A Multi-Modal Distributed Real-Time IoT System for Urban Traffic Control

Zeba Khanam
BT Security Research, Adastral Park, UK

Vejey Pradeep Suresh Achari
Keele University, Keele, UK

Issam Boukhennoufa
University of Essex, Colchester, UK

Anish Jindal
Durham University, Durham, UK

Amit Kumar Singh
University of Essex, Colchester, UK

Abstract

Traffic congestion is one of the growing urban problem with associated problems like fuel wastage, loss of lives, and slow productivity. The existing traffic system uses programming logic control (PLC) with round-robin scheduling algorithm. Recent works have proposed IoT-based frameworks that use traffic density of each lane to control traffic movement, but they suffer from low accuracy due to lack of emergency vehicle image datasets for training deep neural networks. In this paper, we propose a novel distributed IoT framework that is based on two observations. The first observation is major structural changes to road are rare. This observation is exploited by proposing a novel two stage vehicle detector that is able to achieve 77% vehicle detection accuracy on UA-DETRAC dataset. The second observation is emergency vehicle have distinct siren sound that is detected using a novel acoustic detection algorithm on an edge device. The proposed system is able to detect emergency vehicles with an average accuracy of 99.4%.

2012 ACM Subject Classification Computer systems organization → Real-time system architecture

Keywords and phrases Vehicle Detection, Deep Neural Network, Traffic Control, Edge Computing, Emergency Vehicle Detection, Sliding Window

Digital Object Identifier 10.4230/OASIcs.NG-RES.2024.2

Category Invited Paper

1 Introduction

One of the major problems plaguing urban cities is traffic congestion. This problem stems from lopsided growth in road infrastructure in comparison to traffic volume. The visible devastating effects on travelers and urban cities in general are increased global carbon footprint, low productivity, fuel wastage; resource depletion, loss of lives, and economic downfall. To address this pressing issue, governments have invested heavily in infrastructure upgradation with complex civil construction of bridges, roads, and new lanes addition. However, this solution has further complicated the issue. Big cities like London that have been implementing this solution are currently reeling under issues like urban sprawl, pollution, and stalling [8]. As the number of on-road vehicles is projected to increase manifold, this problem is going to intensify. With the global emphasis to cut the scope 1 emission due to road transportation for a sustainable future, an intelligent urban traffic system is the need of the hour.

Most of the cities deploy a traditional traffic control system that uses an array of programming logic controllers (PLC) and a round-robin scheduling algorithm to allow traffic for each lane to pass in a circular fashion [7]. However, few pilot studies have been conducted on the deployment of Internet of Things (IoT)-driven intelligent traffic control systems. Most of these studies have been part of the deployment of connected and automated vehicles and self-driving cars [16, 14]. Given the uncertainty around the future of self-driving cars and the urgency of decarbonization targets, there is a need for an IoT system for current on-road vehicles. Few recent IoT frameworks have used recent technologies like infrared cameras and GPS [3], RFID [6], Bluetooth and Zigbee [15]. However, these technologies require the deployment of an array of sensors in vehicles. These solutions require large investment and energy to revamp the existing vehicles and the accuracy of detection of traffic density estimation is not substantially high due to their susceptibility to noise from the environment [12].

To overcome the aforementioned problems, we proposed a novel framework, IoT based Intelligent Urban Traffic System (I^2UTS) for traffic light control that leveraged the existing CCTV network for designing the IoT-based framework for traffic light control [1]. CCTV videos have proven to be efficient and economical in past. There are several works that have explored the potential of CCTV camera networks as input sensors for solving traffic problems like predicting accidents, monitoring spatio-temporal behavior of pedestrians, and the detection of firearms and knives [4]. I^2UTS framework used CCTV video and state-of-the-art CNN to estimate the traffic density. Traffic density was a major input to control traffic lights. To address the privacy concern and computational resource requirements, Yolo v3 with a darknet backbone was deployed over edge device, Raspberry Pi, alongside our novel scheduling algorithm. Even though, I^2UTS was able to achieve 68.10% vehicle detection accuracy, it inherited problems associated with end-to-end convolutional neural networks (CNNs) that rendered practical difficulties in real-time traffic network analysis.

The first major problem associated with I^2UTS was the inability to find datasets with emergency vehicles. One of the novel contributions of our previous research was to account for the different traffic signal times incase emergency vehicles like police cars, firefighter trucks, and ambulances are part of the traffic scheduling algorithm. Since, the proposed CNNs Yolo v3- Efficient Net is dependent on labeled data, the availability of labeled dataset of the emergency vehicle was a challenge. Finding such a dataset is cumbersome given the rare occurrence of emergency vehicles in traffic conditions. The proposal of emergency vehicle datasets has been an area of research. Researchers have turned to various resources like google search alongside manual annotation of 1500 images, manual filtering of Kaggle dataset images, and youtube streams [13]. Given the amalgamation of various sources of image acquisition, these datasets suffer from large viewpoint variations, which renders them unsuitable for our IoT framework where the input sensor, CCTV camera, has a fixed viewpoint. Apart from viewpoint variation, another problem in the detection of emergency vehicles is weather variation. Inclement weather conditions present in the images often decrease vehicle detection accuracy significantly [5]. Since, the speed of non-emergency vehicles is comparatively slow, their occurrence on CCTV images for a fixed time frame is higher. This increases their detection accuracy alongside the availability of a large set of labeled images in varying weather as a training input for YOLO V3. These problems clearly illustrate the inhibition of using RGB cameras for emergency vehicles. This serves as a major motivation to re-investigate our previous work I^2UTS for emergency vehicle detection. Emergency vehicles all over the world use loud-noise making noise, sirens, to alert the traffic of passage. In this work, we propose a multi-modal distributed IoT framework that detects the distinguished sound property of emergency vehicles alongside the image-based traffic density estimation.

The second major problem associated with I^2UTS is the high variance in mean average precision (mAP) across different classes of vehicle detectors. Although, the major outlier was the class *"van"* where mAP was 37.49%. Further, False Discovery Rate (FDR) was 63.23%, which clearly indicates that the proposed YOLOV3-Effecient net has high false positive (FP) in comparison to true positive (TP) for *"van"*. YOLO is a preferred single-stage object detector in comparison to its counterparts two-stage detectors like RCNN due to its faster performance on edge devices. However, YOLO struggles to localize smaller objects/vehicles and identify the optimum number of clusters. This is the main reason that I^2UTS has a high misclassified bus stops as vans due to similar features. Single-stage detectors classify and localize objects in a single shot using dense sampling whereas two-stage detector consists of an additional preliminary stage of region proposal. The region proposal stage indeed increases the performance of object detector but are computationally expensive. However, the CCTV camera on road is fixed and viewpoint variation in the images captured is hardly possible. Changes in road infrastructure are also minimal. Considering, this advantage, we propose a novel two-stage detector road-based YOLO (*"R-YOLO"*) that confines the search of vehicles to the road with an increased performance accuracy comparable to two-stage object detector(like RCNN) and low computational resource usage like single stage detector (like YOLO).

The remainder of the paper is organised as follows. Section II describes the proposed IoT framework. The experimental evaluation and results of the proposed distributed system is detailed in Section III. Finally, the conclusion is presented in Section IV.

2 Proposed Distributed IoT framework

The proposed distributed IoT based urban traffic management system contains two edge parts: 1) S_1: Vehicle Detector that uses Deep Neural Network, R-CNN. 2) S_2: Emergency Vehicle Detector that uses acoustic sliding window approach to detect siren's of emergency vehicle.

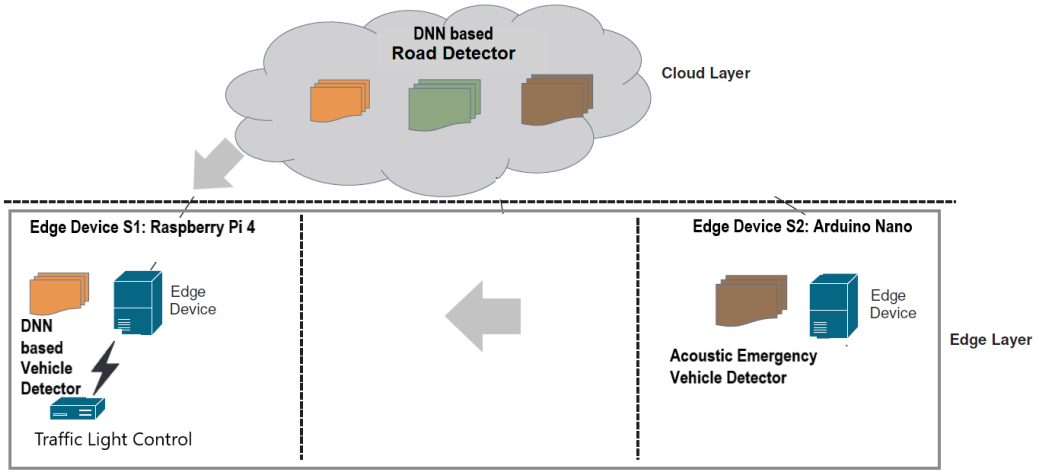


Figure 1 A multi-modal IoT Distributed Framework.

2.1 Edge Devices and Sensors

The previous works deployed NVIDIA Jetson TX2 that is a computationally powerful edge device with the dual support of CPU and GPU [2]. However, the cost of Jetson Nano is 24 times more than average cost of other edge devices. Given the economic feasibility contraint of the system, we use Raspberry PI 4 as the edge device S_1 for vision algorithms with moderate memory of 4 GB and powerful Quad core cortex-A72 (Arm-8) 1.5GHz processor.

The acoustics emergency vehicle detector uses Arduino Nano as Edge device S_2. Arduino Nano is an open source micro-controller based on ATmega328P architecture. With a flash memory of 32 KB, 16 Analog Pins and 22 I/O pins, SEN0232 noise meter is used. SEN0232 uses an instrument circuit and a low noise microphone, with a measuring decibel value ranges from 30dBA to 130dBA, accurately measuring noise level of the surrounding environment.

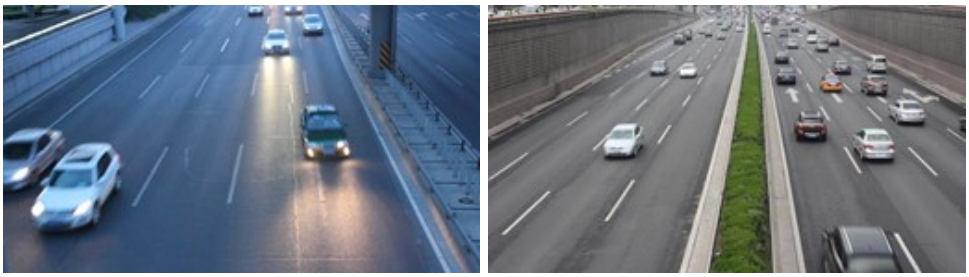
2.2 System Overview

A higher level overview of the distributed system is as follows.

1. **Cloud Layer: Road Detector**
   a. Train Faster-RCNN network on cloud to detect the vehicles.
   b. Detect the object road and Estimate the road mask.
   c. Train the YOLO v3 network with Efficient net as a backbone on cloud to detect the road.
   d. Transfer the trained weights to the Faster RCNN deployed on the edge device S_1 for testing.
   e. Transfer road mask to the edge device S_2 for road extraction.
2. **Edge S_2: Acoustic Emergency Vehicle Detector**
   a. Detect Siren Sound from continuous noise level monitoring.
   b. Send the emergency vehicle trigger to Edge Device S_1.
3. **Edge S_1: Vehicle Detector**
   a. Estimate the traffic density of each road lane by detecting vehicles using road mask and YOLOv3 vehcile detector.
   b. Use the parameters generated in previous steps as an input to the proposed traffic control algorithm.

2.2.1 Dataset

Many prior studies have utilized the KITTI and COCO datasets to train the YOLOv3 network. However, a more recent dataset, UA-DETRAC [18], developed by the University of Albany for object detection and tracking, is more profound as a benchmark dataset for real-world multi-object tracking. This dataset comprises traffic camera videos, recorded at 25 fps over a span of 10 hours, with a resolution of 960 x 540 pixels. The footage originates from 24 distinct locations in Beijing and Tianjin, China, offering a diverse and challenging environment. UA-DETRAC includes approximately 1.21 million labeled bounding boxes representing 8250 vehicles across four classes: car, van, bus, road and others. Figure 2 illustrates some CCTV images from the dataset. An additional strength of utilizing UA-DETRAC is its representation of multi-class weather conditions and variations in day and night illumination. This dataset is used for both the purposes-road and vehicle detection. For both the tasks, the dataset is divided into training, validation, and testing sets following a 70:15:15 ratio, respectively.



(a) (b)

Figure 2 Sample CCTV images from UA-DETRAC [18].

2.2.2 DNN for Road Detection

For road detection and classification, we use YOLO v3 CNN model [9] trained on the backbone of Efficient Net [17]. Unlike the traditional YOLO v3 model [9] which has used DarkNet-53 network as backbone, our network has a high object detection accuracy at low inference latency. Changes in road infrastructures are minimal, so we can assume CCTV images have advantage of viewpoint invariation. Since vehicles runs on road, the bounding boxes of vehicles are embedded within the bounding box of road, pooling of region of interests, we select the road bounding box. Once the bounding box is selected, we calculate *road mask*. Road Mask is a binary image that is applied to CCTV images to mask out remaining image (converts the pixels to black) except road object. Road Mask image is helpful in limiting regions of interest. Fig 3b, 3d, and 3f show the CCTV image obtain after applying road mask.

2.2.3 DNN for Vehicle detection

For vehicle detection and classification into five annotation classes: bus, car, vans, road, and others, we use Faster R-CNN model [10]. Faster R-CNN in comparison to traditional regional based CNN [11] is manifold times faster due to region proposal network. However, its pooling characteristic is also extremely beneficial. Faster RCNN has a better accuracy than single shot detectors like YOLO. However, the inference speed is extremely slow in comparison to YOLO, rendering it unsuitable for real-time object detection on edge devices. To overcome this, we limit the scope by inputing the image with road mask. Therefore, Faster R-CNN will now only detect object on road, significantly reducing the image area, number of objects, in turn reducing the processing time. Few instances of vehicle detection are illustrated in Figure 3.

2.2.4 Acoustic Emergency Vehicle Detection Framework

Most sirens are rated at around 124 dB when measured 10 feet in front of the sound source. As the distance from the siren doubles, the sound pressure of the siren will drop by approximately 6dB. This concept is known as the "inverse square law." In our system, Data is collected with a frequency of 50Hz and a sliding window technique is employed to detect the emergency vehicle. A sliding window computes the area of the sound noise level over a certain period of time as shown in Figure 4. When the area exceeds a predefined value the detection algorithm dispatches the emergency light sequence. When it does not, normal sequence is carried out. The area value is computed using the formula:

$$Area = Frequency \times sound\ level \tag{1}$$

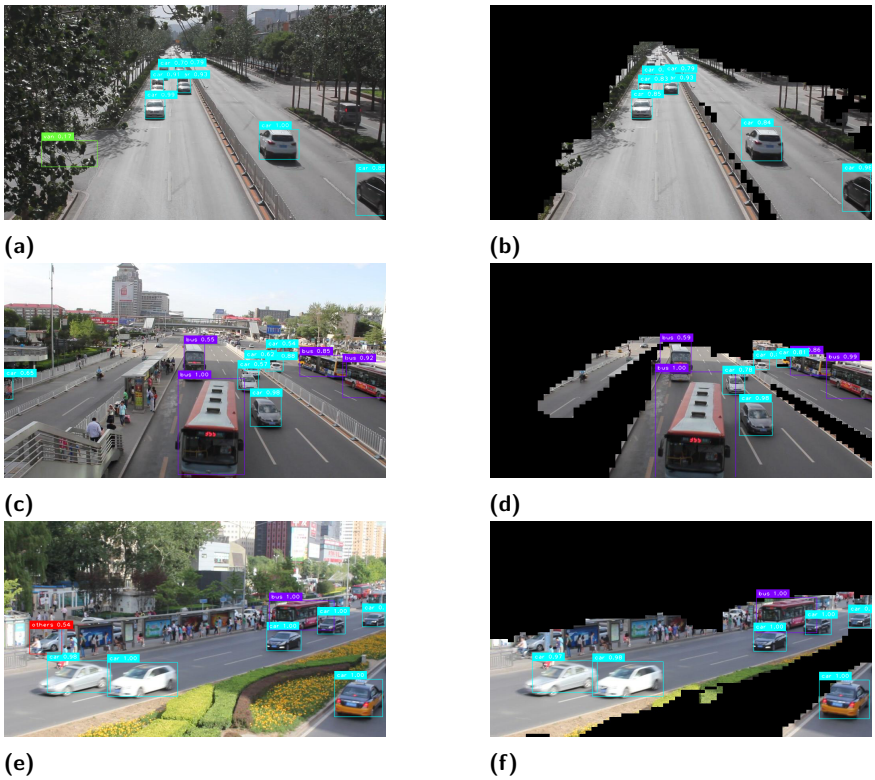


Figure 3 Examples of vehicle detection by a), c), e) by I2UTS and in b), d), f) by our proposed system on same CCTV images from UA-DETRAC [18]. Proposed system performs masking while leaving road.

The system is continuously computing the area over the window length, noise values are summed together over a defined window. The highest sound level is set 100dB, this value has been chosen to take into account different distances of the ambulance from the sound sensor. It has been chosen as it is the average noise level when taking into account 80 feet (25m) as the noise level varies between 124dB and 76dB.

Figure 4 Sample Sliding Window for Emergency vehicle's siren.

Once the siren is detect, a trigger is generated in Edge device S_2. This is sent to edge device S_1 as shown in Figure 1. Edge Device S_1 estimates the vehicle density of each lane using vision algorithm explained in Section 2.2.3. Traffic Control Algorithm proposed by us in our earlier work in I^2UTS uses both these parameters to calculate traffic light sequence again on edge device S_1. The mutli-modal nature of our framework that encapsulates both acoustic and vision sensor and processing to build a resilient system.

3 Experimentation and Results

In this section, we evaluate the performance of our proposed distrubuted IoT framework. The weights of the Faster RCNN and YOLO v3- Effecient Net are trained on a cloud server with Intel i7-9th generation as main processor alongside Nvidia 1660Ti GPU on Linux 18.04 operating system. CUDA 10.1 with Cudnn 7 libraries were used for parallel computation on GPU. The edge device S_1, Raspberry Pi 4, has Quad core cortex-A72 (Arm-8) 1.5GHz processor, 4GB RAM and OpenGL ES 3.0 graphics with Raspbian Buster as the operating system.

The experimental setup for edge system for edge comprises of a SEN0232 sound meter that is connected through a SPI serial connection to an Arduino Nano. This edge device is further connected to Raspberry Pi either wirelessly or through USB. Raspberry Pi is edge device that is responsible for managing the traffic light sequence.

3.1 Emergency Vehicle Detection

Experiments were conducted to find the optimal window lengths. To do so multiple window sizes were chosen starting from 0.5s to 10s. After each trial the window length is incremented by 0.5s. For each window size, ten different sounds are played, of which three correspond to sirens sound of emergency vehicles (police, ambulance, fire-truck) while the rest are different urban noises. The detection time is measured, as well as detection accuracy. The overall operation is repeated 100 times. The detection time D_t is the difference between the time at which the siren is first detected T_d and the time at which the sound is played T_p.

$$D_t = T_d - T_p \tag{2}$$

The accuracy of detection is defined as the number of the truly predicted siren sounds P_s and the truly predicted urban noises P_u divided by the number all the tests N.

$$Accuracy = \frac{P_s + P_u}{N} \tag{3}$$

Table 1 Accuracy and detection time per window size.

Window Length(s)	0.5	2	3.5	5	6.5	8	9.5
Detection Accuracy (%)	63.6	88.4	94.5	99.5	97.4	99.2	99.4
Detection Time (s)	0.54	2.032	3.524	5.031	6.521	8.041	9.523

Results are shown in Table 1, accuracy is very low for short window sizes. The main reason behind this is that it detects numerous urban noises as being siren sounds. Short window sizes make the detection algorithm act as a threshold detection. Accuracy increases with the window length to attain a maximum of 99.5% at 5s and it stays at the same level relatively. From 1000 segments, only 5 sound segments were miss-classified, and these

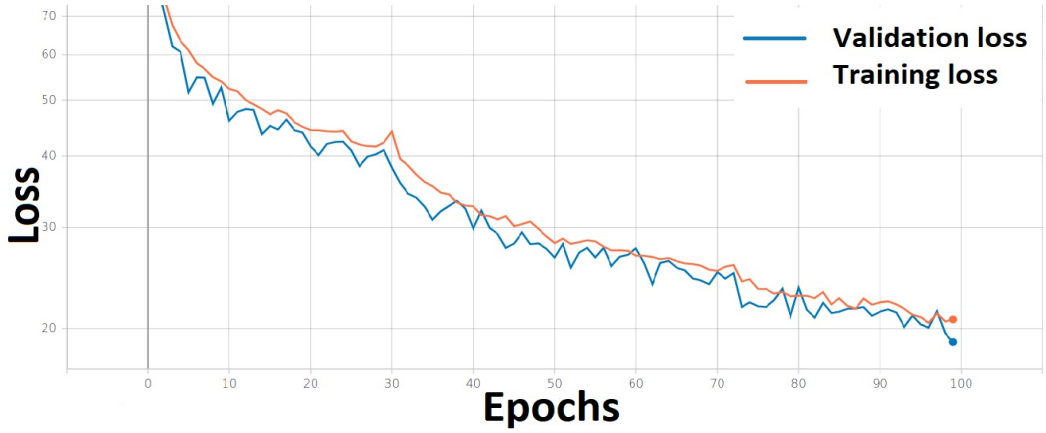


Figure 5 Number of Epoch v/s Loss Difference [1].

segments belong to the urban noises' sounds. The detection time for the different experiments takes an average time of 0.03s ± 0.007s. For 5s the maximum accuracy is reached, and it is selected to be utilised. Another reason is that it is more effective to have a quicker detection.

3.2 Vehicle Detection

While training and fine-tuning the hyper-parameters of both the DNNs for vehicle and road detection, we ensure model reaches optima by neither overfiting nor underfiting. The most important hyperparameter that we need to fine-tune will be number of epochs. To determine this, we plot number of epochs with respect to validation and training loss difference. Figure 5 shows that the loss value's difference is lowest around 100^{th} epoch.

The other important metric to measure efficiency of vehicle detection system is inference time. The inference time of our DNN on the edge divice S_1, Raspberry Pi varied between 1.55 – 2.3 sec per frame. This is comparable to state-of-the-art I^2UTS framework which had inference time of 1.45 – 1.57 sec per frame. The power consumption of edge device (Raspberry Pi 4) per second on different loads is presented in Table 2. The input voltage and current to edge device was DC 5.1V and 3A.

Table 2 Power consumption of IoT device on different loads.

Parameters	Current (Amps)	Voltage (Volts)	Power (Watts)
IoT device not connected to monitor	0.76	5.8	3.12
IoT device connected to monitor	0.78	5.8	3.45
IoT device running only detector	1.34	5.23	6.87
IoT device running detector with connected monitor	1.4	5.23	7.182

The highest power consumption observed was 7.18 W when the detector ran alongside a monitor, constituting only half of the input power supplied. When connected to a traffic camera, the power consumption reduced to 6.87 W. The mean average precision (mAP) for vehicle detection DNN is 79.5% in comparison to 65.10% achieved by state-of-the-art framework I^2UTS. If both the metrics inference time and accuracy are looked together, we can easily say that our proposed novel two-stage detector R-YOLO is able to achieve better accuracy in similar inference time.

4 Conclusion

This paper proposes a distributed IoT framework for urban traffic management system using the CCTV camera and sound sensor. The framework uses the two important observations in urban traffic control: 1) Structural Changes to road are minimum. 2) Emergency Vehicles have distinct sound. The novel two stage detector exploits the first observation by detecting road in first stage and vehicles in second stage. The detector achieves 79.5% accuracy that can further be enhanced by training the network on multiple datasets with CCTV footage with viewpoint and illumination variation. The second observation is implemented using acoustic sliding window detection algorithm achieving 99.4% accuracy.

References

1 Vejey Pradeep Suresh Achari, Zeba Khanam, Amit Kumar Singh, Anish Jindal, Alok Prakash, and Neeraj Kumar. I 2 UTS: An IoT based intelligent urban traffic system. In *2021 IEEE 22nd International Conference on High Performance Switching and Routing (HPSR)*, pages 1–6. IEEE, 2021.

2 Stephan Patrick Baller, Anshul Jindal, Mohak Chadha, and Michael Gerndt. DeepEdgeBench: Benchmarking deep neural networks on edge devices. In *2021 IEEE International Conference on Cloud Engineering (IC2E)*, pages 20–30. IEEE, 2021.

3 Sayalee Deshmukh and SB Vanjale. Iot based traffic signal control for reducing time delay of an emergency vehicle using gps. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–3. IEEE, 2018.

4 Michał Grega, Andrzej Matiolański, Piotr Guzik, and Mikołaj Leszczuk. Automated detection of firearms and knives in a CCTV image. *Sensors*, 16(1):47, 2016.

5 Jose Carlos Villarreal Guerra, Zeba Khanam, Shoaib Ehsan, Rustam Stolkin, and Klaus McDonald-Maier. Weather classification: A new multi-class dataset, data augmentation approach and comprehensive evaluations of convolutional neural networks. In *2018 NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, pages 305–310. IEEE, 2018.

6 Yeong-Lin Lai, Yung-Hua Chou, and Li-Chih Chang. An intelligent IoT emergency vehicle warning system using RFID and Wi-Fi technologies for emergency medical services. *Technology and health care*, 26(1):43–55, 2018.

7 Vamsi Paruchuri, Sriram Chellappan, and Rathinasamy B Lenin. Arrival time based traffic signal optimization for intelligent transportation systems. In *2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA)*, pages 703–709. IEEE, 2013.

8 Sonam Pathak and Manish Pandey. Smart cities: Review of characteristics, composition, challenges and technologies. In *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, pages 871–876. IEEE, 2021.

9 Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint*, 2018. `arXiv:1804.02767`.

10 Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

11 Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

12 David Dorantes Romero, Anton Satria Prabuwono, A Hasniaty, et al. A review of sensing techniques for real-time traffic surveillance. *Journal of applied sciences*, 11(1):192–198, 2011.

13 Shuvendu Roy and Md Sakif Rahman. Emergency vehicle detection on heavy traffic road from CCTV footage using deep convolutional neural network. In *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pages 1–6. IEEE, 2019.

14 Mohd Saifuzzaman, Nazmun Nessa Moon, and Fernaz Narin Nur. Iot based street lighting and traffic management system. In *2017 IEEE region 10 humanitarian technology conference (R10-HTC)*, pages 121–124. IEEE, 2017.

15 Rajeshwari Sundar, Santhoshs Hebbar, and Varaprasad Golla. Implementing intelligent traffic control system for congestion control, ambulance clearance, and stolen vehicle detection. *IEEE Sensors Journal*, 15(2):1109–1113, 2014.

16 Mehal Zaman Talukder, Sheikh Shadab Towqir, Arifur Rahman Remon, and Hasan U Zaman. An IoT based automated traffic control system with real-time update capability. In *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–6. IEEE, 2017.

17 Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.

18 Longyin Wen, Dawei Du, Zhaowei Cai, Zhen Lei, Ming-Ching Chang, Honggang Qi, Jongwoo Lim, Ming-Hsuan Yang, and Siwei Lyu. UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking. *Computer Vision and Image Understanding*, 193:102907, 2020.