Wireless Networks
https://doi.org/10.1007/s11276-022-03076-9

Multi-aspect detection and classification with multi-feed dynamic frame skipping in vehicle of internet things

Usman Ahmed[1] · Jerry Chun-Wei Lin[1] · Gautam Srivastava[2,3]

Accepted: 11 July 2022

Abstract
Consumer demand for automobiles is changing because of the vehicle's dependability and utility, and the superb design and high comfort make the vehicle a wealthy object class. The creation of object classes necessitates the creation of more sophisticated computer vision models. However, the critical issue is image quality, determined by lighting conditions, viewing angle, and physical vehicle construction. This work focuses on creating and implementing a deep learning-based traffic analysis system. Using a variety of video feeds and vehicle information, the developed model recognizes, categorizes, and counts vehicles in real-time traffic flow. The dynamic skipping method offered in the developed model speeds up the processing of a lengthy video stream while ensuring that the video picture is delivered accurately to the viewer. In real-time traffic, standard vehicle retrieval may assist in determining the make, model, and year of the vehicle. Previous MobileNet and VGG19 models achieved F-values of 0.81 and 0.91, respectively. However, the proposed solution raises MobileNet's frame rate from 71.2 to 89.17 and VGG19's frame rate from 48.2 to 59.14. The method may be applied to a wide range of applications that require a dedicated zone to monitor real-time data analysis and normal multimedia operations.

Keywords Vehicle classification · Deep learning · Fine-grained classification · Vehicular traffic · Sensor data

1 Introduction

With 5G technology, current innovative vehicles are networked and distributed systems creating the Internet of Vehicles (IoV). IoV provides autonomous driving, real-time video analytics, and traffic and vehicle management. Increase traffic efficiency, driving safety, and passenger comfort. In-vehicle processing power and interconnect capacity between vehicle and cloud are limited for IoV applications. Edge processing leverages processing resources near IoT nodes to deliver services quickly and with fewer cloud visits, which can be time-consuming or unreliable. Edge computing enables low-latency service delivery for safety- and mission-critical applications such as autonomous driving and non-critical applications such as infotainment.

Intelligent Transportation Systems (ITS) use AI to manage and control traffic. ITS has attracted interest from the transportation industry and other businesses and organizations because it can solve transportation problems. With Big Data and the Internet of Things (IoT), access to big traffic data, including photographs, videos, and texts, is becoming easier. Big Data processing and data analysis using video and trajectory data can help model human behavior, analyze traffic routes, project traffic patterns and trends, and explore hotspots. Due to the unique characteristics of multimedia data, the complexity of human behavior, and the diversity of ITS trajectory patterns, such

✉ Jerry Chun-Wei Lin
jerrylin@ieee.org

Usman Ahmed
Usman.Ahmed@hvl.no

Gautam Srivastava
SRIVASTAVAG@brandonu.ca

[1] Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, 5063 Bergen, Norway

[2] Department of Mathematics and Computer Science, Brandon University, Brandon, Canada

[3] Research Center for Interneural Computing, China Medical University, Taichung 40402, Taiwan

Published online: 02 August 2022

as spatiotemporal attributes, cognitive and behavioral modeling, and evidence-based decision-making, several problems arise. Future research needs more inventive methods that depend on ITS application conditions [1]. ITS is steeped in multimedia cognitive computing. In today's world of Deep Learning and multimedia Big Data, ITS needs breakthrough multimedia cognitive computing approaches and applications.

Figure 1 shows the communication mechanism of the Internet of Drones (IoD) (V2I: vehicle to infrastructure, V2V: vehicle to vehicle, I2I: infrastructure to infrastructure, UAV: unmanned aerial vehicle). The communication shows that the Internet of Things data volume flows through V2V or V2I. It shows the extraction of video content near the smart Internet of Things network. The V2I receives and sends UAV data to the I2I. This information is sent to V2V in different scenarios. The multimedia data is used to provide live video feeds to data centers. After storing and processing the data, management can act quickly under various conditions. With 5G and 6G networks, the Internet of Drones (IoD) has become critical for surveillance applications. Smart traffic monitoring requires urban innovations, vehicle classification, and traffic data. They provide information to traffic management for law enforcement, crime prevention and control, increasing traffic safety, better planning of traffic infrastructure, and minimizing congestion and accidents.

This paper applies state-of-the-art deep learning methods to solve real-world difficulties in fine-grained vehicle categorization. Fine-tuning pre-trained CNNs has led to outstanding performance on several challenging classification tasks in computer vision. However, pre-trained models such as AlexNet and VGG are unsuitable for real-time applications due to their huge memory requirements and lengthy training procedures. Due to MobileNet's simplified design, we chose a smaller CNN architecture with incredibly few training parameters, execution times, and memory requirements. Extensive testing on a huge and diverse dataset has shown promising results, suggesting state-of-the-art deep learning techniques that can be used for fine-grained categorization of vehicles. In addition, there is a lack of qualitative data on car manufacturers, types of automobile models, and generations of car models. We have compiled a large and diverse dataset based on automakers, the types of models available, and the year each model was first introduced. Researchers in academia are using this dataset to develop and verify various computer vision and deep learning approaches. Below are some of the specific contributions made by the paper:

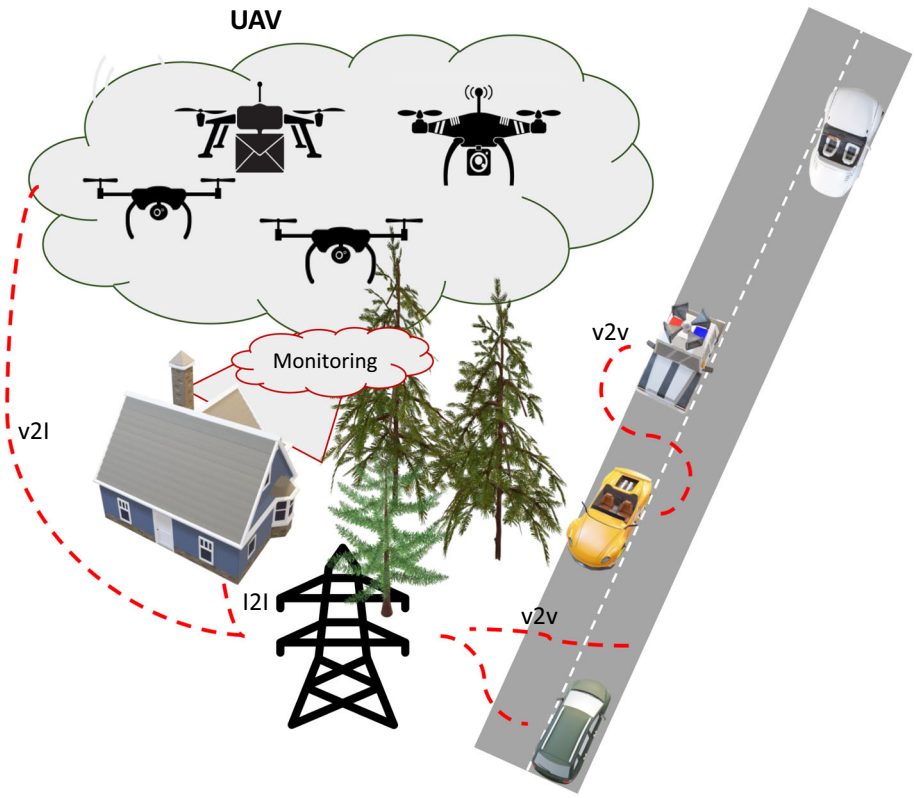

Fig. 1 The overview of the Multi-feed scenario

1. A hierarchical feature learning strategy with multi-level vehicle type identification is accomplished by embedding features and semantic information.
2. Improve memory-efficient and dynamic frame-skipping for multiple feed object detection.
3. Extensive experiments reveal that the developed model outperforms existing state-of-the-art approaches.

The remainder of the paper is organized as follows: Sect. 2 reviews current methods for fine-grained vehicle classification described in the literature. Section 3 presents the proposed method, dataset, experimental protocol, and evaluation metrics. Section 4 presents the experimental results as well as observations, followed by a conclusion in Sect. 5. Finally, a section on future directions this research may take finalizes the paper in Sect. 6.

2 Related work

Deep learning, often referred to as DL, has already significantly increased the power of computer vision. This is the result of recent developments in GPU technology, enormous amounts of training data, and algorithmic improvements. Recent applications of advanced deep learning algorithms for feature extraction [2], and object identification in computer vision difficulties include image and video classification and forensic analysis. The use of advances in machine vision and artificial intelligence has also helped improve law enforcement on the road. Machine vision and machine learning techniques have been developed for use in traffic surveillance, activity monitoring, traffic anomaly detection, driver assistance, and other traffic management approaches [3]. Several attempts have been made to overcome the barriers associated with conventional vehicle detection and categorization challenges. Autonomous license plate recognition (ALPR) and human inspection-based vehicle detection technologies fall short of real-time requirements. Identifying and classifying automobiles using these approaches takes time and effort due to the wide range of vehicle brands, models, and decades.

The rise of smart cities has heightened interest in fine-grained vehicle categorization. Several problems exist in fine-grained vehicle categorization [4], including intra-class similarity, viewing angles, and illumination. Several initiatives have been started recently. Traffic control, traffic flow analysis, traffic composition, and other operations are part of an intelligent transportation system (ITS). Many research studies have utilized CNN to classify vehicles [5]. Using preprocessed pictures and data augmentation, In [6] detects traffic congestion using a dataset of approximately 30,000 photographs. Traficant, based on residual learning, is given for traffic congestion detection.

Deep learning has already been shown to significantly improve the performance of computer vision techniques due to technological advances such as graphics processing units (GPU), the availability of large amounts of training data, and algorithm optimization [7]. Recent applications of the method known as deep learning include feature extraction, object recognition, and categorization. Traffic regulation has also benefited from recent developments in artificial intelligence (AI) and machine vision technology. Various methods of image processing and machine learning have been developed for the purposes of traffic monitoring, activity monitoring, traffic anomaly detection, driver assistance, traffic behavior analysis, monitoring, and traffic management [8, 9]. In response to the growing concern about road safety, several approaches have been proposed to circumvent the challenges posed by traditional vehicle identification and categorization methods. Traditional methods of vehicle identification and categorization, such as human inspection or automatic license plate recognition, are unable to keep pace with the demands of real-time operations. Vehicle identification and categorization is a process that is both time consuming and labor intensive due to a large number of different vehicle types, models, and generations. These tactics complicate the process.

Deep learning was also used to classify flowers [10], pets [11], and automobiles [4]. With the growth of smart cities, the emphasis has switched to fine-grained vehicle categorization in recent years. Several hurdles to fine-grained vehicle classification include significant intra-class similarity, uncontrolled perspectives, and variable lighting conditions. In recent years, many initiatives have been started in this area. Wang et al. [12] created a deep learning system for recognizing vehicle parts based on their location. A localization network provides segmentation masks or component locations for improved categorization. This model learns discriminative region attention and regional representations of characteristics at different sizes without requiring annotated training regions. Lam et al. [13] presented a search-based framework for fine-grained recognition in which CNN feature maps are employed to build the search space for component identification and classification. Pooling is an essential component of deep learning architectures. To improve fine-grained categorization, low-rank bilinear feature pooling was used in a compact CNN classification framework [14]. A polynomial kernel-based predictor was utilized to extract high-order statistical data from convolutional activations, which was subsequently fed into a standard CNN for fine-grained classification [15].

The primary purpose of CNN training is to maximize the backpropagation method's loss function to extract generic

and robust features for general classification and recognition tasks. However, when it comes to fine-grained vehicle categorization, this frequently results in duplicate characteristics that are too fitted to the model. A lightweight CNN model [16]. That extracts compact, low-dimensional features may perform remarkably well on training data while improving the network's generalization. Furthermore, with a CNN design, the layer connecting the convolutional layers to the fully connected layer comprises many neurons and training parameters. Biglari et al. [17] pioneered channel max pooling, a method for integrating several feature maps from distinct channels into a single feature map. This method minimizes the number of train parameters while enhancing generalization. Ma et al. [12] used channel max-pooling in CNNs to detect fine-grained cars and empirically evaluated the efficiency and generalization of two datasets.

Chang et al. [18] provide sparse encoding and autoencoding techniques for data with visual occlusions. Following vehicle identification, neural learning is used to classify vehicles (car, van, bus). A saliency map feature lowers the computational cost by selecting target regions and then labelling the output, which is subsequently sent to the ResNet50 (CNN) model for vehicle classification. However, it only categorizes vehicles into three types: cars, trucks, and buses [19]. The proposed work uses the Stanford Cars-196, Label-Me, UIUC-Sports, and CIFAR-10 datasets to train eight classes using VGG16 and DenseNet161 CNN models [20]. In addition to solving the vanishing gradient problem, they suggest a new loss function called dual cross-entropy loss. Some studies employed less complicated but less accurate procedures than CNN. A feature is assigned to the cluster centre using K-means clustering. The features are extracted using a DNN. Using high- and low-resolution pictures, the features are calculated and combined [21]. Other vehicle categories are covered. The authors claim that this system can adapt to changing image resolutions and learn vehicle features.

A cascade ensemble classifier based on MLP and K-Nearest Neighbor (K-NN) is provided for categorizing vehicle kinds. Our hierarchical classifier achieves 97.8% reliability [22] with only five vehicle kinds. Valid et al. [23] develop a computationally efficient approach for length-based vehicle categorisation with 99.98 percent accuracy. This method is easy and transferrable. However, it cannot detect automobiles of similar length or modified vehicles. R-CNN classifies automobiles using VGG16 models. They utilize the MIT and Caltech vehicle datasets for training and testing. The results demonstrate over 80% identification accuracy for seven specific vehicle kinds. This paper presents a ResNet-based vehicle categorization and localization technique using traffic surveillance video [5]. It employs a MIOvision traffic dataset with 11 categories to further characterize the model. Cooperative fine-tuning (CFN) is implemented using CNN (DropCNN). The authors claim their model outperforms VGG16, AlexNet, and ResNet50.

3 Methodology

It is difficult to distinguish between uninteresting things in traffic monitoring. For example, analyzing all video feeds and evaluating the results might take a long time if a traffic analyzer searched for a certain car by manufacturer, model, and year of production. This review method requires at least as much time as the length of the video feed. This wastes time analyzing data and results from a lack of appropriate screening technologies. The option to look for automobiles and utilize the same screening methods as the complete video might help solve this issue. The system may be able to identify and process traffic. After finding the data, traffic analysis might examine the results for the higher-level agency. This technique takes far less time than watching the complete film. This technology can also discriminate between various car model makers, kinds, and years.

Figure 2 illustrates the communication strategy for multiple feeds in an IoD context. The communication shows how the Internet of Things (IoT) enables vehicle-to-vehicle (V2V) as well as vehicle-to-infrastructure (V2I) data exchange. It shows the extraction of multimedia video from a network connected to the Smart Internet of Vehicular Things. Data from the environment flows to UAVs that communicate with the *V2I*, which collects and sends data to the *I2I*. This data is then transmitted to the *V2V* in various settings. The data centres get a live video feed of all multimedia content. After storing the data, management may be able to act quickly. The Internet of Drones is vital for surveillance applications now that high-speed mobile networks like 5G and 6G are available. Intelligent traffic monitoring in intelligent cities requires efficient vehicle categorization and traffic data to help traffic management make informed judgments about law enforcement, crime prevention, traffic safety, traffic infrastructure development, and congestion and accident reduction.

This work tackles the topic of fine-grained vehicle categorization in real-time utilizing media sources and cutting-edge deep learning techniques. Pre-trained CNNs have excelled at tough classification jobs in various computer vision challenges. Because of their significant memory needs and lengthy training methods, models like AlexNet [24] and VGG [25] are unsuitable for real-time applications. We picked the MobileNet CNN architecture because it has fewer movement parameters, execution durations, and memory needs. We have extensively tested state-of-

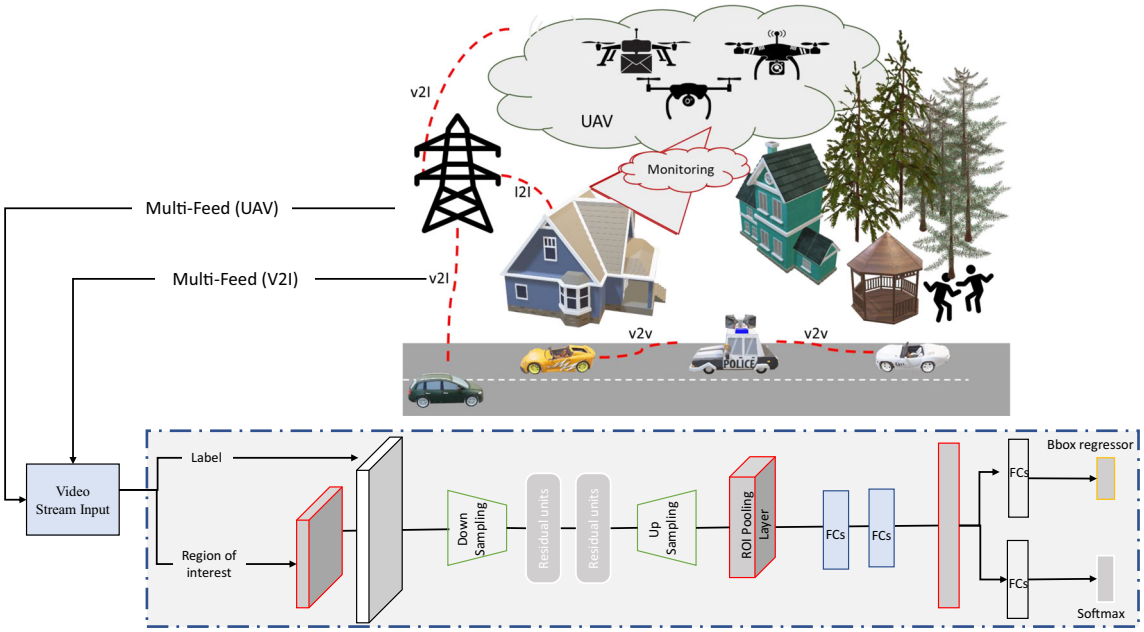

Fig. 2 The Multi-feed Fine grained vehicle classification

the-art Deep Learning approaches for fine-grained vehicle classification on a large and diverse dataset, with encouraging results [26]. Moreover, qualitative data on car manufacturers, model kinds, and model generations are lacking. We have compiled a large and diversified collection of automobile manufacturers, models, and generations. This would allow researchers to design and test computer vision and deep learning techniques. Specifically, the proposed model aims to improve average accuracy, reduce processing time, and reduce the false-negative rate in vehicle recognition.

3.1 Dataset

This study's data will come from a web source. Web-based data sources include car forums, search engines, and public websites. Different scenarios assess the impacted model's response to a problem [27]. The data sets' properties allow for examining underlying problems under various picture quality settings. The fundamental issue is distortion in various image quality settings. A shot near a moving object becomes gradually distorted [27]. When a high-quality camera examines the image slowly, the form is exhibited with incredible precision and quality.

3.2 Data augmentation

Augmentation enhances the performance of the learning process by augmenting the training set with new instances. When a learning model has accumulated sufficient data, it may be extended to generate more accurate results. Data collection and labelling is a time-consuming and costly procedure. Businesses may be able to cut operational and data processing expenses by utilizing data augmentation. The augmentation strategy adds new training patterns to the training set to address data scarcity, reduce overfitting, promote variety, boost generalization, and resolve the class imbalance problem. Overfitting happens when the model excels at learning the training patterns but struggles to learn new ones. This may be addressed by using extension strategies to diversify the training patterns. The data mainly was included to prevent our proposed system from being too close to the training data [28]. We enhanced our data using rotation, latitude shift, height shift, shear, zoom, horizontal flip, and brightness range.

3.3 Transfer learning

Transfer learning is a subset of machine learning that uses a model generated for a specific task as a foundation. Two powerful frameworks, MobileNet and Resnet52, have already been developed and made publicly available in TensorFlow. Both frameworks are based on an Imagenet model that has been trained and fine-tuned on a dataset of over 1.2 million RGB photos. The transfer learning process was integrated into the initial and final phases of the frameworks. The linked convolutional layers are overlaid [29]. Convolutional layers retrieve information from images containing a variety of data types. This approach has been used in several scientific publications dealing with image analysis topics such as classification.

As mentioned in Fig. 2, we used different methods for comparing and improving the computational costs. The convolution, max-pooling, and flattening layers form different designs, which will be explained in the next section. The convolution layer extracts image features using filters, while the max-pooling layer reduces the image size while extracting features from feature maps. Downsampling uses a simple convolutional architecture to abstract the visual input. Upsampling expands the abstract image representations to fit the input image. Flattening converts feature maps into 1D tensors (one for each class). The last one is the soft-max layer. Fully connected layers look the same in any network. All layers use ReLu activation except the last classification layer, which uses softmax. In both neural transfer learning techniques, the pre-trained model is used as a feature extractor for the linked features. The goal of using a pre-trained network is to optimize CNNs without compromising state-of-the-art performance. Fine-tuning involves making changes to both the first and last layers. The fully connected layers of the custom class replace the previous layer. The backpropagation algorithm becomes convergent and adapts to the new classification objective when this happens.

3.4 Multi-feed fine-grained classification

Automobiles are classified into five distinct 51 categories. 70–30% of the data is utilized for training and 30% for testing. The categorization algorithm is trained on images of automobiles collected from various perspectives. MobileNet outscored Resnet considerably in the ImageNet classification test [29]. However, more sophisticated networks incorporating attention processes may be employed for activities involving many classes. A subsequent experiment is conducted but with natural surveillance data for fine-grained categorization. The data collection includes instances of 44, 481 automobile models that class 51. 70% to 30% of these photographs are maintained for educational or testing purposes. The photographs of the automobiles are all front views shot in various weather circumstances, including rain, fog, and night. This experiment employs the same three network architectures utilized in the online natural data application, such as ResNet and MobileNet, as seen in reffig2 and reffig3. Additionally, these networks were trained on ImageNet classification tasks and tested using a single-centre crop [29]. The cropping limits advised here leave a 7% gap on both sides of the automobile images and boxes. After that, the photos are downsized to a resolution of 256*x*256 pixels. Each of the three networks operated without a hitch. The finding implies that, despite significant contextual changes, the frontal perspective enables relatively fine-grained categorization.

Table 1 Dynamic frame skipping. (n: represents number of feature vector saved for the comparison)

Video ID	Total frames	Skipping ratio	n=5	n=10
1	2017	1/7	204	651
2	4781	1/6	1810	2681
3	7109	1/5	2390	2819

3.5 Dynamic frame skipping

In this section, we discuss an innovative method for reducing multiple feeds. The model reduces the computational cost of the proposed framework by dynamically skipping redundant classifications. The transfer learning method detects the object from the multi-feed. We used the transfer learning method for object detection to tune the network to the type of vehicle detection. The data is manually labeled and then enhanced during the surveillance. Then we use the detected frames for frame skipping when searching for relevant information. We compare consecutive frames structurally. We skip frames that are fundamentally identical to previous frames and do not require re-routing by the faster RCNN using a 90 threshold (Algorithm 1, line 3). This value was determined by trial and error and helped exclude frames with a high degree of similarity. The difference between successive frames is defined by the dynamic movement of the video scene with respect to the direction of the camera. We set the threshold to $structsim = 0.9$ because image noise increases the similarity of irregular images (Algorithm 1, lines 1-7). The threshold value of 0.9 accounts for the problem of image noise. Additionally, we classified the object based on the probability $t > 0.9$ associated with the backbone network (where higher probability is better). If $t>$ is 0.9, the image is discarded as a duplicate at the classification phase (Algorithm 1, lines 8-15). Each feature map is held in memory for a period of n classifications until no other feature map of the image to be classified matches it. In this experiment, we kept a feature vector in memory for non-similar 5, 10, and 20 classifications (Algorithm 1, lines 11-15). Since most concurrent frames are similar and can be skipped without data loss, dynamic frame skipping reduces computation time by nearly 70% to 90%, as described in Table 1 and Algorithm 1.

Algorithm 1 Proposed Algorithm with Muti-Frame Skipping

Require: Multi-feed *Backbone* with targeted *class*.
Ensure: $output_{frame}$ with object region and name.

$Frames \leftarrow [Last_{frame}, First_{frame}]$
while $frames \in Multi - feed$ **do**
  if $Similarity_{strucutre}\ (frames,\ Last_{frame}) \geq 0.9$ **then**
    continue;
  else
    $Frame \leftarrow Last_{frame}$;
  end if
  for $all\ object, name, Thresold \in Model(frames)$ **do**
    if $name == Model - class\ and\ thresold_1 > 0.9$ **then**
      $output_{frame}$ ←Region of interest detection with coordinates $output_{frame}$;
      $Ouput_{class}, threshold_2 = Backbone(object)$;
    else
      $Output_{frame}$ ← Region of interest detection on UNKNOWN object ;
    end if
  end for
end while
Return $Output_{frame}$.

3.6 Duplicate classification avoidance

To prevent classifications from duplicating, we calculated three distinct types of features for each categorization. Histograms of colours A colour histogram is a graphical depiction of a photograph's colour distribution. It is a phrase related to how colours are distributed throughout all of the pixels in an image. As a result, it is unique to each image and may be utilized as a feature map to differentiate across photographs. We determined an image's spatial properties by compressing it to a shallow resolution of 32*x*32 over all three colour channels. Histogram of oriented gradients *(HOG)* features: It possesses the following qualities; namely, HOG is a sort of feature representation in which an image's pixel-based information is translated into gradients with varying orientations. Already categorized image characteristics are utilized to avoid clashes with the *Unknown* class. These characteristics are generated for each picture and stored in memory before the backbone network classifying them. They are then matched with future photos, which are classified based on the cosine similarity of their feature map, as defined in Eq. (1).

$$similarity(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{||x||||y||} \tag{1}$$

While storing more information in memory increases the system's computational complexity, it also aids in avoiding duplicate frames. This is because adding features to memory increases the number of cosine comparisons at each stage. Due to the extra complexity, more duplicates are eliminated, but the procedure is also slower. Avoiding duplicate categories has the unintended consequence of increasing the complexity necessary to compute and compare automotive attributes. This is acceptable since the added time complexity of computing and comparing characteristics overcomes the added difficulty of duplication. A limiting situation would be a collision or speed bump near the surveillance cameras, undoubtedly interrupting traffic flow and leading the algorithm to mislabel similar automobiles. As a result, the additional temporal complexity associated with avoiding repetitive categorization is necessary to avoid certain boundary conditions, as previously noted.

3.7 Evaluation metrics

A series of experiments are conducted to evaluate the proposed model's performance. The classification performance is evaluated using Precision, Recall as well as F-measure. The following equations are used to calculate the metrics. Accuracy is the percentage of correctly identified samples relative to the total number of samples.

$$\text{Accuracy} = \frac{\#\ of\ correctly\ classified\ samples}{Total\ samples} \tag{2}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

$$\text{F1-Score} = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)}, \tag{5}$$

where the number of classified outputs as true positive *TP*, false positive *FP*, as well as false negative *FN*.

4 Experimental results

Extending the experiments reported in Sect. 3, we also conduct tests on fine-grained car classification (manufacturer type, year, and model) utilizing the complete dataset and numerous deep models to assess the models' capability on these tasks. The collection is organized into three sections (manufacturer type, year, and model) and contains 52,083 photos of 51 vehicle types.

For the classification tasks, the three network architectures EfficientNETB0 [30], MobileNet [31], NASNET-Mobile [32], and VGG with 16 and 19 variants [33]. These models are pre-trained on the ImageNet classification problem [34] as well as fine-tuned for each task using the same mini-batch size, epochs, as well as learning rates. The predictions of the deep models are made with a single frame. For our experiments, we use the Tensorflow system. The dataset divides objects into 51 distinct categories. 70% of the data is utilized for training and 30% for testing. The classification model is trained using photographs of automobiles taken from various angles. Figure 3 depicts the three networks' performance. MobileNet surpasses Resnet by 6.0 percent, similar to their classification performance on ImageNet. Another experiment is carried out to determine the accuracy of fine-grained classification. All items are frontal views in various environments, including rain, fog, and darkness. This experiment employs the same three network architectures, including ResNet and MobileNet. Additionally, these networks were trained on ImageNet classification tasks, and the tests use a single clipping. Cropping is performed on item images and boxes using the given frame, which includes roughly 7% padding on each side. These images are then downsized to a resolution of 256×256 pixels. Each of the three networks demonstrated an unusually high degree of accuracy. The findings indicate that, despite the high degree of contextual divergence, the front view considerably increases the feasibility of fine-grained classification tasks.

VGG employs deep, convolutional neural networks (up to 19 weighting layers). As illustrated in Fig. 3, increasing the depth of representation increases classification accuracy, and a typical ConvNet architecture may attain peak performance on the ImageNet challenge dataset. The method generalizes well over a wide range of objects and equals or outperforms the performance of more complicated identification pipelines based on flatter visual representations. Our results again show the relevance of visual representation depth, with increased accuracy of 0.90 for 51 items. The VGG19 model outperformed the other models.

Mobilenet demonstrates that efficient models exist for embedded image processing applications as well as archives with 0.81 F-measure, as seen in Figs. 3, 4, and 3 but with longer processing time. MobileNets are based on a simplified architecture that generates lightweight deep neural networks with depth separable convolutions. The hyperparameters enable the model maker to select the best model for their application based on the limitations of the problem. We present significant research on the tradeoffs between resources and accuracy and good ImageNet classification performance compared to other regularly used models. MobileNets employ width and resolution multipliers to reduce size and latency while retaining

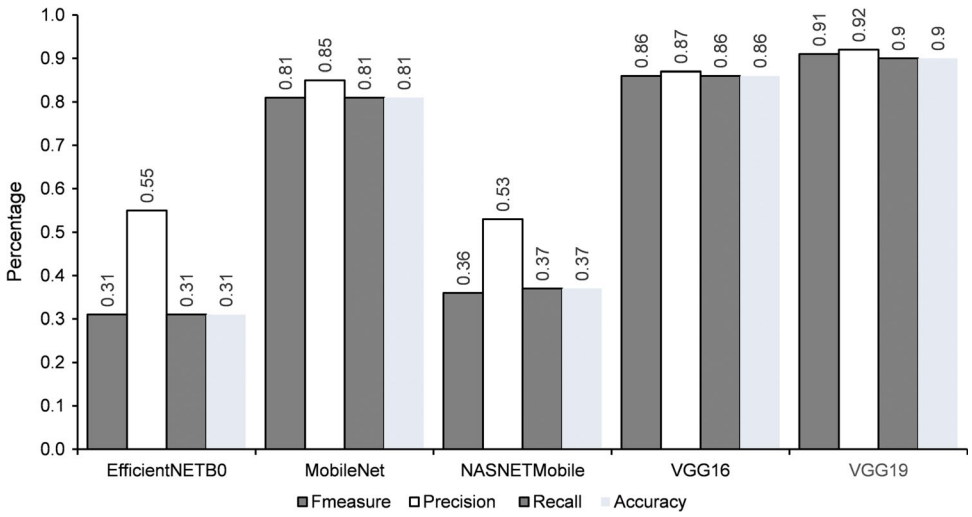

Fig. 3 F-measure of number of architecture

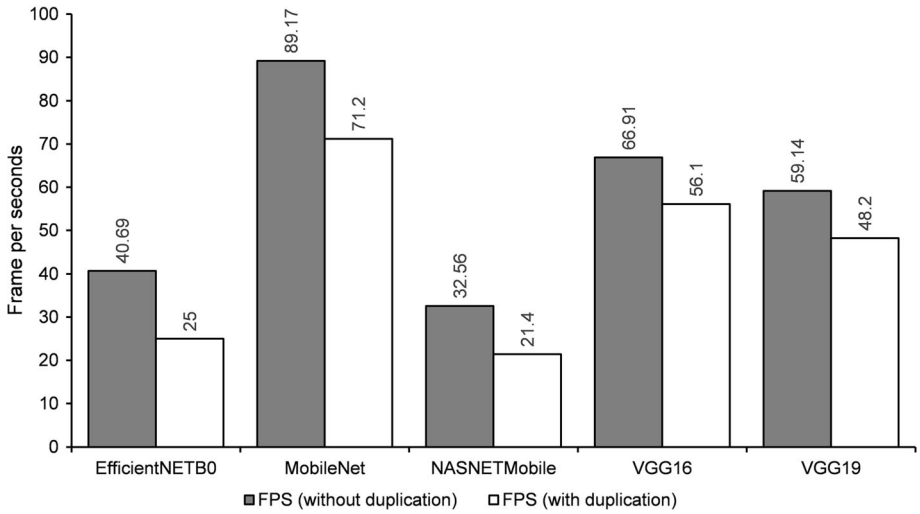

Fig. 4 Frame per seconds comparison

acceptable accuracy. MobileNets that excel in terms of size, speed, and accuracy.

Convolutional Neural Networks (ConvNets) are often developed with a restricted resource budget and then scaled up for increased accuracy as more resources become available. Discover how an optimal mix of network depth, breadth, and resolution may be able to contribute to performance benefits. Based on this, it employs a simple but extremely effective composite coefficient to scale all depth/width/resolution dimensions evenly. This strategy's efficiency and efficacy in MobileNets and ResNets. EfficientNets are convolutional neural networks that exceed prior ConvNets in accuracy and efficiency. However, as shown in Figs. 3, 4, and 5, the technique is incapable of achieving high accuracy. The EfficientNet model can be scaled successfully at mobile size with fewer parameters and processing time, but it does not generalize well to ImageNet using this hybrid scaling approach.

As mentioned in Table 2, Mobilenet, VGG-16, and VGG-19 performed better. The class activation map for models with different scaling techniques is shown in Fig. 4 to show why frame duplication elimination performs better. However, the frame rates of the mobile network perform better with a measure of 0.81 and the frame rates with enhancement are 89.17 per second. However, the VGG19 has the highest performance in type detection. As shown in Figs. 3 and 4, the duplication reduction model can achieve high frame rates while focusing on more relevant regions with more extensive object details. The other models, on the other hand, either lack object details or cannot capture all objects in the images.

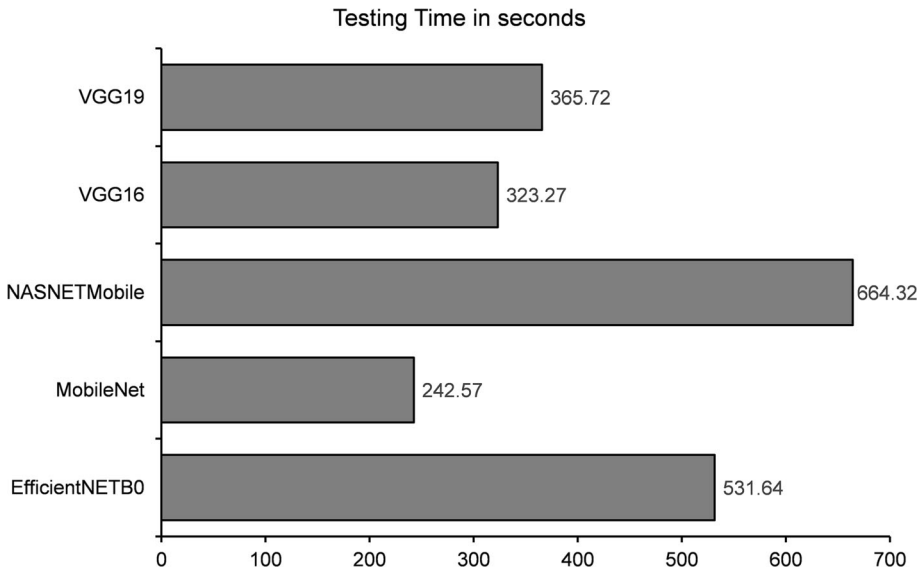

Fig. 5 Testing time comparison

Table 2 Effects of the proposed model on architectures results

Methods	F-measure	FPS (without duplication)	FPS (with duplication)	Testing in seconds
EfficientNETB0	0.31	40.69	25	531.64
MobileNet	0.81	**89.17**	**71.2**	**242.57**
NASNETMobile	0.36	32.56	21.4	664.32
VGG16	0.86	66.91	56.1	323.27
VGG19	**0.91**	59.14	48.2	365.72

Bold values indicate the best results per column

Low inter-class or high intra-class variance leads to mis-classification. The score shows mis-classification due to low inter-class variance. Classes have comparable forms and frontal features. Transfer learning and frame skipping were evaluated with the dataset. First, fine-tune the deep neural networks and extract high-level features from the layers and the classification layer. VGG19 fine-tuning outperformed feature extraction accuracy. High intra-class variance or low inter-class variance resulted in some mis-classifications after implementing the proposed technique. Using a deeper classification network reduces the intra-class variance and increases the inter-class distance. Adding more instances per class with transfer learning can improve model performance. Super-resolution algorithms can also improve low-resolution surveillance.

The IoT is collecting video from healthcare, smart city, and surveillance. Video analytics requires the right perspective, format, and quality. The rapid increase in population and cars in cities in developing countries leads to traffic infractions, vehicle theft, congestion, and accidents. Vehicle classification and traffic data must be accurate and efficient to improve law enforcement, crime control and prevention, traffic safety, transportation infrastructure, and congestion and accidents. Safe cities projects have installed CCTV cameras on a large scale for traffic monitoring and management. The huge amounts of video data from CCTV cameras provide AI solutions for traffic monitoring and surveillance solutions. This study identifies cars by type, year, and manufacturer. The model is trained with user-defined data by transfer learning. Then, the model can use object categorization to store RAM. We talked about data collection and feed frames. Sharing and checking the database can save processing costs. The sensitive data of multiple acquisitions can be preprocessed on-site.

Any technology can support health care and border security. A drone network can collect temperature, humidity, video, acceleration, ultrasound, proximity, and gas data to detect and analyze various objects. Due to processing, battery life and other factors, drones cannot cover large areas quickly. With query-based search, analyzing a large amount of drone data can help detect an occurrence. Military bases, transportation groups, and energy utilities may use this method to manage a smart grid. The smart power grid is a multimedia network application. Energy generation and distribution must be intelligent. Numerous media provide security for power generation and distribution systems that require continuous power flow. Multimedia monitors personnel for technical and non-technical problems, including energy theft and infrastructure damage. Multimedia can detect temperature and humidity changes that cause power outages.

5 Conclusion

This study presents a skipping technique for multimedia frames that may be utilized effectively for frame processing in-vehicle networks. A-frame skipping strategy based on a multi-class classifier is presented to help us reduce the time required to query frames in multimedia networks. As a basis, we employ a deep learning-based model. Then, to lower the computational cost, the object recognition algorithm employs a similarity-based frame skipping mechanism. The video footage from several incidents is then processed to establish the vehicle's make, year, and manufacturer. The model may be capable of reducing the number of frames required for object detection while enhancing object detection. Numerous grain properties are dynamically examined in a vehicle surveillance situation to skip frames. Extensive investigation and comparison demonstrate that our technique effectively uses dynamic frame skipping and important information identification. The approach has the potential for further development, application, and study in various multimedia sensing tasks. The dynamic architectural search approach will be enhanced in the future to consider more input frames, domain kinds, and computational and energy resources.

6 Future work

Dynamic frame skipping in-vehicle monitoring considers multiple attributes. Extensive investigation and comparisons show that our technique dynamically skips frames and detects relevant information. The method can be applied to numerous multimedia tasks that require a careful search for relevant information with limited resources. Future studies will adapt the dynamic architectural search to the input frames, the type of domain, and the computational and energy resources. We would also like to incorporate the semi-supervised learning approach to improve architectural performance and frame rates.

Funding Open access funding provided by Western Norway University Of Applied Sciences.

References

1. Zhang, F., Xu, M., & Xu, C. (2022). Tell, imagine, and search: End-to-end learning for composing text and image to image retrieval. *ACM Transactions on Multimedia Computing, Communications and Applications (TOMM), 18*(2), 1–23.
2. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1725–1732
3. Impedovo, D., Balducci, F., Dentamaro, V., & Pirlo, G. (2019). Vehicular traffic congestion classification by visual features and deep learning approaches: A comparison. *Sensors, 19*(23), 5213.
4. Ma, Z., Chang, D., Xie, J., Ding, Y., Wen, S., Li, X., Si, Z., & Guo, J. (2019). Fine-grained vehicle classification with channel max pooling modified cnns. *IEEE Transactions on Vehicular Technology, 68*(4), 3224–3233.
5. Jung, H., Choi, M.K., Jung, J., Lee, J.H., Kwon, S., Young Jung, W. (2017). Resnet-based vehicle classification and localization in traffic surveillance systems. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 61–67
6. Wang, P., Hao, W., Sun, Z., Wang, S., Tan, E., Li, L., & Jin, Y. (2018). Regional detection of traffic congestion using in a large-scale surveillance system via deep residual trafficnet. *IEEE Access, 6*, 68910–68919.
7. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.
8. Yousaf, A., Khan, M. J., Khan, M. J., Siddiqui, A. M., & Khurshid, K. (2020). A robust and efficient convolutional deep learning framework for age-invariant face recognition. *Expert Systems, 37*(3), 12503.
9. Ahmad, H. M., Khan, M. J., Yousaf, A., Ghuffar, S., & Khurshid, K. (2020). Deep learning: A breakthrough in medical imaging. *Current Medical Imaging, 16*(8), 946–956.
10. Elhoseiny, M., Elgammal, A., & Saleh, B. (2016). Write a classifier: Predicting visual classifiers from unstructured text. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(12), 2539–2553.
11. Zhang, Y., Wei, X.-S., Wu, J., Cai, J., Lu, J., Nguyen, V.-A., & Do, M. N. (2016). Weakly supervised fine-grained categorization with part-based image representation. *IEEE Transactions on Image Processing, 25*(4), 1713–1725.
12. Wang, Y., Morariu, V.I., Davis, L.S. (2018). Learning a discriminative filter bank within a cnn for fine-grained recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4148–4157
13. Lam, M., Mahasseni, B., Todorovic, S.: Fine-grained recognition as hsnet search for informative image parts. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2520–2529 (2017)
14. Kong, S., Fowlkes, C. (2017). Low-rank bilinear pooling for fine-grained classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 365–374
15. Cai, S., Zuo, W., Zhang, L. (2017). Higher-order integration of hierarchical convolutional activations for fine-grained visual categorization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 511–520
16. Zhang, Q., Zhuo, L., Zhang, S., Li, J., Zhang, H., Li, X. (2018). Fine-grained vehicle recognition using lightweight convolutional neural network with combined learning strategy. In: 2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM), pp. 1–5 IEEE
17. Biglari, M., Soleimani, A., & Hassanpour, H. (2017). A cascaded part-based system for fine-grained vehicle classification. *IEEE Transactions on Intelligent Transportation Systems, 19*(1), 273–283.
18. Chang, J., Wang, L., Meng, G., Xiang, S., & Pan, C. (2018). Vision-based occlusion handling and vehicle classification for traffic surveillance systems. *IEEE Intelligent Transportation Systems Magazine, 10*(2), 80–92.
19. Li, Y., Song, B., Kang, X., Du, X., & Guizani, M. (2018). Vehicle-type detection based on compressed sensing and deep learning in vehicular networks. *Sensors, 18*(12), 4500.
20. Li, X., Yu, L., Chang, D., Ma, Z., & Cao, J. (2019). Dual cross-entropy loss for small-sample fine-grained vehicle classification. *IEEE Transactions on Vehicular Technology, 68*(5), 4204–4212.
21. Santhosh, K. K., Dogra, D. P., & Roy, P. P. (2018). Temporal unknown incremental clustering model for analysis of traffic surveillance videos. *IEEE Transactions on Intelligent Transportation Systems, 20*(5), 1762–1773.
22. Lian, J., Zhang, J., Gan, T., Jiang, S. (2018). Vehicle type classification using hierarchical classifiers. In: Journal of Physics: Conference Series, 1069 p. 012099 IOP Publishing
23. Balid, W., Tafish, H., & Refai, H. H. (2017). Intelligent vehicle counting and classification sensor for real-time traffic surveillance. *IEEE Transactions on Intelligent Transportation Systems, 19*(6), 1784–1794.
24. Dutta, T., Soni, A., Gona, P., & Gupta, H. P. (2021). Real testbed for autonomous anomaly detection in power grid using low-cost unmanned aerial vehicles and aerial imaging. *IEEE MultiMedia, 25*, 81.

25. Simonyan, K., Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations
26. Kniaza, V., & Moshkantseva, P. (2021). Object re-identification using multimodal aerial imagery and conditional adversarial networks. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 44*, 2.
27. Chen, L., Liu, F., Zhao, Y., Wang, W., Yuan, X., Zhu, J. (2020). Valid: A comprehensive virtual aerial image dataset. In: 2020 IEEE International Conference on Robotics and Automation, pp. 2009–2016 IEEE
28. Wang, J., & Perez, L. (2017). The effectiveness of data augmentation in image classification using deep learning. *Convolutional Neural Networks Visual Recognition, 11*, 1–8.
29. Zhang, C., Benz, P., Argaw, D.M., Lee, S., Kim, J., Rameau, F., Bazin, J.-C., Kweon, I.S. (2021). Resnet or densenet? introducing dense shortcuts to resnet. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 3550–3559
30. Tan, M., Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning, pp. 6105–6114 PMLR
31. . Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. http://arxiv.org/abs/1704.04861
32. Zoph, B., Vasudevan, V., Shlens, J., Le, Q.V. (2018). Learning transferable architectures for scalable image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8697–8710
33. Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. http://arxiv.org/abs/1409.1556
34. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems, 25*, 1097–1105.

Usman Ahmed is a Ph.D. candidate at the Western Norway University of Applied Sciences (HVL). He has rich experience in building and scaling high-performance systems based on data mining, natural language processing, and machine learning. His research interests are sequential data mining, heterogeneous computing, natural language processing, recommendation systems, and machine learning.

Jerry Chun-Wei Lin received his Ph.D. from the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan in 2010. He is currently a full Professor with the Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway. He has published more than 500+ research articles in refereed journals (IEEE TKDE, IEEE TCYB, IEEE TII, IEEE TITS, IEEE TIAS, IEEE TETCI, IEEE SysJ, IEEE SensJ, IEEE IOTJ, ACM TKDD, ACM TDS, ACM TMIS, ACM TOIT, ACM TIST) and international conferences (IEEE ICDE, IEEE ICDM, PKDD, PAKDD), 11 edited books, as well as 33 patents (held and filed, 3 US patents). His research interests include data mining, soft computing, artificial intelligence and machine learning, and privacy preserving and security technologies. He is the Editor-in-Chief of the International Journal of Data Science and Pattern Recognition, the Associate Editor for several journals including IEEE TNNLS, IEEE TCYB, IEEE TDSC, among others. He has recognized as the most cited Chinese Researcher respectively in 2018, 2019, 2020, and 2021 by Scopus/Elsevier. He is the Fellow of IET (FIET), ACM Distinguished Member (Scientist) and IEEE Senior Member.

Gautam Srivastava was awarded his B.Sc. degree from Briar Cliff University in the U.S.A. in the year 2004, followed by his M.Sc. and Ph.D. degrees from the University of Victoria in Victoria, British Columbia, Canada in the years 2006 and 2012, respectively. He then taught for 3 years at the University of Victoria in the Department of Computer Science, where he was regarded as one of the top undergraduate professors in the Computer Science Course Instruction at the University. From there in the year 2014, he joined a tenure-track position at Brandon University in Brandon, Manitoba, Canada, where he currently is active in various professional and scholarly activities. He was promoted to the rank of Associate Professor in January 2018. Dr. G, as he is popularly known, is active in research in the field of Cryptography, Data Mining, Security and Privacy, and Blockchain Technology. In his 5 years as a research academic, he has published a total of 150 papers in high-impact conferences in many countries and in high status journals (SCI, SCIE) and has also delivered invited guest lectures on Big Data,Cloud Computing, Internet of Things, and Cryptography at many universities worldwide. He is an Editor of several SCI/SCIE journals. He currently has active research projects with other academics in Taiwan, Singapore, Canada, Czech Republic, Poland, and the U.S.A. He is an IEEE Senior Member and also an Associate editor of the world renowned IEEE Access journal.