frontiers | Frontiers in Computer Science

TYPE Original Research
PUBLISHED 24 April 2025
DOI 10.3389/fcomp.2025.1561899

OPEN ACCESS

EDITED BY
Koorosh Gharehbaghi,
RMIT University, Australia

REVIEWED BY
Shahzad Ashraf,
DHA Suffa University, Pakistan
Amin Hosseinian-Far,
University of Hertfordshire, United Kingdom

*CORRESPONDENCE
Fawad Hussain
✉ fawad.hussain@uettaxila.edu.pk

†These authors have contributed equally to this work and share first authorship

RECEIVED 16 January 2025
ACCEPTED 31 March 2025
PUBLISHED 24 April 2025

CITATION
Hayee S, Hussain F, Yousaf MH, Yasir M, Ahmad S and Viriri S (2025) A benchmark dataset and methodology for fine grained vehicle make and model classification. *Front. Comput. Sci.* 7:1561899. doi: 10.3389/fcomp.2025.1561899

A benchmark dataset and methodology for fine grained vehicle make and model classification

Sobia Hayee[1†], Fawad Hussain[1*], Muhammad Haroon Yousaf[1,2], Muhammad Yasir[3†], Shahzor Ahmad[4] and Serestina Viriri[5]

[1]Department of Computer Engineering, University of Engineering and Technology, Taxila, Pakistan, [2]Swarm Robotics Lab, National Centre for Robotics and Automation (NCRA), University of Engineering and Technology, Taxila, Pakistan, [3]Department of Electrical Engineering, Lahore Leads University, Lahore, Pakistan, [4]Synapsify, Islamabad, Pakistan, [5]School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Durban, South Africa

Urban transportation management increasingly relies on Intelligent Transportation Systems (ITS), where Vehicle Make and Model Recognition (VMMR) plays a vital role in surveillance, traffic monitoring, and infrastructure planning. However, traffic conditions in developing nations such as Pakistan present unique challenges due to unstructured driving practices and lack of lane discipline. We introduce a large VMMR dataset for Pakistan's traffic dynamics to address these challenges. This dataset comprises 129,000 images across 94 vehicle classes. We collected the dataset through web scraping and overhead traffic video recording, followed by an iterative semi-automated annotation process to ensure quality and reliability. For evaluation, we perform a fine-grained analysis using modern deep-learning architectures, including VGG, EfficientNet, and Vision Transformers. Experimental results are obtained through model simulations. These results establish a new benchmark in vision-based traffic analytics for developing countries. Our best-performing model achieves an accuracy of 97.3%, demonstrating the potential of the data set to advance ITS applications.

KEYWORDS

Vehicle Make and Model Recognition (VMMR), Intelligent Transportation Systems (ITS), fine-grained classification, traffic analysis, benchmarking, vehicle data set

1 Introduction

Efficient transportation is a critical challenge in large metropolitan cities. In this regard, Intelligent Transportation Systems (ITS) (Bharadiya, 2023) are essential components of smart city initiatives in urban areas worldwide to achieve optimal, safe, and sustainable utilization of the available transportation infrastructure and achieve maximum traffic efficiency. Automatic vehicle analysis is an important undertaking in any intelligent transportation system, involving vehicle attribute recognition such as vehicle re-identification, vehicle type recognition, and VMMR (vehicle make and model recognition). VMMR has many applications, such as in surveillance for policing and law enforcement, augmenting Automatic License Plate Recognition (ALPR) systems, advanced driver assistance systems (ADAS), electronic toll collection (ETC), self-driving cars, intelligent parking systems, measurement of traffic parameters like vehicle count, speed, and flow, as well as market analysis for car manufacturing companies.

Traffic monitoring through VMMR is a critical tool for collecting statistics that will aid in designing and planning sustainable and efficient transportation infrastructure. Often, simple vehicle counting is insufficient, and the system must capture extended attributes of vehicles. For example, can heavy traffic be separated from lighter vehicles, or can individual vehicles be tracked to determine which routes the drivers typically take? This type of data enables fine-grained analysis and more accurate profiling of users of transportation infrastructure, which is required to guide the design of future upgrades and renovation projects.

VMMR is fraught with complications. The first is vehicle detection; the VMMR system should accurately locate vehicles in video images to perform feature extraction and classification. Numerous vehicle variations, such as color, size, and shape, make the problem difficult. Furthermore, under different lighting conditions and viewpoint variations, the visual properties of vehicles also change dramatically. The next task is to classify the localized image regions into make and model categories. Several issues must be addressed to achieve good classification accuracy. Firstly, the wide range of models and makes seen in practice can render the number of classes to consider rather large, making it a challenging fine-grained classification problem. Next, different models from the same manufacturer (make) frequently share similar shape characteristics and are thus difficult to distinguish. Also, the same model can have various facelifts released by the manufacturer over the years, introducing intra-class variation within this class.

The traditional method of collecting traffic data involves manual data collection and human labor to count vehicles. Alternatively, various technologies could help with the automated detection of vehicles, e.g., inductive loops (Gheorghiu et al., 2021), RADAR, LiDAR, infrared, and acoustic sensors (Wang et al., 2022). Video camera-based traffic monitoring techniques have recently gained popularity due to their widespread availability, low cost, and potential for video analytics-based surveillance applications. In addition, they provide a rich source of information from which researchers can develop new generations of recognition methods.

For a long time, the performance of computer vision techniques was the primary bottleneck for camera-based traffic monitoring systems. However, the advent of deep learning has fundamentally altered the situation. Image classification, in particular, has advanced to an entirely new level in the last decade, approaching human-level accuracy in several domains. The availability of large-scale datasets and computational power have been key players in this transformation.

Most of the available large-scale image-based VMMR datasets come from developed countries (Krause et al., 2013; Sochor et al., 2016; Tafazzoli et al., 2017; Yang et al., 2015). Hence, these data sets provide a good starting point for research purposes. However, traffic dynamics are very different and inconsistent in developing countries such as Pakistan, especially in city thoroughfares. For example, there are no lane markings more often than not, and driving within lanes is not typically the norm. Most existing datasets (e.g., Stanford Cars, CompCars) are designed for structured traffic environments. No existing benchmark dataset captures vehicles in overhead viewpoints under real-world, unstructured traffic conditions as seen in Pakistan and similar developing countries. Our dataset is the first to provide a fine-grained, annotated dataset for VMMR in Pakistani overhead traffic conditions, making it a valuable resource for future ITS and surveillance applications. Key contributions of this research are:

- A novel, large-scale dataset (129,000 images, 94 classes) for overhead vehicle recognition, addressing the lack of datasets tailored to unstructured traffic conditions.
- A semi-automated annotation pipeline that combines deep learning models (VGG) with manual verification for high-quality dataset labeling.
- Benchmark evaluation of several deep learning models (AlexNet, VGG, EfficientNet, ViT) on this dataset, establishing a baseline for future research.
- Insights into overhead VMMR challenges, including occlusions, lighting variations, and dataset scalability.

The rest of the paper is organized as follows: Section 2 presents a detailed review of existing Vehicle Make and Model Recognition (VMMR) methods and datasets. Section 3 outlines the data collection and annotation process, highlighting the methodology used to create a high-quality dataset for overhead traffic analytics. Section 4 presents the experimental setup for baseline model evaluation and thorough dataset analysis. Finally, Section 5 concludes the paper by summarizing the key findings and suggesting potential directions for future advancements in fine-grained VMMR applications.

2 Literature review

We first present existing fine-grained VMMR methods, emphasizing deep learning techniques to contextualize the proposed dataset. In the second section, we review the existing datasets in detail.

2.1 VMMR methods

The general VMMR technique contains feature extraction and classification. Existing literature can be roughly divided into two categories, namely, traditional classification methods, e.g., discriminant analysis (Klecka, 1980), Bayesian methods (Bernardo and Smith, 1994; Ashraf and Ahmed, 2020), and support vector machines (Suthaharan and Suthaharan, 2016), and techniques from the computational intelligence domain, e.g., neural networks (Islam et al., 2019), various combinations of neural networks, fuzzy sets, and genetic algorithms (Gorzalczany, 2012; Arfeen et al., 2021).

2.1.1 Traditional methods

Early approaches to VMMR primarily relied on traditional machine learning techniques, using handcrafted features to represent vehicle images. One idea is to identify cars based on their inherent dimensions, shapes, and textures. These methods rely on the cameras' pose and position. Using shape information, (Gu and Lee, 2013) demonstrated improved car model recognition. Betke et al. (2000) detected and tracked vehicles using symmetry and rear lights, while another approach (Emami et al., 2014; Llorca et al.,

2014) identified cars from backlights, license plates, and global shape features.

Car models are also classified using various features, including low-level features like edge-based, contour point, contourlet transform, corner features, and high-level features like SIFT (Psyllos et al., 2010), SURF (Hsieh et al., 2013), HoG (Dalal and Triggs, 2005), PHOG, and Gabor features Xue (2006). Different studies have shown high accuracy in recognizing car models using these features. For example, Munroe and Madden Munroe and Madden (2005) achieved 97% accuracy while distinguishing between 5 vehicle classes using edge features with K-means. Pearce and Pears (Pearce and Pears, 2011)proposed LNHS for improved corner detection, achieving a 96% correct classification rate. Boonsim and Prakoonwit (2017) addressed night-time identification with a 93% accuracy rate. Clady et al. (2008) developed a method with 93.1% accuracy for identifying vehicle types based on oriented contour points. Baran et al. (2015) used local features like SURF and achieved a 97.2% correct recognition rate for car models. These methods are based on human detection and recognition techniques, utilizing robust and efficient features like HOG, Gabor Filters, and LBP (Shan et al., 2009). Still, they may face severe limitations as the dataset grows regarding the number of very similar classes.

Many VMMR methods follow the intuition that distinguishing features of a fine-grained category, such as the grille of a car, are most naturally represented in 3D object space, which includes both the appearance of the features and their location for an object. Prokaj and Medioni (2009) introduced a top-down, model-based approach for vehicle pose estimation using 3D models, achieving 90% accuracy on 36 classes. Krause et al. (2013) enhanced 2D object representations for 3D applications and utilized 3D CAD models to avoid manual annotation, reaching 94.5% accuracy across 196 car models. Hsiao et al. (2014) used non-parametric 3D curves derived from 2D images for vehicle model representation, with an 87% success rate on an 8-class dataset. Lin Y.-L. et al. (2014) combined 3D model fitting with fine-grained classification for a 90% accuracy on the FG3DCar dataset, highlighting limitations in application scope and performance for large vehicle datasets. Model-based approaches have the advantage of reducing viewpoint dependency in general; however, they remain limited to the basic classes, and producing simple 3D models accurate enough to distinguish between the many makes and models or subtypes of different models appears unlikely to have high success. Furthermore, the inter-class difference is quite significant in vehicle type classification applications, whereas the appearances of various models are very similar in car make and model recognition.

2.1.2 Computationally intelligent methods

The traditional methods laid the groundwork for developing more sophisticated computer vision systems in VMMR. The advent of deep learning and convolutional neural networks (CNNs) marked a significant turning point in VMMR research. Recent literature reveals that convolutional neural networks (CNNs) have established a new benchmark in fine-grained visual classification (Sochor et al., 2016). Pioneering studies by Liu and Wang (2017) and Yang et al. (2015) have underscored the effectiveness of CNNs in this domain. They introduced GoogleNet, an early pre-trained deep learning model, and surpassed conventional methods in fine-grained vehicle classification.

Early investigations primarily explored auxiliary networks to capture local-level information for fine-grained classification. Krause et al. (2015) proposed a method without annotations of parts, using alignment and segmentation concepts to identify informative parts. Similarly, Xiao et al. (2015) employed multiple attention mechanisms to extract relevant image details, integrating them to train deep networks. Zhang et al. (2016) introduced an automatic recognition approach without annotations of objects or parts, extracting distinctive filter responses and learning specific patterns.

Further advancements addressed constraints in pose-normalized representations for fine-grained classification. Zhang et al. Zhang et al. (2014) incorporated semantic part localization in CNNs, achieving state-of-the-art performance. Fu et al. (2017) proposed a recurrent attention model, learning discriminative region attention on multiple scales without bounding boxes. Furthermore, a novel part-stacked CNN (Huang et al., 2016) simultaneously encoded object- and part-level cues to model subtle differences.

Spatially Weighted Pooling (SWP) layers introduced by Hu et al. (2017) in CNNs enhanced feature pooling by learning discriminative spatial units, outperforming prior methods. Ma et al. (2019) enhanced CNN generalization by inserting a Channel Max Pooling (CMP) layer, significantly reducing parameters while improving classification accuracy. Lightweight CNNs (Zhang et al., 2018) optimized parameters through pre-training and fine-tuning on the VMMR dataset, achieving notable results.

Innovative architectures were devised to consolidate informative image parts. Lam et al. (2017) proposed a heuristic function to score proposals unified via LSTM networks into a deep recurrent architecture. Lin et al. (2015) introduced a valve linkage function (VLF) enhancing back-propagation, particularly in deep localization and alignment systems.

Several loss functions were introduced to enhance neural network performance. Deep CNNs with Large-Margin Softmax (L-softmax) loss Liu et al. (2016b) improved feature discriminability, while Center Loss (Wen et al., 2016) facilitated inter-class dispersion and intra-class compactness. Focal Loss addressed class imbalance, focusing on hard-set examples, while a novel loss function proposed by Lin et al. (2017) penalized misclassification probabilities.

Hayee et al. (2023) proposed a model that extracts deep features through the FC layer of a fine-tuned CNN and produces the features that best describe a vehicle for fine-grained vehicle classification using the Fisher discriminative least squares regression (FDLSR) module.

In conclusion, recent advances in CNN architectures, loss functions, and feature optimization techniques have significantly increased fine-grained visual classification performance, particularly in vehicle recognition tasks.

These advantages come with many challenges, including calculation complexity, where deep CNNs with numerous layers and parameters have a significant computational overhead (Yu et al., 2020). Other challenges include optimizing CNN models for fine-grained classification to balance computational speed and

TABLE 1 Summary of datasets for fine-grained vehicle classification.

No.	Dataset name	Year	Number of images	Number of classes	Salient features
1.	Stanford Cars	2013	16,185	196	• Fine-grained categories include make, model, and year. • High-resolution images
2.	CompCars	2015	Over 200,000	1,716	• Surveillance-natured and web-natured data. • Categories include make and model. • Viewpoint annotations available
3.	VeRi-776	2016	Over 50,000	776	• Real-world urban surveillance data. • Suitable for vehicle re-identification. • Attributes include bounding boxes
4.	VehicleID	2016	221,763	26,267	• Large-scale dataset suitable for vehicle re-identification. • Single frontal view per vehicle
5.	PKU VehicleID	2016	Over 110,000	13,134	• Focuses on vehicle re-identification. • Images captured from multiple real-world scenarios
6.	BoxCars 116k	2018	116,000	Over 500	• Benchmark for fine-grained recognition. • 3D bounding boxes available. • Attributes include multiple angles and conditions

accuracy degradation (Habib and Qureshi, 2022) and capturing subtle differences between classes, which can lead to complex models with many parameters. This complexity can lead to overfitting, especially given the typically smaller datasets available for fine-grained classification tasks (Yun et al., 2023).

Training deep CNNs for fine-grained classification demands a large amount of data to avoid overfitting, making it difficult to train these networks effectively with a small dataset (Yu et al., 2020).

2.2 VMMR datasets

Most of the earlier fine-grained image classification benchmark datasets contain only a few thousand (or less) training images. These vehicle data sets only include a selection of brands and models (Pearce and Pears, 2011), or only categorize cars at a high level (ie SUV, truck, sedan) (Ma and Grimson, 2005), and are only helpful for vehicle-related tasks such as identification and pose estimation (Lin Y.-L. et al., 2014). Most of the data sets focus on organized driving situations. This approach is generally associated with well-defined infrastructure, such as lanes, clearly defined traffic participant groups, a variety of motifs or backgrounds of the same, and strict adherence to traffic regulations.

A unique data set is required for road scene comprehension in unstructured situations when the assumptions mentioned above are mostly unmet. It is also essential to have many images for each class to cover the vast range of view angle and illumination changes and the pretty wide range of appearance changes within the same category. Fine-grained car classification (FGCV) datasets are typically categorized into web-image, surveillance-image, and hybrid datasets. Web image datasets collect car images from public websites featuring various car models in various poses and viewpoints, leading to significant appearance variations. Surveillance image data sets consist of car images captured by fixed surveillance cameras in different traffic scenarios, primarily from the front view. Hybrid data sets combine elements from both the web and surveillance images. Unlike traditional classification datasets, fine-grained car datasets can be divided into categories based on various attributes. One of the first benchmark data sets researchers still use today is Stanford Cars (Krause et al., 2013). The authors present a dataset containing 16,000+ images of 196 vehicle classes, including make, model, and year. The data is split almost 50/50 between training and testing images, with 8,144 training images and 8,041 testing images. The image resolution is 360 × 240 pixels. The CompCars dataset (Yang et al., 2015) is another widely used dataset. This dataset has two categories: web-nature and surveillance-nature. The web-natured part comprises 136,727 vehicles from 153 car brands with 1716 car models, photographed from various angles and covering many commercial car models from the last ten years. Most of these models are Chinese, while the surveillance part comprises 44,481 frontal images of vehicles captured by surveillance cameras. The CompCars dataset was created for fine-grained automobile categorization, attribute prediction, and verification. Liu et al. (2016a) presented the "VehicleID" dataset, which contains 221,763 images of 26,267 vehicles. The dataset includes images of 250 vehicle models, with 110,178 images of 13,134 vehicles for training and 111,585 images of 13,133 vehicles for testing. Multiple real-world surveillance cameras capture the images in the data set in a small city in China during the day. Each vehicle in the data set has at least two images, including more than 900,000 labeled images with vehicle model information. The dataset is suitable for vehicle search-related tasks and is organized to assist in vehicle retrieval and re-identification experiments.

VMMRdb (Vehicle Make and Model Recognition Database) (Tafazzoli et al., 2017) is a large-scale dataset designed for vehicle make and model recognition, a specific application of fine-grained classification. The dataset has over nine million images, making it one of the most extensive publicly available datasets

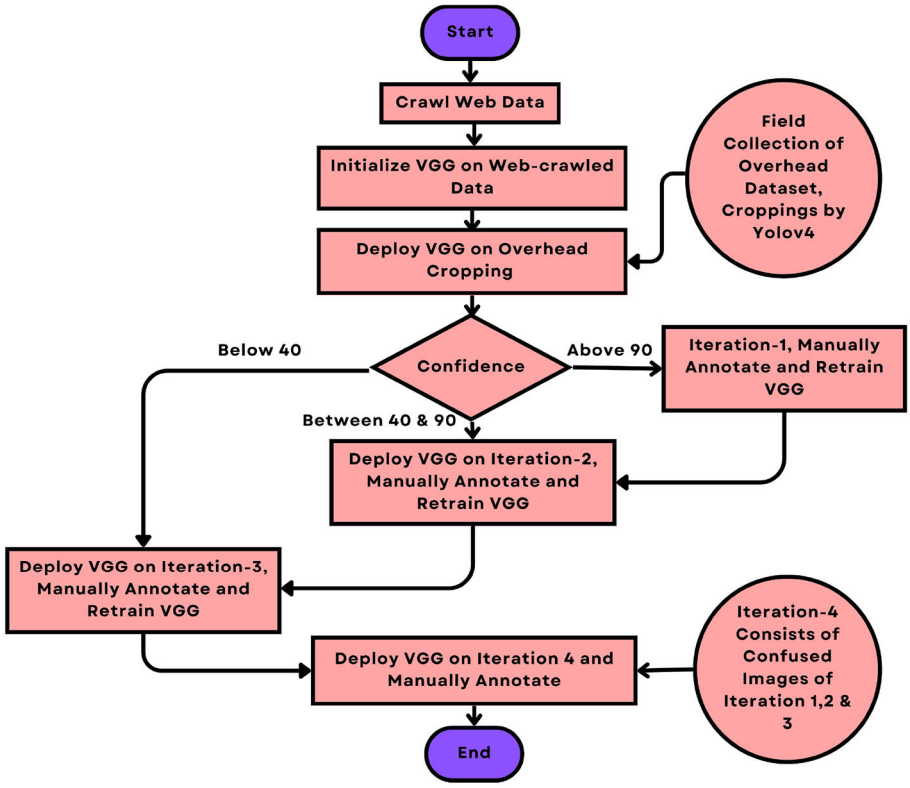

FIGURE 1
Dataset collection procedure.

TABLE 2 Web crawled data statistics.

Number of classes	94
Total number of images	36,164
Average number of images per class	385
Minimum number of images per class	321
Maximum number of images per class	1,005

for vehicle recognition. Covers more than 2,950 models and years combined, providing many vehicles for recognition tasks. Images from the Internet in multiple countries and environments contribute to diversity. This includes cars from different periods, conditions (new or used), and modifications that reflect real-world variability. Sochor et al. (2018) presented BoxCars116K. The dataset contains 116,286 images of 27,496 vehicles. There are 693 fine-grained classes in the dataset, which include make, model, submodel, and model year information. In the hard split, there are 11,653 tracks (51,691 images) for training and 11,125 tracks (39,149 images) for testing. The data set includes information on the 3D boundary box for each vehicle, an image with a background mask extracted by background subtraction, and various attributes such as boundary boxes, vehicle types, and colors.

Recent vehicle data sets such as EuroCity (Braun et al., 2019) focus on pedestrians but include vehicle annotations in urban traffic scenes across European cities, and KITTI-360 (Liao et al., 2022) provide 360-degree view annotations, including vehicles. These datasets are not vehicle-specific and are not suitable for fine-grained classification tasks. A summary of notable VMMR datasets is given in Table 1.

Few or no datasets are available on vehicles such as autorickshaws, tempo, trucks, etc. In addition, the images in the dataset should be taken in varied weather conditions, including daylight, evening, and night. The dataset must have various illumination variations, distances, viewpoints, etc.

A dataset that accurately trains a CNN-based classifier for fine-grained image classification must have the following properties:

- High-Quality Annotations: The data set should contain true and dispositioned annotations for each image. This is especially true for fine-grained classification, where small feature changes might incorrectly classify an object (Wah et al., 2011).
- Diverse Representations: The images should represent various settings, poses, lights, and occlusions to provide loads of pattern variations to prevent the model from getting trapped on a limited set of attributes (Krause et al., 2013).
- Balanced Class Distribution: Diversity in training data is important to avoid that the model is biased towards classes with a greater presence in the data set (the representations of its classes should be as close to each other as possible) (Zhou et al., 2016).

FIGURE 2
Honda City fifth generation from web-crawled data.

TABLE 3 Video data statistics.

Duration of video data	15 h
Total number of video clips	74
Total number of frames	1.8 million
Frames per second	30/60
Conditions	Early morning, afternoon, late evening, sunset
Number of locations	6

- Sufficient Sample Size: For a deep learning model, the number of images should be the same per class, which is crucial to achieving efficient training (Sun et al., 2014).
- Consistent Annotation Criteria: The basis for annotating and categorizing pictures must be the same in all instances to prevent learning from being interfered with by insufficient labels or inconsistent properties (Van Horn et al., 2015).
- Presence of Hierarchical Labels: Fine-grained classification can be improved by adding hierarchical labels (Valan et al., 2019)
- Multiple Instances Per Class: This enables the model to classify more diverse and robust features. The data should contain several examples per class with pictures of the subject studied taken in different situations and conditions (Deng et al., 2009).

Together, these properties will achieve the desired result of a dataset that accurately trains a deep learning (CNN)-based classifier. The inputs deal with specific points regarding the fine-grained dataset model, such as the data collection, annotation, and utilization process associated with these challenges.

3 Proposed dataset

The article presents a novel data set comprising 129,000 images in 94 vehicle classes explicitly tailored to unique traffic conditions in Pakistan. Due to the large dataset size, we primarily stored it on a local external hard disk; this enabled efficient data retrieval and processing. A copy of the dataset was also uploaded to Microsoft OneDrive for remote access and backup. This ensures data security and protects against hardware failure losses. The research demonstrates a robust methodology to collect and classify vehicular data. It performs well even under challenging conditions, such as occlusions and low lighting. This highlights the resilience of the approach to common issues in collecting overhead traffic data for congested road scenarios. The following section describes in detail the dataset collection mechanism, its quantitative and qualitative analysis, and, as a final step, the traffic analytics performed on the collected dataset.

3.1 Data collection

Two types of dataset collection have been conducted. The first type involves web-crawled data, which was gathered using a web scraping tool and consists primarily of front-view and rear-view images. The second type is derived from various overhead traffic videos recorded in different locations within the twin cities of Rawalpindi and Islamabad, each lasting fifteen minutes. This approach aims to include images captured from an overhead viewpoint. Figure 1 shows collection procedure. To date, there are 94 common vehicle categories on Pakistani roads.

3.1.1 Web-crawled data

A scraping tool is developed to scrape images from the Internet. The scrapper takes vehicle names and tags as input and then scraps data from Google, Yahoo, and Bing, depending on the nature of the tags. The scraped data are saved in the form of raw images. There are a lot of repetition and out-of-context images from the desired ones.

The raw images are manually cleaned and annotated. Repetitions and out-of-context images are removed. The Web-crawled dataset consists of 36,164 images with 94 different vehicle classes. Its statistics are shown in Table 2.

Figure 2 shows some images of Honda City fifth generation from web-crawled data.

3.1.2 Overhead-image data

There is minimal vehicle image data on the Internet from overhead viewpoints. To remedy this, almost 15 hours of video data

FIGURE 3
Honda City fifth generation from overhead image data.

TABLE 4 Cropped images placed in each iteration.

Iteration	Confidence	No of images
1	90–100	31,118
2	40–89	56,462
3	0–39	15,862

were collected from different locations in Islamabad. The statistics of this video data set are shown in Table 3.

We took ten-second sections from each video to use as test sequences to create our dataset. Using a version of the YOLO-v4 (Jiang et al., 2022) software trained on the MS COCO dataset (Lin T. -Y. et al., 2014), we specifically focused on identifying specific types of vehicles: bikes, bicycles, buses, trucks, and cars. We then extracted sections of images from this 15-hour collection of video footage. We only kept the image sections that were 64 by 64 pixels or larger to ensure good image quality, and we picked these sections from every 100th frame. In this way, we could avoid collecting too many similar images. From the ten-second test sequences, we gathered 1,744 image sections. For the training sequences, which comprise the rest of the video duration, we obtained 103,442 image sections. Together, we collected 105,186 image sections from the 15-h video dataset. These image sections are shown in Figure 3, showing examples of image croppings we took from the videos.

We trained VGG (Vedaldi and Zisserman, 2016) on the complete web-crawled dataset and deployed it on overhead-cropped images. Depending on the model's confidence level with each crop, we segregated the overhead dataset into three different iterations. The crops above 90 percent confidence are placed in Iteration 1, those with confidence between 40 and 90 percent confidence are placed in Iteration 2, and those below 40 percent confidence are placed in Iteration 3. Table 4 shows the total number of crops placed in each iteration.

The VGG model, trained on a web-crawled dataset, poorly classified overhead croppings. To remedy this problem, we manually cleaned the overhead data, iteratively trained the VGG model, and deployed it in the next iteration. Images with a confidence level greater than 90 were included in the first iteration, images with a confidence level between 40 and 90 were included in the second iteration, and images with fewer than 40 were included in the third iteration.

Each iteration produces cleaned images, garbage images, and ambiguous images. Starting from web-crawled data, cleaned images are combined with the dataset from the previous iteration and used to retrain VGG. Garbage images cannot be used and are discarded. Ambiguous images need deeper visualization and are segregated separately. These ambiguous images are included in the final iteration of the VGG. Annotations are performed manually in each iteration. The statistics of each iteration are summarized in Table 5.

Figures 4, 5 show some ambiguous and discarded images.

After cleaning the overhead crops, 92,836 total overhead images were cleaned. Combined with the web-crawled dataset, 129,000 images of 94 classes were finally collected. The data set is collected iteratively. VGG and AlexNet are used in parallel for the collection, and VGG performed better. We used a small VGG (Ioannou et al., 2015). It has 11 layers with 13,766,754 parameters, compared to VGG-16, which has almost 138 million. The second CNN architecture implemented is AlexNet (Alom et al., 2018), consisting of 8 layers with five convolutional layers and three fully connected ones. There are 58,671,966 parameters in it. The accuracy of the model to classify images from each iteration and the time consumed by its correction are shown in Table 6.

In an iterative process, a significant decrease in the time required for manual image cleaning is observed. Initially, considerable time was expended due to redundant and irrelevant web-crawled images. Subsequently, time was also consumed in addressing issues related to viewpoint overhead. However, the time expenditure decreased as the process advanced, improving the model's accuracy.

4 Dataset analysis and results

We analyzed our data set, focusing on its utility, reliability, and scalability to establish it as a vehicle manufacturer and model recognition benchmark.

TABLE 5 Data cleaning statistics.

Iteration	Total images	Training accuracy	Validation accuracy	Training loss	Validation loss	Cleaned images	Ambiguous images	Discarded images
1	61,616	93.15 %	86.01 %	0.2084	0.6674	25,452	5,290	376
2	103,139	92.49 %	88.09 %	0.2281	0.4873	41,523	12,924	2,015
3	114,395	93.23 %	88.90 %	0.2029	0.4863	11,256	3,583	1,023
4	127,500	93.33 %	88.35 %	0.1986	0.5132	13,105	7,834	858

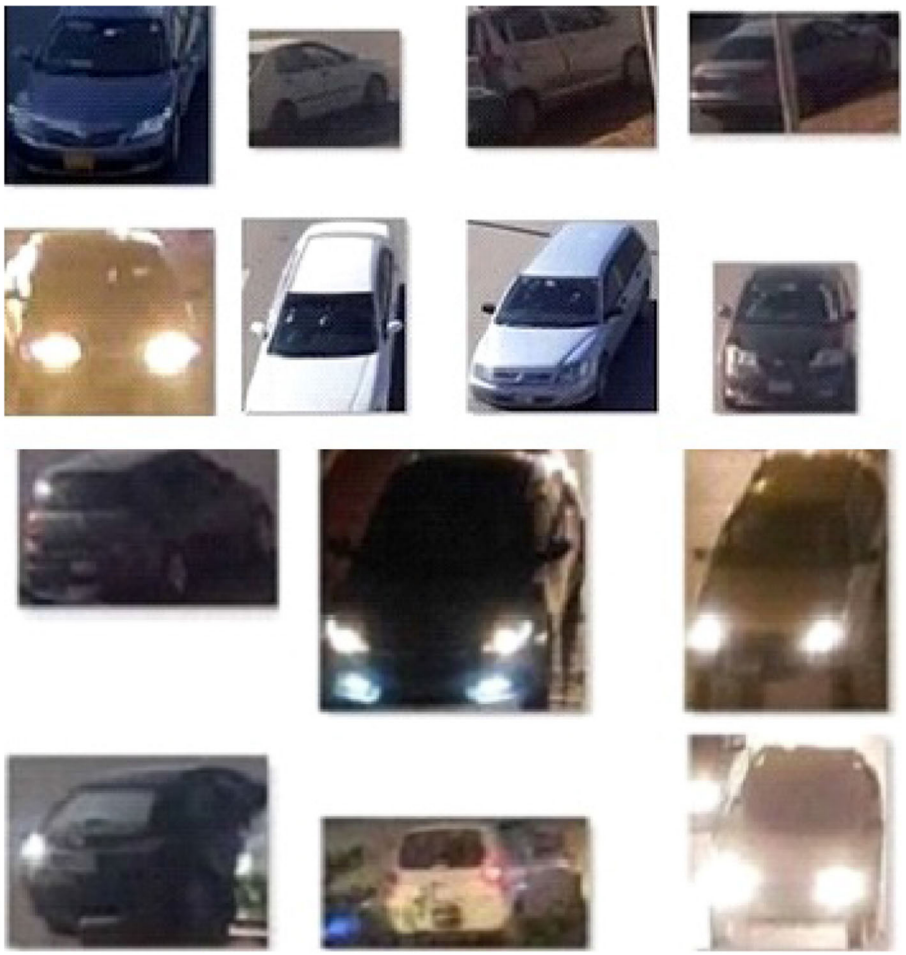

FIGURE 4
Ambiguous images from each iteration.

4.1 Data annotation quality

Data annotation is carried out semi-automatically. As a first step, the web-crawled data with available annotations is manually rechecked. These double-checked class labels annotate new images in the next iteration. The resulting annotations are manually confirmed once again. This process is carried out in each iteration. Details of each iteration are given in Section 3.1. A strict naming convention is followed to ensure consistency in label naming. This will reduce errors and improve the reliability of the data set. The naming convention followed is Make_Model_Generation; for example, Honda_City_5 means Honda City 5th generation.

4.2 Optimum image size and data augmentation

For VMMR datasets, the image size is crucial to balance accuracy, processing power, and storage. Commonly recommended image sizes are; Small Resolution (64x64 to 128x128 pixels) (Tang et al., 2015), medium Resolution (224 × 224 to 256 × 256 pixels) (Krizhevsky et al., 2017) and high resolution (512 × 512 to 1,024 × 1,024 pixels) (Li et al., 2023). Since our data set is intended for overhead traffic analytics, we did not consider high-resolution images, as they require significant computational resources and larger storage.

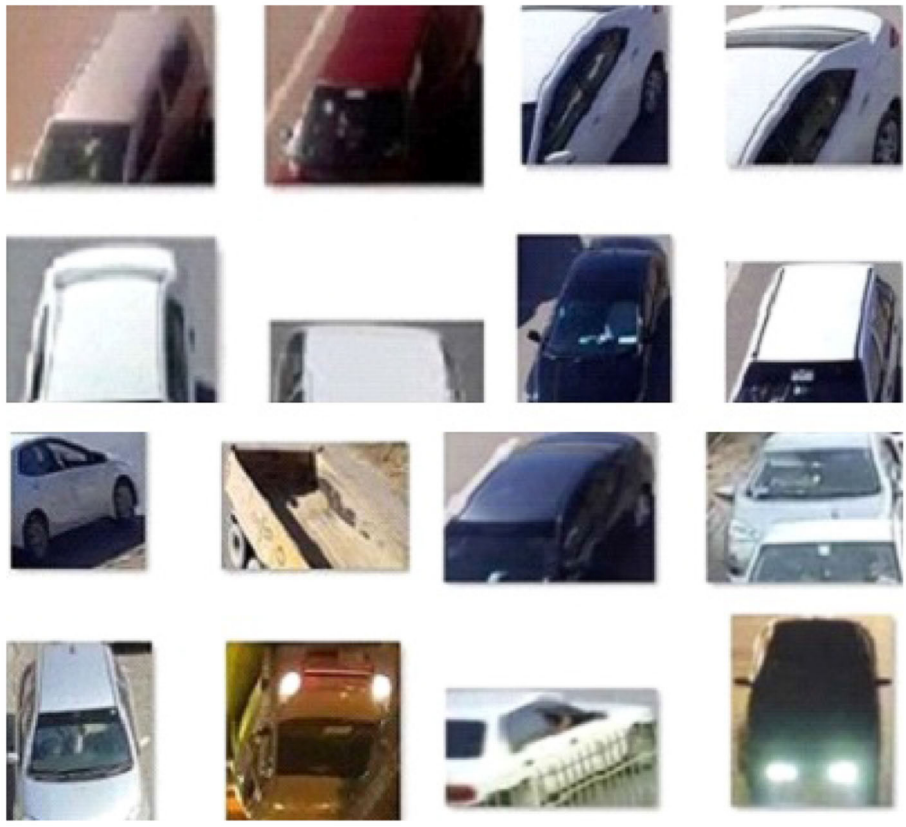

FIGURE 5
Discarded images from each iteration.

TABLE 6 Model accuracy in each iteration and correction time.

Iteration	Model accuracy	Number of images	Correction time	Correction time per image (sec)
1	25.18 %	25,452	18 days,7 h/day	18
2	77.70 %	41,523	20 days, 7 h/day	12
3	85.77 %	11,256	4 days, 8 h/day	10
4	86.40 %	13,105	5 days, 8 h/day	7

TABLE 7 Image sizes after each iteration.

Image size	Total images	Greater than 227 × 227	Less than 227 × 227	Average size
Web crawled	36,164	24,205	11,959	777 × 538
Iteration 1	25,452	6,208	19,244	207 × 222
Iteration 2	41,523	10,135	31,388	214 × 212
Iteration 3	11,256	2,769	8,487	217 × 209
Iteration 4	13,105	2,338	10,767	179 × 173
Last 10 s	1,500	465	1,035	241 × 243
Total data	129,000	46,120	82,880	368 × 301

After each iteration, the data set has web-crawled images and overhead traffic images. Table 7 shows image sizes after each iteration.

Different experiments were performed to choose the image size (small or medium resolution) and set the model parameters. We trained a small VGG with a 64 × 64 and 227 × 227 input image

size to decide on the optimum image size. The statistics of both experiments are shown in Table 8.

Based on the experimental results, a model size of 227 × 227 was chosen. These results are consistent with the fact that very low-resolution images are not ideal for capturing detailed make- and model characteristics but can work for primary classification.

We also performed data augmentation to increase the number of variants in the dataset. Data augmentation is applied to each image in the dataset to increase the diversity of the data and improve the model's generalization ability. The details of the augmentation parameters, their values, and their functions are given in Table 9.

TABLE 8 Image size experiment results.

Image size	227 × 227	64 × 64
Training accuracy	93.38 %	86.83 %
Validation accuracy	82.22 %	76.92 %
Training loss	0.2071	1.0051
Validation loss	0.9623	3.6805

TABLE 9 Data augmentation parameters.

Parameter	Value	Operation
Rotation	rotation_range=0.1	Random rotation of each image within a range of ±10% of 360 degrees
Shift	width_shift_range=0.1, height_shift_range=0.1	Shift the image horizontally and vertically by up to 10% of the image's width or height, respectively
Shear	shear_range=0.2	Shear (slant the shape of an object) transformation up to 20 degrees
Zoom	zoom_range=0.2	Random zooming to the image, either zooming in or out by up to 20%
Horizontal flip	horizontal_flip=True	Each image is flipped horizontally, doubling the number of unique samples by creating mirror images
Fill	fill_mode="nearest"	Newly created empty space from transformations is filled by the nearest pixel values along the edges, creating a seamless effect

These transformations introduce variations in rotation, position, shear, zoom, and flipping to each image. This results in a more diverse dataset, which makes the model robust to different orientations, positions, and scales. Hence, it improves its ability to generalize in real-world scenarios.

4.3 Baseline model training and evaluation

We train several popular Convolutional Neural Network (CNN) architectures as baseline models to evaluate data set performance. To assess the flexibility of our proposed dataset, we trained and evaluated AlexNet, VGG, and EfficientNet (Tan, 2019) multiple times, and their quantitative results are represented as 50 epochs per iteration per the standard dataset. We used stochastic gradient descent and batch normalization as regularizers. The learning rate of fully connected layers is kept at 0.0001, and we have trained no model for more than 200 epochs.

We created a standard test dataset to evaluate the models that have been trained iteratively. This standard test data set includes the following:

- 5% of the validation dataset
- 5% of the complete test dataset
- 1,500 images from the test sequence videos

The standard test data set contains 14,154 images and is used to evaluate the performance of the trained models. The statistics of all three models for each iteration are shown in Table 10.

The training plots of all four models for all iterations are shown below in Figure 6.

The models trained on data collected after the fourth iteration are the final models. We notice that VGG performs better in all iterations than AlexNet. EfficientNet performs significantly better than older architectures such as AlexNet and VGG for fine-grained classification tasks. This is due to its design principles that emphasize efficiency and scalability while maintaining high performance (Tan, 2019).

4.4 Scalability analysis

We analyze scalability to examine how well our data set performs across models of different sizes, complexities, and

TABLE 10 Model performance metrics after each iteration.

Model	AlexNet				VGG				EfficientNet			
Iteration	1	2	3	4	1	2	3	4	1	2	3	4
Training accuracy (%)	93.15	95.14	95.62	95.43	96.34	92.49	93.23	93.33	88.64	96.35	98.90	100.00
Validation accuracy (%)	68.60	84.46	84.36	85.15	78.18	88.09	88.90	88.35	70.03	83.97	98.01	97.30
Training loss	0.1182	0.1458	0.1360	0.1423	0.2084	0.2281	0.2029	0.1986	0.1930	0.037	0.011	0.004
Validation loss	1.2738	0.8794	0.9977	0.9477	0.6674	0.4873	0.4863	0.5132	0.6674	0.017	0.006	0.010

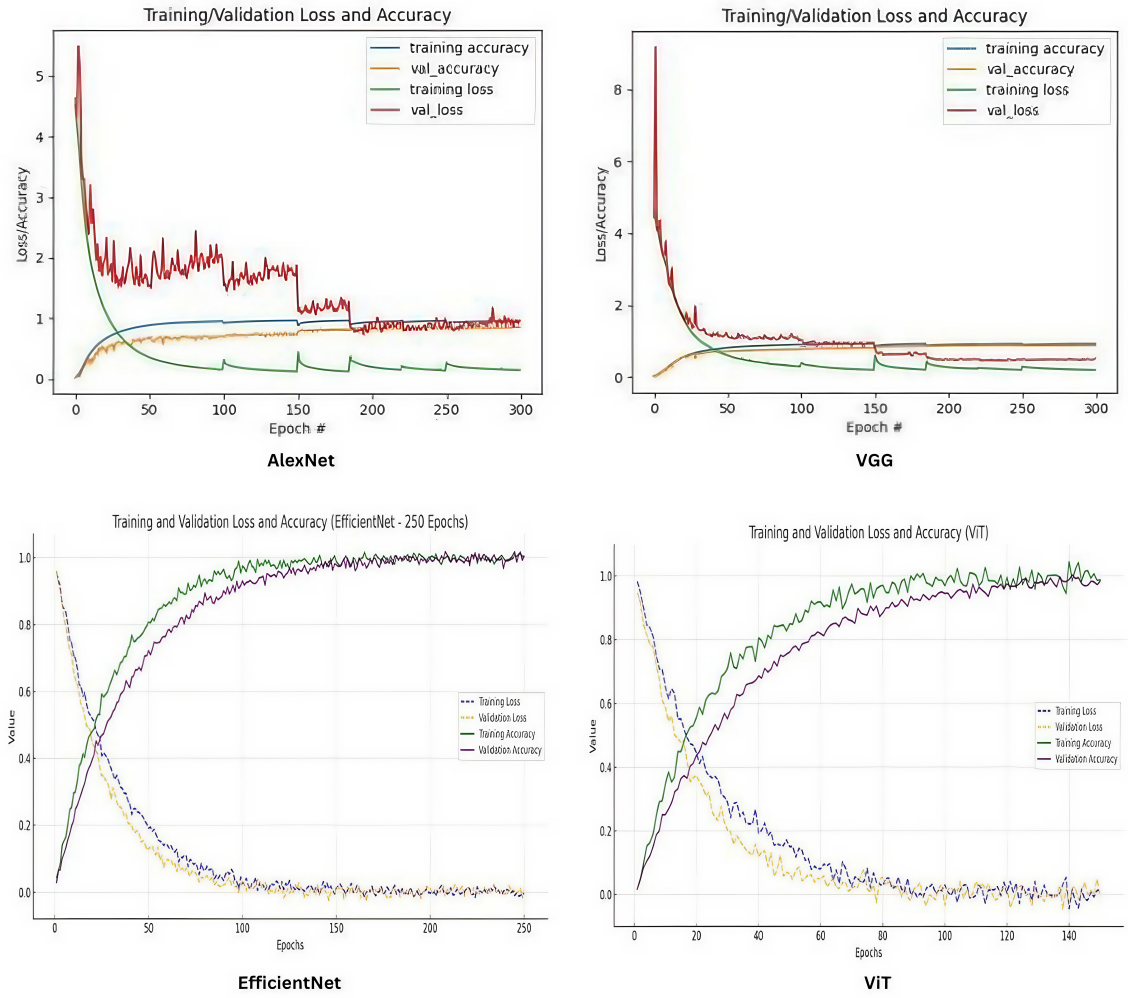

FIGURE 6
Overall training plots.

architectures. In the previous section, our dataset performed well for various CNN models, from the classic AlexNet architecture to the recent EfficientNet. Traditionally, transformer architectures are employed for natural language processing, but recently, they have shown remarkable performance in image classification tasks. Vision Transformed (ViT) (Dosovitskiy, 2020) is highly expressive and can capture complex patterns that CNNs might miss. ViT splits images into patches instead of using convolutional filters. A dataset with rich feature variety is required to validate these patches. ViT splits images into patches instead of using convolutional filters. A data set with rich feature variety is necessary for these patches to be useful. ViT testing can highlight the strength of critical feature representations (such as vehicle make and model details). ViT-B/16 was deployed; hyperparameters are listed in Table 11. The model statistics after every 50 epochs are shown in Table 12.

ViT performs well due to the self-attention mechanism (Touvron et al., 2021), which helps to capture subtle differences between images. Both EfficientNet and ViT show comparable performance. ViT can outperform EfficientNet if the dataset is extensive because it can better capture global relationships. The accuracy of EfficientNet improves more steadily, particularly in the early epochs, while ViT requires more epochs to achieve similar accuracy. ViT eventually catches up, but EfficientNet performs better in the initial stages.

We observed that Vision Transformers (ViT) outperformed Convolutional Neural Networks (CNNs) in fine-grained vehicle make and model recognition. While CNNs excel in capturing local spatial hierarchies, ViTs leverage self-attention mechanisms to model global dependencies, making them more effective for detailed classification. CNNs employ weight sharing, which makes them more efficient; ViTs require extensive pretraining and are computationally more demanding. The study also highlighted CNNs' limitations, including difficulty capturing long-range dependencies, fixed kernel constraints, and reduced performance distinguishing visually similar vehicle models. This underscores the importance of balancing accuracy and computational cost in real-world applications (Khalil et al., 2023).

Google Colab was used for model training, while local systems were used for image cropping and testing. In particular, cropping

TABLE 11 Hyperparameters for ViT-B/16.

Hyperparameter	Value	Description
Patch size	16 × 16	The input image is divided into patches of this size
Embedding dimension	768	Size of the embedding for each patch
Number of transformer layers	12	The depth of the transformer, representing the number of encoder layers
Dropout rate	0.1	Dropout rate used for regularization to prevent overfitting
Learning rate	0.0003	Initial learning rate
Batch size	1,024	Number of samples per batch used in training

TABLE 12 ViT statistics after each iteration.

	Iteration 1	Iteration 2	Iteration 3	Iteration 4
Training accuracy	80.94 %	96.5 %	98.2%	99.88%
Validation accuracy	70.08 %	93.99 %	95.3%	98.53%
Training loss	0.1141	0.035	0.009	0.006
Validation loss	0.0763	0.0112	0.023	0.021

of approximately one image per second, training of roughly 68 images per second, and testing of nearly eight images per second were achieved. With the final validation accuracy of VGG at 88.35%, AlexNet at 85.15 %, EfficientNet at 97.3%, and ViT at 98.5%.

5 Conclusions and suggestions for future work

We have established that, before our investigation, no benchmark data set was available for vision-based traffic analytics in the Pakistani context. Furthermore, no pre-trained deep neural network existed to categorize overhead traffic image data effectively.

To address this problem, we collected a large-scale dataset through extensive field data collection. The data set was then annotated using an iterative, semi-automated approach that employed existing deep neural networks in a novel and conscientious manner. The classification results obtained using existing standard deep neural networks demonstrated promising outcomes and have the potential to facilitate insightful traffic analytics.

The data set comprises 129,000 images placed into 94 different classes and is now available as a benchmark for future research focused on fine-grained vehicle classification and traffic analytics. This data set provides a valuable resource for researchers and practitioners seeking to advance the field of vision-based traffic analytics in the Pakistani setting, thus contributing to developing more effective and efficient traffic management strategies.

The dataset can now be used as a benchmark for further traffic analytics. It can be used to explore tracking algorithms to achieve real-time traffic analytics by exploring multiclass object detectors. It can also be used for a single-pass pipeline. The hardware implementation of the algorithms mentioned above can be done. Single-image-based speed estimation can also be processed further.

The paper sets a new benchmark for vision-based traffic analytics in developing countries by providing a comprehensive and well-annotated dataset. This data set can be a valuable resource for future research and practical applications in ITS.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

SH: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. FH: Investigation, Methodology, Supervision, Validation, Writing – original draft, Writing – review & editing. MHY: Conceptualization, Formal analysis, Methodology, Project administration, Supervision, Validation, Writing – original draft, Writing – review & editing. MY: Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Writing – original draft, Writing – review & editing. SA: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. SV: Investigation, Project administration, Supervision, Validation, Writing – original draft, Writing – review & editing.

Funding

The author(s) declare that no financial support was received for the research and/or publication of this article.

Conflict of interest

SA was employed at Synapsify.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that no Gen AI was used in the creation of this manuscript.

Publisher's note

References

Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., et al. (2018). The history began from alexnet: a comprehensive survey on deep learning approaches. *arXiv* [preprint] arXiv:1803.01164. doi: 10.48550/arXiv.1803.01164

Arfeen, Z. A., Sheikh, U. U., Azam, M. K., Hassan, R., Faisal Shehzad, H. M., Ashraf, S., et al. (2021). A comprehensive review of modern trends in optimization techniques applied to hybrid microgrid systems. *Concurr. Comp.: Pract. Exp*. 33:e6165. doi: 10.1002/cpe.6165

Ashraf, S., and Ahmed, T. (2020). "Sagacious intrusion detection strategy in sensor network," in *2020 International Conference on UK-China Emerging Technologies (UCET)* (IEEE), 1–4.

Baran, R., Glowacz, A., and Matiolanski, A. (2015). The efficient real-and non-real-time make and model recognition of cars. *Multimed. Tools Appl*. 74, 4269–4288. doi: 10.1007/s11042-013-1545-2

Bernardo, J. M., and Smith, A. F. (1994). *Bayesian Theory Wiley*. New York: Wiley, 49. doi: 10.1002/9780470316870

Betke, M., Haritaoglu, E., and Davis, L. S. (2000). Real-time multiple vehicle detection and tracking from a moving vehicle. *Mach. Vis. Appl*. 12, 69–83. doi: 10.1007/s001380050126

Bharadiya, J. P. (2023). Artificial intelligence in transportation systems a critical review. *Am. J. Comp. Eng*. 6, 35–45. doi: 10.47672/ajce.1487

Boonsim, N., and Prakoonwit, S. (2017). Car make and model recognition under limited lighting conditions at night. *Pattern Analy. Appl*. 20, 1195–1207. doi: 10.1007/s10044-016-0559-6

Braun, M., Krebs, S., Flohr, F., and Gavrila, D. M. (2019). Eurocity persons: a novel benchmark for person detection in traffic scenes. *IEEE Trans. Pattern Anal. Mach. Intell*. 41, 1844–1861. doi: 10.1109/TPAMI.2019.2897684

Clady, X., Negri, P., Milgram, M., and Poulenard, R. (2008). "Multi-class vehicle type recognition system," in *Artificial Neural Networks in Pattern Recognition: Third IAPR Workshop, ANNPR 2008* (Paris, France: Springer), 228–239.

Dalal, N., and Triggs, B. (2005). "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (San Diego, CA: IEEE), 886–893.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition* (Miami, FL: IEEE), 248–255.

Dosovitskiy, A. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* [preprint] arXiv:2010.11929. doi: 10.48550/arXiv.2010.11929

Emami, H., Fathi, M., and Raahemifar, K. (2014). Real time vehicle make and model recognition based on hierarchical classification. *Int. J. Mach. Learn. Comp*. 4:142. doi: 10.7763/IJMLC.2014.V4.402

Fu, J., Zheng, H., and Mei, T. (2017). "Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu, HI: IEEE), 4438–4446.

Gheorghiu, R. A., Iordache, V., and Stan, V. A. (2021). "Urban traffic detectors-comparison between inductive loop and magnetic sensors," in *2021 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)* (Pitesti: IEEE), 1–4.

Gorzalczany, M. B. (2012). *Computational Intelligence Systems and Applications: Neuro-Fuzzy and Fuzzy Neural Synergisms*. Springer Science & Business Media.

Gu, H.-Z., and Lee, S.-Y. (2013). Car model recognition by utilizing symmetric property to overcome severe pose variation. *Mach. Vis. Appl*. 24:255–274. doi: 10.1007/s00138-012-0414-8

Habib, G., and Qureshi, S. (2022). Optimization and acceleration of convolutional neural networks: a survey. *J. King Saud Univer.-Comp. Inform. Sci*. 34, 4244–4268. doi: 10.1016/j.jksuci.2020.10.004

Hayee, S., Hussain, F., and Yousaf, M. H. (2023). A novel fdlsr-based technique for view-independent vehicle make and model recognition. *Sensors* 23:7920. doi: 10.3390/s23187920

Hsiao, E., Sinha, S. N., Ramnath, K., Baker, S., Zitnick, L., and Szeliski, R. (2014). "Car make and model recognition using 3d curve alignment," in *IEEE Winter Conference on Applications of Computer Vision* (Steamboat Springs, CO: IEEE), 1.

Hsieh, J.-W., Chen, L.-C., Chen, D.-Y., and Cheng, S.-C. (2013). "Vehicle make and model recognition using symmetrical surf," in *2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance* (Krakow: IEEE), 472–477.

Hu, Q., Wang, H., Li, T., and Shen, C. (2017). Deep cnns with spatially weighted pooling for fine-grained car recognition. *IEEE Trans. Intellig. Transp. Syst*. 18, 3147–3156. doi: 10.1109/TITS.2017.2679114

Huang, S., Xu, Z., Tao, D., and Zhang, Y. (2016). "Part-stacked cnn for fine-grained visual categorization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV: IEEE), 1173–1182.

Ioannou, Y., Robertson, D., Shotton, J., Cipolla, R., and Criminisi, A. (2015). Training cnns with low-rank filters for efficient image classification. *arXiv* [preprint] arXiv:1511.06744. doi: 10.48550/arXiv.1511.06744

Islam, M., Chen, G., and Jin, S. (2019). An overview of neural network. *Am. J. Neural Netw. Appl*. 5:7–11. doi: 10.11648/j.ajnna.20190501.12

Jiang, P., Ergu, D., Liu, F., Cai, Y., and Ma, B. (2022). A review of yolo algorithm developments. *Procedia Comput. Sci*. 199, 1066–1073. doi: 10.1016/j.procs.2022.01.135

Khalil, M., Khalil, A., and Ngom, A. (2023). A comprehensive study of vision transformers in image classification tasks. *arXiv* [preprint] arXiv:2312.01232. doi: 10.48550/arXiv.2312.01232

Klecka, W. R. (1980). *Discriminant Analysis, Volume 19*. Thousands Oak: SAGE

Krause, J., Jin, H., Yang, J., and Fei-Fei, L. (2015). "Fine-grained recognition without part annotations," in *Proceedings of the IEEE Conference On Computer Vision and Pattern Recognition* (Boston, MA: IEEE), 5546–5555.

Krause, J., Stark, M., Deng, J., and Fei-Fei, L. (2013). "3D object representations for fine-grained categorization," in *Proceedings of the IEEE International Conference on Computer Vision Workshops* (Sydney, NSW: IEEE), 554–561.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Commun. ACM* 60, 84–90. doi: 10.1145/3065386

Lam, M., Mahasseni, B., and Todorovic, S. (2017). "Fine-grained recognition as hsnet search for informative image parts," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu, HI: IEEE), 2520–2529.

Li, J., Cong, Y., Zhou, L., Tian, Z., and Qiu, J. (2023). Super-resolution-based part collaboration network for vehicle re-identification. *World Wide Web* 26, 519–538. doi: 10.1007/s11280-022-01060-z

Liao, Y., Xie, J., and Geiger, A. (2022). Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2D and 3D. *IEEE Trans. Pattern Anal. Mach. Intell*. 45, 3292–3310. doi: 10.1109/TPAMI.2022.3179507

Lin, D., Shen, X., Lu, C., and Jia, J. (2015). "Deep LAC: deep localization, alignment and classification for fine-grained recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Boston, MA: IEEE), 1666–1674. doi: 10.1109/CVPR.2015.7298775

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision* (Venice: IEEE), 2980–2988. doi: 10.1109/ICCV.2017.324

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014a). "Microsoft coco: Common objects in context," in *Computer Vision-ECCV 2014: 13th European Conference* (Zurich: Springer) 740-755.

Lin, Y.-L., Morariu, V. I., Hsu, W., and Davis, L. S. (2014b). "Jointly optimizing 3d model fitting and fine-grained classification," in *Computer Vision-ECCV 2014: 13th European Conference* (Zurich: Springer), 466–480.

Liu, D., and Wang, Y. (2017). *Monza: Image Classification of Vehicle Make and Model Using Convolutional Neural Networks and Transfer Learning*. Stanford, CA: Stanford University.

Liu, H., Tian, Y., Yang, Y., Pang, L., and Huang, T. (2016a). "Deep relative distance learning: Tell the difference between similar vehicles," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*(Las Vegas, NV,: IEEE), 2167–2175.

Liu, W., Wen, Y., Yu, Z., and Yang, M. (2016b). Large-margin softmax loss for convolutional neural networks. *arXiv* [preprint] arXiv:1612.02295. doi: 10.48550/arXiv.1612.02295

Llorca, D. F., Colás, D., Daza, I. G., Parra, I., and Sotelo, M. (2014). "Vehicle model recognition using geometry and appearance of car emblems from rear view images," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (Qingdao: IEEE), 3094–3099.

Ma, X., and Grimson, W. E. L. (2005). "Edge-based rich representation for vehicle classification," in *Tenth IEEE International Conference on Computer Vision (ICCV'05)* (Beijing: IEEE), 1185–1192.

Ma, Z., Chang, D., Xie, J., Ding, Y., Wen, S., Li, X., et al. (2019). Fine-grained vehicle classification with channel max pooling modified cnns. *IEEE Tran. Vehicular Technol.* 68, 3224–3233. doi: 10.1109/TVT.2019.2899972

Munroe, D. T., and Madden, M. G. (2005). "Multi-class and single-classclassification approaches to vehicle model recognition from images," in *Proc. AICS* (Munster: Springer), 1–11.

Pearce, G., and Pears, N. (2011). "Automatic make and model recognition from frontal images of cars," in *2011 8th IEEE International conference on advanced video and signal based surveillance (AVSS)* (Klagenfurt: IEEE), 373–378.

Prokaj, J., and Medioni, G. (2009). "3-D model based vehicle recognition," in *2009 Workshop on Applications of Computer Vision (WACV)* (Snowbird, UT: IEEE), 1–7.

Psyllos, A. P., Anagnostopoulos, C.-N. E., and Kayafas, E. (2010). Vehicle logo recognition using a sift-based enhanced matching scheme. *IEEE Trans. Intellig. Transport. Syst*. 11, 322–328. doi: 10.1109/TITS.2010.2042714

Shan, C., Gong, S., and McOwan, P. W. (2009). Facial expression recognition based on local binary patterns: a comprehensive study. *Image Vis. Comput*. 27, 803–816. doi: 10.1016/j.imavis.2008.08.005

Sochor, J., Herout, A., and Havel, J. (2016). "Boxcars: 3D boxes as CNN input for improved fine-grained vehicle recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV: IEEE), 3006–3015.

Sochor, J., Špaňhel, J., and Herout, A. (2018). Boxcars: improving fine-grained recognition of vehicles using 3-D bounding boxes in traffic surveillance. *IEEE Trans. Intellig. Transport. Syst*. 20:97–108. doi: 10.1109/TITS.2018.2799228

Sun, Y., Wang, X., and Tang, X. (2014). "Deep learning face representation from predicting 10,000 classes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE Computer Society), 1891–1898.

Suthaharan, S., and Suthaharan, S. (2016). "Support vector machine," in *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*, 207–235.

Tafazzoli, F., Frigui, H., and Nishiyama, K. (2017). "A large and diverse dataset for improved vehicle make and model recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (Honolulu, HI: IEEE), 1–8.

Tan, M. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *arXiv* [preprint] arXiv:1905.11946. doi: 10.48550/arXiv.1905.11946

Tang, J., Deng, C., and Huang, G.-B. (2015). Extreme learning machine for multilayer perceptron. *IEEE trans. Neural Netw. Learn. Syst*. 27, 809–821. doi: 10.1109/TNNLS.2015.2424995

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. (2021). "Training data-efficient image transformers & distillation through attention," in *International Conference on Machine Learning* (New York: PMLR), 10347–10357.

Valan, M., Makonyi, K., Maki, A., Vondráček, D., and Ronquist, F. (2019). Automated taxonomic identification of insects with expert-level accuracy using effective feature transfer from convolutional networks. *Syst. Biol*. 68, 876–895. doi: 10.1093/sysbio/syz014

Van Horn, G., Branson, S., Farrell, R., Haber, S., Barry, J., Ipeirotis, P., et al. (2015). "Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Boston, MA: IEEE), 595–604.

Vedaldi, A., and Zisserman, A. (2016). *VGG Convolutional Neural Networks Practical*. Oxford: Department of Engineering Science, University of Oxford, 66.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. (2011). *The Caltech-UCSD Birds-200-2011 Dataset*. California Institute of Technology.

Wang, Z., Zhan, J., Duan, C., Guan, X., Lu, P., and Yang, K. (2022). A review of vehicle detection techniques for intelligent vehicles. *IEEE Trans. Neural Netw. Learn. Syst*. 34, 3811–3831. doi: 10.1109/TNNLS.2021.3128968

Wen, Y., Zhang, K., Li, Z., and Qiao, Y. (2016). "A discriminative feature learning approach for deep face recognition," in *Computer Vision-ECCV 2016: 14th European Conference* (Amsterdam: Springer), 499–515.

Xiao, T., Xu, Y., Yang, K., Zhang, J., Peng, Y., and Zhang, Z. (2015). "The application of two-level attention models in deep convolutional neural network for fine-grained image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Boston, MA: IEEE), 842–850.

Xue, W. (2006). Facial expression recognition based on gabor filter and svm. *Chin. J. Electron*. 15:809.

Yang, L., Luo, P., Change Loy, C., and Tang, X. (2015). "A large-scale car dataset for fine-grained categorization and verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Boston, MA: IEEE), 3973–3981.

Yu, D., Xu, Q., Guo, H., Zhao, C., Lin, Y., and Li, D. (2020). An efficient and lightweight convolutional neural network for remote sensing image scene classification. *Sensors* 20:1999. doi: 10.3390/s20071999

Yun, S.-H., Kim, H.-J., Ryu, J.-K., and Kim, S.-C. (2023). Fine-grained motion recognition in at-home fitness monitoring with smartwatch: a comparative analysis of explainable deep neural networks. *Healthcare* 11:940. doi: 10.3390/healthcare11070940

Zhang, N., Donahue, J., Girshick, R., and Darrell, T. (2014). "Part-based r-cnns for fine-grained category detection," in *Computer Vision-ECCV 2014: 13th European Conference* (Zurich: Springer), 834–849.

Zhang, Q., Zhuo, L., Zhang, S., Li, J., Zhang, H., and 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)* (Xi'an: IEEE), 1–5.

Zhang, X., Xiong, H., Zhou, W., Lin, W., and Tian, Q. (2016). "Picking deep filter responses for fine-grained image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV: IEEE), 1134–1142.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). "Learning deep features for discriminative localization," in *Proceedings of the IEEE Conference on Computer Vision and pattern recognition* (Las Vegas, NV: IEEE), 2921–2929.