INVITED PAPER

Edge Video Analytics for Public Safety: A Review

This article provides a survey on applications, algorithms, and platforms that have been proposed to facilitate edge video analytics for public safety.

By Qingyang Zhang, *Student Member IEEE*, Hui Sun, Xiaopei Wu, and Hong Zhong

ABSTRACT | With the installation of enormous public safety and transportation infrastructure cameras, video analytics has come to play an essential part in public safety. Typically, video analytics is to collectively leverage the advanced computer vision (CV) and artificial intelligence (AI) to solve the four-W problem. That is to identify Who has done something (What) at a specific place (Where) at some time (When). According to the difference of latency requirements, video analytics can be applied to postevent retrospective analysis, such as archive management, search, forensic investigation and real-time live video stream analysis, such as situation awareness, alerting, and interested object (criminal suspect/missing vehicle) detection. The latter is characterized as having higher requirements on hardware resources as the sophisticated image processing algorithms under the hood. However, analyzing large-scale live video streams on the Cloud is impractical as the edge solution that conducts the video analytics on (or close to) the camera provides a silvering light. Analyzing live video streams on the edge is not trivial due to the constrained hardware resources on edge. The AI-dominated video analytics requires higher bandwidth, consumes considerable CPU/GPU resources for processing, and demands larger memory for caching. In this paper, we review the applications, algorithms, and solutions that have been proposed recently to facilitate edge video analytics for public safety.

KEYWORDS | Edge computing; public safety; video analytics

I. INTRODUCTION

Video surveillance and video analytics have been substantially growing from practical needs in the past decade, being driven by a wide range of applications in public safety [1], [2], e.g., identifying crimes in a city-wide video surveillance system or monitoring firefighting safety in fireground control centers by sharing the firefighting's video view. In the past, most video surveillance systems transmitted the video to a public or private cloud for video analysis. However, as the scale of a video surveillance system increases, a huge number of video data transmission and computation-intensive video analysis bring an overwhelmed burden for cloud-based solutions on computing and network infrastructures [3]. Moreover, for mobile cameras used in fire departments and emergency medical services (EMS), the video data are not always reliable due to the unstable network connection. They also cannot reach the cloud-based video analytics services in time and they are only able to undertake lightweight video analysis on board and have a postanalysis in the cloud. Currently, with large-scale video surveillance, more and more intelligent cameras are used in public safety systems [4]. It enables part of the video analytics workload, e.g., face detection, to execute on the edge devices, reduces the computational burden of the cloud, and saves a large volume of video data transmission to the cloud, which coincides with edge video analytics. However, current intelligent cameras can only analyze videos with built-in fixed algorithms [5].

Inspired by emerging edge computing [6], [7] (also known as fog computing [8], mobile edge computing [9], or cloudlet [10]), edge video analytics refers

Manuscript received February 8, 2019; revised April 7, 2019; accepted May 8, 2019. Date of publication July 30, 2019; date of current version August 5, 2019. The work of Q. Zhang, H. Sun, and H. Zhong was supported in part by the National Natural Science Foundation of China under Grant 61572001, Grant 61702004, and Grant 61872001, in part by the Key Technology R&D Program of Anhui Province under Grant 1704d0802193, and in part by the Natural Science Foundation of Anhui Province under Grant 1708085QF160. The work of X. Wu was supported in part by the National Science Foundation (NSF) under Grant CNS1741635. *(Corresponding author: Hui Sun.)*

Q. Zhang, **H. Sun**, and **H. Zhong** are with the School of Computer Science and Technology, Anhui University, Hefei 230039, China (e-mail: qingyang@thecarlab.org; sunhui@ahu.edu.cn; zhongh@ahu.edu.cn).

X. Wu is with the Department of Computer Science, Wayne State University, Detroit, MI 48202 USA (e-mail: xiaopei.wu@wayne.edu).

Digital Object Identifier 10.1109/JPROC.2019.2925910

to performing part or all of the video analytics workload on the edge devices, i.e., cameras and edge servers. The noteworthy advantages of edge video analytics for public safety (EVAPS) include: 1) low data transmission overhead; 2) low response latency; and 3) enabling various unprecedented applications achievable. In particular, for mobile cameras carried by police, unmanned aerial vehicles (UAVs), and firefighting robots, the edge video analytics can avoid a large number of data transmission by mobile devices in an unstable network on connection and bandwidth [11], [12]. Taking edge video analytics on body-worn cameras as an example, it enables surrounding objects and incidents to be captured and reported automatically without high latency on data transmission and processing on the cloud, which can significantly improve public safety as well as the police officers' safety [13]. However, how to optimally and dynamically offload workloads in an edge-cloud environment with real-time video analysis is still an insoluble problem in both industry and academia.

In this paper, we review the applications, video analytics algorithms, and platforms, which have been proposed or deployed recently to facilitate EVAPS. First, we review a variety of public safety applications, applied in four departments, including the police department, transportation department, fire department, and EMS, and explain how they work or what their positive influences are on public safety, e.g., reducing the crime rate. However, as far as we know, the EVAPS applications are unevenly developed. At present, the most popular and successful public safety application is crime identification [4], [14], by recognizing faces in city-wide video surveillance systems. For EMS, edge video analytics is only used in a few scenarios, e.g., patient monitoring. Second, to understand how a video is analyzed and which part of the workload could be offloaded to the edge, we dissect the general video analytics process as well as review typical video analytics algorithms used in these special applications in the field of public safety. Face recognition for police departments, vehicle recognition for transportation departments, and flame detection for fire departments are examined. At last, we review proposed platforms or systems for EVAPS, parts of which have partially supported workload offloading and could play a role of reference for other EVAPS frameworks, platforms, systems, and applications.

The remainder of this paper is organized as follows. We first introduce video analytics for public safety and edge computing, followed by a discussion about how EVAPS benefits from edge video analytics in Section II. The public safety applications and related video analytics technique are reviewed in Sections III and IV, respectively. We review recent video analytics platforms in Section V, which can be used for the improvement of public safety, in terms of cloud-based and emerging edge-based systems. For the development of EVAPS, we discuss and present several points as future works in Section VI. Finally, we conclude this paper in Section VII.

II. EDGE VIDEO ANALYTICS FOR PUBLIC SAFETY

Typically, video analytics is to collectively leverage the advanced computer vision (CV) and artificial intelligence (AI) to solve the four-W problem. That is to identify Who has done something (What) at a specific place (Where) at some time (When) [1]. Therefore, video analytics techniques are widely adopted in public safety-related departments, including the police department [4], [13], transportation department [15], [16], fire department [17], and EMS [18]. Taking face detection and recognition technique as an example, it enables identification of crimes in a video, and this could serve as a major boon to transform many public safety applications. The primary application is that of identifying a crime in city-wide video surveillance systems, used by the police department. Also, the fire department can leverage face detection to find survivals in video captured by infrared (IR) thermal cameras in fireground filled with smoke.

However, the sources of public safety video data are exploding and on the move, including video surveillance cameras, body-worn cameras, dash cameras, UAVs, robot cameras, and so on. It leads to a high burden for centralized cloud-based solutions with high requirements on computing capability, storage capability, and network bandwidth. Moreover, as the development of hardware, more and more cameras are promising to equip with a powerful computing unit, e.g., GPU, field-programmable gate array (FPGA), and even AI chip [19], [20]. In this case, the concept of edge video analytics is proposed to transform public safety applications.

The emerging edge computing refers to "the enabling technologies allowing computation to be performed at the edge of the network, on downstream data on behalf of cloud services and upstream data on behalf of Internet of Everything service" [6]. Note that the edges are complements of the existing cloud-computing model. It is promising for latency-sensitive applications by leveraging resources at the proximal edge of data sources instead of in the remote cloud, which saves considerable time on data transmission. Inspired by edge computing, edge video analytics allows video analysis to be performed at the edge close to the cameras. As shown in Fig. 1, the video analytics workload is distributed on the path of cameras to the cloud.

Instead of traditionally transmitting video data to the cloud, edge video analytics enables edge devices to perform various detection algorithms for different public safety applications, e.g., face detection for the police department and vehicle detection for the transportation department. In addition, the results of object tracking algorithms can be used to control cameras directly or blurring algorithms protect the privacy of citizens. Finally, valuable information in videos are extracted and sent to the cloud for further analysis, e.g., depicting a person's trajectory through multiple cameras. Here, the extraction

Table 1 Typical Applications/Events Benefited From/Based on Video Analytics Techniques

Department	Name	Description	Used typical algorithms
Police	Skynet Project in China [4]	In 2018, a BBC reporter was "catched" in seven minutes, while over 170 million cameras have been installed in China.	Motion detection, face recognition, person re-identification, etc.
	Surat in India [14]	It alerts police in real time when suspicious criminals or activities are detected in the surveillance, resulting in 27% crime rate reduction.	Face detection, activity recognition, etc.
	AMBER Alert Assistant [21]	An automatic solution for finding missing children in AMBER alert by recognizing kidnapper's license plate (LP) number in city-wide cameras.	Motion detection, LP detection, LP number recognition.
	Arroy *et al.* [22]	It detects suspicious behaviors in real time in shopping malls.	Behavior recognition, etc.
Transportation	Automatic License Plate Recognition (ALPR) System	ALPR helps identify suspect in Indiana incident involving vehicle that nearly strikes police chief [23].	Vehicle detection, LP detection, LP number recognition, etc.
		ALPR leads to recovery of stolen car, arrest in Louisiana [24].	
	Ki *et al.* [25]	The system can understand behaviors and detect conflicts and accidents to control traffic signals and reduce traffic congestion.	Vehicle detection, traffic flow analysis.
Fire	Yuan *et al.* [26], [27]	Video analytics techniques are applied to UAVs to monitor forest fire.	Motion detection, flame detection.
	Ma *et al.* [28]	UAVs are used to detect early forest fire by detecting smoke.	Motion/smoke/flame detection.
	Thermite [29]	A robot is small enough to enter fireground but strong enough to carry a fire suppression system and a robotic arm for downed victim extraction.	Smoke detection, flame detection, object recognition.
EMS	ETHAN [18]	Cameras are installed on ambulances for telemedicine.	Video encoding and transmission.
	STREMS [30]	A system for pre-hospital aims to improve the quality of EMS using wearable cameras and video-based telemedicine technologies.	Video encoding and transmission.
	Rougier *et al.* [31]	A surveillance system detects fallen people for emergency care.	Motion detection, fall detection.

process effectively reduces the data transmission to the cloud and therefore the transmission cost, in terms of traffic, bandwidth, and payment. A simple but typical public safety application, promising to be benefited by edge video analytics, is that of cropping out and transmitting the interested area of a video, e.g., injury, instead of the full view of video, which significantly improves the quality of service (QoS) of video transmission, especially in a moving ambulance with an unstable network.

In addition, in some countries, e.g., China, the law enforcement requires the video data to be stored for several months. Edge video analytics can also reduce the cost of these parts. In most cases, since the video analytics have been performed, it can avoid the live video data transmission unless the officer requires this feature for activating. It means the video data can be compressed by various techniques, resulting in a low data transmission cost and storage cost for the cloud.

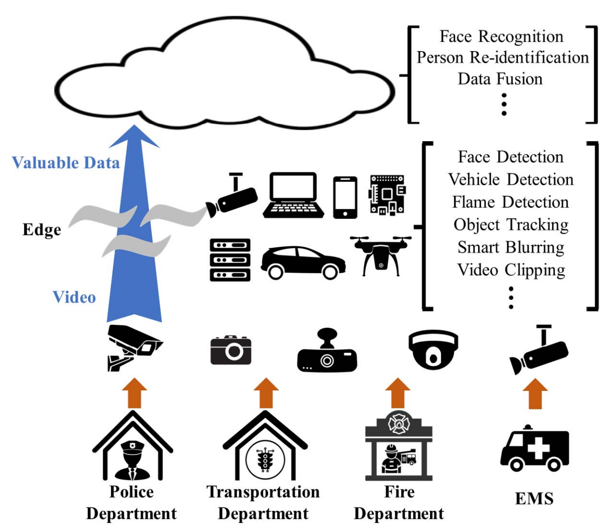


Fig. 1. *Overview of EVAPS.*

III. APPLICATIONS IN EVAPS

In this section, we review a few typical applications of public safety around the world, used in the police department, transportation department, fire department, and EMS. The typical applications or events benefited from video analytics techniques are listed in Table 1.

A. Police Applications

The biggest application domain of video analytics in the police department is intelligent video surveillance, which utilizes various video analytics techniques to recognize humans, vehicles, and even criminal activities, e.g., a human with a criminal history or shooting event. In this section, we introduce several typical applications in the police department, which have improved public safety.

Due to the capability of improving public safety, an increasing number of video surveillance systems are being generated to protect people's safety around the world [32], and a huge number of video data are processed by video analytics techniques. According to the British Broadcasting Corporation (BBC) report [4], a British reporter was identified in just 7 min by a camera in China after Chinese police officers added his face photograph to the database; China was reported to have a huge surveillance network of over 170 million cameras with 400 million new cameras expected to be installed in the next three years. The key technique here is video-based face recognition.

Similarly, the Singapore government built a Safe City Test Bed, utilized since 2013, in which video surveillance and video analytics techniques are widely employed, aiming to improve the city safety [33]. Moreover, a face recognition technique called NeoFace Watch [34] is deployed in Surat, India, which enables the police to be alerted in real time regarding suspicious criminals or activity detected in the surveillance zone. As a result, the crime rate has dropped 27%, and 150 cases were solved after the system had been deployed [14]. In addition, an application that enables a large-variety of rarely occurring activities to be detected is also important for public safety, such as counterterrorism and crowd monitoring.

Moreover, the video surveillance systems are not only used to find a criminal person but also help the citizen, e.g., finding missing children by recognizing the missing children in city-wide video surveillance. Zhang *et al.* [21] have also proposed a kidnapper tracking application to enhance the America's Missing: Broadcasting Emergency Response (AMBER) alert system by tracking the kidnapper's vehicle. Furthermore, many other applications have been proposed about video analytics in a video surveillance system. Arroyo *et al.* [22] proposed a complete application of the real-time detection of potentially suspicious behaviors in shopping malls.

Along with video surveillance, there exist mobile camera-based public safety applications, e.g., body-worn cameras and UAV-based cameras, which are widely used in the police department. For example, Motlagh *et al.* [35] proposed an application that leverages UAVs to obtain facial images, quickly. Moreover, Wang *et al.* [36] also mentioned an application that leverages UAVs to detect survivors in the ocean after a shipwreck. Furthermore, Zhang *et al.* [11] envisioned an application where the surrounding objects and incident can be captured and reported automatically by body-worn cameras. Also, the police can be alerted in a real-time manner when he or she encounters a dangerous person with a criminal history or when a shooting or fire event happens.

B. Transportation Applications

Video surveillance with vision-driven techniques [37] is widely deployed in our cities as intelligent transportation systems (ITS) [15] to detect and track vehicles passing through controlled areas to detect anomalous public safety events, e.g., congestion, speeding violations, illegal driving behaviors, and so on. In this paper, we mainly present three vision-based techniques in ITS, i.e., automatic license plate recognition (ALPR), traffic analysis, and in-vehicle behavior recognition.

The ALPR system is one of the essential components in ITS to analyze and track the vehicles in cities [38], [39]. In law enforcement, ALPR systems are widely adopted by agencies throughout the nation to enhance their enforcement and investigative capabilities, expand their collection of relevant data, and expedite the tedious and time-consuming process of comparing vehicle LPs with lists of stolen, wanted, and other vehicles of interest. As reported in [40], the ALPR alert in California provided the investigative lead in a Kansas murder case [41], and ALPR detection led to the recovery of a stolen car in Louisiana [23]. Moreover, ALPR helped identify the suspect in an Indiana incident involving a vehicle that nearly strikes a police chief [24]. In addition, ALPR systems can be employed in the security control of restricted areas, highway electronic toll collection, red light violation enforcement, parking management systems, and so on [16].

Traffic analysis is mostly used to obtain traffic flow at the intersection, which is one of the high portion incident sites where vehicles and pedestrians interact in a variety of behaviors [42]. Ki and Lee [25] designed a system that can understand participant behaviors and detect conflicts and accidents as well as control traffic signals based on analyzed traffic parameters to improve mobility and transportation safety and reduce traffic congestion in ITS. In addition, anomaly behaviors also impact public safety, and video-based techniques are employing CV approaches to analyze and identify the vehicles' motion pattern and find anomaly behavior, e.g., parking violations, illegal turns, illegal lane change on the highway, violation of traffic line, and so on [43].

In addition, the behaviors of drivers and passengers are the most important elements affecting public safety, e.g., drowsiness and distraction are two main reasons for traffic accidents [44], which seriously affect public safety. In this case, the surveillance system can be used to recognize these dangerous behaviors, e.g., seat belt violations [45] and illegal cell phone usage while driving [46]. With the development of ridesharing services, e.g., Uber, a big challenge remains in guaranteeing the safety of passenger and driver. Liu *et al.* [47] designed an attack detection application to ensure in-vehicle safety, which can recognize speech in a vehicle and detect driving behaviors, i.e., abnormal trajectory, and capture video while in danger, as well as upload video to the cloud for further analysis.

C. Firefighting Applications

The work on video analytics in the fire department focuses on fire monitoring, i.e., forest fire monitoring and firefighting scene. Various cameras carried by firefighters or mobile devices, e.g., hand-held IR cameras with firefighters, multiple cameras with firefighting robots, and UAVs, collect most of the video data. However, due to the limitation of the network, most of the data are processed at the devices, such as robot, or the edge of the network, i.e., the local control center.

Due to limitations of infrastructure in a forest, it is difficult to monitor the entire place through traditional methods, relying on watchtowers, human observers, and satellites. The UAV is expected to change this dilemma due to its mobile characteristics for forest fire surveillance. Many UAV-based systems [17], [48] have been proposed and they usually equip not only IR cameras, due to the convenience on capturing fire by temperature, but also visual cameras for smoke detection. For example, Yuan *et al.* [26], [27] proposed to detect a flame in a forest based on the visual and thermal video, captured by UAVs' cameras. Moreover, smoke detection is usually used for an early warning as studied by Ma *et al.* [28].

The firefighting robot usually equips with multiple cameras, i.e., visual cameras and IR cameras, for sensing environment, and feedback the control of components, e.g., a nozzle for extinguishment, and a communication module for remote monitoring and controlling. For example,

Table 2 Part of Algorithms Used in Public Safety Applications (LP: License Plate)

Stage	Police	Transportation	Fire	EMS
Video Decoding	H.264 decoding, H.265 decoding, H.323 decoding, etc.			
Pre-processing	Image enhancement, noise reduction, lens correction, color adjustments, etc.			
Image Segmentation	Motion detection, edge detection, histogram-based methods, thresholding, etc.			
Object Detection	Face detection	Vehicle/LP detection	Flame/smoke/water detection	Fall detection
Object Recognition	Face recognition	LP number recognition	Object detection	–
Object Tracking	Single object detection, multiple object detection, etc.			
Data Fusion	Person re-identification	Vehicle re-identification	Nozzle control	–

the commercial firefighting robot Thermite [29] is small enough to get through doors, scoot down hallways, and even use elevators but strong enough to carry a robust dual fire suppression system and a robotic arm for downed victim extraction. Kim *et al.* [49] proposed a firefighting robot integrating IR camera and radar sensor to locate and measure distances to objects in the thermal video data. To reduce the searching time for the fire source in smoky indoor environments, Kim *et al.* [50] also proposed to locate the fire source according to the direction of smoke in the thermal image, detected based on the Bayesian estimation. To autonomously extinguish, the firefighting robot usually recognizes the spray and feedback from the control of the nozzle [51].

D. EMS Applications

Health care includes responsive emergency care and regular care [52], [53]. EMS systems provide transportation and medical care to maximize the survival probability of patients. In this paper, we mainly focus on the edge video-based EMS, which is an important subcategory in health care, requiring much real-time operation to be better guaranteed by the emerging edge computing. We classify the services as prehospital EMS and patient emergency monitoring services.

Most situations require paramedics with specific skills and knowledge, which are common in the hospital but less at the prehospital EMS. Video-based telemedicine is one of the most effective ways to improve the quality of prehospital care. The Houston Fire Department launched a pilot project, ETHAN [18]. It is a real-time video chat-based screening system that allows paramedics to participate in real-time video consultation of medical control physicians. Wu *et al.* [30] studied the application of wearable sensing, smart mobile devices, and video technology in EMS and then proposed an efficient, intelligent real-time emergency system for prehospital EMS to improve the quality of EMS using various sensors and live video streaming. In addition, several glass-enabled EMS applications have been developed [54]–[57], and the manner of wearing the glasses allows users to keep working while performing a remote video. For example, a project using Google Glass for paramedics was launched in 2014, but Google Glass rapidly indicated a lack of connectivity stability and very short battery life [58].

Moreover, several patient emergency monitoring applications used video-based systems to monitor and recognize people's different activities [59]–[61], e.g., fall detection. Currently, fall motion is one of the main causes of injuries for the elderly. Rougier *et al.* [62] designed an eight-IP-camera surveillance system for fall detection based on human shape deformation. In addition, depth cameras such as Kinect were studied in [63] and [64] to develop a monitor system with high accuracy.

IV. ALGORITHMS IN EVAPS

The video analysis algorithms are usually computation-intensive and offloading all workloads to edge devices brings an overwhelming burden. In this section, to understand which parts of the workload could be offloaded to the edge, we briefly introduce the general processes of video analytics in different public safety applications, followed by the introductions of several common algorithms. Then, for these four public safety application categories, we introduce typical algorithms, followed by a brief discussion.

A. General Process of Video Analytics

The process of video analytics usually can be divided into several stages as shown in Table 2. To better understand the video analytic effect on video, we take a typical video analytics process with face recognition (shown in Fig. 2) as an example. Generally, as the video analytics progresses on the edge, the amount of output data reduce, and demanded workload increases. The function of these stages is described as follows. In particular, we introduce these common algorithms here, and then, respectively, introduce the special algorithms appearing in these four categories of public safety application.

1) Video Decoding: Currently, cameras are able to provide a real-time streaming protocol (RTSP) or real-time messaging protocol (RTMP)-based interface for pulling video data. Thus, the first step is decoding the video data into a series of frames with different parameters, i.e., resolution and frames per second (FPS). There are many video encoding formats, such as H.264 [65], MPEG, H.265, and so on.

2) Preprocessing: In this paper, we consider all operations, between video decoding and image segmentation, to be preprocessing operations, which include various image editing operations, such as image enhancement, noise reduction, lens correction, and so on. Thus, in general, these operations are used to improve the quality of an

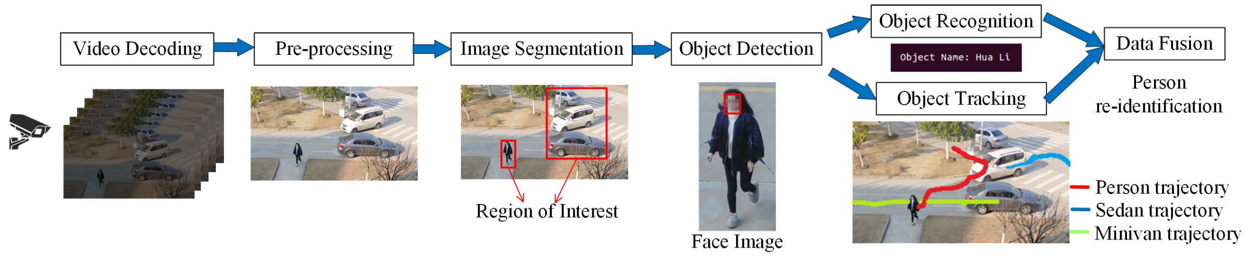


Fig. 2. ***Typical process of video analytics.***

image and correct the image from distortions. For instance, an image, captured by an inclined traffic camera in the dark, can be made lighter through image enhancement and twisted to a normal image for remaining operations.

3) Image Segmentation: Image segmentation is the process of partitioning an image into multiple segments (e.g., motion, foreground, and background). This type of operation is usually used to segment suspicious regions in video sequences. Motion detection is widely used in EVAPS, which is the process of detecting motion regions. The typical motion detection algorithms can be classified into three classifications as follows.

1) The background difference method [66] is one of the most commonly used methods, which detects the motion region based on the difference between the current frame and the background image.
2) The optical flow method [67]–[69] can detect the object with independent motion, which uses the optical flow characteristics of moving objects with time. The advantage of this method is that it can detect independent moving targets even in the presence of camera motion, e.g., cameras in UAVs.
3) The frame difference method [70] is less affected by light changes while it uses pixel-based time difference and thresholding between two or three adjacent frames in a continuous video sequence to extract motion regions.

4) Object Detection: This process usually aims to classify the regions of interest (ROIs) into a certain class (e.g., humans, buildings, or cars). The popular objects in public safety include the face, pedestrian, vehicle, flame, and smoke and can be detected by corresponding detection algorithms. The details are introduced in different application scenarios.

5) Object Recognition: The goal of object recognition technologies is classifying the observed objects into semantically meaningful categories. For example, the convolutional neural network (CNN)-based object recognition model, `Inception v3`, could recognize 1000 types of objects in images.

6) Object Tracking: Object tracking is the process of locating a moving object (or multiple objects) over time in a video. Thus, it can save repeating recognition operations, resulting in the reduction of total video analysis latency. The object tracking is also used in these four scenarios, and the typical object tracking algorithms can be classified into several categories. For example, a region-based tracking method usually aims to track foreground regions or blobs after background subtraction, and feature-based tracking is using various features to track the object in a video, e.g., histogram of oriented gradient (HOG) [71], Haar-like [72], color [73], and edges [74]. In addition, models based on deep CNN have dominated recent visual tracking research [75]–[77]. In addition, the multiple object tracking algorithm has been a hot topic in recent years [78], and more difficult but useful in real video surveillance systems than single object tracking algorithms.

7) Data Fusion: Data fusion is used to obtain more consistent and useful information by integrating several video analytic results from different video sources than that provided by any individual one. A typical application using data fusion is identifying the same person in multiple city cameras, which is also named person reidentification.

In addition, the following are also important in video analytics.

1) Generally, image segmentation extracts the regions, probably including interested objects, in images, e.g., a region of 200 $\times$ 200 extracted from a image of 1920 $\times$ 1080. Then, object detection algorithms detect whether an object is contained in a fixed-size window, e.g., 30 $\times$ 30, and scan the region using a sliding window method.
2) In the above-mentioned seven stages, several stages can be integrated into one step in deep learning (DL)-based algorithms (models), which can directly recognize the object in an image without image segmentation and object detection operations, by inserting an object detection layer in a neural network with selective search. However, the regular CV algorithms also play an important role in video analytics, especially in preprocessing and object tracking domains.

B. Algorithms in Police

The most popular and basic algorithms used in the police department are face detection and face recognition. Moreover, activity detection is also important, while a series of rarely occurring activities might be preludes of terrorist attacks, as reported in [79]. Furthermore, person reidentification technologies are also important for automatic searching of a person's presence across multiple cameras.

1) Face Detection: Face detection is used to find the facial ROI in images, usually with a less computational overhead, significantly reducing the overhead of the data transmission, and is thus suitable to be offloaded to the edge. The development of face detection technique can be divided into three stages. First, face detection algorithms use a template matching technique to match a face template image to each ROI in the image to determine if there was a face. The machine learning technique has also been used to detect a face image, including neural networks [80] and support vector machines (SVM). Second, Viola and Jones [81] designed a novel face detection algorithm, which proposed the AdaBoost framework-based classifier (cascade classifier) leveraging a Haar-like feature, constructing a strong classifier with high accuracy through multiple simple weak classifiers. This type of cascade classifiers allows a strong subclassifier to eliminate a large number of nonface images in the initial simple classifier. Third, many DL models are proposed. Cascade CNN [82] uses convolutional layers instead of the classifiers in a cascade classifier, and multitask cascaded convolutional network (MTCNN) [83] works in a similar way, but is more clever and reasonable. R-CNN [84] is a breakthrough work on proposing a novel model to avoid using the sliding windows with high performance, and fast R-CNN [85] and faster R-CNN [86] are improvements to R-CNN. Based on R-CNN, face R-CNN [87] is proposed, which is optimized for the particularity of face detection.

2) Face Recognition: At the beginning of researching about face detection, many types of algorithms are used to distinguish different faces, e.g., geometric feature-based algorithms, template-based matching algorithms, subspace algorithms, and so on. Two representative works are EigenFace [88] and FisherFace [89]. In the second stage, artificial feature and classifier are used together to recognize a face image. The common classifiers include neural networks, SVM [90], Bayesian [91], etc., and many features are used in face recognition problems, e.g., HOG, scale-invariant feature transform (SIFT), Gabor, local binary pattern [92], [93], and so on. Currently, many DL models are proposed and have been deployed in real systems. DeepFace [94] was proposed by Facebook in 2014, and it is the foundation work of deep CNN in face recognition and achieves an accuracy of 97.35% on Labeled Faces in the Wild (LFW) database. After that, Google proposed FaceNet [95] in 2015 with an accuracy of 99.63% on the LFW database. Sun *et al.* [96]–[98] also proposed a set of face recognition models, i.e., DeepID1, DeepID2, and DeepID3, respectively. However, the trained models are too large with a lot of computing and storage overheads to be offloaded into resource constrained edge devices, thus face recognition is performed in the cloud, generally.

3) Others: In addition to the above-mentioned algorithms, there are many other algorithms widely used in EVAPS. Activity recognition algorithms [99] aim to recognize the actions and goals of one or more people from a series of observations on the people's actions and the environmental conditions, which can be used to detect shooting event. Person reidentification algorithm is used to find an individual in diverse locations over different nonoverlapping camera views [100], enabling the capability of tracking one person in the city-wide environment.

C. Algorithms in Transportation

In this section, we mainly introduce the ALPR algorithms, vehicle detection, and behaviors analysis, which are widely used in transportation applications.

1) ALPR Algorithms: To recognize an LP, three stages are commonly applied, LP detection, character segmentation, and character recognition, in which the latter two can use a sweeping optical character recognition (OCR) engine to recognize an LP number [101]. After applying image segmentation, e.g., motion detection, motion areas are generated. Then, edge detection is used to select a few suspicious LP areas. Many operators are defined such as Sobel, Laplacian-of-Gaussian (LoG), Canny, and Prewitt [102] in edge detection. Then, many classifiers are proposed, such as SVM and pattern recognition [101]. In addition, many DL models are proposed. Meng *et al.* [103] proposed a CNN model named LocateNet with ten layers for predicting the four vertices coordinates in detection. Selmi *et al.* [104] integrated the DL architecture represented by the CNN model to filter and distinguish between LPs and non-LPs, performing LP detection. Similar to face detection, ALPR can be offloaded to the edge. Nevertheless, it should be noted that ALPR is a time-consuming task, and we can reduce its time consummation with piezoelectric sensors in road, which avoids a large number of invalid ALPR tasks.

2) Vehicle Detection: Vehicle detection is capable of detecting vehicles, measuring traffic parameters, and analyzing vehicles from images or video clips. Typically, the features in target samples are first extracted, and then classifiers are used to detect vehicles on extracted features, e.g., the Bayesian classifier on color features [81], the boosted classifier on Haar-like features [105], and AdaBoost classifiers on HOG features [106]. Moreover, Wu and Juang [107] proposed an adaptive vehicle detection method. It used the histogram equalization to remove the effects from light and weather. Then, the ROI is checked by the difference channel image in the RGB image and can be detected by the mechanism of merging and splitting moving objects. In addition, the And-Or graph model is employed to detect a vehicle by means of the vehicle's window edge, taillights, LP color, contour, and texture features [108], [109]. With the development of AI, DL algorithms are also proposed. Generally, vehicle detection algorithms are suitable to be offloaded to the edge.

3) Others: Behaviors analysis on the vehicle, traffic behavior, and even drivers are also important for public safety. The traditional process also includes several steps, i.e., feature extraction and classification. For example, traffic behavior of incident detection algorithms can be divided into pattern recognition-based algorithms [110], statistic-based algorithms [111], AI-based algorithms, and so on. Detecting the seat belt violation and illegal cell phone usage while driving is commonly behavior analytic [112].

D. Algorithms in Firefighting

In firefighting, video analytics is usually used to enhance vision for understanding fireground and monitor forest fire; therefore we introduce the algorithms in these two scenarios, i.e., flame detection, smoke detection, and object recognition. Note that the biggest difference in these scenarios is the usage of IR cameras, so the algorithms used in image segmentation stages are also different from other scenarios. Based on our observations, flame detection and smoke detection are generally executed on the edge due to limited network connections, as well as image segmentation, which is executed before detection algorithms.

1) Image Segmentation: Due to the mobility of camera carriers, motion detection cannot work well and several algorithms are proposed. For visual video, the color of the flame and smoke is a typical and useful characteristic to segment and detect the candidate regions in images [113], [114]. For thermal video, the temperature intensity captured by IR cameras is a useful characteristic for image segmentation [115].

2) Flame Detection: After segmenting the ROIs, flame detection algorithms [116] can be executed to finally decide whether the region is a flame image or not by feeding values of features to various classifiers, i.e., SVM [117], Bayesian classifiers [50], Markov models, and blob counter method [118]. Generally, several features could be used in flame recognition, such as texture [117], flickering, and motion vector [118]. Most flame detection algorithms rely on several features at the same time. For example, Yuan [119] comprehensively consider the factors of color, shape fluctuation, and growth rate based on a Gaussian mixture model. Moreover, several CNN-based flame detection models [120], [121] have been proposed without image segmentation.

3) Smoke Detection: Color-based algorithms are more suitable due to the motion of cameras. Then, most of the approaches use color and motion features to detect whether the candidate region is a smoke. In particular, since the color of smoke is kaleidoscopic, a preprocessing operation is prepared to enhance the smoke color nonlinearly in work [114], and then, the saliency of smoke is measured to estimate the candidate smoke region along with motion energy. Furthermore, a small number of DL-based approaches are proposed, such as CNN based [122] and deep normalization and convolutional neural network based [123].

4) Others: Person recognition algorithms are also used for searching victims. For example, Ulrich *et al.* [124] proposed to recognize the person using microdoppler and IR camera, which can be used for firefighter and firefighting robots to find a person in fire and smoke environments. To make firefighters easily understand fireground, the CNN-based algorithms are leveraged to recognize objects in an IR camera. In some cases of the firefighting robot domain, data fusion is needed. A typical example is fusing the analytic results of multiple cameras to locate the spray and flames, accurately. For instance, McNeil and Lattimer [51] proposed to use two IR cameras to recognize and locate the spray and the flame. Feeding the locations of the spray and the flame, the robot can adjust the nozzle angle to optimal extinguish.

E. Algorithms in Emergency Medical Service

Currently, a video conference technique is usually used in prehospital EMS for remote helping, which uses video encoding/decoding for data transmission. In addition to the video conference technique, fall detection is used in patient emergency monitoring services, which can be offloaded to the edges, such as cameras and local servers in home, hospital, and so on.

1) Fall Detection: Lee and Mihailidis [125] proposed to use a shape feature vector composed of five elements to represent the silhouette of a person, i.e., center of gravity, the perimeter of the object, the ferret diameter, and velocity of the center of gravity to detect the fallen person. Sehairi *et al.* [126] also proposed a fall detection algorithm based on combining shape features and motion analysis, which compute the vertical velocity of the head without the need of tracking filter since the tracking filter-based methods generally cost more resources and time. Meanwhile, this method avoids dependence of the skin or hair color-based detection to estimate head coordinates, which often limits the capability of such algorithms. Then, a classifier is used to detect whether the person in the video has fallen or not, e.g., SVM. Several neural network models are also proposed and deployed [127], [128]. For example, Alhimale *et al.* [128] implemented an intelligent and video-based fall detection system using a neural network.

In addition to vision-based solutions, several nonvision-based methods are also proposed, such as leveraging accelerometer sensors in smart phones, which also have good performances. However, in this paper, we mainly focus on vision-based solutions.

F. Discussion

We have introduced the general process of video analysis and a few frequently used video analytics algorithms in public safety. As the processing of video analytics, the amount of data that needs to be transmitted decreases. For example, a preprocessed video may be the same size

Table 3 Part of Platforms Introduced in Our Paper

Category	Platform	Workload		Dynamic	API
		Edge	Cloud		
Police	IBM S3 [131]	-	All	N	Y
	Anjum *et al.* [132]	-	All	N	N
	Hu *et al.* [133]	Feature extraction for recognition	Feature matching	N	N
	A3 [21]	LPR	Data fusion	Y	Y
Transportation	Ananthanarayanan *et al.* [3]	All		Y	Y
	SafeShareRide [47]	Video encoding	Video analysis	N	N
Fire	FAST [134]	Flame detection	Data fusion	N	N
	McNeil *et al.* [51]	All	-	N	N
	Kalatzis *et al.* [135]	Non or flame/smoke detection	Flame/smoke detection or non	Y	N
EMS	Teleconsultant [136]	Video encoding	Mouth droop detection	N	N
	STREMS [30]	Video encoding	Video decoding	N	N
General-purpose	VideoStorm	-	All	Y	N
	LAVEA [137]	All		Y	N
	Wang *et al.* [36]	Detection algorithms	Recognition algorithms	Y	N
	Zhang *et al.* [138]	Detection algorithms	Recognition and data fusion algorithms	Y	Y

as the original video or slightly lower than the original video. However, for a video after detecting faces, only facial images need to be transmitted for recognition. Take a full-HD video (1920 × 1080) as an example. Its network bandwidth cost and transmission latency are around 6 Mb/s and 120 ms, respectively, and the size of a facial image may be only 50 kB [129]. Moreover, using a tracking algorithm can further reduce the data transmission, since the face image can be transmitted only when the person appears or walks out of the monitoring area.

Although edge video analytics can bring many benefits, offloading all workloads to the edge is impossible due to various reasons. A typical reason is the many video analytics algorithms consume many computing resources, such as sliding window-based object detection, and various DL models. Take `Inception v3` as an example. The time consumption, respectively, is 153 and 242.8 ms on the Intel i7-6700 CPU and NVIDIA Jetson TX2, respectively, which is usually performed as an on-board edge device for UAVs or cameras. Note that a video may consist of tens of frames, according to the value of FPS. Therefore, in the design phase of a platform, we should consider these tradeoff factors, in terms of network, computational resources, and latency. In the next section, we introduce several recently proposed video analytics platforms, especially edge video analytics platforms.

V. PLATFORMS IN EVAPS

In this section, we introduce the platforms in EVAPS as listed in Table 3, followed by several general-purpose platforms. Here, we classify these platforms based on their detailed implementations in papers, which might be the most suitable domain. Furthermore, a part of them can also be deployed into other domains with a few revisions. Dynamic means that the platform enables workload offloaded between edge–edge or edge–cloud, and air position indicator (API) provides an interface for developing, e.g., customized offloading strategies or customized video analytics algorithms.

A. Police Department

1) Cloud-Based Platforms: As cloud computing has grown rapidly, a number of cloud-based video analytics services are proposed by cloud service providers [138], [139]. In addition, due to the privacy and law of video for public spaces in the police department, most of them use private clouds to handle the video analysis and storage. Thus, many commercial private cloud-based video analytics platforms are proposed.

a) Commercial video analytics platform (IBM S3): IBM proposed a commercial smart video analytics system (S3) [130] as shown in Fig. 3, which uses automatic video analytics techniques to extract information from the surveillance data, e.g., behavior analysis, face recognition, and LP recognition. It should be mentioned that IBM S3 is a general-purpose platform and we introduce it here since a few platforms used by the police department are implemented based on IBM S3, which also means that IBM S3 provides the APIs for developing. Mainly, the first version of IBM S3 includes two components, Smart Surveillance Engine (SSE) and Middleware for Large-Scale Surveillance (MILS), while SSE provides the front-end video analysis capabilities, and MILS provides data management

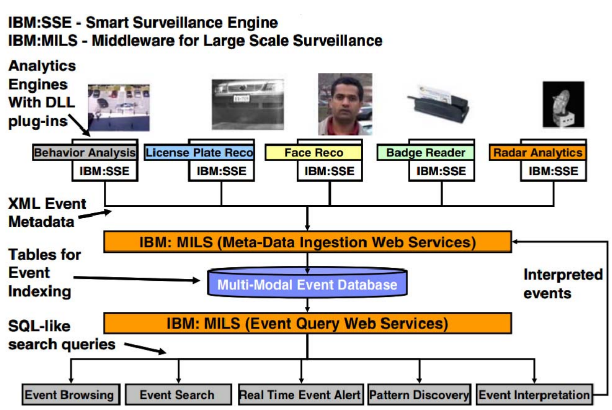


Fig. 3. ***Architecture of IBM S3 [130].***

and retrieval capabilities. As shown in Fig. 3, the SSEs process video data from a variety of cameras and generate real-time alerts and generic event meta-data, e.g., face detection event, object detection event, and object tracking event. To improve the performance of S3, several post-process video analytics algorithms could be implemented, i.e., tracking algorithms, to reduce repeated computing on event recognition. MILS provides the data management services, consisting of metadata ingestion services (MIS), Schema Management Services (SchemaMS), and System Management Services (SystemMS). The MIS allows an engine to ingest events into the MILS's database for indexing and searching. The SchemaMS allows a developer to manage their own metadata schema for video analysis. The SystemMS provides facilities to manage a surveillance system, including camera management (e.g., add/delete a camera), engine management (i.e., start/stop an analysis engine for a camera), user management, and query response (e.g., searching one event in S3 database).

Moreover, S3 provides interfaces for developers, so that developers can develop their own basic video analytics algorithms, e.g., face detection algorithms with higher accuracy, and based on XML format event, high-level event detection algorithms can also be implemented to analyze events, e.g., person reidentification by aggregating the same facial identification in multiple cameras' events. Alternatively, a developer can easily implement a sophisticated event detection, constructed by basic events, by providing a XML-written configuration file. For example, a crowd congestion event can be defined by tens of face detection events. In this case, to suit a real user requirement, Prati *et al.* [140] implemented their own video analytics system based on S3 by implementing a people reidentification combining with multiple camera video, which is important and widely used in the police department.

In addition, many companies, such as IBM, Hikvision, and Dahua, also propose out-of-the-box intelligent cameras with built-in video analytics algorithms, e.g., face detection and line crossing detection, and so on. These built-in algorithms were typically implemented on the server side in the past cloud-based solutions. Currently, a few intelligent camera-based platforms are being proposed. For instance, Shao *et al.* [5] proposed a platform with smart storage and rapid retrieval, which utilizes the temporal-spatial association analytics with respect to the abnormal events in different monitoring.

b) Object detection and classification platform in clouds: In recent years, with the increasing scale of video surveillance and the increasing complexity of video analytics algorithms, e.g., DL models, video analysis on the cloud has become a huge burden. To deal with this problem, Anjum *et al.* [131] proposed an object detection and classification platform in clouds for high-performance video analysis, which provides a scalable solution for video analysis with minimum latency and user intervention, while object detection and classification are the basic tasks

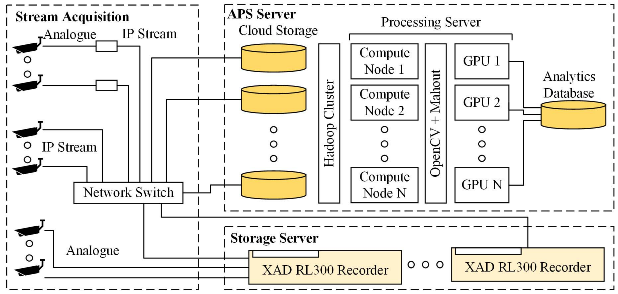


Fig. 4. ***Architecture of Anjum et al. [131] platform.***

in video analytics and the starting point for other complex applications. In their paper, the authors used face/vehicle detection as the video analytics workload, and the experimental results show that face detection is with a higher accuracy, which is the reason we introduce this platform in this category. As shown in Fig. 4, the proposed video analytics platform includes stream acquisition for capturing video streams from cameras, storage server for video stream storage, and analysis processing server (APS) for video analysis. The master of the APS servers handles the video analytics task and sends it to one compute APS node. Although compute APS node receives the task, it requests the video data from the storage server, and decode it into multiple video frames, followed by a multiple-thread analyzing by detailed algorithms, i.e., the feature extraction algorithm and cascade-based classification algorithm in their work.

In the implementation, this multiple-thread analyzing is implemented based on Hadoop MapReduce. A video is decoded into a number of frames as the input of MapReduce. The Map task is used for processing the video frames for object detection and classification, while the Reduce task is used to save the results into the database.

2) Edge-Based Platforms: A lot of video analytics platforms have been built to verify edge computing-based solutions to reduce the burden of the cloud. Here, we introduce two edge-based platforms. One is a hybrid edge–cloud face recognition platform. Another one is an edge-based real-time kidnapper tracking platform.

a) Hybrid face recognition platform: Hu *et al.* [132] proposed an edge-enabled face recognition system as shown in Fig. 5, in which face identifier is computed at the edge (Fog in the figure) and matching is performed at the cloud, avoiding a large volume of video transmission. The proposed system consists of five components, four of which are in the cloud: 1) fog nodes; 2) management server (MS); 3) information server (IS); 4) resolution server (RS); and 5) data center. In their system, part of the workload of video analysis is offloaded to the fog nodes, i.e., video decoding, preprocessing, face detection, and facial feature extraction, and the facial feature vector is used as a face identifier. The MS connects with edge nodes and manages IS and RS in the cloud for

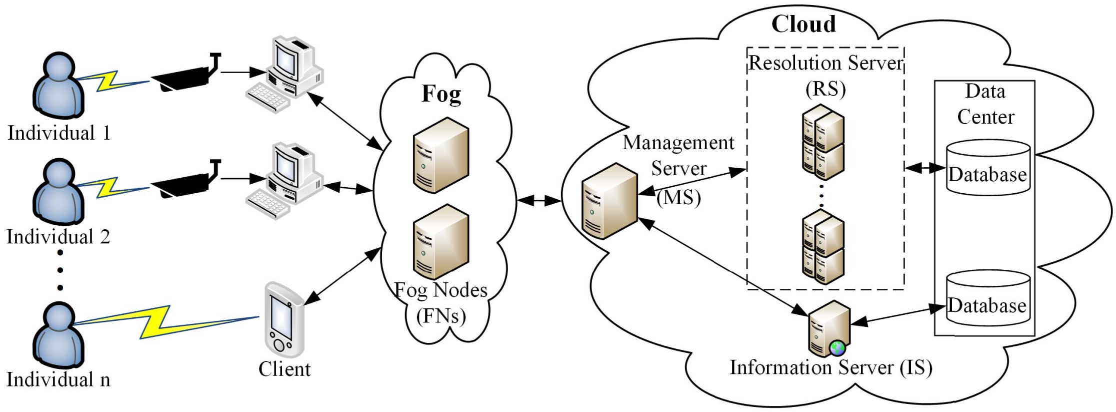


Fig. 5. ***Architecture of Hu et al. [132] platform.***

resources scheduling and computing task allocating. The RS performs detailed face identifier matching and returns an address where an individual's identity information is located, while IS is responsible for managing individuals' identity information. Both RS and IS share the same data center as a database, which has powerful data storage capability. The experimental results show that the proposed edge-based system has a better performance in terms of network transmission and response time.

b) Edge based real-time kidnapper tracking platform: Zhang *et al.* [21] proposed a distributed framework as shown in Fig. 6, (extended from their previous work [129]), and implemented a kidnapper tracking application by tracking the kidnapper's vehicle, called AMBER Alert Assistant (A3), which enables surrounding edge devices to perform real-time video analysis. Fig. 6 illustrates the architecture of the proposed framework, including three layers. The executor management serves as an adapter that allocates computing resources to a job for video analysis, i.e., motion detection and plate recognition in their work. The job management layer, including three major components, is proposed to manage the task and offload its jobs to its own executor management layer or cooperative edge node when its idle computation resources cannot meet more task execution, thus providing dynamic characteristic. Taking A3 as an example, the task is kidnapper vehicle tracking in city-wide cameras, and the jobs are video processes, including video decoding, motion detection, and license plate recognition (LPR), while motion detection is used to reduce the data transmission between different edge nodes and computing latency caused by LP recognition. For one camera, it sends video processing jobs to collaborative edge nodes for real-time video analysis utilizing a job-scheduling module and diffuse the tracking task to surrounding cameras when it does not find the vehicle in its video after a threshold time, utilizing a task dispatch module. Similar to work [131], it also utilizes a multiple thread mechanism for maximizing the use of hardware performance, in which all executors are communicated with a message queue. The up-layer service management is used to discover surrounding cooperative edge nodes and provide an easy-to-use configuration (referring to API) for running task (i.e., configure how to connect a camera with task parameters), as well as security mechanisms for the whole platform.

Based on the proposed platform, the edge nodes can collaboratively track a kidnappers' vehicle in real time. The results also show that the LPR algorithm is a computation-intensive task for edge nodes. Therefore, in an edge-based video analytics platform, we should consider several techniques to reduce the computational burden, e.g., an object-tracking algorithm to avoid repeated LPR operation.

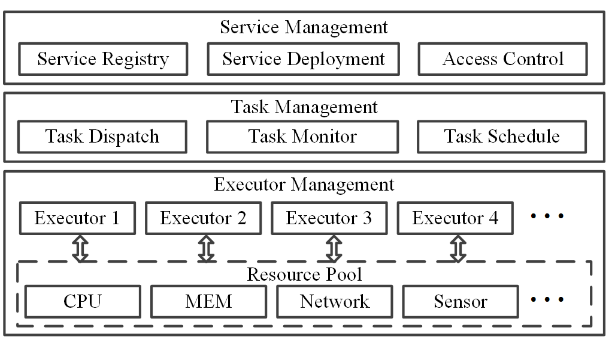


Fig. 6. ***Architecture of extended Firework [21].***

B. Transportation Department

The cloud-based video analytics platforms used in the transportation department are similar to the ones in the police department. For example, by implementing a vehicle-tracking algorithm in IBM S3, it enables IBM S3 to be a transportation platform. Consequently, in this section, we mainly introduce several edge-based platforms, focusing on a real-world platform and a platform for improving public safety for emerging ride-sharing services by analyzing an in-vehicle video.

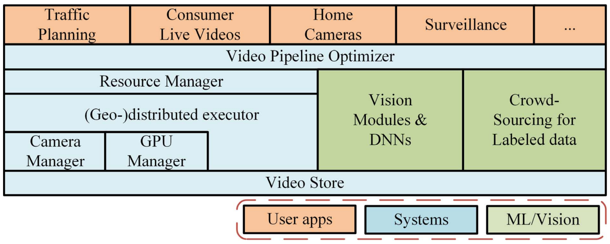


Fig. 7. ***Video analytics software stack, Rocket [3].***

c) Real-world transportation platform: Ananthanarayanan *et al.* [3] proposed a hierarchical geo-distributed infrastructure to support the wide range of video analytics in transportation scenarios, including cameras, edge and private cluster, and public cloud as well as a video analytics stack, called Rocket, as shown in Fig. 7. The video pipeline optimizer (VPO) converts video queries into a video analytics pipeline that includes many vision modules, e.g., a video decoder, followed by an object detector, and an object tracker. The VPO can also estimate the resource-accuracy profile of the query by calculating the total resource cost and accuracy for each configuration of knobs and implementations of each module with the labeled data from crowdsourcing to compute accuracy. The centralized global resource manager (RM) responds for all executing query pipelines and their access to resources, e.g., CPU and GPU compute, network and even parameters of cameras, according to the profile calculated by VPO. RM also periodically determines the best configuration of each query and places components across the available computing nodes (e.g., edge and private clusters and the public cloud). The detailed resource management is implemented by standard operating system mechanisms, camera manager, and GPU manager. In particular, the deep neural network (DNN) execution service (in Vision modules and DNNs) is running on each machine for efficiently handling all DNN requests on GPU.

Based on the proposed software stack rocket, an implemented system can dynamically decide on the placement of the vision modules in the pipeline, i.e., camera, edge and private cluster, or cloud, involving considering capacities of multiple resources, e.g., compute and network. Finally, traffic analytics solutions based on rocket have been actively deployed since 2016. For example, in Bellevue, Washington, a multimodel object counter has run 24/7 to help the city understand and track cars, pedestrians, and bikes, as well as raising alerts on anomalous traffic patterns.

d) SafeShareRide platform: Liu *et al.* proposed SafeShareRide [47], a platform to protect the safety of passenger and driver for ridesharing services. The overall platform of SafeShareRide is shown in Fig. 8, which consists of two components: edge or mobile devices and the cloud. The edge component has a three-stage detection model to detect the attacks happening on a vehicle: 1) the speed recognition model recognizes the key words in live audio, i.e., help; 2) the driving safety detection model determines whether the driving behavior is normal, e.g., zigzag route, utilizing the onboard diagnostics data and other sensors; and 3) video compression is performed at the edge to save upload bandwidth, while video analysis is conducted in the cloud with powerful resources. During these two detection stages, any abnormal event will trigger the third stage detection.

In the SafeShareRide platform, video capture and analysis adopt an edge-cloud collaborative model. The related video clips are compressed and sent to the cloud. In the cloud, two kinds of detection are used for the video analysis, including action detection and object recognition. The action detection is used to detect excessive movements of the driver and passenger, and the object recognition leverages the CNN-based model to recognize objects in the video, e.g., guns and knives. Finally, the alerted videos, i.e., having abnormal movements or dangerous objects, are shared with the law enforcement via a security link.

C. Fire Department

Similar to our introduction of firefighting applications, we also introduce several platforms related to the fireground and forest fire monitoring, which improve public safety. Due to the limitation of the network, most of the platforms are edge-based and have a better performance than the cloud-based one.

1) Platforms for Fireground: To introduce the platforms for fireground, we first introduce an edge-enabled smart

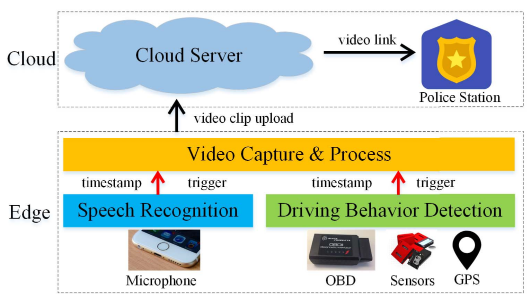


Fig. 8. ***Architecture of SafeShareRide [47].***

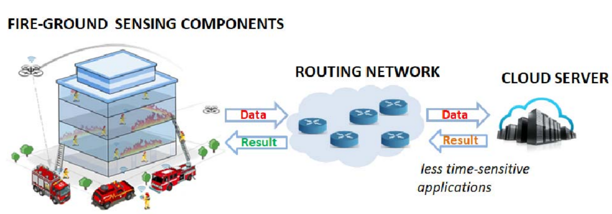


Fig. 9. ***Architecture of edge computing enabled smart firefighting [133].***

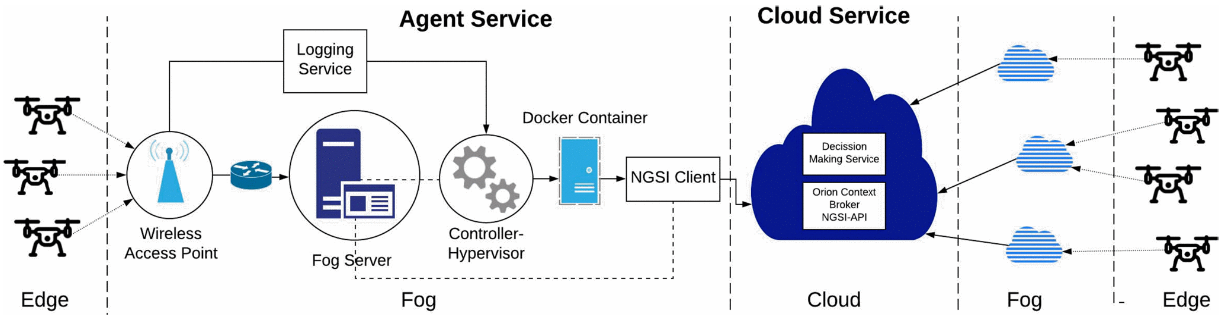


Fig. 10. ***Architecture of edge computing enabled UAV [134].***

firefighting platform, FAST, which leverages edge-based smart devices in fireground to protect the safety of firefighters. Then, we then introduce several firefighting robot platforms, which can be used in fireground.

a) FAST: Edge-enabled smart firefighting platform: To make firefighting smart, Wu *et al.* [133] explored the smart firefighting field utilizing edge computing and discussed the system architecture, as well as built edge-enabled smart firefighting. Fig. 9 illustrates the architecture of the envisioned edge computing-enabled smart firefighting. In fireground, on the fire vehicle, there is a local centralized data center, also referred to as a base station, usually deployed on a laptop, providing the user interface for monitoring and tracking for firefighter safety, and advanced communication systems for various sensors, i.e., 4G, WiFi, and an *ad hoc* wireless network. In this case, the local centralized data center can perform as an edge node, handling primary tasks from sensors, e.g., video analysis, especially for most of the cameras, including existing surveillance cameras, hand-held IR thermal cameras, and cameras carried by UAVs and firefighting robots. Moreover, the edge node can upload results to the cloud or offload the secondary tasks to the cloud, e.g., 3-D mapping for a building.

To simulate an edge-based video analysis in fireground, the authors evaluated the performances of flame detection in different solutions, including cloud-based and edge-based. The data source here could be hand-held cameras, firefighting robots, or UAVs, which can communicate with the edge node via various wireless techniques. Since the latency on data transmission is the same, the authors did not take this latency into account, and the preliminary results show that edge-enabled smart firefighting would significantly increase the system's reactive speed, with a 50% reduction in system latency on average.

b) Firefighting robot platforms: Firefighting robots are widely used in fireground, since it can communicate with trapped and injured victims inside the fire scene, and send video and audio to the control unit describing the fire environment inside the building. Many autonomous extinguishment firefighting robots have been proposed. For example, the robot designed by McNeil and Lattimer [51] consists of a series of video analytics-based components: fire localization, water classification, trajectory modeling, spray localization, pitch and yaw angle estimation, PI control, and visual servo control. Thus, the results of the former four components feedback to the pitch and yaw angle estimation to obtain the control command, which enables control of the nozzle for autonomously extinguishing by adjusting the PI and servos. Moreover, to obtain accurate fire and spray localization, a proposed robot is equipped with multiple cameras, including two IR thermal cameras and one visual camera, thus capturing multiperspective video data. Generally, several of these firefighting robots can be launched to work together collaboratively with the assistance of a remote control unit, held by firefighters or a local commander. In addition, other firefighting robot platforms have been proposed in [49] and [50].

2) Platforms for Forest Fire Monitoring: Because the drone can easily monitor the forest fire in the sky, many UAV-based platforms are proposed to forest fire monitor. Due to the limitation on the network, most UAVs process all or part of the video analytics' workload on board, which can be classified into edge-based platforms.

Kalatzis *et al.* [134] proposed an edge-based UAV platform for forest fire detection as shown in Fig. 10, which consists of the cloud with a powerful resource, edge servers with rich resources, and UAVs with the capability of sensing. The main components for fire monitoring include image classification service for fire/smoke detection using a DNN model, controller-hypervisor for service management (i.e., create, run, scale and stop application-specific virtual services) and decision-making service for detection of an emergency-level situation. In addition, several components are proposed to maintain the running of the system, e.g., logging services for monitoring all parameters (CPU, storage), NSGI Client for data transmission between different layers, and Orion Context Broker for securely maintaining data received from NSGI Client. The authors evaluated the performance of the proposed platform, and the results show that the case, while image classification runs on the edge, has a similar latency with the cloud, but a lower volume of data transmission. Note that the Raspberry Pi 3 Model B as an on-board computing unit is with a lower computation

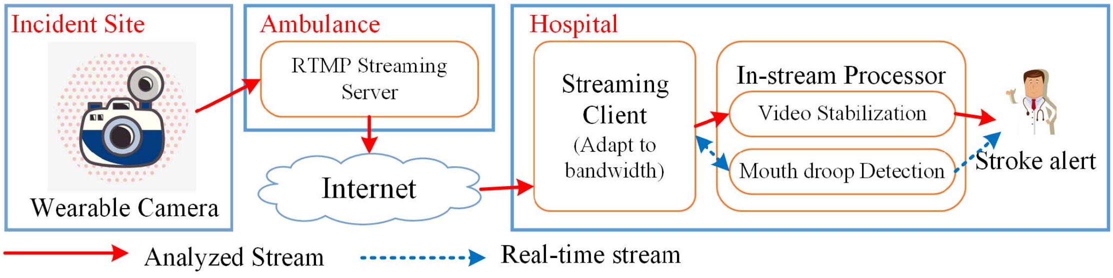


Fig. 11. ***Architecture of Teleconsultant [135].***

capability than NVIDIA Jetson TX2, which is more commonly used in UAV.

Similarly, Luo *et al.* [141] proposed a platform for disaster sensing applications, including fire. The UAV captures video data and feeds a preprocessed video data to a context-aware video scheduler, which decides if the video should be sent to the cloud for more rigorous analysis or not, e.g., object detection. However, due to the limitation of wireless communication, e.g., communication radius, the communication quality decreases as the distance increases. In order to improve the QoS of UAV-based systems, Wang *et al.* [142] proposed a UAV video transmission platform, which uses multiple UAVs as relays coordinated by the edge servers.

In addition, in helicopter-based UAV systems, the helicopter not only has a heavy load capability but also has high-performance computing that can perform all computational intensive tasks on board. Thus, the base station can only perform the role of data fusion, accepting forest fire alarm from UAVs, including pictures and videos. For instance, Merino *et al.* [143] proposed an architecture, consisting of several UAVs and a central station, all of which have two main components: the decision-making system and the perception system. The perception system enables the UAV to carry various devices, e.g., visual cameras, IR cameras, and computation units. The decision-making system aims to autonomously navigate between waypoints, thus, performing certain tasks, implemented by four different mechanisms: task allocation, task planning, coordination, and supervision. Although a UAV performs a fire monitor task, it extracts fire contours by means of on-board IR and visual cameras, and then send such information to the central station, as well as sensed data from various UAV sensors. By the collaboration of multiple UAVs, the bigger areas or complementary views of a fire could be covered/obtained in the view of a central station. Moreover, Pastor *et al.* [144] also proposed a similar architecture, which processes the video data on-board using a powerful computation unit, e.g., FPGA-based unit.

D. Emergency Medical Services

EMS is one of the public services, which provides rapid response, transportation, and emergency medical care for injured patients. Edge video-based EMS is largely dependent on a real-time and efficient platform. Here, we introduce two edge video-based prehospital EMS platforms, while the used video-related algorithms mainly include video encoding/decoding.

1) Teleconsultant Platform: Elgamal *et al.* [135] proposed a telemedicine system, Teleconsultant, which provides near real-time communication and treatment information between paramedics and doctors. As shown in Fig. 11, Teleconsultant deployed at an incident area consists of ambulance and hospital. It is assumed that the camera worn by the paramedic can communicate with the laptop in the ambulance via the *ad hoc* WiFi P2P network, which acts as a video streaming server and delivers the video to hospital customers via the Nginx web server with two modules: RTMP and HTTP Live Streaming. The ambulance installs a wireless base station that allows the laptop to connect wirelessly to the Internet.

In the hospital, the doctor can review the video from the accident site from a desktop, which is equipped with a streaming client and an in-stream processor. The streaming client communicates with the streaming server to obtain the most appropriate bit rate video stream based on the client bandwidth. The in-stream processor is used to decode video and perform arbitrary image processing functions on the decoded frame, e.g., mouth droop detection. Similarly, Rogar *et al.* [54] also proposed a medical platform, which leverages smart glasses as wearable cameras in emergency medical situations. With the platform, paramedics can access medical knowledge and obtain assistance from hospital specialists at the accident site via smart glasses-based video conferencing.

2) Strems Platform: Wu *et al.* [30] proposed an efficient and intelligent prehospital EMS system, which explored the use of wearable sensing, smart mobile devices, and video technology to improve the QoS of EMS. It consists of EMS, cloud center, and hospital, while EMS and hospital are performing the role of edge. The EMS element actually contains wearable sensing devices and a mobile application on the ambulance, which collects data, e.g., electrocardiogram, vital signs, and the actual location of the GPS, as well as captures images or short video clips about the patient, and transmits the data to the cloud center. The hospital has prehospital visual data from

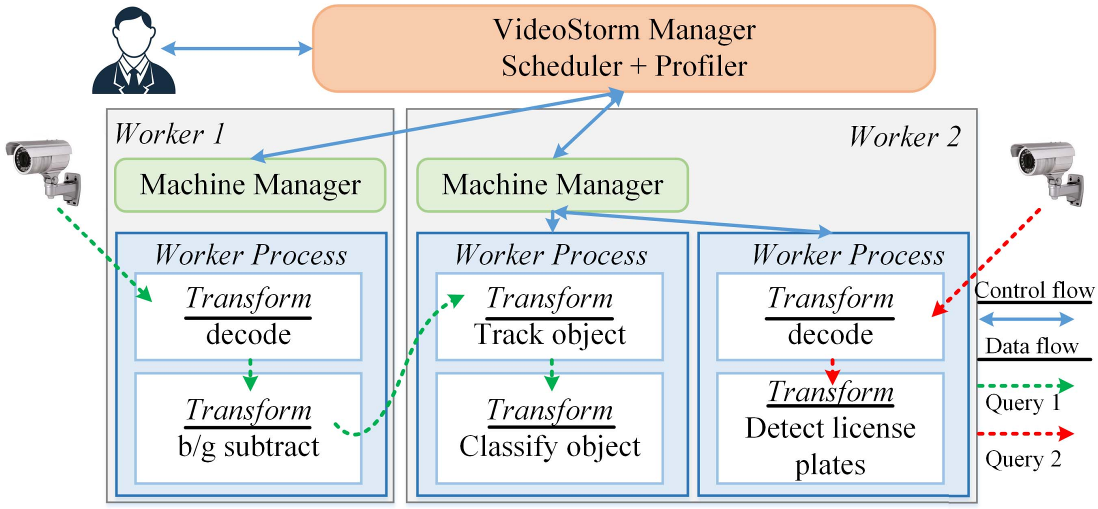


Fig. 12. ***Architecture of VideoStorm [145].***

cloud center and real-time video conference for telemedicine with EMS. The hospital can review all emergency data, shortening the handoff time of incoming patients when they arrive. In addition, paramedics on the EMS can use real-time video communication to conduct additional medical examinations, triplets, or other early medical interventions. The cloud center consists of three parts, i.e., a real-time video server, database system, and user authentication. A particular real-time video server is set up to manage point-to-point video communication. The database system stores all reported ambulance emergency data. User authentication is used to authenticate the user. A real-time database system, i.e., firebase, is used to synchronize the data between the EMS and the hospital, including GPS, electrocardiogram, and so on.

As a preliminary study, the authors evaluated the performances of live video communication between the EMS and a hospital while the ambulances were running at different speeds. The results show that the latency of the high-speed is with an almost 1-s latency and more than 40% frame loss rate for full HD video via User Datagram Protocol (UDP)-based real-time protocol (as part of RTSP), which is inferior to the one of low-speed.

3) Brief Summary: After reviewing many platforms in EMS, we discovered that the video quality is very important for a hospital's doctors to acquire accurate on-site information. Typically, most EMS platforms have not performed video analysis on the edge, and they mainly focus on improving video quality. Thus, video encoding/decoding techniques should be given more attention. In addition, we believe that augmented reality technology is promising to enhance video quality with clearer visual alerts.

E. General-Purpose Platforms

In addition to these domain-special platforms, a few superior video analytics platforms are proposed and most of them can be easily applied to public safety scenarios by implementing public safety video analytics algorithms instead of original video analytics algorithms.

1) VideoStorm Platform and Its Improvements: Video analytics can have very high resource demands to analyze the live video in real time. Microsoft group performed a series of works to improve the accuracy of video data query and the usage of computation in the cloud.

a) VideoStorm: Zhang *et al.* [145] proposed VideoStorm, a video analytics platform that processes thousands of live video streams in the cloud, where resource management is crucial for the improvement of the costs of video analytics with two key characteristics: 1) resource-quality tradeoff with multidimensional configurations and 2) variety in quality and lag goals. Fig. 12 shows the high-level architecture of VideoStorm and the specifications for video queries (i.e., two example queries).

Each query is defined as a directed acyclic graph (DAG) of transforms. Each transform processes a time-ordered message stream (e.g., video frames) and its output is passed to the next processing unit. By implementing detailed video analytics algorithms, the platform could be a domain-specific platform, where all video data are uploaded to the cloud. The VideoStorm consists of a centralized manager and a few worker machines. Each worker machine has a machine manager to manage worker processes, and the machine manager reports the resource usage and status of each transform to the centralized manager. Leveraging reported information, the scheduler on the centralized manager can assign resources for queries.

VideoStorm allows arbitrary DAGs, including multiple inputs and outputs for a transform, so that the scheduler can tradeoff the query configuration and high-quality results by dynamically adjusting with offline profiling and online phases. The offline profiler generates the query resource quality profile, which is used by the online

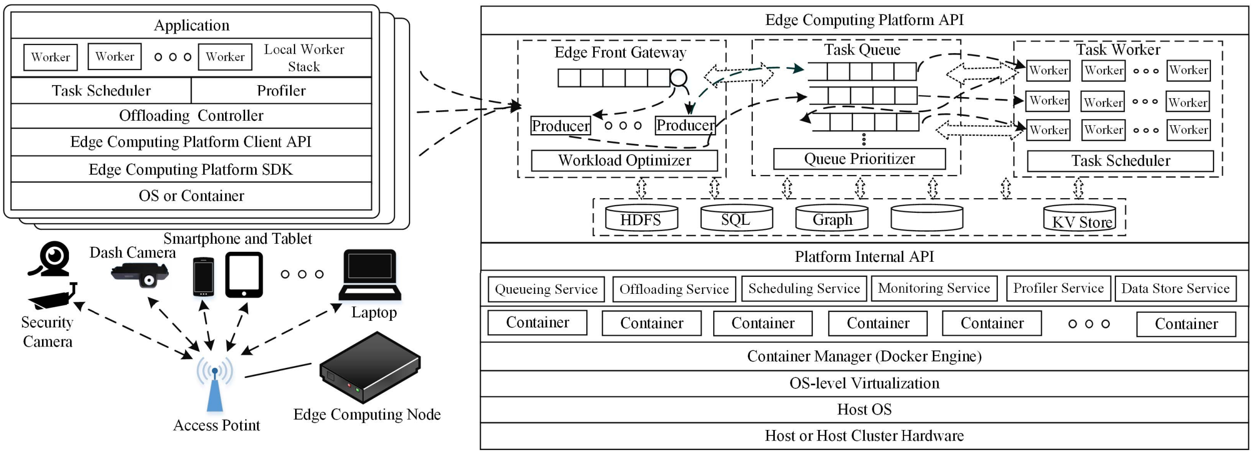


Fig. 13. ***Architecture of LAVEA [136].***

scheduler to allocate resources to queries, aiming to maximize performance on quality and lag. In the online phase, the scheduler periodically adjusts resource allocation, machine placement, and configurations, changes in demand and/or capacity for all running queries based on the forward profile.

b) Chameleon: In addition, Jiang *et al.* [146] designed a controller, Chameleon, which dynamically sets the best configurations for deep convolutional neural network-based video analytic pipelines. Video processing pipelines are adapted over time to avoid low accuracy. However, a naive reprofiling is prohibitively expensive. Instead, Chameleon is proposed to use several techniques to dramatically reduce profiling cost and improve accuracy.

c) VideoEdge: A key assumption in VideoStorm is that there is enough bandwidth to ensure video data transmission from the camera to the cloud. However, it is impossible between a private cluster (as an edge) and the public cloud. Thus, Hung *et al.* [147] proposed VideoEdge to identify the best tradeoff between multiple resources and accuracy, while they found that video analytic queries had many implementation options impacting their resource demands and accuracy of outputs.

d) Summary: The work of Hung *et al.* [147] revealed that VideoStorm only employs the CPU resource in a single cluster to optimize query knobs and resource allocation, and VideoStorm cannot be trivially applied to problem setting (i.e., hierarchical clouds). Chameleon [146] can continuously adjust DNN configurations to optimize accuracy or reduce resources' costs based on the temporal and spatial correlation among the video frames. However, Chameleon does not address the query-merging opportunity, contributing to significant gains in accuracy, which is the goal of the VideoEdge.

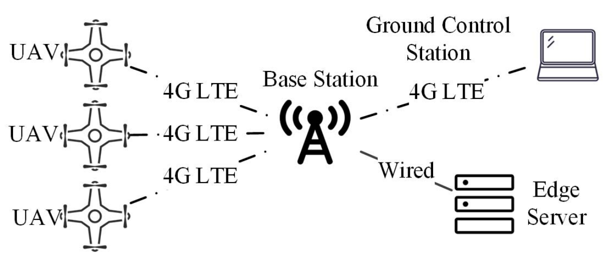


Fig. 14. ***Architecture of UAV surveillance.***

2) LAVEA Platform: Yi *et al.* [136] proposed a latency-aware video analytics platform based on edge computing, called LAVEA, which explores the video analytics task for latency-sensitive applications and performs collaborative video analysis at edges and the cloud. It is easy to implement a public safety application, e.g., an ALPR-based vehicle tracking application by creating a docker image with the ALPR algorithm. The LAVEA (presented in Fig. 13) mainly consists of the edge computing node (ECN) and the edge client (EC). An ECN has many more resources than an EC but limited compared to the cloud. When an EC executes tasks, and an ECN nearby is available, the tasks can be executed locally or remotely, i.e., running on the ECN or the cloud.

ECNs: In LAVEA, ECNs provide edge-computing services to nearby mobile devices while the ECN connected to the same access point or base station with an EC is called an edge front end. The edge front end always performs the role of a master node and coordinator with other edge and cloud nodes. In addition, multiple ECNs can collaborate. To isolate different clients' tasks and resources on ECNs, virtualization technology is used, i.e., docker in implementation, as well as container manager for management. Based on several internal microservices, e.g., queue services and scheduling services, the functions of identifying workloads, managing queue priorities, and scheduling tasks are implemented. Thus, the client can submit tasks via the client API to LAVEA and LAVEA can schedule, execute, and manage these tasks with the collaboration.

EC: This has limited resources but handle requests from numerous clients. Thus, heavy computing tasks could be offloaded to nearby ECNs and only lightweight tasks run locally. The profiler and the offloading controller perform as participants in the profiler and the offloading services, so that an EC can provide offloading information to the edge front-end node and complete the offloading decision.

To implement such nodes in LAVEA, three main services are implemented, including profiler service, monitoring service, and offloading service. The profiler service collects the performance metrics of tasks on different devices, e.g., length of task queue, and the monitor service is used to collect runtime information, e.g., network bandwidth and latency between different devices, so the offloading service can decide the placement of offloading based on collected information by the other two services. Moreover, the authors formulated the offloading problem to an optimization problem, as well as implemented offloading strategies to minimize the response time of video analysis.

3) UAV-Based Platforms: The current police department has started to leverage UAV as a mobile and flying camera to expand the view. A few UAV-based platforms have been proposed, and they have a similar architecture as shown in Fig. 14. As a preliminary work, Qazi *et al.* [148] investigated the technical performance of a UAV-based real-time video surveillance system over 4G local thermal equilibrium (LTE) using the metrics of throughputs, loss rates, and delay in relation to the physical aspects of wireless propagation: multipath propagation loss, shadowing, and fading models. Moreover, Motlagh *et al.* [35] proposed a UAV-based platform for crowd surveillance, in which the UAV (called hexacopter) equips with a Raspberry Pi as an on-board computing unit, and a laptop serves as an edge node. The authors set up an LTE server and both the UAV and the laptop were equipped with an LTE transceiver. Thus, the UAV can transmit raw video data to the laptop or analyze video data locally, transmitting only the recognition result to the laptop, which could be two different edge-based solutions. Since the computing resource is limited in the UAV, the later has a higher performance. In the above-mentioned UAV-based platforms, we needed to implement our domain-specific video analytics algorithms to build our own EVAPS applications.

Wang *et al.* [36] proposed a UAV-based video analytics platform, in which UAVs can transmit all or part of the video captured to a ground-based cloudlet [10] over a wireless network. To reduce bandwidth demand for video transmission from a UAV to the cloudlet, the authors proposed four strategies, EarlyDiscard (ED), just-in-time-learning (JITL), ReachBack (RB), and ContextAware (CA). In the ED strategy, a modified MobileNet DNN model is used to filter video frames in the UAV camera and useful frames are streamed to the remote cloudlet. A typical application scenario is that the UAV detects a facial frame on board and sends that to the cloudlet for recognizing or more accurately detecting. The JITL strategy is based on the ED strategy that uses a cascade filter to distinguish between the ED DNN's true positives and false positives, which can serve to improve the precision of ED. Although UAVs are out for a mission, the cloudlet with more processing power is able to run more accurate DNNs models to identify true positives and false positives. Then, using this information, a small JITL filter is trained and then pushed to the UAVs, instead of training and pushing the original DNN model with high latency. Also, taking the previous case as an example, using the JITL strategy, the UAV can detect a facial frame with increasing accuracy. The RB strategy is designed to allow the cloudlet to request a few previous frames on the UAV, which has been detected as negative frames, but critical for event recognition. This mechanism is particularly useful in the context of activity detection. The CA strategy aims to dynamically choose the optimal filter for a mission. For example, while UAVs are going to search survivors in the ocean after a shipwreck, the results of video analytics in the cloudlet show that the color feature would be better for person detection. Thus, the cloudlet could adjust the model or filter on the UAV to adapt to the scenario. The experiment results based on these four strategies show that a judicious combination of UAV-based processing and edge-based processing could save substantial wireless bandwidth and thus improve scalability, without compromising result accuracy or result latency. Thus, based on this platform, we can easily implement a domain-specific application by providing trained models, such as flame detection model files for fire department applications.

4) Wireless Video Surveillance Platform: In addition, also due to the limitation of network bandwidth of the wireless connection, many edge-based wireless surveillance systems with video analysis are proposed, as pushing video analytics processing on the edge can significantly reduce the bandwidth requirement. For example, Zhang *et al.* [137] proposed and evaluated a novel edge-enabled architecture for wireless video surveillance in which each ECN is connected to one camera to perform simple video analytics functions, and only upload relevant portions of the video to a controller in the cloud. The proposed architecture provides a callback interface for developing its own applications, `frameUtility`, which returns an integer value evaluating the importance of a video frame to that application. Taking people counting application as an example, an implemented `frameUtility` leverages a face detection algorithm to return the number of people in the frame, referring to the value of the frame. Although all ECNs receive queries to count people with a definition of the `frameUtility` function, each ECN calls this function on every frame and sends the returned value to the controller. In addition, multiple ECNs with cameras monitoring the areas form a cluster. The proposed platform effectively fuses the observations from ECNs in a cluster by communicating with each other ECN to further reduce bandwidth. Also, taking the previous application as an example, the `frameUtility` function on two cameras in a cluster returns the same value. Thus, only one video data of them is selected to upload to the controller. Here, two major data fusion strategies are proposed, the basic frame-selection algorithm and the sophisticated frame-selection algorithm. The basic frame-selection algorithm relies on the highest value to select the ECN to upload its images in a cluster with a dynamical adjustment. The sophisticated frame-selection algorithm targets cases where one camera in a cluster sees objects that other cameras miss to select more than one ECN to upload images. The main idea used here is an object reidentification technique. Moreover, the authors proposed an intercluster traffic shaping scheme to avoid network congestion, resulting in maximizing the number of useful objects per second delivered to the controller.

VI. OPEN ISSUES AND FUTURE WORKS

We have described a few public safety applications, underlying video analytics techniques, as well as several potential platforms in previous sections. In this section, we further summarize these issues and bring forward several potential solutions and opportunities worth further research.

A. Edge-Enabled Applications

With the development of special-purpose hardware and custom chips, e.g., AI chips, it enables the computation-intensive AI algorithms to run on the edge devices, e.g., static cameras, as well as various novel applications to be implemented, especially in mobile environments, e.g., body-worn cameras and smart glasses for police. However, novel scenarios and applications bring several new challenges on the architecture. For example, law enforcement officers are equipped with body-worn cameras and are nearby their law enforcement vehicles when they are on duty. Thus, how to design a platform, enabling the vehicle to have a more powerful computing unit as an edge server to perform real-time video analysis for body-worn cameras is still an open question.

B. Algorithm Optimizing

On the one hand, part of video analytics algorithms have a good performance on accuracy, and even exceeded the accuracy of humans, e.g., face recognition. However, the accuracy of existing algorithms should be improved, e.g., person reidentification, and a number of domain-specific algorithms should be developed, e.g., person detection in fireground and criminous activity recognition. In addition, combining multiple features and modals may be potential solutions. For example, multimodal data fusion [149] employed to combine knowledge from edge video and other aspects offer a much more comprehensive and accurate understanding for a decision compared to video-based methods.

On the other hand, although the CNN-based algorithms provide good accuracy, it also takes a lot of computational overhead, resulting in a high latency on resource-limited devices, e.g., cameras. It means that most of the algorithms have to run on the cloud or a powerful device, with an in-time response. Therefore, to enable edge video analytics and reduce the burden of the cloud, how to reduce the computational overhead of CNN-based algorithms is a big challenge. Fortunately, several ideas are promising to deal with this issue, including structure optimization and model compression.

C. Functional Partitioning

In edge video analytics, a key problem is how to obtain an optimized performance by dividing the entire process to perform on different devices, i.e., cameras, edge servers, and the cloud. In current works, most of them are divided at the video analytics levels by experience. For example, Zhang *et al.* [11] divided person recognition into two subprocesses: face detection on the edge and face recognition in the cloud, by exchanging face images instead of video data. It is worthy of attention that Kang *et al.* [150] have evaluated the performance of different layers of several CNN algorithms running on the edge and the cloud, and proved that offloading a part of the CNN workload to the edge and remaindering on the cloud has a better performance than all workload in the cloud. Thus, how to support this type of workload offloading from platform and programming perspectives is a challenge, especially for heterogeneous mobile devices, e.g., smart glasses and body-worn cameras in public safety applications.

D. Dynamic Strategy

Many external factors, e.g., lighting, and internal configurations of cameras, e.g., white balance and sensitometry, often affect imaging of cameras, as well as the accuracy of the video analytics algorithms, e.g., face detection, face recognition, and so on. Thus, preprocessing operations are applied to harmonize these differences between different video data with training data. However, in a city-wide video surveillance system, different cameras usually have different imaging, e.g., with different brightness, and for one camera, the imaging is also changing as time. Therefore, differentiated and dynamical configuring of different cameras as the changes in the external environment are needed to prevent this problem. Moreover, as the law enforcement tasks progress, the environment and mission objectives are changing, which might require different video analytics algorithms as well as DL models for a higher accuracy rate. Thus, designing a flexible architecture to enable dynamically configuring for large-scale video surveillance systems, especially for these systems including mobile devices (e.g., body-worn cameras), remains a challenge.

E. System Operation and Maintenance

According to a study by Seagate Technology LLC [151], 566 petabytes (PB) of data were generated per day by newly installed worldwide video surveillance systems in 2016, and this value is expected to reach 3500 PB by 2023, leading to a few challenges on system operation and maintenance, e.g., finding a fail camera or shortening storage of video data. For instance, a fail camera uninterruptedly uploads useless video data, aggravating the overload in the network and wastes storage in the cloud before the failure is detected by video analytics techniques, e.g., content analysis [152], moving object detection [153], and action analytic [154].

Therefore, an edge-enabled operation and maintenance system is promising to deal with such problems for a large-scale video surveillance system. The key technologies are twofold. First, an edge-enabled failure detection system aims to find the failures in the video surveillance system to reduce the transmission and storage of unworthy video data. Second, an edge-enabled video semantic analysis system can be used to reduce the transmission and storage of unworthiness video data. For example, the video can lower the quality of video while there is no change in the surveillance area.

F. Security & Privacy

Security is important for any system, especially for video surveillance systems. For instance, once a camera is compromised, the attacker can use it as a springboard to attack other devices, the data center, and even other systems in the police department. It means that the edge devices are weak spots in the whole system. The 2016 Dyn cyberattack leveraged a large number of IP cameras to launch a series of distributed-denial-of-service (DDoS) attacks targeting the Domain Name System provider Dyn, resulting in major Internet platforms and services to be unavailable to users in Europe and North America [155]. Various geographically distributed and resource-limited edge devices and servers lack strict security measures, compared with a data center.

Privacy is another problem. The video surveillance systems, especially body-worn cameras, impact police and first responder personnel and how they do their jobs. Therefore, understanding their privacy concerns, even criminal rights protections, is an important challenge. Moreover, the video surveillance systems are usually deployed into city-wide public spaces, and this frequently raises privacy issues of the citizen. The investigation that happened after the Disorder of August 2011 [79], where many images and video clips were provided by the video surveillance, demonstrates the potential power of EVAPS in solving and preventing crimes and acts of terrorism. The video analyzing provides much valuable and useful information and is a compelling need for public safety but might also leak privacy. Thus, when we build EVAPS applications, we also should consider the privacy of the public, especially ethical issues and government regulations. To summarize, we need to better understand how to balance privacy and surveillance requirements. In this case, edge computing-enabled privacy-aware video management systems present as useful and valuable future works, namely, blurring facial regions in videos before uploading.

VII. CONCLUSION

Edge video analytics is motivated by the increasing popularity of edge computing and can be widely adopted in various video surveillance, in terms of static and mobile, to improve the public safety of our daily life. In this paper, we have reviewed recent successful or potential applications of public safety. The reviewed applications show that there is an uneven development in different departments, and edge video analytics is promising to improve public safety. Then, we introduced the general video analytics process and reviewed typical video analytics algorithms used in public safety, followed by a brief discussion, to help understand the benefits of edge video analytics and which parts can be offloaded to the edge. To enable an EVAPS application, a suitable architecture is needed for its public safety application, especially in supporting an optimal and dynamical workload offloading. Thus, we reviewed several video analytics platforms, including domain-specific platforms and general-purpose platforms to show their experience on architecture design. Since the edges are complements of the existing cloud-computing mode, we also reviewed several cloud-based platforms as comparisons. At last, we put forward the challenges and opportunities that are worth working on. We hope this paper gain attention from the community and inspire more research in this direction. ■

REFERENCES

[1] (2017). *First Workshop on Video Analytics in Public Safety*. [Online]. Available: https://www.nist.gov/sites/default/files/documents/2017/01/19/ir_8164.pdf

[2] CV Networking, Cisco Public, San Jose, CA, USA. (2016). *Cisco Global Cloud Index: Forecast and Methodology, 2016–2021 White Paper*. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.html

[3] G. Ananthanarayanan *et al.*, "Real-time video analytics: The killer app for edge computing," *Computer*, vol. 50, no. 10, pp. 58–67, 2017.

[4] BBC News. *In Your Face: China's all-Seeing State—BBC News*. [Online]. Available: https://www.bbc.com/news/av/world-asia-china-42248056/in-your-face-china-s-all-seeing-state

[5] Z. Shao, J. Cai, and Z. Wang, "Smart monitoring cameras driven intelligent processing to big surveillance video data," *IEEE Trans. Big Data*, vol. 4, no. 1, pp. 105–116, Mar. 2018.

[6] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[7] W. Shi and S. Dustdar, "The promise of edge computing," *Computer*, vol. 49, no. 5, pp. 78–81, 2016.

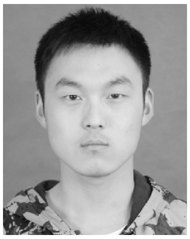
[8] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of

Things," in *Proc. 1st ACM Ed. MCC Workshop Mobile Cloud Comput. (MCC)*, 2012, pp. 13–16.
[9] M. Patel *et al.*, "Mobile-edge computing introductory technical white paper," Mobile-Edge Comput. (MEC) Ind. Initiative, White Paper, 2014.
[10] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct./Dec. 2009.
[11] Q. Zhang, Z. Yu, W. Shi, and H. Zhong, "Demo abstract: EVAPS: Edge video analysis for public safety," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Oct. 2016, pp. 121–122.
[12] Q. Zhang, H. Zhang, J. Wu, and W. Shi, "How edge computing and initial congestion window affect latency of Web-based services: Early experiences with Baidu?" in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Oct. 2018, pp. 393–398.
[13] J. J. Corso *et al.*, "Video analysis for body-worn cameras in law enforcement," *CoRR*, vol. abs/1604.03130, 2016. [Online]. Available: https://arxiv.org/abs/1604.03130
[14] *Here is What Surat City Police did to cut Crime Rate by 27% DATAQUEST*. [Online]. Available: https://www.dqindia.com/here-is-what-surat-city-police-did-to-cut-crime-rate-by-27/
[15] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1624–1639, Dec. 2011.
[16] I. M. Cohen, D. Plecas, and A. V. McCormick, "A report on the utility of the automated licence plate recognition system in British Columbia," School Criminol. Criminal Justice, Univ. College Fraser Valley, Abbotsford, BC, Canada, Tech. Rep., 2013.
[17] R. Charvat, R. Ozburn, S. Bushong, K. Cohen, and M. Kumar, "SIERRA team flight of zephyr UAS at west virginia wild land fire burn," *Infotech Aerosp.*, vol. 2012, p. 2544, Jun. 2012.
[18] M. Quinn. *ETHAN Telemedicine Program Has Prevented 6,000 Unnecessary ED Transports in Houston*. [Online]. Available: https://www.emsworld.com/news/12334469/ethan-telemedicine-program-has-prevented-6-000-unnecessary-ed-transports-in-houston
[19] (2018). *Nvidia Deep Learning Accelerator*. [Online]. Available: http://nvdla.org
[20] Qualcomm. *Qualcomm Announces New Flagship Snapdragon 855 Mobile Platform—A New Decade of 5G, AI, and XR*. [Online]. Available: https://www.qualcomm.com/news/releases/2018/12/05/qualcomm-announces-new-flagship-snapdragon-855-mobile-platform-new-decade
[21] Q. Zhang, Q. Zhang, W. Shi, and H. Zhong, "Distributed collaborative execution on the edges and its application to AMBER alerts," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3580–3593, Oct. 2018.
[22] R. Arroyo, J. J. Yebes, L. M. Bergasa, I. G. Daza, and J. Almazán, "Expert video-surveillance system for real-time detection of suspicious behaviors in shopping malls," *Expert Syst. Appl.*, vol. 42, no. 21, pp. 7991–8005, 2015.
[23] B. Jacobs. *Chicago Man Nearly Strikes Hammond Police Chief While Fleeing Traffic Stop, Video Shows*. [Online]. Available: http://www.chicagotribune.com/suburbs/post-tribune/crime/ct-ptb-hammond-fleeing-suspect-st-0421-story.html
[24] T. Couvillion. *License Plate Reader Spots Stolen Car, Man Arrested After Brief Pursuit*. [Online]. Available: http://www.wbrz.com/news/license-plate-reader-spots-stolen-car-man-arrested-after-brief-pursuit
[25] Y. K. Ki and D. Y. Lee, "A traffic accident recording and reporting model at intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 188–194, Jun. 2007.
[26] C. Yuan, Z. Liu, and Y. Zhang, "Fire detection using infrared images for UAV-based forest fire surveillance," in *Proc. Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, 2017, pp. 567–572.
[27] C. Yuan, Z. Liu, and Y. Zhang, "UAV-based forest fire detection and tracking using image processing techniques," in *Proc. Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, 2015, pp. 639–643.
[28] S. Ma, Y. Zhang, J. Xin, Y. Yi, D. Liu, and H. Liu, "An early forest fire detection method based on unmanned aerial vehicle vision," in *Proc. Chin. Control Decis. Conf.*, 2018, pp. 6344–6349.
[29] *THERMITE RS1-T3 AND RS2-T2—Thermite Fire Fighting Robot*. [Online]. Available: http://www.roboticfirefighters.com/
[30] X. Wu, R. Dunne, Z. Yu, and W. Shi, "STREMS: A smart real-time solution toward enhancing EMS prehospital quality," in *Proc. IEEE/ACM Int. Conf. Connected Health, Appl., Syst. Eng. Technol. (CHASE)*, Jul. 2017, pp. 365–372.
[31] C. Rougier, E. Auvinet, J. Rousseau, M. Mignotte, and J. Meunier, "Fall detection from depth map video sequences," in *Proc. Int. Conf. Smart Homes Health Telematics*, 2011, pp. 121–128.
[32] T. Arikuma and Y. Mochizuki, "Intelligent multimedia surveillance system for safer cities," *APSIPA Trans. Signal Inf. Process.*, vol. 5, p. e3, Mar. 2016.
[33] P. Wang, K. W. Woo, and S. K. Koh, "Special Issue on Solutions for Society-Creating a Safer and More Secure Society For a safer and more secure life Building a Safer City in Singapore," Tech. Rep. [Online]. Available: https://in.nec.com/en_IN/images/140115.pdf
[34] NEC. *Face Recognition*. [Online]. Available: https://www.nec.com/en/global/solutions/safety/facerecognition/NeoFace Watch.html
[35] N. H. Motlagh, M. Bagaa, and T. Taleb, "UAV-based IoT platform: A crowd surveillance use case," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 128–134, Feb. 2017.
[36] J. Wang *et al.*, "Bandwidth-efficient live video analytics for drones via edge computing," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Oct. 2018, pp. 159–173.
[37] A. Sanchez, P. D. Suarez, A. Conci, and E. Nunes, "Video-based distance traffic analysis: Application to vehicle tracking and counting," *Comput. Sci. Eng.*, vol. 13, no. 3, pp. 38–45, 2011.
[38] A. M. Bayen and S. Sastry, "Intelligent transportation systems and infrastructure a series of briefs for smart investments," Tech. Rep., 2017. [Online]. Available: https://escholarship.org/uc/item/77b9h8vk
[39] S. Ozbay and E. Erçelebi, "Automatic vehicle identification by plate recognition," *World Acad. Sci., Eng. Technol.*, vol. 9, no. 41, pp. 222–225, 2005.
[40] International Association of Chiefs of Police. (2017). *News Stories About Law Enforcement ALPR Successes September 2017—September, 2018*. [Online]. Available: https://www.theiacp.org/sites/default/files/ALPR%20Success%20News%20Stories%202018.pdf
[41] E. Mathews. *Maine Man Bound Over for Trial in Connection With Tescott Man's Murder*. [Online]. Available: http://www.salina.com/news/20180129/maine-man-bound-over-for-trial-in-connection-with-tescott-mans-murder
[42] F. Alves, M. Ferreira, and C. Santos, "Vision based automatic traffic condition interpretation," in *Proc. 8th IEEE Int. Conf. Ind. Inform.*, Jul. 2010, pp. 549–556.
[43] J. Wu, Z.-M. Cui, J.-M. Chen, and G.-M. Zhang, "A survey on video-based vehicle behavior analysis algorithms," *J. Multimedia*, vol. 7, no. 3, pp. 223–230, 2012.
[44] S. Kaplan, M. A. Guvensan, A. G. Yavuz, and Y. Karalurt, "Driver behavior analysis for safe driving: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 3017–3032, Dec. 2015.
[45] H. Guo, H. Lin, S. Zhang, and S. Li, "Image-based seat belt detection," in *Proc. IEEE Int. Conf. Veh. Electron. Saf.*, Jul. 2011, pp. 161–164.
[46] K. Seshadri, F. Juefei-Xu, D. K. Pal, M. Savvides, and C. P. Thor, "Driver cell phone usage detection on Strategic Highway Research Program (SHRP2) face view videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2015, pp. 35–43.
[47] L. Liu, X. Zhang, M. Qiao, and W. Shi, "SafeShareRide: Edge-based attack detection in ridesharing services," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Oct. 2018, pp. 1–13.
[48] S. R. Kukreti, M. Kumar, and K. Cohen, "Genetic fuzzy based target geo-localization using unmanned aerial systems for firefighting applications," in *Proc. AIAA Inf. Syst.-AIAA Infotech Aerosp.*, 2018, p. 2136.
[49] J.-H. Kim, J. W. Starr, and B. Y. Lattimer, "Firefighting robot stereo infrared vision and radar sensor fusion for imaging through smoke," *Fire Technol.*, vol. 51, no. 4, pp. 823–845, 2015.
[50] J.-H. Kim, Y. Sung, and B. Y. Lattimer, "Bayesian estimation based real-time fire-heading in smoke-filled indoor environments using thermal imagery," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 5231–5236.
[51] J. G. McNeil and B. Y. Lattimer, "Robotic fire suppression through autonomous feedback control," *Fire Technol.*, vol. 53, no. 3, pp. 1171–1199, 2017.
[52] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K.-S. Kwak, "The Internet of Things for health care: A comprehensive survey," *IEEE Access*, vol. 3, pp. 678–708, Jun. 2015.
[53] S. I. Lakkis and M. Elshakankiri, "IoT based emergency and operational services in medical care systems," in *Proc. IEEE Internet Things Bus. Models, Users, Netw.*, 2017, pp. 1–5.
[54] R. Schaer, T. Melly, H. Müller, and A. Widmer, "Using smart glasses in medical emergency situations, a qualitative pilot study," in *Proc. IEEE Wireless Health (WH)*, Oct. 2016, pp. 54–58.
[55] A. Widmer, R. Schaer, D. Markonis, and H. Müller, "Facilitating medical information search using Google Glass connected to a content-based medical image retrieval system," in *Proc. 36th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2014, pp. 4507–4510.
[56] O. J. Muensterer, M. Lacher, C. Zoeller, M. Bronstein, and J. Kübler, "Google Glass in pediatric surgery: An exploratory study," *Int. J. Surg.*, vol. 12, no. 4, pp. 281–289, 2014.
[57] U.-V. Albrecht *et al.*, "Google Glass for documentation of medical findings: Evaluation in forensic medicine," *J. Med. Internet Res.*, vol. 16, no. 2, p. e53, 2014.
[58] A. Widmer and H. Müller, "Using Google Glass to enhance pre-hospital care," *Swiss Med. Informat.*, vol. 30, pp. 1–4, Sep. 2014.
[59] Z. Zhou *et al.*, "A real-time system for in-home activity monitoring of elders," in *Proc. Elect. Comput. Eng. Publications (MU)*, 2009, pp. 6115–6118.
[60] P. Perego, M. Tarabini, M. Bocciolone, and G. Andreoni, "SMARTA: Smart ambiente and wearable home monitoring for elderly," in *Proc. Int. Internet Things Summit*. Cham, Switzerland: Springer, 2015, pp. 502–507.
[61] L. Ren, Q. Zhang, W. Shi, and Y. Peng, "Edge-based personal computing services: Fall detection as a pilot study," *Computing*, vol. 101, no. 8, pp. 1199–1223, Jan. 2019. doi: 10.1007/s00607-018-00697-x.
[62] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "Robust video surveillance for fall detection based on human shape deformation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 5, pp. 611–622, May 2011.
[63] G. Mastorakis and D. Makris, "Fall detection system using Kinect's infrared sensor," *J. Real-Time Image Process.*, vol. 9, no. 4, pp. 635–646, 2014.
[64] E. E. Stone and M. Skubic, "Fall detection in homes of older adults using the Microsoft Kinect," *IEEE J. Biomed. Health Informat.*, vol. 19, no. 1, pp. 290–301, Jan. 2015.
[65] (2017). *H.264/MPEG-4 AVC*. [Online]. Available: https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC
[66] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, vol. 2, 2004,

pp. 28–31.
[67] A. Verri, S. Uras, and E. De Micheli, "Motion segmentation from optical flow," in *Proc. Alvey Vis. Conf.*, 1989, pp. 1–6.
[68] J. Kim, X. Wang, H. Wang, C. Zhu, and D. Kim, "Fast moving object detection with non-stationary background," *Multimedia Tools Appl.*, vol. 67, no. 1, pp. 311–335, Nov. 2013.
[69] J. Kim, G. Ye, and D. Kim, "Moving object detection under free-moving camera," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 4669–4672.
[70] M. Seki, H. Fujiwara, and K. Sumi, "A robust background subtraction method for changing background," in *Proc. 5th IEEE Workshop Appl. Comput. Vis.*, Dec. 2000, pp. 207–213.
[71] S. Agarwal, A. Awan, and D. Roth, "Learning to detect objects in images via a sparse, part-based representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1475–1490, Nov. 2004.
[72] Y. Freund, "Boosting a weak learning algorithm by majority," *Inf. Comput.*, vol. 121, no. 2, pp. 256–285, Sep. 1995.
[73] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2002, pp. 661–675.
[74] M. Yokoyama and T. Poggio, "A contour-based moving object detection and tracking," in *Proc. IEEE Int. Workshop Vis. Surveill. Perform. Eval. Tracking Surveill.*, Oct. 2005, pp. 271–276.
[75] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 850–865.
[76] D. Held, S. Thrun, and S. Savarese, "Learning to track at 100 FPS with deep regression networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 749–765.
[77] D. Zhang, H. Maei, X. Wang, and Y.-F. Wang, "Deep reinforcement learning for visual object tracking in videos," 2017, *arXiv:1701.08936*. [Online]. Available: https://arxiv.org/abs/1701.08936
[78] M. Kristan *et al.*, "The visual object tracking VOT2015 challenge results," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV) Workshops*, Dec. 2015, pp. 564–586.
[79] M. P. Service, "Four days in August: Strategic review into the disorder of August 2011," Tech. Rep., 2012.
[80] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 1, pp. 23–38, Jan. 1998.
[81] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Dec. 2001, p. 1.
[82] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5325–5334.
[83] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.
[84] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.
[85] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. ICCV*, Sep. 2015, pp. 1440–1448.
[86] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst. (NIPS)*. Cambridge, MA, USA: MIT Press, 2015, pp. 91–99.
[87] H. Wang, Z. Li, X. Ji, and Y. Wang, "Face R-CNN," *CoRR*, vol. abs/1706.01061, 2017. [Online]. Available: https://arxiv.org/abs/1706.01061
[88] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognit. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.
[89] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.
[90] G. Guo, S. Z. Li, and K. Chan, "Face recognition by support vector machines," in *Proc. 4th IEEE Int. Conf. Autom. Face Gesture Recognit.*, Mar. 2000, pp. 196–201.
[91] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun, "Bayesian face revisited: A joint formulation," in *Computer Vision—ECCV*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Germany: Springer, 2012, pp. 566–579.
[92] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.
[93] D. Chen, X. Cao, F. Wen, and J. Sun, "Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 3025–3032.
[94] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 1701–1708.
[95] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.
[96] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 1891–1898.
[97] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 1988–1996.
[98] Y. Sun, D. Liang, X. Wang, and X. Tang, "DeepID3: Face recognition with very deep neural networks," *CoRR*, vol. abs/1502.00873, 2015. [Online]. Available: https://arxiv.org/abs/1502.00873
[99] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, D. Lin, and X. Tang, "Temporal action detection with structured segment networks," *CoRR*, vol. abs/1704.06228, 2017. [Online]. Available: https://arxiv.org/abs/1704.06228
[100] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2360–2367.
[101] (2017). *OpenALPR*. [Online]. Available: https://www.openalpr.com/
[102] G. Sharma, "Performance analysis of vehicle number plate recognition system using template matching techniques," *J. Inf. Technol. Softw. Eng.*, vol. 8, no. 2, p. 232, 2018.
[103] A. Meng, W. Yang, Z. Xu, H. Huang, L. Huang, and C. Ying, "A robust and efficient method for license plate recognition," in *Proc. IEEE 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 1713–1718.
[104] Z. Selmi, M. Ben Halima, and A. M. Alimi, "Deep learning system for automatic license plate detection and recognition," in *Proc. IEEE 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, vol. 1, Nov. 2017, pp. 1132–1138.
[105] L.-W. Tsai, J.-W. Hsieh, and K.-C. Fan, "Vehicle detection using normalized color and edge map," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 850–864, Mar. 2007.
[106] G. Yan, M. Yu, Y. Yu, and L. Fan, "Real-time vehicle detection using histograms of oriented gradients and AdaBoost classification," *Optik*, vol. 127, no. 19, pp. 7941–7951, 2016.
[107] B.-F. Wu and J.-H. Juang, "Adaptive vehicle detector approach for complex environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 817–827, Jun. 2012.
[108] Y. Li, M. J. Er, and D. Shen, "A novel approach for vehicle detection using an AND–OR-graph-based multiscale model," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2284–2289, Aug. 2015.
[109] Y. Li and F.-Y. Wang, "Vehicle detection based on AND–OR graph and hybrid image templates for complex urban traffic conditions," *Transp. Res. C, Emerg. Technol.*, vol. 51, pp. 19–28, Feb. 2015.
[110] R. Wang and S. M. Lukic, "Review of driving conditions prediction and driving style recognition based control algorithms for hybrid electric vehicles," in *Proc. IEEE Vehicle Power Propuls. Conf. (VPPC)*, Sep. 2011, pp. 1–7.
[111] S. Tang and H. Gao, "Traffic-incident detection-algorithm based on nonparametric regression," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 38–42, Mar. 2005.
[112] Y. Artan, O. Bulan, R. P. Loce, and P. Paul, "Passenger compartment violation detection in HOV/HOT lanes," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 395–405, Feb. 2016.
[113] J.-H. Kim and B. Y. Lattimer, "Real-time probabilistic classification of fire and smoke using thermal imagery for intelligent firefighting robot," *Fire Saf. J.*, vol. 72, pp. 40–49, Feb. 2015.
[114] Y. Jia, J. Yuan, J. Wang, J. Fang, Q. Zhang, and Y. Zhang, "A saliency-based method for early smoke detection in video sequences," *Fire Technol.*, vol. 52, no. 5, pp. 1271–1292, 2016.
[115] J.-H. Kim, S. Jo, and B. Y. Lattimer, "Feature selection for intelligent firefighting robot classification of fire, smoke, and thermal reflections using thermal infrared images," *J. Sensors*, vol. 2016, Aug. 2016, Art. no. 8410731.
[116] Y. H. Habiboğlu, O. Günay, and A. E. Çetin, "Covariance matrix-based fire and flame detection method in video," *Mach. Vis. Appl.*, vol. 23, no. 6, pp. 1103–1113, 2012.
[117] K. Dimitropoulos, P. Barmpoutis, and N. Grammalidis, "Spatio-temporal flame modeling and dynamic texture analysis for automatic video-based fire detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 2, pp. 339–351, Feb. 2015.
[118] C. Yuan, Z. Liu, and Y. Zhang, "Vision-based forest fire detection in aerial images for firefighting using UAVs," in *Proc. Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, 2016, pp. 1200–1205.
[119] F. Yuan, "An integrated fire detection and suppression system based on widely available video surveillance," *Mach. Vis. Appl.*, vol. 21, no. 6, pp. 941–948, 2010.
[120] S. Frizzi, R. Kaabi, M. Bouchouicha, J.-M. Ginoux, E. Moreau, and F. Fnaiech, "Convolutional neural network for video fire and smoke detection," in *Proc. 42nd Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2016, pp. 877–882.
[121] K. Muhammad, J. Ahmad, and S. W. Baik, "Early fire detection using convolutional neural networks during surveillance for effective disaster management," *Neurocomputing*, vol. 288, pp. 30–42, May 2018.
[122] C. Tao, J. Zhang, and P. Wang, "Smoke detection based on deep convolutional neural networks," in *Proc. Int. Conf. Ind. Inform.-Comput. Technol., Intell. Technol., Ind. Inf. Integr.*, 2016, pp. 150–153.
[123] Z. Yin, B. Wan, F. Yuan, X. Xia, and J. Shi, "A deep normalization and convolutional neural network for image smoke detection," *IEEE Access*, vol. 5, pp. 18429–18438, 2017.
[124] M. Ulrich, T. Hess, S. Abdulatif, and B. Yang, "Person recognition based on micro-Doppler and thermal infrared camera fusion for firefighting," in *Proc. 21st Int. Conf. Inf. Fusion (FUSION)*, 2018, pp. 919–926.
[125] T. Lee and A. Mihailidis, "An intelligent emergency response system: Preliminary development and testing of automated fall detection," *J. Telemedicine Telecare*, vol. 11, no. 4, pp. 194–198, 2005.
[126] K. Sehairi, F. Chouireb, and J. Meunier, "Elderly fall detection system based on multiple shape features and motion analysis," in *Proc. Int. Conf. Intell. Syst. Comput. Vis. (ISCV)*, 2018, pp. 1–8.
[127] S. Wang, L. Chen, Z. Zhou, X. Sun, and J. Dong, "Human fall detection in surveillance video based

on PCANet," *Multimedia Tools Appl.*, vol. 75, no. 19, pp. 11603–11613, 2016.
[128] L. Alhimale, H. Zedan, and A. Al-Bayatti, "The implementation of an intelligent and video-based fall detection system using a neural network," *Appl. Soft Comput.*, vol. 18, pp. 59–69, May 2014.
[129] Q. Zhang, Q. Zhang, W. Shi, and H. Zhong, "Firework: Data processing and sharing for hybrid cloud-edge analytics," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 9, pp. 2004–2017, Sep. 2018.
[130] Y.-L. Tian, L. Brown, A. Hampapur, M. Lu, A. Senior, and C.-F. Shu, "IBM smart surveillance system (S3): Event based video surveillance system with an open and extensible framework," *Mach. Vis. Appl.*, vol. 19, nos. 5–6, pp. 315–327, 2008.
[131] A. Anjum, T. Abdullah, M. Tariq, Y. Baltaci, and N. Antonopoulos, "Video stream analysis in clouds: An object detection and classification framework for high performance video analytics," *IEEE Trans. Cloud Comput.*, to be published.
[132] P. Hu, H. Ning, T. Qiu, Y. Zhang, and X. Lou, "Fog computing based face identification and resolution scheme in Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 1910–1920, Aug. 2017.
[133] X. Wu, R. Dunne, Q. Zhang, and W. Shi, "Edge computing enabled smart firefighting: Opportunities and challenges," in *Proc. 5th ACM/IEEE Workshop Hot Topics Web Syst. Technol. (HotWeb)*, Oct. 2017, pp. 11-1–11-6.
[134] N. Kalatzis, M. Avgeris, D. Dechouniotis, K. Papadakis-Vlachopapadopoulos, I. Roussaki, and S. Papavassiliou, "Edge computing in IoT ecosystems for UAV-enabled early fire detection," in *Proc. IEEE Int. Conf. Smart Comput. (SMARTCOMP)*, Jun. 2018, pp. 106–114.
[135] T. Elgamal, B. Chen, and K. Nahrstedt, "Teleconsultant: Communication and analysis of wearable videos in emergency medical environment," in *Proc. ACM Multimedia Conf.*, 2017, pp. 1241–1242.
[136] S. Yi, Z. Hao, Q. Zhang, Q. Zhang, W. Shi, and Q. Li, "LAVEA: Latency-aware video analytics on edge computing platform," in *Proc. 2nd ACM/IEEE Symp. Edge Comput. (SEC)*, Oct. 2017, pp. 15-1–15-13.
[137] T. Zhang, A. Chowdhery, P. Bahl, K. Jamieson, and S. Banerjee, "The design and implementation of a wireless video surveillance system," in *Proc. ACM 21st Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2015, pp. 426–438.
[138] Google. *Cloud Video Intelligence*. [Online]. Available: https://cloud.google.com/video-intelligence/
[139] Huawei. *Video Cloud Solution—Huawei Solutions*. [Online]. Available: https://e.huawei.com/hk/solutions/business-needs/data-center/storage/video-storage
[140] A. Prati, R. Vezzani, M. Fornaciari, and R. Cucchiara, "Intelligent video surveillance as a service," in *Intelligent Multimedia Surveillance*. Berlin, Germany: Springer, 2013, pp. 1–16.
[141] C. Luo, J. Nightingale, E. Asemota, and C. Grecos, "A UAV-cloud system for disaster sensing applications," in *Proc. IEEE 81st Veh. Technol. Conf. (VTC Spring)*, May 2015, pp. 1–5.
[142] X. Wang, A. Chowdhery, and M. Chiang, "Networked drone cameras for sports streaming," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2017, pp. 308–318.
[143] L. Merino, F. Caballero, J. R. Martínez-de-Dios, I. Maza, and A. Ollero, "An unmanned aircraft system for automatic forest fire monitoring and measurement," *J. Intell. Robot. Syst.*, vol. 65, nos. 1–4, pp. 533–548, 2012.
[144] E. Pastor, C. Barrado, P. Royo, E. Santamaria, J. Lopez, and E. Salami, "Architecture for a helicopter-based unmanned aerial systems wildfire surveillance system," *Geocarto Int.*, vol. 26, no. 2, pp. 113–131, 2011.
[145] H. Zhang, G. Ananthanarayanan, P. Bodik, M. Philipose, P. Bahl, and M. J. Freedman, "Live video analytics at scale with approximation and delay-tolerance," in *Proc. 14th USENIX Symp. Netw. Syst. Des. Implement. (NSDI)*, 2017, pp. 377–392.
[146] J. Jiang, G. Ananthanarayanan, P. Bodik, S. Sen, and I. Stoica, "Chameleon: Scalable adaptation of video analytics," in *Proc. Conf. ACM Special Interest Group Data Commun.*, 2018, pp. 253–266.
[147] C.-C. Hung *et al.*, "VideoEdge: Processing camera streams using hierarchical clusters," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Oct. 2018, pp. 115–131.
[148] S. Qazi, A. S. Siddiqui, and A. I. Wagan, "UAV based real time video surveillance over 4G LTE," in *Proc. Int. Conf. Open Source Syst. Technol. (ICOSST)*, 2015, pp. 141–145.
[149] D. Lahat, T. Adali, and C. Jutten, "Multimodal data fusion: An overview of methods, challenges, and prospects," *Proc. IEEE*, vol. 103, no. 9, pp. 1449–1477, Sep. 2015.
[150] Y. Kang *et al.*, "Neurosurgeon: Collaborative intelligence between the cloud and mobile edge," in *Proc. ACM 22nd Int. Conf. Archit. Support Program. Lang. Oper. Syst.*, 2017, pp. 615–629.
[151] Market Report Future. *Video Surveillance Storage Market Research Report—Forecast to 2023*. [Online]. Available: https://www.marketresearchfuture.com/reports/video-surveillance-storage-market-5848
[152] N. Dimitrova, H.-J. Zhang, B. Shahraray, I. Sezan, T. Huang, and A. Zakhor, "Applications of video-content analysis and retrieval," *IEEE MultimediaMag.*, vol. 9, no. 3, pp. 42–55, Jul. 2002.
[153] K. A. Joshi and D. G. Thakore, "A survey on moving object detection and tracking in video surveillance system," *Int. J. Soft Comput. Eng.*, vol. 2, no. 3, pp. 44–48, 2012.
[154] B. Yogameena and K. S. Priya, "Synoptic video based human crowd behavior analysis for forensic video surveillance," in *Proc. Int. Conf. Adv. Pattern Recognit.*, 2015, pp. 1–6.
[155] *Large DDoS Attacks Cause Outages at Twitter, Spotify, and Other Sites | TechCrunch*. [Online]. Available: https://techcrunch.com/2016/10/21/many-sites-including-twitter-and-spotify-suffering-outage/

ABOUT THE AUTHORS



Qingyang Zhang (Student Member, IEEE) received the B.Eng. degree in computer science and technology from Anhui University, Hefei, China, in 2014, where he is currently working toward the Ph.D. degree.

His current research interests include edge computing, computer systems, and security.

Xiaopei Wu received the B.Sc. degree in computer science from Luoyang Normal University, Luoyang, China, in 2005, and the M.Sc. degree and the Ph.D. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, China, in 2007 and 2012, respectively.

From 2016 to 2018, she was a Research Scientist with Wayne State University, Detroit, MI, USA. Her current research interests include edge computing, big data, and natural language processing.

Hui Sun received the Ph.D. degree from Huazhong University Science and Technology, Wuhan, China, in 2012.

He is currently an Assistant Professor with the School of Computer Science and Technology, Anhui University, Hefei, China. His current research interests include computer systems, edge computing, performance evaluation, and nonvolatile memory-based storage systems.

Hong Zhong received the B.S. degree in applied mathematics from Anhui University, Hefei, China, in 1986, and the Ph.D. degree in computer science and technology from the University of Science and Technology of China, Hefei, in 2005.

She is currently a Professor and a Ph.D. Advisor with Anhui University. Her current research interests include applied cryptography, Internet of Things (IoT) security, vehicular *ad hoc* network, and software-defined networking (SDN).