 3 s by using the proposed deep learning framework.

Akilan et al. [39] proposed a multi-view perceptual field codec convolutional neural network (MvRF-CNN) model, which contributes mainly to the use of multiple views of convolution kernels with residual feature fusions in the early, middle and later stages of the codec (EnDEC) architecture. The model is still effective in complex traffic environments (dynamic background, camera jitter, night video, bad weather, etc.).

Shobha et al. [40] proposed a deep learning adaptive active network subdivision model for vehicle division. The proposed solution includes three stages: subtraction based on an adaptive background model, active network subnet using a CNN and optimization using an extended topology active network (ETAN) to extract data from the CNN results. Adaptive background modeling is based on the adaptive gain function, which is composed of pixels of frames in the video. The gain function can compensate for the shadow and lighting problems that affect the vehicle division. The deep learning-assisted topology active network deformable model can provide higher partition accuracy in the presence of occlusion, cluttered background and traffic density changes.

Ma et al. [41] proposed an intelligent collaborative visual perception system for networked vehicles. The driving video is collected from the vehicle and transmitted to the cloud for visual perception using a CNN. In order to receive and process multiple vehicle videos synchronously, several data pipeline CNN frameworks are developed; their frame structures are shown in Figure 4. Considering the bandwidth consumption and transmission delay, the resolution adaptive strategy and frame rate control strategy are designed, and IPv6 routing is used to transmit between the vehicle and the cloud. The system helps drivers to make better and safer decisions by enhancing their perception.

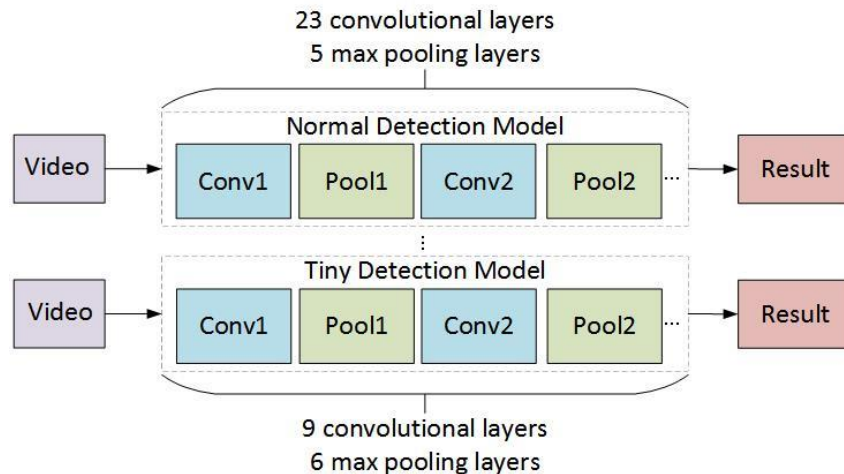


Figure 4. Multi-data pipeline CNN framework.

Guo et al. [42] combined video compression sensing (CVS) with a CNN, and proposed a correlation analysis model in the measurement domain called CVS-CNN. They used CNN instead of the pseudo-inverse transformation of the measurement matrix, and established the correlation between the measured values of the blocks to be estimated and the measured values of adjacent non-overlapping blocks, which can analyze the time correlation of video frames in the measurement domain. The processing speed, accuracy and robustness are obviously better than similar video frame correlation analysis methods.

Jeong et al. [43] proposed an intrusion detection method based on feature generation and CNN for the first time to detect attacks on audio and video transmission protocol (AVTP) streams in a network based on automobile Ethernet. In order to evaluate the intrusion detection system, a physical test platform based on BroadR-Reach and captured real AVTP packets was built.

Madhumitha et al. [44] proposed a new heuristic unimodal method based on a vision system to estimate the collision priority of vehicles on the road. Priority is estimated from the point of view that it may be equipped with a vision-based driver assistance system or the self-driving vehicle itself. Consider crowdsourced videos from YouTube about vehicle collisions captured by vehicle onboard cameras: in order to detect the moving vehicle in the video, the pre-training object detection model and tracking algorithm based on CNN are used.

Wang et al. [45] proposed a robust hierarchical deep learning for promoting congestion detection. In this method, a deep network is designed for hierarchical semantic feature extraction. Different from the traditional deep regression network, which usually uses the mean square error directly as the loss function, robust metric learning is used to train the network effectively. On this basis, multiple networks are combined together to further improve the generalization ability. Extensive experiments have been carried out and it is proved that the proposed model is effective.

A region-based CNN algorithm (RCNN) was proposed by Girshick et al. in 2014 [46]. The author applies a CNN to the field of target detection for the first time. The algorithm is mainly composed of four parts: candidate region generation, feature extraction, category judgment and location refinement. Because of its excellent performance, RCNN is widely used in the field of video applications over CVIS.

Priyadharshini et al. [47] proposed an RCNN algorithm to deal with the aggregation of vehicle data from real-time video streams which improves the detection accuracy by 3.5% compared with a CNN. Kamran et al. [48] prepared datasets containing real data (taken from military performance videos) and toy data (downloaded from YouTube videos). The datasets were divided into three main types: military vehicles, non-military vehicles and other non-vehicle objects. In order to analyze the adequacy of the prepared datasets, we use algorithms such as RCNN target detection to distinguish between military and non-military vehicles. The experimental results show that using the customized/prepared datasets to train the deep architecture can identify seven kinds of military vehicles and four kinds of non-military vehicles.

You Only Look Once (YOLO) was proposed by Joseph et al. in 2015 [49]. YOLO is a target detection algorithm based on CNN, which means that target features can be detected only once. Because of this feature, YOLO is favored by researchers in the field of real-time video target detection. Seal et al. [50] devoted themselves to developing benchmark applications for real-time traffic incident identification and related traffic management using real-time congestion-aware navigation of intelligent vehicles with video feeds, proposing an algorithm framework based on YOLO for identifying and classifying traffic events, such as traffic accidents and congestion and building a fog layer between edge nodes and clouds to make distributed computing (servers) and provide storage closer to edge nodes. Experiments show that the total latency or response time of the YOLO-based fog-cloud platform is lower than that of the pure cloud platform.

Pham et al. [51] proposed a target detection framework based on YOLO to detect and track a large number of highly mobile vehicles, which is also considered as the region of interest (ROI) in vehicle optical camera communication (OCC) systems. The author tested the method on a rainy night on a Korean highway to analyze the effectiveness of this method in the vehicle OCC system. Humberto et al. [52] proposed a real-time statistical algorithm framework for YOLO vehicle classification based on edge AI. Experiments show that the algorithm is helpful to provide timely traffic information.

Song et al. [53] aimed to use real-time traffic data collected by traffic surveillance video and image recognition to explore the relationship between the spatio-temporal pattern of vehicle types and numbers in different urban functional areas and the emission of traffic-related air pollutants. Video-based YOLO detection technology is used to analyze the data, and air pollution is quantified by pollutant emission coefficient.

Sreekumar et al. [54] proposed a deep learning model based on YOLOv2, and applied the model to traffic detection scenarios. In order to reduce the network burden and eliminate

the deployment of the backbone of the network at the intersection, it is recommended that the traffic video data be processed at the edge of the network without transmitting big data back to the cloud. In order to improve the frame rate of edge processing, this paper further proposed a depth object tracking algorithm based on an adaptive multimodal model which is robust to object occlusion and changing lighting conditions.

Huang et al. [55] aimed to develop an efficient system that can meet the needs of video analysis based on dynamic content and expand to large-scale traffic camera video data. The proposed system used a YOLO-based deep learning method to identify objects in video data. This information was then processed and analyzed by the analysis layer implemented using Spark and Hive. Compared with the traditional method, the accuracy of this system can reach more than 80%.

3.2.2. RNN

There is a correlation between the data when the data has sequence characteristics and the accurate feature representation cannot be obtained by analyzing and extracting features from a single data. In order to solve this problem, the RNN was proposed by Pineda et al. [56]. When an RNN is faced with time series data, according to its special network structure, the information of the previous time can be mapped non-linearly to the next time, and the dependency relationship in the data is maintained, which gives RNN the ability to remember and process the previous data information so that it can extract features from the sequence data. For RNN, its weight parameters are shared during a training process.

After follow-up practice, it is found that it is difficult for RNN to deal with long-term and long-distance dependence problems, which often lead to RNN gradient disappearance or gradient explosion [57]. The long-term short-term memory (LSTM) model proposed by Hochreiter et al. [58] can deal with the above problems well. The structure of LSTM is relatively complex, and its memory unit includes three kinds of gating units: the forgetting gate, the input gate and the output gate. The forgetting gate can filter the historical data and discard the information that feels useless in the historical information, the input gate can control the storage of part of the current data into the memory unit and part of the stored information in the output control memory unit is output at the current time. Because of the above features, LSTM can better store useful information and plays an important role in solving the problems of long-term and long-distance dependence. As a further study of LSTM, gated cyclic unit neural networks (GRUs) couple the input gate and the forgetting gate into a gated unit, which means the reduction of parameters and the reduction of computational complexity. Therefore, compared with LSTM, GRU has better training efficiency.

Adita et al. [59] designed a sequence-to-sequence deep learning model called DeepChannel that is based on a codec. It can predict the change of wireless signal strength in the future according to the past signal strength data. This article considers two different versions of DeepChannel; the first and second versions use LSTM and GRU as their basic unit structures, respectively. Different from the previous work of designing models for specific network settings, DeepChannel has strong adaptability and can predict future channel conditions for different networks, sampling rates, mobility modes and communication standards.

Yang et al. [60] proposed a unified driver behavior modeling system for multi-scale behavior recognition. The driver behavior recognition system aims to identify the physical and mental state of the driver simultaneously based on the deep encoder–decoder framework. The model learns to recognize three different time scales of driver behavior: mirror examination and facial expression state, as well as two psychological behaviors, including intention and emotion. The encoder module is based on CNN and is used to capture spatial information from the input video stream. Then, several decoders for different driver state estimations are proposed using RNN based on full connection (FC) and LSTM. The model framework is shown in Figure 5.

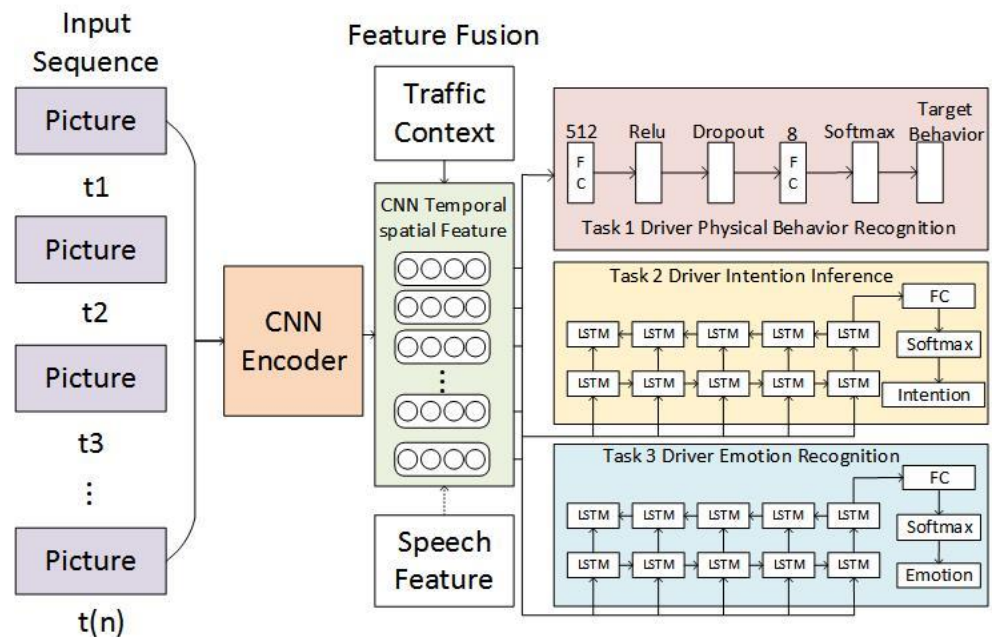


Figure 5. Multi-task learning model framework.

4. Video Application Based on Deep Reinforcement Learning Algorithm

Deep reinforcement learning is a combination of deep learning and reinforcement learning. With deep learning [26], through a gradual, layer-by-layer process, low-level data input is transformed into high-level abstract feature representations through nonlinear transformations, and deep learning pays more attention to the perception and expression of environmental feedback datasets. Reinforcement learning [61] obtains the optimal strategy to maximize the cumulative reward through the interaction between the agent and the environment. It focuses more on the strategies to deal with the problems through autonomous feedback learning. Deep reinforcement learning converts high-dimensional datasets of environmental feedback into abstract state features with the help of deep learning. Then, the optimal strategy is obtained with the help of reinforcement learning autonomous learning, which is used to solve the strategy problem in a complex environment.

In the video application environment over CVIS, the complexity of the environment is influenced by traffic flow, the speed of the vehicle, wireless channel resources, signal strength and so on. The environmental state space is often high-dimensional, and traditional reinforcement learning algorithms have difficulty dealing with high-dimensional state space and continuous action space. With the performance of deep learning in high-dimensional data input processing, the above problems can be easily solved. Therefore, deep reinforcement learning has gradually become the most-covered direction of researchers in the field of video transmission under CVIS. According to the solution strategies based on different methods and numbers of agents, they are divided into the following three categories:

1. Deep reinforcement learning algorithm based on value function: the algorithm mainly evaluates the Q-value generated by all actions, selects the action according to the Q-value and obtains the optimal strategy indirectly through the value function;
2. Policy gradient algorithm: the algorithm parameterizes the strategy, uses the weight parameters of the depth neural network to represent the strategy, optimizes the strategy through the gradient method and constantly modifies the parameters and gradually obtains the optimal strategy. The policy gradient algorithm is an algorithm to solve the optimal policy directly;
3. Multi-agent deep reinforcement learning: multiple agents choose the corresponding actions according to the current environment, establish different reward functions according to the relationship between agents and solve the optimal strategy.

Through the above three ways, this paper divided and classified the video application algorithms over CVIS based on deep reinforcement learning, and its characteristics and advantages were summarized.

4.1. Deep Reinforcement Learning Algorithm Based on Value Function

After the Deep Q-Network (DQN) was first proposed by Mnih et al. [62] in 2013, deep reinforcement learning has attracted increased attention. With the deepening of video transmission research over CVIS, DQN is widely used to solve video business-related problems.

Because of the large dimension of state space, the Q-value of traditional reinforcement learning is difficult to predict. The deep neural network can extract eigenvalues through multiple network layers to reduce the dimension, so DQN approximately represents the Q-value function through the deep convolutional network (DNN). The evaluated Q-value function may be unstable and deviated. The target network and the experience playback mechanism proposed by Mnih et al. [63] solve this issue.

In order to improve the revenue of RSU operators, Ahmed et al. [64] proposed a scenario in which vehicles covered by an RSU cache off-the-shelf content (video streams, etc.) to the RSU, and the RSU then provides services for newly arrived vehicles, which is defined as an MDP. The author defined the environmental state as the vehicle location information, the request content and the available content information, defined the action as whether or not to distribute the content and defined the reward as the available income. The DQN algorithm is used to solve the MDP problem; the structure of the DQN algorithm is shown in Figure 6. First, observe the state space: DNN extracts the abstract representation through the state space, RSU selects the action to execute and the environment gives feedback (the vehicle delivery cost). The DQN model proposed by the author is superior to other methods in terms of total income, cost and service rate.

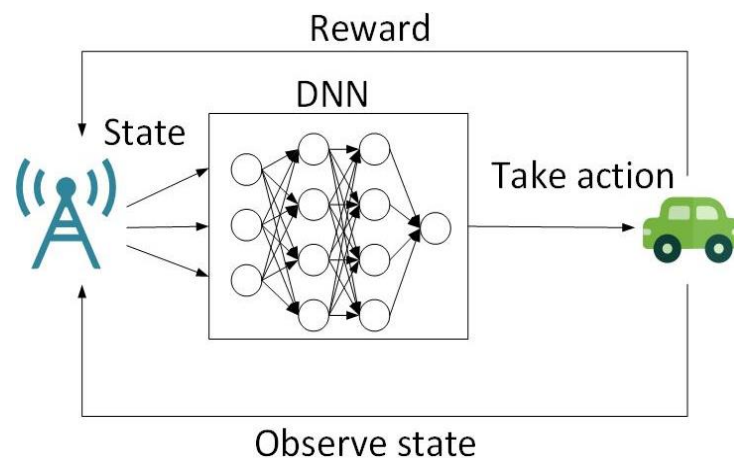


Figure 6. The structure of the DQN algorithm.

Atallah et al. [65] solve the security and QoS problems of green, balanced, linked and efficient vehicle networks. Using the DQN model, the completion request rate is increased by 10–25%, the average request delay is reduced by 10–15% and the total network life is increased by 5–65%.

Nassar et al. [66] developed a network slicing model based on the fog node (FN) cluster coordinated with the edge controller (EC) to solve the resource allocation problem. The author described the problem as an MDP, and the state space is defined as the resource block information, the action is defined as whether the edge controller sends the resource block to the corresponding FN cluster for processing and the reward is defined as the key performance indicator (KPI). Through DQN, self-adaptive learning is the best slicing strategy to maximize KPI, so as to achieve efficient resource allocation.

Pan et al. [67] proposed an asynchronous federated DQN-based and URLLC-aware computation offloading algorithm (ASTEROID) to maximize throughput under long-term

URLLC constraints. In this paper, a URLLC constraint model, based on extreme value theory, is established. Secondly, Lyapunov is used to optimize the decomposition of task unloading and computing resource allocation. Finally, an asynchronous federated DQN (AF-DQN) algorithm is proposed to solve the problem of vehicle-side task unloading. Its state space is defined as the backlog of information in actual and virtual queues, the action is defined as whether the task is unloaded and the reward is defined as the throughput. Simulation results show that ASTEROID achieves excellent throughput and URLLC performance.

Sun et al. [68] proposed a novel slicing framework and optimization solution based on dynamic reinforcement learning for efficient resource allocation in virtual networks with D2D-based V2V communication. The goal is to balance resource utilization and QOS satisfaction across multiple slices. The problem is described as an MDP whose state space is defined as the QOS satisfaction, resource utilization and the number of resource blocks occupied by each slice; the action is defined as the proportion of resources allocated to the slice and the reward is defined as the composition of resource utilization and QOS satisfaction. The optimal strategy is obtained by using the DQN model, and the performance results of the resource utilization, QOS satisfaction and throughput are given.

Tan et al. [69] studied joint communication, caching and computing design issues to achieve excellent operation and cost efficiency for in-vehicle networks. Because of the high complexity of CVIS, the author used the multi-time scale framework [70] to develop deep reinforcement learning to reduce the complexity. The multi-time scale framework consists of two models. The model for each cycle is called the large time scale DQN model, and the model for selecting the action decision for each time slot is called the small time scale DQN model. Numerical results are given to prove its performance gain.

To maximize the long-term sum ratio of all vehicles, Wang et al. [71] proposed a hybrid architecture which consists of centralized decision making and distributed resource sharing (C-Decision scheme). Each vehicle uses a deep neural network to compress its observed information, and then feeds it back to the central decision-making unit. The centralized decision-making unit uses DQN to allocate resources, and then sends the decision results to all vehicles. In order to promote distributed resource sharing, the author proposed a distributed decision and spectrum sharing architecture (D-Decision scheme) for each V2V link. Simulation results show that the proposed C-Decision and D-Decision schemes can achieve near-optimal performance and are robust to feedback interval variation, input noise and feedback noise.

Zhang et al. [72] studied the federated optimization of transmission mode selection and resource allocation for cellular V2X communication. The problem is expressed as an MDP whose state space is defined as the receiving interference power, the resource block neighbor information of the previous subframe, the channel gain, the current load and the remaining time to meet the delay threshold; the actions are defined as the resource block allocation, communication mode selection and transmission power level; the rewards are defined as the sum of the total capacity benefits, the unmet capacity penalties and the effects of reliability and delay requirements. A decentralized algorithm based on DQN is proposed to maximize the total capacity of vehicles to infrastructure users and to meet the delay and reliability requirements of V2V communication.

Hasselt et al. [73] proposed a deep double Q-network on the premise of an in-depth study of the double Q-learning algorithm. The traditional DQN may overestimate the Q-value, but DDQN reduced the overestimation by improving the training algorithm, so as to obtain a more accurate Q-value and optimize the performance of the model. Wang et al. [74] proposed a competitive network structure as a network model of DQN; its structure is shown in Figure 7. The competitive network structure divided the abstract features extracted by DNN into two places, in which the state value function represents the value of the environment itself, and its value does not change according to the action choice, while the action dominance function represents the additional reward value after the execution of the selected action. When the two are combined, the Q-value is generated

for each action. The duel DQN improves the prediction accuracy of the value function by improving the structure of the model.

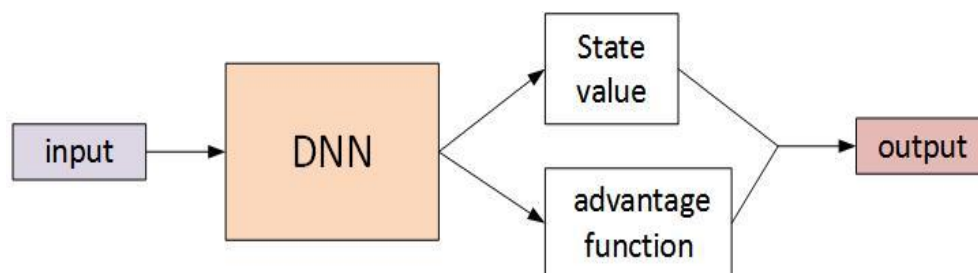


Figure 7. The DQN algorithm structure based on a competitive architecture.

In order to improve the performance of vehicle network, He et al. [75] proposed an integrated framework which can realize the dynamic allocation of network, cache and computing resources. The resource allocation strategy of the framework is described as a joint optimization problem. Because the state space is very complex, the traditional DQN algorithm has difficulty effectively approximating the Q-value function to obtain the optimal strategy. The author combined the deep double Q-network (DDQN) and the duel DQN algorithm to find the optimal strategy to solve the above problems. The effectiveness of the proposed scheme is proved by the simulation results of different system parameters.

Malektaji et al. [76] considered an edge content delivery network with vehicle nodes, and proposed using the DDQN model to solve the problem of content migration. The DQN model defined the state space as the priority information of the content and the delivery status of the content, the actions are defined as migrating, caching and deleting content and the reward is defined as the access delay. This model reduces content access latency by 70% compared with traditional policies.

Aiming to minimize the user cost, Nan et al. [77] studied the user-centered content delivery problem with service delay constraints in vehicle networking. The optimal content delivery strategy problem is modeled as an MDP which defines the state space as service time, vehicle location information, content cache status and signal strength; the action is defined as whether the vehicle obtains content delivery from the RSU or cloud and the cost is defined as the cost paid by the vehicle for content. The DDQN algorithm is proposed to solve the above problems, and can achieve the best performance while minimizing the cost.

4.2. Policy Gradient Algorithm

The complexity of the DQN seeks out the optimal policy by estimating that the Q-value indirectly increases exponentially when the dimension of continuous or discrete action space is large, so some researchers have focused on solving the optimal strategy directly. The stochastic policy gradient algorithm was proposed in 1999, which parameterizes the policy and updates the policy parameters by calculating the policy gradient. The deterministic policy gradient (DPG) algorithm, on the basis of a random policy gradient, was proposed in 2014. However, because all kinds of policy gradient algorithms need a large number of datasets to provide training—otherwise they will fall into the local optimal situation—the problem can be well solved by the combination of an AC algorithm and a policy gradient. The AC algorithm framework is shown in Figure 8. The AC algorithm is based on the idea of time difference (TD). The policy network is an actor who selects the action according to the environment state. The value function network is a critic who evaluates the action through the value function and generates the time difference error signal to update the improved policy network and the value function network.

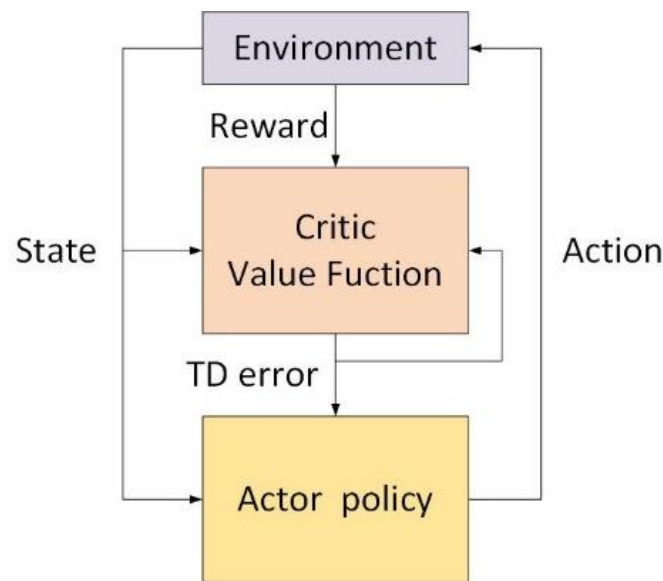


Figure 8. The AC algorithm framework.

Lillicrap et al. [78] combined a deep neural network with DPG for the first time in 2016, and proposed a deep deterministic policy gradient (DDPG) algorithm based on an AC framework, which makes up for the deficiency that DRL is unable to find the optimal strategy when it comes to continuous action spaces.

To minimize the content delivery latency, Dai et al. [79] designed an optimized vehicle edge caching and content delivery strategy using DRL. Edge caching and content delivery problems are described as MDPs; the state space is defined as an adjacent vehicle set, link transmission rate, cached content information and vehicle driving direction; the actions are defined as whether content is cached, the request content ratio and the bandwidth allocation; and the reward is defined as the content delivery delay. The DDPG algorithm is proposed to solve the above problems. Compared with the two benchmark solutions, the numerical results show the effectiveness of the proposed DDPG-based algorithm.

Fang et al. [80] aimed to maximize video quality while reducing delay and rate switching in traffic fog computing (VFC) systems by jointly optimizing vehicle scheduling, rate selection, computing and spectrum resource allocation. Considering the dynamic characteristics of the vehicle network and the available computing/spectrum resources, the problem is modeled as an MDP. Due to the action space of an MDP being a mixture of multi-dimensional continuous variables and discrete variables, this study uses a soft-ACDRL algorithm to solve the MDP problem, and its algorithm structure is shown in Figure 9. Lyu et al. [81] proposed a dual-scale DRL framework based on a soft-ACDRL algorithm to solve the problem of joint communication, computing and cache resource allocation, reducing costs and improving user satisfaction as a result.

The asynchronous advantage ACDRL algorithm (A3C) was proposed by Mnih [82]. Because of its multi-core CPU and multi-threading technology, the A3C algorithm makes multiple agents collect experience samples at the same time, and the correlation between samples is very low. It adopts the training mode of on-policy, and replaces the experience playback mechanism in the ACDRL algorithm by multi-line parallel and asynchronous updating parameters. The researchers have proved that the iterative speed of the algorithm is obviously better than that of the ACDRL algorithm. Jiang et al. [83] proposed a video analysis framework that integrates multi-access edge computing and blockchain technology into IoAV to maximize the throughput of blockchain systems and reduce the latency of MEC systems. The joint optimization problem is modeled as a Markov decision process (MDP) and is solved by the A3C algorithm based on DRL. Khan et al. [84] studied the correlation between vehicles and RSUs in millimeter wave (mmWave) communication networks. Their aim was to maximize the data transmission rate per vehicle user (VUE)

while ensuring that all VUE rates were not lower than the minimum rate and had a low signaling overhead. The association problem is expressed as an MDP, and a low-complexity algorithm based on A3CDRL, which is similar to the solution of the proposed optimization problem, is proposed.

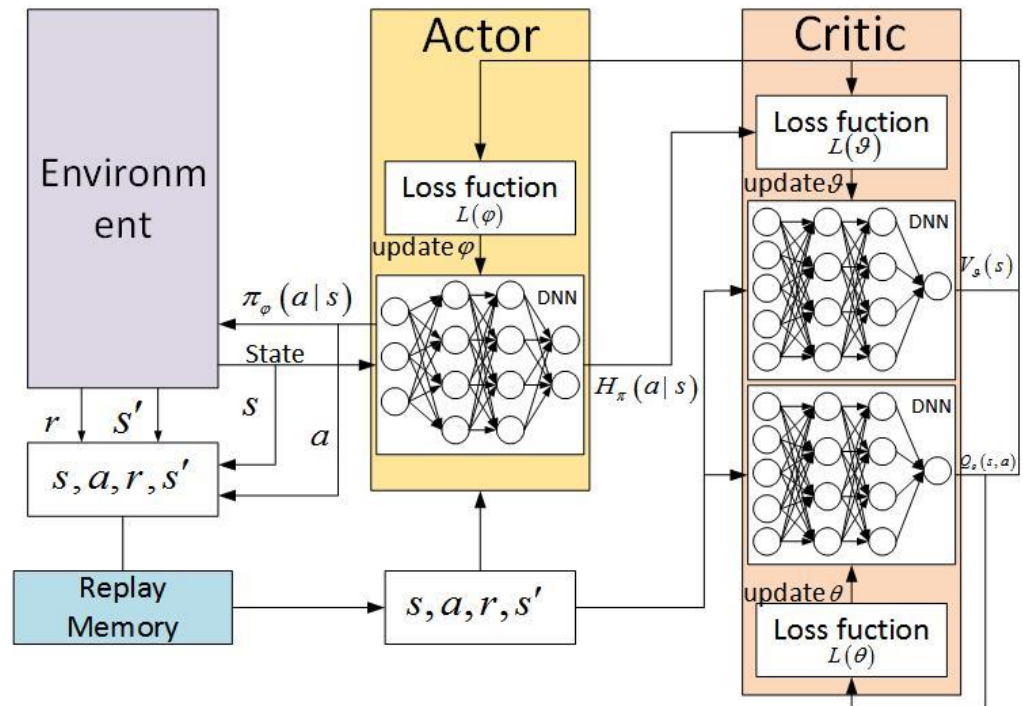


Figure 9. The structure of the soft-ACDRL algorithm.

As researching the DDPG algorithm is becoming an increasingly popular task, a large number of researchers have applied this algorithm to the field of video transmission over CVIS. This paper summarizes the research methods based on DDPG in Table 1.

4.3. Multi-Agent Deep Reinforcement Learning

The DDPG algorithm becomes helpless when faced with more than one agent. Tampuu et al. [85] proposed a DRL model of cooperation and competition among multi-agents, in which each agent is trained by independent DQN algorithm, and rewards are designed according to the goal of each agent. According to the different return functions designed, multi-agent deep reinforcement learning can be divided into three ways: complete cooperation, complete competition and mutual cooperation and competition. When the environment of video transmission over CVIS is so complex that a single agent cannot solve the problems, multi-agent DRL enters the attention of researchers.

Zhu et al. [86] proposed a computing offload scheme, based on multi-agent DRL, which minimizes the total task processing delay in the long run. In order to evaluate the performance of the proposed unloading scheme, a large number of simulations were carried out. Simulation results verify the effectiveness and superiority of the proposed scheme.

Amine et al. [87] studied the problem of service migration in MEC-enabled vehicle networks to minimize the total service delay and migration costs. In order to obtain an effective solution, it is modeled as a multi-agent MDP and solved by the deep Q-learning (DQL) algorithm. Simulation results show that the proposed scheme achieves near-optimal performance.

Table 1. Video application methods over CVIS based on DDPG.

Author	State	Action	Reward	Research Content
Chen [88]	Unload target, computing resource information, available bandwidth and vehicle location information.	Whether to uninstall target and computing resources and available bandwidth allocation.	Task completion utility.	Intelligent allocation of edge computing resources and channel resources.
Dai [89]	Communication rate, energy consumption, driving direction, content delivery delay and storage capacity.	Match cache pair.	System utility.	Implement blockchain-enabling content cache security and privacy protection.
Kwon [90]	Preloaded video size, cache occupancy, average video quality and location information.	Power allocation and cache allocation.	Quality change, packet loss and video interruption.	Millimeter wave-based station power control and active buffer allocation method.
Lan [91]	Computing and caching capabilities.	Calculate the uninstall and service caching policy.	Task processing delay and energy consumption.	Optimization of joint computing offload and service cache.
Yun [92]	Location information, transmission queue status, receiving area status, number of blocks and average quality.	Whether to unload, unload quantity and block quality.	Quality enhancement, avoiding packet loss and video freeze.	Video streaming transmission scheme for mobile-aware vehicle network.
Zhang [93]	Video data transmission status, local view set status and vehicle association status with RSU.	A combination of the view set and memory allocation.	Download cost, data loss and video freeze cost.	Active caching of multi-view 3D video in 5G network.

Kumar et al. [94] used deep learning technology to predict the movement pattern of vehicles. An architecture composed of a centralized decision-making unit and distributed channel allocation is proposed to maximize the spectrum efficiency of all related vehicles. In order to achieve the above goals, the authors used DQN and A2C technology and further integrated LSTM [58] into DQN and A2C technology. The performance of the proposed scheme is verified by simulation.

In order to solve the problem of the abnormal growth of computational complexity and the explosive growth of state space and action space in multitasking and multi-agent (MTMA) environments, Xu et al. [95] used a value decomposition mechanism (VDM) to decompose complex value functions into several small blocks, and then calculated each sub-function separately. The proposed paradigm can greatly accelerate the learning speed and help significantly in a smaller state and action space, but will not degrade the performance. Simulation results showed the effectiveness of the proposed mechanism.

5. Datasets and Model Performance Evaluation Metrics

Most video datasets are collected through realistic means. Generally speaking, the dataset is divided into a training set, a validation set and a test set. Specifically, in the training phase, the model parameters are trained and tuned using the training and validation sets, while the test set is used in the final testing phase. The performance evaluation metrics can more intuitively observe the advantages and disadvantages of the model. According to different optimization standards, it was divided into three types of performance evaluation indicators and listed the performance advantages of some distinct methods in video applications over CVIS based on deep learning. In this section, we summarize existing datasets and performance evaluation metrics for video applications.

5.1. Datasets

The full name of NGSIM is Next Generation Simulation, which is a data collection project initiated by the Federal Highway Administration that includes data on construction, vehicle trajectory prediction, driver intent recognition, automatic driving decision planning, etc. By

applying image processing algorithms to recorded traffic flow videos, Jeon et al. [38] construct a trajectory database of 18 variables related to vehicle state, and test the performance of its proposed deep learning architecture for predicting future traffic scenarios.

The COCO dataset contains 80 different object categories, and this dataset is used mainly for object detection, segmentation, etc. Seal et al. [50] used 2 CNN architectures to classify data: YOLOv3.11 (for 11 classes related to traffic congestion) for traffic object recognition; and YOLOv3.6 (for 6 classes related to traffic events).

Jeong et al. [43] tested its intrusion detection methods on its own carefully created automotive Ethernet intrusion dataset. The dataset was recorded in the format of a PCAP file so that it can be viewed using popular programming libraries and packet analyzers.

Shobha et al. [40] tested the performance of the proposed deep learning-assisted ETAN segmentation on the MIO-TCD dataset. The MIO-TCD dataset is a standard benchmark dataset for vehicle localization and identification. The dataset has about 500,000 images of vehicles taken from US roads at different times of the day. From this dataset, we selected 100 images to test the performance of the proposed work.

Ma et al. [41] used VOC2007, VOC2012 and videos captured in real driving scenarios, which contain 16,551 vehicles, buses, pedestrians, motorcycles and bicycles, to train the model.

5.2. Average Delay

The average delay can effectively reflect whether the video stream can be transmitted successfully in a limited time. The average delay includes mainly the average transmission delay, the average queuing delay, the average processing delay and the average propagation delay. When the video file is large, the article will consider primarily the average transmission delay and the average propagation delay. When the number of files is large, the article will consider the average queuing delay and the average propagation delay. The total latency or response time of the deep learning-based fog cloud computing framework proposed by Seal et al. [50] is reduced by 79.7% compared to cloud platforms. Compared with the greedy power conservation algorithm [96] (GPC) and the greedy priority departure algorithm (GPDV), the DQN-based scheduling algorithm proposed by Atallah et al. [65] reduces the average delay by 10.2% and 21.1%, respectively.

Lan et al. [91] compared the average processing latency of all schemes. In the DDPG and offload-only schemes, as the computing power of the fog server increased, the delay caused by task processing decreased. Our proposed scheme reduces latency by approximately 18.8% compared to the offload-only scheme. This is because the offload-only processing model does not take into account the impact of service caching on network performance.

5.3. QOE

QOE reflects the gap between the actual situation and user expectations. Compared with the unicast-based routing algorithm, the cache hit ratio (CHR) of the DBN-based IRA proposed by Zhang et al. [27] increases by 19%. Compared with the traditional AC algorithm [61] (TACA), the standard policy gradient-based resource allocation algorithm (SPGRA) and the non-learning scheme [97], the soft-ACDRL algorithm proposed by Fang et al. [80] converges faster, and the performance improvement related to video quality is close to 15.2%, 16.5% and 32.5%, respectively.

5.4. Accuracy, Recall, Precision and F-Measure

Accuracy, recall, precision and F-measure are indispensable parts of evaluations of target detection models, vehicle division models and so on. First, we defined the binary prediction categories as follows:

1. True Positive (TP): the positive result is predicted to be positive;
2. True Negative (TN): the negative result is predicted to be negative;
3. False Positive (FP): the negative result is predicted to be positive;
4. False Negative (FN): the negative result is predicted to be negative.

The corresponding operation formula is given as follows:

$$ACC = \frac{TP + TN}{total} \quad (1)$$

$$REC = \frac{TP}{TP + FN} \quad (2)$$

$$PEC = \frac{TP}{TP + FP} \quad (3)$$

$$F = \frac{(\alpha^2 + 1)PEC \cdot REC}{\alpha^2(PEC + REC)} \quad (4)$$

where α is the adjustable parameter.

Accuracy is the most common performance evaluation metric; it can describe the quality of the prediction model. Recall and precision usually need to be considered comprehensively, so a comprehensive evaluation metric F-measure is proposed which is the weighted sum average of the recall and precision, and its value is proportional to the performance of the model. In this paper, the performance and characteristics of the representative methods based on deep learning over CVIS are summarized in Table 2.

Table 2. Performance evaluation and characteristics of representative methods based on deep learning. Let ‘-’ indicate that the data was not available.

Author	Accuracy	Recall	Precision	F-Measure	Method
Chen [36]	0.8	-	-	-	DL.
Kumar [37]	0.98	-	-	0.87	R-CNN
Priyadarshini [47]	0.964	-	-	-	R-CNN
Huang [55]	0.8	-	-	-	YOLO
Jeon [38]	0.9	-	-	-	CNN and LSTM
Akilan [39]	-	-	-	0.95	MvRF-CNN
Ma [41]	-	-	0.7364	-	CNN
Jeong [43]	-	0.9949	-	0.9704	CNN
Seal [50]	-	-	0.515	-	YOLOv3
Kamran [48]	-	-	0.6279	-	R-CNN
Sreekumar [54]	-	-	0.71	-	YOLOv2
Humberto [52]	-	-	0.579	-	YOLOv3

6. Discussion

Up to now, there is little research on solving traditional problems based on deep learning, and the field of video application over CVIS is still in its infancy. Compared with the traditional methods, the video application method based on deep learning has lower average delay, higher QOE and more accurate detection and is more robust to the complex CVIS environment. Therefore, the use of deep learning algorithms to solve video business problems over CVIS is worthy of extensive research. The development trend of this research in the future is discussed as follows:

1. In order to restore the real environment and reflect more real data characteristics when building a model based on deep learning, it is a problem worthy of in-depth study to integrate the traffic characteristics such as the high-speed mobility of vehicles and the short life of channel links;
2. Due to the access environment of workshop communication being open, it will face the risk of malicious attacks in the CVIS environment [98,99]. Therefore, the security of vehicle communication needs to be paid enough attention, and it is necessary to design algorithms based on deep learning to solve network security problems and avoid video content transmission errors, identity authentication failures and network intrusion;

3. When facing large-scale traffic datasets, the model based on deep learning is slightly insufficient in terms of training speed and precision, so how to improve the training speed on the premise of ensuring precision is a problem that needs to be paid attention, so as to improve the training efficiency of the deep learning model;
4. Because of the single training environment, the deep learning algorithm has poor model transfer in the complex and changeable CVIS environment. Therefore, improving the generalization ability of the deep learning model is also the direction of optimizing the deep learning algorithm.

To sum up, firstly, we summarized the current video application methods based on deep learning over CVIS, and then described and analyzed the shortcomings and related problems in the existing research; finally, we discussed the possible development trends of this research in the future. From the simulation results of most studies at present, the deep learning algorithm performs very well compared with the traditional algorithm, which still has the potential for optimization. Therefore, it can be predicted that solving traditional problems through deep learning algorithms will become the mainstream trend in video applications over CVIS.

Author Contributions: Conceptualization, L.D.; methodology, B.S.; validation, L.D, Y.J. and B.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Key Research and Development Program of China (2021YFB2601401), and the Natural Science Foundation of Shaanxi Province, China (2019JQ-264).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used were obtained from public open-source datasets from: 1. NGSIM: <https://data.transportation.gov/Automobiles/Next-Generation-Simulation-NGSIM-Vehicle-Trajectory/8ect-6jqj> (accessed on 26 April 2022); 2. COCO dataset: <http://cocodataset.org>; 3. automotive Ethernet intrusion dataset: <https://doi.org/10.21227/1yr3-q009> (accessed on 26 April 2022); 4. MIO-TCD dataset: <https://www.kaggle.com/yash88600/miotcd-dataset-50000-imagesclassification> (accessed on 26 April 2022); 5. VOC2007: <http://host.robots.ox.ac.uk/pascal/VOC/voc2007/index.html> (accessed on 26 April 2022); and 6. VOC2012: <http://host.robots.ox.ac.uk/pascal/VOC/voc2012/index.html> (accessed on 26 April 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhou, H.; Xu, W.; Chen, J.; Wang, W. Evolutionary V2X Technologies Toward the Internet of Vehicles: Challenges and Opportunities. *Proc. IEEE* **2020**, *108*, 308–323. [\[CrossRef\]](#)
2. Hinton, G.E.; Osindero, S.; Teh, Y.-W. A Fast Learning Algorithm for Deep Belief Nets. *Neural Comput.* **2006**, *18*, 1527–1554. [\[CrossRef\]](#)
3. Jiang, X.; Song, T.; Shi, W.; Katayama, T.; Shimamoto, T.; Wang, L. Fast Coding Unit Size Decision Based on Probabilistic Graphical Model in High Efficiency Video Coding Inter Prediction. *IEICE Trans. Inf. Syst.* **2016**, *99*, 2836–2839. [\[CrossRef\]](#)
4. Jiang, X.; Feng, J.; Song, T.; Katayama, T. Low-Complexity and Hardware-Friendly H.265/HEVC Encoder for Vehicular Ad-Hoc Networks. *Sensors* **2019**, *19*, 1927. [\[CrossRef\]](#)
5. Jiang, X.; Song, T.; Zhu, D.; Katayama, T.; Wang, L. Quality-Oriented Perceptual HEVC Based on the Spatiotemporal Saliency Detection Model. *Entropy* **2019**, *21*, 165. [\[CrossRef\]](#)
6. Cai, L.; Xiang, S.; Luo, Y.; Pan, J. Scalable Modulation for Video Transmission in Wireless Networks. *IEEE Trans. Veh. Technol.* **2011**, *60*, 4314–4323. [\[CrossRef\]](#)
7. Belyaev, E.; Vinel, A.; Surak, A.; Gabbouj, M.; Jonsson, M.; Egiazarian, K. Robust vehicle-to-infrastructure video transmission for road surveillance applications. *IEEE Trans. Veh. Technol.* **2014**, *64*, 2991–3003. [\[CrossRef\]](#)
8. Zhang, X.; Lv, T.; Yang, S. Near-Optimal Layer Placement for Scalable Videos in Cache-Enabled Small-Cell Networks. *IEEE Trans. Veh. Technol.* **2018**, *67*, 9047–9051. [\[CrossRef\]](#)
9. Ali, A.; Kwak, K.S.; Tran, N.H.; Han, Z.; Niyato, D.; Zeshan, F.; Gul, M.T.; Suh, D.Y. RaptorQ-Based Efficient Multimedia Transmission Over Cooperative Cellular Cognitive Radio Networks. *IEEE Trans. Veh. Technol.* **2018**, *67*, 7275–7289. [\[CrossRef\]](#)
10. Zaidi, S.; Bitam, S.; Mellouk, A. Enhanced Adaptive Sub-Packet Forward Error Correction Mechanism for Video Streaming in VANET. In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016; pp. 1–6. [\[CrossRef\]](#)

11. Zhu, C.; Huo, Y.; Zhang, B.; Zhang, R.; El-Hajjar, M.; Hanzo, L. Adaptive-Truncated-HARQ-Aided Layered Video Streaming Relying on Interlayer FEC Coding. *IEEE Trans. Veh. Technol.* **2016**, *65*, 1506–1521. [[CrossRef](#)]
12. Joshi, J.; Jain, K.; Agarwal, Y.; Deka, M.J.; Tuteja, P. COMIP: Cluster based overlay and fast handoff mobile IP system for video streaming in VANET's. In Proceedings of the 2015 IEEE 3rd International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA), Kuala Lumpur, Malaysia, 24–25 November 2015; pp. 1–6. [[CrossRef](#)]
13. Yaacoub, E.; Filali, F.; Abu-Dayya, A. QoE Enhancement of SVC Video Streaming Over Vehicular Networks Using Cooperative LTE/802.11p Communications. *IEEE J. Sel. Top. Signal Processing* **2015**, *9*, 37–49. [[CrossRef](#)]
14. Ben Brahim, M.; Hameed Mir, Z.; Znaidi, W.; Filali, F.; Hamdi, N. QoS-Aware Video Transmission Over Hybrid Wireless Network for Connected Vehicles. *IEEE Access* **2017**, *5*, 8313–8323. [[CrossRef](#)]
15. Shen, X.; Li, J.; Chen, L.; Chen, J.; He, S. Heterogeneous LTE/DSRC Approach to Support Real-time Vehicular Communications. In Proceedings of the 2018 10th International Conference on Advanced Infocomm Technology (ICAIT), Stockholm, Sweden, 12–15 August 2018; pp. 122–127. [[CrossRef](#)]
16. Xie, H.; Boukerche, A.; Loureiro, A.A.F. A Multipath Video Streaming Solution for Vehicular Networks with Link Disjoint and Node-disjoint. *IEEE Trans. Parallel Distrib. Syst.* **2015**, *26*, 3223–3235. [[CrossRef](#)]
17. Aliyu, A.; Abdullah, A.H.; Aslam, N.; Altameem, A.; Radzi, R.Z.; Kharel, R.; Mahmud, M.; Prakash, S.; Joda, U.M. Interference-Aware Multipath Video Streaming in Vehicular Environments. *IEEE Access* **2018**, *6*, 47610–47626. [[CrossRef](#)]
18. Guo, Y.; Yang, Q.; Yu, F.R.; Leung, V.C.M. Cache-Enabled Adaptive Video Streaming Over Vehicular Networks: A Dynamic Approach. *IEEE Trans. Veh. Technol.* **2018**, *67*, 5445–5459. [[CrossRef](#)]
19. Liu, X.; Zhao, N.; Yu, F.R.; Chen, Y.; Tang, J.; Leung, V.C.M. Cooperative Video Transmission Strategies via Caching in Small-Cell Networks. *IEEE Trans. Veh. Technol.* **2018**, *67*, 12204–12217. [[CrossRef](#)]
20. Liu, Z.; Song, H.; Pan, D. Distributed Video Content Caching Policy with Deep Learning Approaches for D2D Communication. *IEEE Trans. Veh. Technol.* **2020**, *69*, 15644–15655. [[CrossRef](#)]
21. Sun, R.; Wang, Y.; Cheng, N.; Lyu, L.; Zhang, S.; Zhou, H.; Shen, X. QoE-Driven Transmission-Aware Cache Placement and Cooperative Beamforming Design in Cloud-RANs. *IEEE Trans. Veh. Technol.* **2020**, *69*, 636–650. [[CrossRef](#)]
22. Sriram, R.T.; Teja, T.; Chikkakrishna, N.K. Application of Machine Learning Techniques in Delay Modelling for Heterogenous Signalized Intersections. In Proceedings of the 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 7–9 October 2020; pp. 618–622. [[CrossRef](#)]
23. Olayode, I.O.; Tartibu, L.K.; Okwu, M.O. Prediction and modeling of traffic flow of human-driven vehicles at a signalized road intersection using artificial neural network model: A South African road transportation system scenario. *Transp. Eng.* **2021**, *6*, 100095. [[CrossRef](#)]
24. Erfani, S.M.; Rajasegarar, S.; Karunasekera, S.; Leckie, C. High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. *Pattern Recognit.* **2016**, *58*, 121–134. [[CrossRef](#)]
25. Bengio, Y.; Delalleau, O. *On the Expressive Power of Deep Architectures*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 18–36. [[CrossRef](#)]
26. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)]
27. Zhang, T.; Chen, X.; Xu, C. Intelligent Routing Algorithm Based on Deep Belief Network for Multimedia Service in Knowledge Centric VANETs. In Proceedings of the 2018 International Conference on Networking and Network Applications (NaNA), Xi'an, China, 12–15 October 2018; pp. 1–6. [[CrossRef](#)]
28. Hao, H.; Xu, C.; Wang, M.; Xie, H.; Liu, Y.; Wu, D.O. Knowledge-centric proactive edge caching over mobile content distribution network. In Proceedings of the IEEE INFOCOM 2018—IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Honolulu, HI, USA, 15–19 April 2018; pp. 450–455. [[CrossRef](#)]
29. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [[CrossRef](#)]
30. Rawat, W.; Wang, Z. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Comput.* **2017**, *29*, 2352–2449. [[CrossRef](#)]
31. Gu, J.; Wang, Z.; Kuen, J.; Ma, L.; Shahroudy, A.; Shuai, B.; Liu, T.; Wang, X.; Wang, G. Recent Advances in Convolutional Neural Networks. *Pattern Recognit.* **2015**, *77*, 354–377. [[CrossRef](#)]
32. Sainath, T.N.; Mohamed, A.; Kingsbury, B.; Ramabhadran, B. Deep convolutional neural networks for LVCSR. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 8614–8618. [[CrossRef](#)]
33. Wang, Z. Applications of Objective Image Quality Assessment Methods. *Signal Process. Mag. IEEE* **2011**, *28*, 137–142. [[CrossRef](#)]
34. Samil, K.; Merve, K.Y.; Kadir, K.; Ferhat, S.R.; Gultekin, B.; Hazım, K.E. How Image Degradations Affect Deep CNN-based Face Recognition. *arXiv* **2016**, arXiv:1608.05246. [[CrossRef](#)]
35. Varga, D. No-Reference Image Quality Assessment with Convolutional Neural Networks and Decision Fusion. *Appl. Sci.* **2022**, *12*, 101. [[CrossRef](#)]
36. Chen, H.; Aggarwal, P.; Taha, T.M.; Chodavarapu, V.P. Improving Inertial Sensor by Reducing Errors using Deep Learning Methodology. In Proceedings of the IEEE National Aerospace and Electronics Conference (NAECON), Dayton, OH, USA, 23–26 July 2018; pp. 197–202. [[CrossRef](#)]

37. Kumar, K.N.; Pawar, D.S.; Mohan, C.K. Open-air Off-street Vehicle Parking Management System Using Deep Neural Networks: A Case Study. In Proceedings of the 2022 14th International Conference on COMMunication Systems & NETWORKS (COMSNETS), Bangalore, India, 4–8 January 2022; pp. 800–805. [\[CrossRef\]](#)
38. Jeon, H.S.; Kum, D.S.; Jeong, W.Y. Traffic Scene Prediction via Deep Learning: Introduction of Multi-Channel Occupancy Grid Map as a Scene Representation. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1496–1501. [\[CrossRef\]](#)
39. Akilan, T.; Wu, Q.M.J.; Zhang, W. Video Foreground Extraction Using Multi-View Receptive Field and Encoder–Decoder DCNN for Traffic and Surveillance Applications. *IEEE Trans. Veh. Technol.* **2019**, *68*, 9478–9493. [\[CrossRef\]](#)
40. Shobha, B.S.; Deepu, R. Deep learning assisted active net segmentation of vehicles for smart traffic management. *Glob. Transit. Proc.* **2021**, *2*, 282–286. [\[CrossRef\]](#)
41. Ma, B.; Xiu, L.; Zhu, K.; Zhang, L. An Intelligent Cooperative Vision Perception System for Connected Vehicles via IPv6. In Proceedings of the 2018 IEEE/CIC International Conference on Communications in China (ICCC), Beijing, China, 16–18 August 2018; pp. 485–489. [\[CrossRef\]](#)
42. Guo, J.; Song, B.; Yu, F.R.; Chi, Y.; Yuen, C. Fast Video Frame Correlation Analysis for Vehicular Networks by Using CVS–CNN. *IEEE Trans. Veh. Technol.* **2019**, *68*, 6286–6292. [\[CrossRef\]](#)
43. Jeong, S.; Jeon, B.; Chung, B.; Kim, H.K. Convolutional neural network-based intrusion detection system for AVTP streams in automotive Ethernet-based networks. *Veh. Commun.* **2021**, *29*, 100338. [\[CrossRef\]](#)
44. Madhumitha, G.; Senthilnathan, R.; Ayaz, K.M.; Vignesh, J.; Madhu, K. Estimation of Collision Priority on Traffic Videos using Deep Learning. In Proceedings of the 2020 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT), Hyderabad, India, 20–21 December 2020; pp. 1–6. [\[CrossRef\]](#)
45. Wang, Q.; Wan, J.; Li, X. Robust Hierarchical Deep Learning for Vehicular Management. *IEEE Trans. Veh. Technol.* **2019**, *68*, 4148–4156. [\[CrossRef\]](#)
46. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [\[CrossRef\]](#)
47. Priyadharshini, P.; Karthikeyan, P. Vehicle Data Aggregation from Highway Video of Madurai City Using Convolution Neural Network. *Procedia Comput. Sci.* **2020**, *171*, 1642–1650. [\[CrossRef\]](#)
48. Kamran, F.; Shahzad, M.; Shafait, F. Automated Military Vehicle Detection from Low-Altitude Aerial Images. In Proceedings of the International Conference on Digital Image Computing—Techniques and Applications (DICTA), Canberra, Australia, 10–13 December 2018; pp. 741–748. [\[CrossRef\]](#)
49. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. *arXiv* **2015**, arXiv:1506.02640. [\[CrossRef\]](#)
50. Seal, A.; Bhattacharya, S.; Mukherjee, A. Fog computing for real-time accident identification and related congestion control. In Proceedings of the 2019 IEEE International Systems Conference (SysCon), Orlando, FL, USA, 8–11 April 2019. [\[CrossRef\]](#)
51. Tung Lam, P.; Huy, N.; Hoan, N.; Van Hoa, N.; Van, B.; Yeong Min, J. Object Detection Framework for High Mobility Vehicles Tracking in Night-time. In Proceedings of the 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), Fukuoka, Japan, 19–21 February 2020; pp. 133–135. [\[CrossRef\]](#)
52. Farias, H.; Solar, M.; Ortiz, D. A Deep Learning Tool to Classify Vehicles in Real Time. In Proceedings of the 2021 8th International Conference on eDemocracy & eGovernment (ICEDEG), Quinto, Ecuador, 28–30 July 2021; pp. 215–218. [\[CrossRef\]](#)
53. Song, J.; Zhao, C.; Lin, T.; Li, X.; Prishchepov, A.V. Spatio-temporal patterns of traffic-related air pollutant emissions in different urban functional zones estimated by real-time video and deep learning technique. *J. Clean. Prod.* **2019**, *238*, 117881. [\[CrossRef\]](#)
54. Sreekumar, U.K.; Devaraj, R.; Li, Q.; Liu, K. Real-Time Traffic Pattern Collection and Analysis Model for Intelligent Traffic Intersection. In Proceedings of the 2018 IEEE International Conference on Edge Computing (EDGE), San Francisco, CA, USA, 2–7 July 2018; pp. 140–143. [\[CrossRef\]](#)
55. Huang, L.; Xu, W.J.; Liu, S.; Pandey, V.; Juri, N.R. Enabling Versatile Analysis of Large Scale Traffic Video Data with Deep Learning and HiveQL. In Proceedings of the IEEE International Conference on Big Data (IEEE Big Data), Boston, MA, USA, 11–14 December 2017; pp. 1153–1162. [\[CrossRef\]](#)
56. Pineda, F.J. Generalization of back-propagation to recurrent neural networks. *Phys. Rev. Letters.* **1987**, *59*, 2229–2232. [\[CrossRef\]](#)
57. Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166. [\[CrossRef\]](#)
58. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [\[CrossRef\]](#)
59. Kulkarni, A.; Seetharam, A.; Ramesh, A.; Herath, J.D. DeepChannel: Wireless Channel Quality Prediction Using Deep Learning. *IEEE Trans. Veh. Technol.* **2020**, *69*, 443–456. [\[CrossRef\]](#)
60. Xing, Y.; Lv, C.; Cao, D.; Velenis, E. Multi-scale driver behavior modeling based on deep spatial-temporal representation for intelligent vehicles. *Transp. Res. Part C Emerg. Technol.* **2021**, *130*, 103288. [\[CrossRef\]](#)
61. Sutton, R.S.; Barto, A.G. Reinforcement Learning: An Introduction. *IEEE Trans. Neural Netw.* **1998**, *9*, 1054. [\[CrossRef\]](#)
62. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; Riedmiller, M. Playing Atari with Deep Reinforcement Learning. *arXiv* **2013**, arXiv:1312.5602. [\[CrossRef\]](#)

63. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Hado, G.; Schaul, T.; Pong, P.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [[CrossRef](#)] [[PubMed](#)]
64. Al-Hilo, A.; Ebrahimi, D.; Sharafeddine, S.; Assi, C. Vehicle-Assisted RSU Caching Using Deep Reinforcement Learning. *IEEE Trans. Emerg. Top. Comput.* **2021**. [[CrossRef](#)]
65. Atallah, R.F.; Assi, C.M.; Khabbaz, M.J. Scheduling the Operation of a Connected Vehicular Network Using Deep Reinforcement Learning. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 1669–1682. [[CrossRef](#)]
66. Nassar, A.; Yilmaz, Y. Deep Reinforcement Learning for Adaptive Network Slicing in 5G for Intelligent Vehicular Systems and Smart Cities. *IEEE Internet Things J.* **2022**, *9*, 222–235. [[CrossRef](#)]
67. Pan, C.; Wang, Z.; Liao, H.; Zhou, Z.; Wang, X.; Tariq, M.; Al-Otaibi, S. Asynchronous Federated Deep Reinforcement Learning-Based URLLC-Aware Computation Offloading in Space-Assisted Vehicular Networks. *IEEE Trans. Intell. Transp. Syst.* **2022**. [[CrossRef](#)]
68. Sun, G.; Boateng, G.O.; Ayepah-Mensah, D.; Liu, G.; Wei, J. Autonomous Resource Slicing for Virtualized Vehicular Networks with D2D Communications Based on Deep Reinforcement Learning. *IEEE Syst. J.* **2020**, *14*, 4694–4705. [[CrossRef](#)]
69. Tan, L.T.; Hu, R.Q. Mobility-Aware Edge Caching and Computing in Vehicle Networks: A Deep Reinforcement Learning. *IEEE Trans. Veh. Technol.* **2018**, *67*, 10190–10203. [[CrossRef](#)]
70. Hyeong Soo, C.; Fard, P.J.; Marcus, S.I.; Shayman, M. Multitime scale Markov decision processes. *IEEE Trans. Autom. Control.* **2003**, *48*, 976–987. [[CrossRef](#)]
71. Wang, L.; Ye, H.; Liang, L.; Li, G.Y. Learn to Compress CSI and Allocate Resources in Vehicular Networks. *IEEE Trans. Commun.* **2020**, *68*, 3640–3653. [[CrossRef](#)]
72. Zhang, X.; Peng, M.; Yan, S.; Sun, Y. Deep-Reinforcement-Learning-Based Mode Selection and Resource Allocation for Cellular V2X Communications. *IEEE Internet Things J.* **2020**, *7*, 6380–6391. [[CrossRef](#)]
73. van Hasselt, H.; Guez, A.; Silver, D. Deep Reinforcement Learning with Double Q-Learning. In Proceedings of the 30th AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; pp. 2094–2100. [[CrossRef](#)]
74. Wang, Z.Y.; Schaul, T.; Hessel, M.; van Hasselt, H.; Lanctot, M.; de Freitas, N. Dueling Network Architectures for Deep Reinforcement Learning. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016. [[CrossRef](#)]
75. He, Y.; Zhao, N.; Yin, H. Integrated Networking, Caching, and Computing for Connected Vehicles: A Deep Reinforcement Learning Approach. *IEEE Trans. Veh. Technol.* **2018**, *67*, 44–55. [[CrossRef](#)]
76. Malektaji, S.; Ebrahimpzadeh, A.; Elbiaze, H.; Glitho, R.H.; Kianpisheh, S. Deep Reinforcement Learning-Based Content Migration for Edge Content Delivery Networks with Vehicular Nodes. *IEEE Trans. Netw. Serv. Manag.* **2021**, *18*, 3415–3431. [[CrossRef](#)]
77. Nan, Z.; Jia, Y.; Ren, Z.; Chen, Z.; Liang, L. Delay-Aware Content Delivery with Deep Reinforcement Learning in Internet of Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2021**. [[CrossRef](#)]
78. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. *arXiv* **2015**, arXiv:1509.02971. [[CrossRef](#)]
79. Dai, Y.; Xu, D.; Lu, Y.; Maharjan, S.; Zhang, Y. Deep Reinforcement Learning for Edge Caching and Content Delivery in Internet of Vehicles. In Proceedings of the 2019 IEEE/CIC International Conference on Communications in China (ICCC), Changchun, China, 11–13 August 2019; pp. 134–139. [[CrossRef](#)]
80. Fu, F.; Kang, Y.; Zhang, Z.; Yu, F.R.; Wu, T. Soft Actor-Critic DRL for Live Transcoding and Streaming in Vehicular Fog-Computing-Enabled IoV. *IEEE Internet Things J.* **2021**, *8*, 1308–1321. [[CrossRef](#)]
81. Lyu, Z.; Wang, Y.; Liu, M.; Chen, Y. Service-Driven Resource Management in Vehicular Networks Based on Deep Reinforcement Learning. In Proceedings of the 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, London, UK, 31 August–3 September 2020; pp. 1–6. [[CrossRef](#)]
82. Mnih, V.; Badia, A.P.; Mirza, M.; Graves, A.; Harley, T.; Lillicrap, T.P. Asynchronous Methods for Deep Reinforcement Learning. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016. [[CrossRef](#)]
83. Jiang, X.; Yu, F.R.; Song, T.; Leung, V.C.M. Intelligent Resource Allocation for Video Analytics in Blockchain-Enabled Internet of Autonomous Vehicles with Edge Computing. *IEEE Internet Things J.* **2020**. [[CrossRef](#)]
84. Khan, H.; Elgabli, A.; Samarakoon, S.; Bennis, M.; Hong, C.S. Reinforcement Learning-Based Vehicle-Cell Association Algorithm for Highly Mobile Millimeter Wave Communication. *IEEE Trans. Cogn. Commun. Netw.* **2019**, *5*, 1073–1085. [[CrossRef](#)]
85. Tampuu, A.; Matiisen, T.; Kodelja, D.; Kuzovkin, I.; Korjus, K.; Aru, J.; Aru, J.; Vicente, R. Multiagent Cooperation and Competition with Deep Reinforcement Learning. *arXiv* **2015**, arXiv:1511.08779. [[CrossRef](#)]
86. Zhu, X.; Luo, Y.; Liu, A.; Bhuiyan, M.Z.A.; Zhang, S. Multiagent Deep Reinforcement Learning for Vehicular Computation Offloading in IoT. *IEEE Internet Things J.* **2021**, *8*, 9763–9773. [[CrossRef](#)]
87. Abouaomar, A.; Mlika, Z.; Filali, A.; Cherkaoui, S.; Kobbane, A. A Deep Reinforcement Learning Approach for Service Migration in MEC-enabled Vehicular Networks. In Proceedings of the 2021 IEEE 46th Conference on Local Computer Networks (LCN), Edmonton, AB, Canada, 4–7 October 2021; pp. 273–280. [[CrossRef](#)]
88. Chen, Y.; Wu, H. Resource Allocation for Edge Collaboration with Deep Deterministic Policy Gradient in Smart Railway. In Proceedings of the 2021 IEEE 4th International Conference on Electronics Technology (ICET), Chengdu, China, 7–10 May 2021; pp. 1163–1167. [[CrossRef](#)]

89. Dai, Y.; Xu, D.; Zhang, K.; Maharjan, S.; Zhang, Y. Deep Reinforcement Learning and Permissioned Blockchain for Content Caching in Vehicular Edge Computing and Networks. *IEEE Trans. Veh. Technol.* **2020**, *69*, 4312–4324. [[CrossRef](#)]
90. Kwon, D.; Kim, J.; Mohaisen, D.A.; Lee, W. Self-adaptive power control with deep reinforcement learning for millimeter-wave Internet-of-vehicles video caching. *J. Commun. Netw.* **2020**, *22*, 326–337. [[CrossRef](#)]
91. Lan, D.; Taherkordi, A.; Eliassen, F.; Liu, L. Deep Reinforcement Learning for Computation Offloading and Caching in Fog-Based Vehicular Networks. In Proceedings of the 2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Delhi, India, 10–13 December 2020; pp. 622–630. [[CrossRef](#)]
92. Yun, W.J.; Kwon, D.; Choi, M.; Kim, J.; Caire, G.; Molisch, A.F. Quality-Aware Deep Reinforcement Learning for Streaming in Infrastructure-Assisted Connected Vehicles. *IEEE Trans. Veh. Technol.* **2021**. [[CrossRef](#)]
93. Zhang, Z.; Yang, Y.; Hua, M.; Li, C.; Huang, Y.; Yang, L. Proactive Caching for Vehicular Multi-View 3D Video Streaming via Deep Reinforcement Learning. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 2693–2706. [[CrossRef](#)]
94. Kumar, A.S.; Zhao, L.; Fernando, X. Multi-Agent Deep Reinforcement Learning-Empowered Channel Allocation in Vehicular Networks. *IEEE Trans. Veh. Technol.* **2022**, *71*, 1726–1736. [[CrossRef](#)]
95. Xu, S.; Guo, C.; Hu, R.Q.; Qian, Y. Value Decomposition based Multi-Task Multi-Agent Deep Reinforcement Learning in Vehicular Networks. In Proceedings of the GLOBECOM 2020—2020 IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; pp. 1–6. [[CrossRef](#)]
96. Hammad, A.A.; Todd, T.D.; Karakostas, G.; Zhao, D. Downlink Traffic Scheduling in Green Vehicular Roadside Infrastructure. *IEEE Trans. Veh. Technol.* **2013**, *62*, 1289–1302. [[CrossRef](#)]
97. Wei, Y.; Yu, F.R.; Song, M.; Han, Z. User Scheduling and Resource Allocation in HetNets with Hybrid Energy Supply: An Actor-Critic Reinforcement Learning Approach. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 680–692. [[CrossRef](#)]
98. Lu, R.; Zhang, L.; Ni, J.; Fang, Y. 5G Vehicle-to-Everything Services: Gearing Up for Security and Privacy. *Proc. IEEE* **2020**, *108*, 373–389. [[CrossRef](#)]
99. Tang, F.; Kawamoto, Y.; Kato, N.; Liu, J. Future Intelligent and Secure Vehicular Network Toward 6G: Machine-Learning Approaches. *Proc. IEEE* **2020**, *108*, 292–307. [[CrossRef](#)]