

Vehicle Color Identification Framework using Pixel-level Color Estimation from Segmentation Masks of Car Parts

Klearchos Stavrothanasopoulos

CERTH-ITI

Thessaloniki, Greece

klearchos_stav@iti.gr

Theodora Tsikrika

CERTH-ITI

Thessaloniki, Greece

theodora.tsikrika@iti.gr

Konstantinos Gkountakos

CERTH-ITI

Thessaloniki, Greece

gountakos@iti.gr

Stefanos Vrochidis

CERTH-ITI

Thessaloniki, Greece

stefanos@iti.gr

Konstantinos Ioannidis

CERTH-ITI

Thessaloniki, Greece

kioannid@iti.gr

Ioannis Kompatsiaris

CERTH-ITI

Thessaloniki, Greece

ikom@iti.gr

Abstract—Color comprises one of the most significant and dominant cues for various applications. As one of the most noticeable and stable attributes of vehicles, color can constitute a valuable and reliable key component in several practices of intelligent surveillance systems. In this paper, we propose a deep-learning-based framework that combines semantic segmentation masks with pixels clustering for automatic vehicle color recognition. Different from conventional methods, which usually consider only the features of the vehicle’s front side, the proposed algorithm is able for view-independent color identification, which is more effective for the surveillance tasks. To the best of our knowledge, this is the first work that employs semantic segmentation masks along with color clustering for the extraction of the vehicle’s color representative parts and the recognition of the dominant color, respectively. In order to evaluate the performance of the proposed method, we introduce a challenging multi-view dataset of 500 car-related RGB images extending the publicly available DSMLR Car Parts dataset for vehicle parts segmentation. The experiments demonstrate that the proposed approach achieves excellent performance and accurate results. To facilitate further research, the evaluation dataset and the pre-trained models will be released at <https://github.com> (URL will be provided after the publication acceptance).

Index Terms—vehicle color identification, semantic segmentation, intelligent surveillance

I. INTRODUCTION

Vehicle recognition is a key component of intelligent surveillance systems. However, the identification of a vehicle is not always a straightforward procedure. During the last decades, several vehicle properties such as model [1], license plate [2], type [3] and logo [4] have been the core research objects in this domain. Still, most of these properties are not always fully apparent due to occlusions and problematic viewpoint angle; thus are difficult to discern in some contexts due to noise, i.e. blur, and distortions. Different from these characteristics, color covers a substantially wider area of the vehicle body and is less susceptible to such interference factors. Due to these advantageous attributes, automatic vehicle

color recognition is an important and promising research issue in the context of intelligent surveillance systems.

Vehicle color has been employed in various surveillance-related applications such as law enforcement [5], criminal detection [6] and video surveillance [7]. However, some factors hampering the current advancements in vehicle color identification, increasing the inherent challenge. First, the weather and light conditions may dramatically affect the color of a vehicle and create visible color variations. In order to overcome this challenge, preprocessing techniques such as haze removal [8] and color contrast [9] are usually applied to the original images. Despite their popularity, these methods are not robust enough when images are captured in complex natural scenes and achieve limited improvements in the final predictions.

Furthermore, these techniques create a bottleneck to the pipeline and can not be easily adopted in an end-to-end framework for near real-time surveillance applications. The second factor is related to the global context connection, an important feature for fine-grained and view-independent vehicle color detection, which most of the aforementioned methods do not assess in their implementation [10], [11]. A vehicle’s color is specified by the dominant color of its surface, albeit certain portions do not have the same RGB value as the main body. Thus, the identification framework must consider the vehicle’s various parts that provide the most representative color information. The last factor is associated with the limitations that reside in the proposed vehicle color datasets. Previous datasets’ limited categories hinder the development of color recognition technology, with the recognised vehicle colors limited to eight basic colors. However, colors belonging to the same fundamental color family have significant variances in natural light, causing the color identification technologies to be unstable and lowering the accuracy of their predictions.

To address the above issues, we introduce a challenging

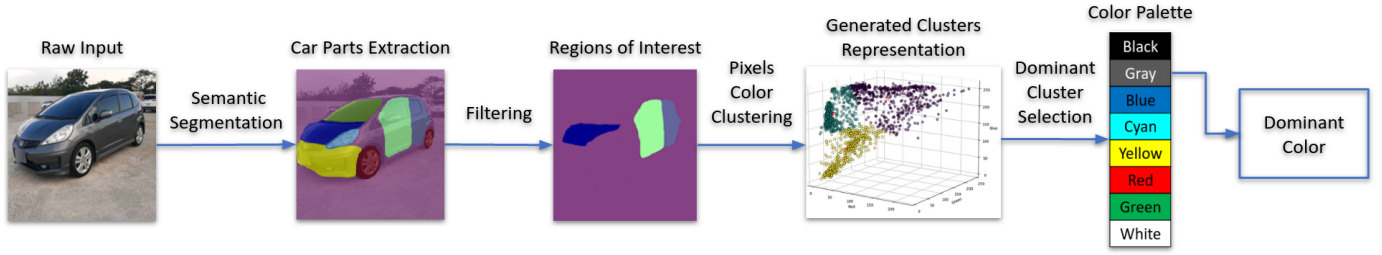


Fig. 1. The architecture illustration of the proposed vehicle color identification framework. Given a raw image as input, semantic segmentation is performed to extract the different parts of the vehicle. The output is filtered to acquire the regions related to the problem, and clustering is applied to generate clusters of pixels. The value of the dominant's cluster representative data point is compared to the predefined color palette to determine the dominant color of the vehicle.

dataset for view-independent identification of vehicle color by extending the public benchmark of [12]. The dataset includes 500 vehicle images and covers several scenes with different viewing angles and environmental changes. In order to extract the most representative regions of a vehicle, we fine-tune a deep-learning-based model for segmenting the vehicle's main parts. During training, the model relies on pixel-level annotations and learns the features of 18 different car parts. Then, the semantic segmentation output masks are processed using an unsupervised clustering model that groups the pixel values in clusters of colors. The dominant cluster (the cluster with the most samples) is considered for the final step of the color identification process. We evaluate our proposed method on the introduced dataset via experimenting with several segmentation and clustering algorithms.

The major contributions and innovation points of this paper are summarised as follows:

- We propose a view-independent color identification framework that considers the different parts of the vehicle by utilising variant segmentation sub-masks and estimating the dominant color through pixels clustering.
- We extend an already existing vehicle parts dataset with additional color annotations to be applicable to the vehicles' color identification domain (target domain), including 500 RGB images manually annotated on eight main color categories, depicting vehicles of various types shot on multiple views.
- Extensive evaluation tests are conducted considering multiple semantic segmentation techniques and clustering approaches for the dominant color extraction.

II. RELATED WORK

In this section, our discussion mainly focuses on two groups of methods most relevant to the proposed work. In the first part, the most significant semantic segmentation techniques are presented since the semantic segmentation module is instrumental for the proposed framework. In the second part, the works reported in the recent literature relevant to the vehicle color identification problem are analysed.

A. Semantic Segmentation

Image segmentation can be considered as the task of pixel classification with semantic labels (semantic segmentation) or

the localisation and delineation of each individual object of interest in an image (instance segmentation). Semantic segmentation associates every pixel of an image with a class label (i.e., person, flower, car, bike), while instance segmentation distinguishes different instances of the same class (i.e., individual cars). During the last decades, many approaches for image segmentation have been introduced, from the earliest methods using simple algorithms [13]–[16] to the latest advanced works based on deep learning architectures [17]–[22].

Recent years have witnessed outstanding advances in the challenging semantic image segmentation task, especially with the development of deep neural networks. In the seminal work of [23], the concept of the Fully Convolutional Network (FCN) is introduced. FCN achieves accurate pixel-wise predictions by transforming the DCNN's fully connected layers for classification to convolutional layers. However, the large downsampling factor of the input resolution results in relatively coarse output. To overcome these restrictions of the FCNs, various methods based on the Markov Random Fields (MRFs) and its variant Conditional Random Fields (CRFs) were developed, improving predictions and producing refined results [24]–[28].

Several other approaches were focused on enlarging the receptive field of neural networks and enhancing contextual aggregation. In this direction, Deeplab [29] and Dilation [30] introduced the dilated (atrous) convolution. Due to the simplicity of the dilation convolution, various novel and effective components relied on this capacity. Atrous Spatial Pyramid Pooling (ASPP) is utilised in [29] and [31], where Deeplab v2 and v3 models embed contextual information by adopting dilated convolutions with multiple atrous rates. PSPNet [32] devises a Pyramid Pooling Module (PPM) on the dilation backbone, using pooling operators with different kernel sizes to collect effective contextual features containing information of different scales. Meanwhile, some methods introduce also the attention mechanisms to capture long-range context based on the dilation backbone. In [17], the authors proposed the Point-wise Spatial Attention Network (PSANet), a new architecture that uses a predicted attention map to generate rich pixel-wise context. In Criss-Cross Network (CCNet) [21], the criss-cross attention module is introduced, capturing the contextual information of all the pixels on a specific path.

A self-attention mechanism is proposed in CPNet [22], which extracts the intra-class and inter-class contextual dependencies, achieving improved feature representation.

Besides the dilation based architectures, another large category of segmentation methods was developed adopting the Encoder-Decoder backbone. Extra top-down and lateral connections are adopted in this architecture to capture the high-resolution feature maps in the decoder part. Unet [33] concatenated the output from low-level layers with higher ones for information fusion. In DeeplabV3+ [19], the authors adopt an encoder-decoder structure to recover spatial information through pooling features at different resolutions and upsampling operations. In Refinet [20], the boundaries of salient objects are located, creating an enhanced saliency map with more accurate spatial details. Laplacian Pyramid Reconstruction and Refinement (LRR) method [34] adopts the Laplacian reconstruction pyramid [35] to harvest detailed context with step-wise reconstruction. Discriminative Feature Network (DFN) [36] incorporates an attention module, using global pooling to recover global information. The authors in [18] proposed a fast and efficient convolutional neural network, ESPNet, based on the Efficient Spatial Pyramid (ESP) convolutional module. Lastly, the Bilateral Segmentation Network (BiSeNet) was proposed in [37], a memory-efficient approach that achieves real-time predictions by extracting local-global information from high-resolution images.

B. Color Identification

In the last decades, many research works were focused on the challenging vehicle color recognition task, applying several techniques to achieve satisfactory results. The authors in [10] proposed a feature context method based on the Bag of Words (BoW) concept combining different color histograms of various color spaces to identify vehicles' color. They created a new dataset by capturing footage from urban roads, which contains a total of 15,601 vehicle images. Several preprocessing techniques were used to improve the quality of the images, such as haze removal and color contrast. Their algorithm achieved 90.68% accuracy using the Support Vector Machine (SVM) classifier. In [11], a new CNN-based color recognition method, Colornet, is introduced. Colornet, which was evaluated on a custom dataset consisting of images captured from city surveillance cameras, outperformed AlexNet [38] and GoogLeNet [39] in the color classification task (eight classes), achieving an accuracy of 95.74%. In [40], a deep learning architecture for vehicle color recognition is proposed that fuses CNNs with the spatial pyramid strategy [41]. This method was validated on the dataset from [10], producing improved results and increased recognition accuracy. Researchers in [42] trained a CNN model to perform classification based on color distribution. Besides the standard RGB space, other color spaces were also examined, including the Hue-Saturation-Value (HSV) and the International Commission on illumination (CIELAB) spaces. Their results on Chen's dataset [10] using eight vehicle color classes showed an accuracy rate of 94.47%. In [43], the authors applied a CNN architecture to classify vehicle

types and color. Their experiments were performed using 914 vehicle images collected from surveillance videos.

In comparison to previous works where decision trees, random forest and DNN classifiers were utilised, the results of this approach showed improvements of 1.8% and 0.8% on the classification of vehicle type (four classes) and color (seven classes), respectively. In [44], a CNN based solution with a reduced number of convolutional layers is proposed. This lightweight architecture implements three convolutional layers, followed by a Global Average Pooling (GAP) layer connected directly to the classification layer. The extracted feature map is divided using SPM (Spatial Pyramid Matching), and then every SPM region is used to a vector of feature representation. Their experiments were performed on the dataset of [10], achieving an accuracy score of 95.41% on the validation data.

Despite their great performance, the aforementioned methods have one common drawback, considering only the front side of the vehicle for the identification of its color. In this paper, we follow a different approach for vehicle color identification to tackle this limitation, focusing on multi-view color recognition. The regions of interest are extracted using vehicle semantic segmentation, and the color identification is achieved by the dominant color extraction through clustering applied on the segmented regions of interest (Fig. 1). This is the first work to tackle the challenge of vehicle color identification through this approach to the best of our knowledge.

The rest of the paper is organised as follows. Section III outlines the proposed method, while Section IV summarises the conducted experiments and the evaluation results of the introduced framework. Finally, Section V concludes this work.

III. PROPOSED METHOD

In this section, the methodology of the proposed framework is presented, starting with a high-level overview, followed by a detailed explanation of the semantic segmentation and the pixel clustering modules.

A. Methodology

Different from human cognition, Machine Learning (ML) based computer vision approaches cannot consider only the colorised parts of a vehicle in order to identify its dominant color; thus, in this section, we present the methodology of the proposed color identification framework that extracts the different parts of the vehicle by utilising variant segmentation sub-masks, discards the ambiguous parts' pixels and applies color clustering on the filtered pixel values in order to estimate the vehicle's dominant color.

The proposed methodology takes into account the color of particular vehicle parts. In general, vehicles consist of some standard parts such as wheels, doors, hood, windshields, bumpers, trunk and mirrors. Most of these parts (doors, hood, trunk) are usually painted under the same color scheme defining the vehicle's main color. However, a vehicle image processed by an ML algorithm has many parts with no color information (wheels, windshields, background), or this

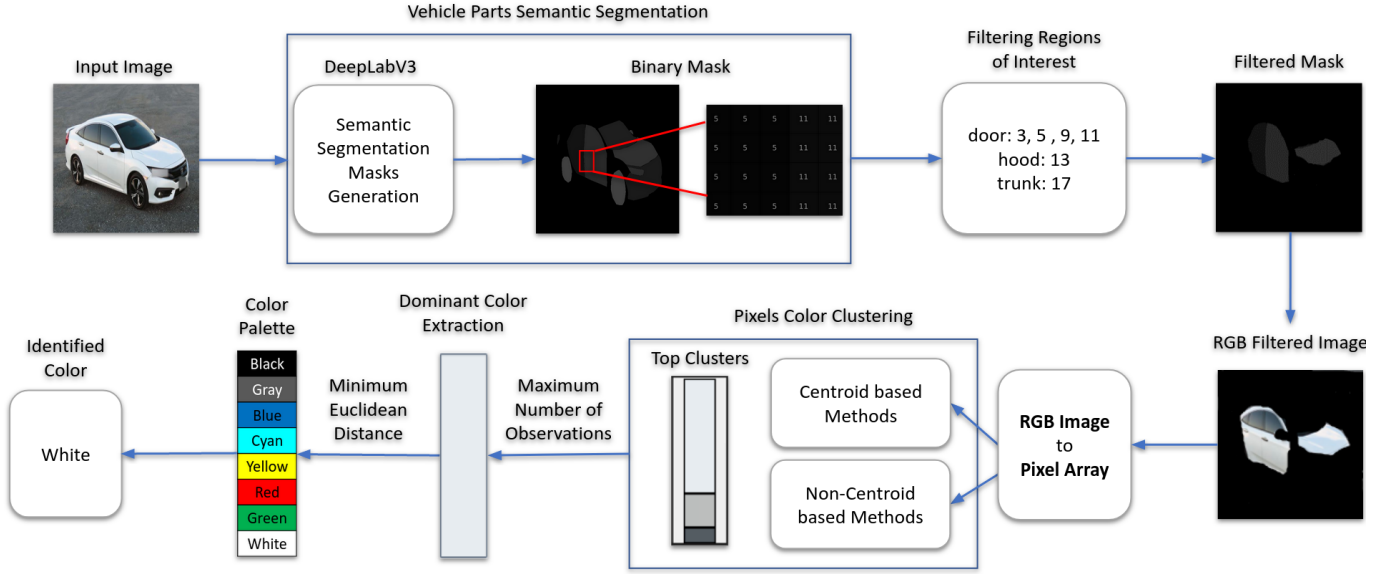


Fig. 2. The step-by-step representation of the proposed vehicle color identification framework. Firstly, the input vehicle image is processed by the semantic segmentation module (DeepLabV3 [31]). Then, a filtering operation is performed on the segmentation masks to keep only the parts considered as the main indicators of the vehicle's color. The corresponding RGB image pixels are extracted and converted to an array to be fed to the next module. Then, the pixel clustering module creates groups of pixels considering the RGB value of each data point. In the final step, the dominant cluster's distance to the closest color of the predefined color palette is calculated to identify the object's dominant color.

information is different from the vehicle's main color (non-colored or different colored bumpers, mirrors). This can be misleading for the ML models and may lead them to learn wrong features during training, producing incorrect predictions and lowering their accuracy.

The architecture of the proposed method, shown in Fig. 2, can be divided into two main parts: the vehicle parts semantic segmentation module and the color classification module. In the first step, the semantic segmentation model generates the pixel masks of the object's different parts (in our scenario vehicle parts). The pixels of these masks contain valuable information for the object's color. As next step, we filter the initial semantic segmentation masks and keep the ones that can be considered containing more relevant information for color estimation. At the second part of the proposed method, the RGB pixel values of the original image are extracted using the filtered masks and fed as input into a clustering model. The algorithm splits the pixels into groups based on their RGB value, and clusters are formed. At the final step, these clusters are utilised in order to calculate the most close color to the predefined color map and identify the object's dominant color.

B. Semantic Segmentation Module

In contrast to most traditional vehicle color recognition methodologies [10], [11], our work focuses on multi-view vehicle color extraction; thus, the accurate semantic segmentation of the vehicle parts is crucial. In the proposed method, the CNN-based semantic segmentation architecture of [31] is adopted and trained in order to estimate the optimal weights for extracting more accurate visual representations. For vehicle parts semantic segmentation, the model is trained on the

dataset proposed in [12]. The network uses ResNet-50 [45] as its backbone for feature extraction. The first few layers of this model have a structure similar to the FCN [23], with four layers with 3, 4, 23 and 3 bottleneck units, respectively. The classifier that follows starts with a 1×1 convolution with batch normalisation accompanied by a ReLU activation function and the output is fed into the ASPP network [29]. The ASPP Network, based on pyramid pooling, is used to classify each pixel with its corresponding pixel-level classification. The convolution operations in the pyramid are 3×3 with different dilation rates. This is followed by adaptive average pooling for global context and four convolution operations with batch normalisation and ReLU activation steps. All convolutions are 1×1 except for the next-to-last convolution, which is a 3×3 operation.

In the proposed framework, the semantic segmentation module processes the input image as a first step, and a semantic label is assigned to each pixel. Then, the predicted semantic segmentation masks are filtered in order to keep only the vehicle parts that can be considered as the main indicators of the vehicle's color. The labelled parts containing non-valuable features for color recognition are removed to avoid the extraction of misleading features during the dominant color identification process, which can lead to reduced model performance. The procedures of segmentation and filtering are illustrated in Fig. 2. The filtered mask and the final RGB image are extracted according to equations (1) and (2) :

$$\text{final_mask}_{ij} = \begin{cases} \text{True} & \text{if label (hood — door — trunk)} \\ \text{False} & \text{else} \end{cases} \quad (1)$$

$$\text{final_image}_{ij} = \begin{cases} \text{rgb_image}[i, j] & \text{if final_mask}_{ij} = \text{True} \\ \text{None} & \text{else} \end{cases} \quad (2)$$

where i, j index the pixel position on each image, $i = 0, \dots, W$ and $j = 0, \dots, H$ (W and H are the width and height of the image, respectively) and $\text{rgb_image}[i, j]$ is the (r,g,b) value of the pixel in the i, j position of the original image.

In Eq.(1) the produced final_mask_{ij} is *True* when the annotated mask label is equal to **hood**, **door** or **trunk**, otherwise the value is *False*. In Eq.(2), the value of final_image_{ij} is set as equal to the original $\text{rgb_image}[i, j]$ value when the final_mask_{ij} is equal to *True*, otherwise the value of final_image_{ij} is set to *None*.

C. Pixel Clustering Module

Considering the extracted Regions of Interest (RoIs) from the original image, the next step includes a clustering algorithm used to divide the pixels into groups, taking into account each pixel's (r,g,b) value.

In the proposed method, two approaches were explored in order to create clusters of colors: algorithms with a predefined number of clusters and algorithms that create clusters without a limitation of maximum groups number. The first approach requires defining the specific number of groups to be discovered in the data, whereas the second one requires the specification of some minimum distance between observations in which examples may be identified as “close” or “connected”. Apart from this categorisation, the clustering algorithms can also be split into centroid based and non-centroid based ones. In the centroid based methods, each cluster can be represented from its centroid value, while in the non-centroid based techniques, where there is no notion of a centroid, we are forced to develop another way to summarise each cluster by extracting the mean value of the cluster's data points.

At the last step of our framework, we extract the cluster with the largest amount of data points of a similar color- the dominant cluster- and calculate the Euclidean distance between the dominant cluster's representative color and each color existing on the predefined color palette. This computation is performed using the equation (3):

$$\text{color_distance}_i = \sqrt{(R - r_i)^2 + (G - g_i)^2 + (B - b_i)^2} \quad (3)$$

where N is the number of colors in our palette, R, G, B are the values of the cluster's delegate point and r, g, b are the values of each color in the palette, with $i = 1, 2, \dots, N$. In the centroid based algorithms this procedure is utilising the dominant cluster's centroid as dominant color. In contrast, in

the other methods we calculate the mean value of the dominant cluster's points (pixels). Finally, the identified dominant color is defined as the one with the smallest distance from the dominant's cluster representative (R,G,B) point.

IV. EXPERIMENTAL EVALUATION

The datasets used during the experiments are firstly presented in this section, highlighting the dataset we extended with color annotations for the training and evaluation of the proposed framework. Next, the setup of the semantic segmentation and color clustering experiments along with the evaluation metrics are demonstrated. In the last part of this section, the evaluation results are discussed.

The environment configuration for the experiments is Ubuntu 20.04 operating system, with a memory of more than 32GB DDR4 and an Nvidia RTX 2060 Ti GPU. The programming language used for the algorithms is Python, while the training and testing of the deep learning models are implemented on the TensorFlow Deep Learning framework [46].

A. Datasets

DSMLR Car Parts dataset was introduced in [12]. The dataset consists of 500 vehicle images of three categories: sedans, pickups and Sports Utility Vehicles (SUVs). The images depict vehicles in multiple viewpoints (front, back and angled), with each image containing a single vehicle. Two types of vehicle parts annotations are provided, segmentation masks and bounding boxes, following the MS COCO Dataset format [47] and labeled under a list of 18 vehicle parts: back_bumper, back_glass, back_left_door, back_left_light, back_right_door, back_right_light, front_bumper, front_glass, front_left_door, front_left_light, front_right_door, front_right_light, hood, left_mirror, right_mirror, tailgate, trunk, and wheel (wheel and tire). All parts of the images including sensitive identification information were anonymised, i.e., car license plate, faces. The number of instances per category is presented in Table I, while some representative samples of the original images along with their annotation masks are illustrated in Fig. 3. The resolution of both images and their masks equals 512×512 pixels. The dataset is challenging for its variability in illumination, viewpoint and vehicle type.

Vehicle Color dataset provided in [10] is one of the commonly used datasets for comparative experiments. The dataset contains 15,601 images of trucks, sedans, and buses covering eight colors: black, blue, cyan, gray, green, red, white, and yellow. The dataset images are shot using the equipment installed on the urban roads, having Full HD resolution (1920×1080 pixels), with each image containing a single vehicle captured from the frontal view. The dataset is characterised by varying environments, such as illumination and various weather conditions, which brings greater challenges to identify the color of the vehicles rightly. However, this dataset does not fit the problem's evaluation due to the lack of annotation masks for vehicle parts segmentation and its limitation regarding the single camera view on the vehicle's

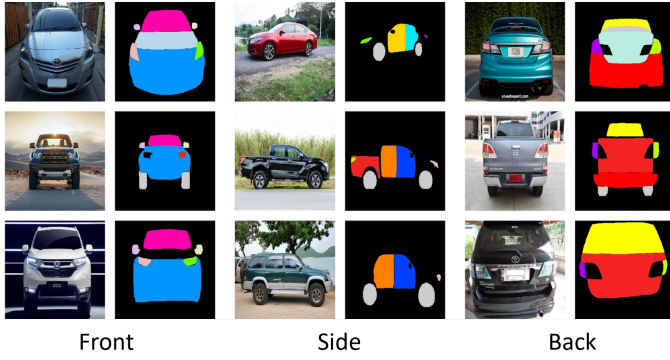


Fig. 3. Samples of DSMLR Car Parts dataset [12] depicting vehicles captured from the front, side, and back views both for the original images and the ground truth segmentation masks.

frontal. The proposed framework is designed to focus on the view-independent vehicle color recognition to accommodate more realistic surveillance scenarios..

TABLE I
DSMLR CAR PARTS DATASET INSTANCES PER CATEGORY.

Category	Number of instances		
	Train	Test	Total
background	400	100	500
back_bumper	82	17	99
back_glass	110	21	131
back_left_door	86	14	100
back_left_light	148	19	167
back_right_door	75	15	90
back_right_light	121	19	140
front_bumper	198	61	259
front_glass	216	72	288
front_left_door	96	18	114
front_left_light	231	56	287
front_right_door	82	23	105
front_right_light	204	61	265
hood	218	69	287
left_mirror	241	50	291
right_mirror	232	52	284
tailgate	45	6	51
trunk	81	16	97
wheel	218	58	276
Total	3084	747	3831

B. Experimental Setup

Several semantic segmentation algorithms and clustering techniques were examined for optimising the proposed color identification framework. As a first step, four semantic segmentation models were tested and evaluated in order to find the method that produces the best results on the task of car parts segmentation. As a second step, we experimented with five clustering techniques and evaluated their performance on the new extended dataset to discover the best performing method in the task of clustering and color identification.

1) *Vehicle Parts Segmentation*: In order to find the most suitable model architecture for the semantic segmentation module of our framework, we have experimented with a total of four state-of-the-art architectures, two dilation based

and two encoder-decoder based architectures: PSPNet [32], DeepLabv3 [31], SegNet [53] and UNet [33]. The models were trained using stochastic gradient descent as the optimiser with four examples per batch, momentum of 0.9 and weight decay of 0.0001. The train set of [12] was used, consisting of 400 vehicle images captured from various angles (front, back, side) including 2,684 annotated car parts. In the training process, several augmentation techniques such as rotation, random crop and random scale were used to increase the segmentation accuracy of the models. Flip augmentation was prohibitive in our case since it would create false features mainly around the vertical axis. We set an initial learning rate of 0.01, which is reduced continuously by a factor of 10 every N iterations, where N is determined in relation to the total number of training iterations of each model.

2) *Clustering and Color Identification*: In order to identify the color of a vehicle, we have tested a total of five clustering techniques, two that require a predefined number of clusters and three without this requirement: K-means clustering [48], agglomerative clustering [49], Mean Shift clustering [52], Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [50], and Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) [51]. During the experiments, the algorithms K-means, BIRCH and Mean Shift were considered as centroid based techniques while agglomerative clustering and DBSCAN as non-centroid. For each clustering technique, we experimented with the main configurable parameters by tuning them to find the most efficient setup for our problem.

In **K-means** method, it is necessary to provide the algorithm with the number of clusters to be discovered. In this direction, we have experimented with several values for this parameter, also considering the choice between random or pre-calculated initialisation of the centroids. For evaluating the **Agglomerative** clustering technique, we have explored two tunable parameters: the number of clusters and the agglomeration method. In this clustering algorithm, which is a part of the broader class of hierarchical clustering methods, the option of predefining the number of clusters is also provided. However,

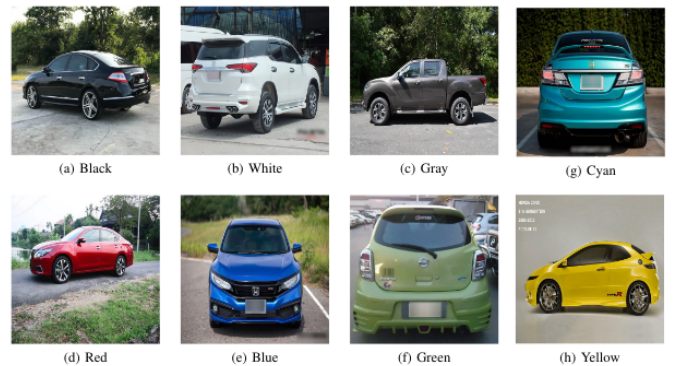


Fig. 4. Proposed extended dataset samples with color labels for each color category.

TABLE II
NUMBER OF SAMPLES PER COLOR CATEGORY.

Color	White	Gray	Black	Red	Blue	Green	Yellow	Cyan
Number of samples	203	155	76	32	19	7	6	2

TABLE III
PERFORMANCE COMPARISON OF CLUSTERING MODELS ON PROPOSED DATASET.

Model	White	Black	Gray	Red	Green	Blue	Yellow	Cyan	Average	Weighted Average
K-means [48]	0.8571	0.8289	0.8968	0.9063	0.5714	0.8421	1.000	1.000	0.8628	0.8660
Agglomerative [49]	0.8374	0.5921	0.8839	0.9063	0.7143	0.7895	1.000	0.5000	0.7779	0.8160
DBScan [50]	0.4680	0.5000	0.9613	0.9375	0.7143	0.9474	1.000	0.5000	0.75356	0.6840
BIRCH [51]	0.8867	0.9211	0.9161	0.9688	0.8571	0.8947	1.000	1.000	0.9306	0.9080
Mean Shift [52]	0.6650	0.4079	0.9484	0.9375	0.8571	0.7895	1.000	0.5000	0.7632	0.7420

this is not a mandatory parameter since the algorithm starts by treating each object as a singleton cluster and successively merges pairs of clusters until all clusters have been merged into one big cluster, but it is required in order to avoid this scenario. The other parameter we examined is the agglomeration method, also known as linkage criterion. The linkage criterion determines which distance to use between sets of observation. The algorithm will merge the pairs of clusters that minimise this criterion. In the **Mean Shift** method experiments -a hierarchical clustering algorithm- we tested its main tunable parameter, the bandwidth value. Different from K-means, Mean Shift does not require prior knowledge of the number of clusters and does not constrain the shape of the clusters. The algorithm automatically sets the number of clusters relying on this parameter which dictates the region's size to search through.

Regarding **DBSCAN**, two essential parameters were considered during the evaluation: epsilon and the minimum number of points. Epsilon is the most significant DBSCAN parameter since it defines the maximum distance between two samples to be considered neighbours and part of the same cluster. The minimum number of points refers to the minimum number required to form a dense region. In **BIRCH**, the algorithm builds a tree called Clustering Feature Tree (CFT) for the given data. In this regard, we have experimented with the branching factor parameter, which defines the maximum number of CF subclusters in tree node.

Moreover, we tested several values of the threshold parameter that determines the maximum radius value of the subcluster obtained by merging a new sample and the closest subcluster; otherwise, a new subcluster is started. Setting this value to be very low promotes splitting and vice-versa. Finally, for the evaluation of the color identification framework, the clustering techniques of both categories were applied to the filtered output of the vehicle part segmentation process.

C. Evaluation Metrics

In order to evaluate the performance of the semantic segmentation model architecture, we used the mean Intersection over Union (IoU) metric. It is a segmentation performance

parameter that measures the overlap between two objects by calculating the ratio of intersection and union with ground truth masks. This metric is also known as Jaccard Index [54] and calculated according to Eq.4:

$$IoU = \frac{Intersection}{Union} = \frac{TP}{TP + FP + FN} \quad (4)$$

Where TP denotes true positive, FP denotes false positive, FN denotes false negative and IoU denotes Intersection over Union value. To evaluate our color identification framework, we adopt the precision metric to measure the per class performance and the weighted average.

Dataset Annotation. To the best of our knowledge, no existing public benchmark provides both pixel annotations for the vehicle parts' segmentation task and classification labels for the vehicle color recognition task. In this paper, we extend the work of [12] to train and evaluate an end-to-end framework that performs vehicle parts semantic segmentation as the first step and color classification as the second step. Specifically, we have manually annotated the 500 RGB car-related images from the original dataset with dominant color labels using an eight common car color-palette similarly to the other state-of-the-art approaches [10], [11], [40], [42] including black, white, gray, red, green, blue, cyan and yellow. Furthermore, multi-label color annotations are provided for the multi-colored vehicles. The images depict several types of vehicles captured from different viewing angles, in contrast to most vehicle color recognition works [10], [11] which are limited just to the frontal view of the vehicle. The proposed dataset is challenging due to the noise caused by illumination, variation, haze, overexposure and multi-colored vehicles. The number of samples per color category is illustrated in Table II. Indicative samples of the extended annotated color dataset are illustrated in Fig. 4.

In the experiments, we keep the train/test split of the original paper for the semantic segmentation module's training and evaluation, where 400 images (80%) are used for training and 100 images (20%) are used for testing. The whole dataset is used to evaluate the color clustering and dominant color extraction.

D. Evaluation Results

For evaluating of the vehicle parts semantic segmentation task, we use 20% of the DSMLR dataset [12], which were not used during the training process. In Table IV, the evaluation results from the four different semantic segmentation models are summarised. Specifically, in the first six rows, we present the results for the parts mainly used for the vehicle’s color recognition, with the IoU scores of the rest presented afterwards. Four different mean values are calculated to showcase the models’ overall performance and highlight the best performing one. It should be noted that the mean IoU extracted from the parts of interest is greater than the calculated result for the rest parts for all the models. This information provides important insight regarding the subset of parts considered for the vehicle color identification framework. In the last row of Table IV, the frequency weighted IoU is calculated to acquire a more fair comparison between the models since the DSMLR Car Parts dataset is strongly unbalanced. As demonstrated in the results, DeepLab outperforms the other methods showing best overall performance, achieving top score in 15 out of 19 categories and most importantly in all six categories that are used for the vehicle color identification; thus, we select DeepLab as the backbone of the proposed framework out of the four methods trained and evaluated to the DSMLR dataset.

Using the RGB pixels equivalent of the filtered pixel masks extracted from the DeepLab, we evaluated five clustering techniques and the corresponding results listed in Table III. The accuracy achieved by each model per color category and the overall accuracy is provided to define the best clustering method for this task. The results demonstrate that the BIRCH [51] method generates more accurate clusters for the vehicle color identification since the best overall performance is obtained using this technique, acquiring top score in six out of eight color categories. The similarity of the results in some categories such as yellow, cyan, green can be explained considering the small number of samples of these classes.

In Fig. 5, vehicle color recognition indicative results of

TABLE IV
PERFORMANCE COMPARISON OF SEGMENTATION MODELS ON CAR PARTS DATASET [12].

Classes	Method			
	PSPNet [32]	SegNet [53]	UNet [33]	DeepLabv3 [31]
hood	0.6973	0.6733	0.7191	0.7385
trunk	0.4902	0.3419	0.3166	0.5430
back_left_door	0.2133	0.2075	0.2761	0.3791
back_right_door	0.1758	0.2404	0.2281	0.3256
front_left_door	0.1930	0.1904	0.1998	0.3252
front_right_door	0.2714	0.2578	0.2832	0.3667
background	0.8665	0.8674	0.8767	0.8881
back_bumper	0.4534	0.5014	0.5350	0.6080
front_bumper	0.6513	0.6625	0.6671	0.6822
back_glass	0.4335	0.2577	0.2984	0.5655
back_left_light	0.1541	0.0546	0.1334	0.3817
back_right_light	0.2219	0.2376	0.2542	0.4716
front_glass	0.3066	0.3075	0.3426	0.3062
front_left_light	0.2063	0.2720	0.3038	0.1195
front_right_light	0.013	0.0372	0.0569	0.1296
left_mirror	0.1051	0.0846	0.992	0.1003
right_mirror	0.1842	0.2208	0.1433	0.1262
tailgate	0.0291	0.0993	0.1147	0.2871
wheel	0.6336	0.6469	0.6645	0.6895
mean IoU - Parts of Interest	0.3402	0.3352	0.3538	0.4464
mean IoU - Rest Parts	0.3275	0.3269	0.3454	0.4120
mean IoU - Total	0.3073	0.2996	0.3187	0.3970
frequency weighted IoU - Parts of Interest	0.4229	0.4127	0.4381	0.5138

TABLE V
PERFORMANCE COMPARISON WITH STATE-OF-THE-ART METHOD ON THE PROPOSED DATASET.

Models	Accuracy
Method [42] (pre-trained on dataset from [10])	0.5846
Method [42] (pre-trained on dataset from [10] + fine-tuned on the proposed dataset)	0.7645
Ours (DeepLabV3 + Clustering)	0.9306

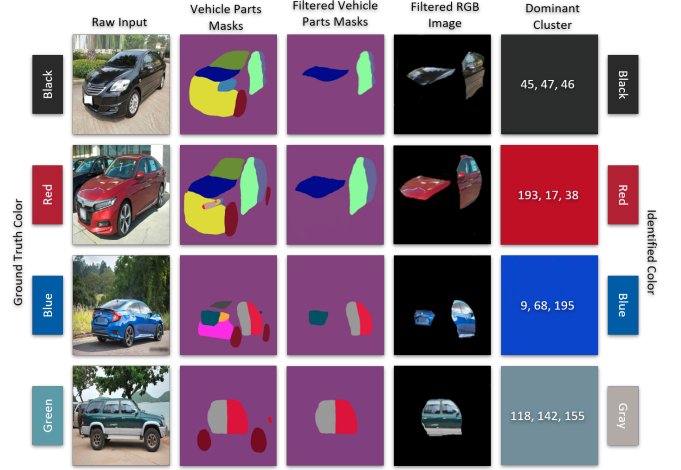


Fig. 5. Proposed framework’s vehicle color recognition indicative results. From left to right, the input vehicle images and the respective ground truth labels are shown. Then the vehicle parts masks generated by the segmentation module, the masks produced after the filtering operation and the corresponding RGB images are presented. Finally, the dominant cluster’s color value extracted from the pixel clustering module and the identified color are depicted. An incorrect color identification case is displayed in the last row where the framework wrongly predicts the vehicle’s color as Gray instead of Green.

the proposed framework, using DeepLab to predict car parts masks and BIRCH for pixel clustering, are demonstrated. The output of the most significant processes is illustrated for better comprehension of each module’s role with the final detection output provided in the last column of the figure. Although the proposed framework performs well under various viewpoints and challenging conditions, there is still room for further improvement. An incorrect color identification case is displayed in the last row where the framework wrongly predicts the vehicle’s color as Gray instead of Green.

Table V shows the performance comparison on the color recognition task between our proposed method and a CNN based state-of-the-art strategy [42]. The state-of-the-art model was initially evaluated on the proposed dataset using a pre-trained version of the model using the dataset of [10]. As illustrated in the first row of Table V, our method outperforms the state-of-the-art approach significantly. In order to facilitate the superiority of the proposed framework in the multi-view scenario against the current state-of-the-art approaches, we fine-tuned and evaluated the model presented in [42] on the new extended dataset. The results are provided in the second row of Table V, showing that our method outperforms the

other model by overall accuracy of 16.61%. The final results manifest that taking into account the parts of a vehicle, especially the most color representative ones, can lead to significant performance improvement in the vehicle's color identification task in the context of multi-view scenarios.

V. CONCLUSION

This work focuses on the problem of view-independent vehicle color identification by proposing a novel vehicle color identification framework combining the semantic segmentation of vehicle parts with the clustering of pixels' color value. Various segmentation algorithms and clustering methods have been explored and tested. Since the previous works are limited due to the restricted datasets and ineffective learning methods, we propose a challenging vehicle color dataset based on the DSMLR Car Parts Dataset, which combines pixel-level annotated vehicle parts with vehicle color labels. The evaluation on the proposed dataset demonstrates the effectiveness of our approach in realistic multi-view scenarios under several environment conditions, by achieving high accuracy of 93.06% and outperforming the most relevant state-of-the-art method applied on our dataset by 16.61%. Our future work includes the pixel-level annotation of a large-scale vehicle parts dataset combined with dominant color labels, further evaluating our approach and extending our current experiments in more colorspace.

ACKNOWLEDGMENT

This work was supported by the projects CREST (H2020-833464) and INFINITY (H2020-883293) funded by the European Commission.

REFERENCES

- [1] Y. Gao and H. J. Lee, "Local tiled deep networks for recognition of vehicle make and model," *Sensors*, vol. 16, no. 2, p. 226, 2016.
- [2] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and E. Kayafas, "A license plate-recognition algorithm for intelligent transportation system applications," *IEEE Transactions on Intelligent transportation systems*, vol. 7, no. 3, pp. 377–392, 2006.
- [3] Y. Tang, C. Zhang, R. Gu, P. Li, and B. Yang, "Vehicle detection and recognition for intelligent traffic surveillance system," *Multimedia tools and applications*, vol. 76, no. 4, pp. 5817–5832, 2017.
- [4] R. Chen, M. Hawes, L. Mihaylova, J. Xiao, and W. Liu, "Vehicle logo recognition by spatial-sift combined with logistic regression," in *2016 19th International Conference on Information Fusion (FUSION)*. IEEE, 2016, pp. 1228–1235.
- [5] G. S. Becker and G. J. Stigler, "Law enforcement, malfeasance, and compensation of enforcers," *The Journal of Legal Studies*, vol. 3, no. 1, pp. 1–18, 1974.
- [6] E. Elaad, A. Ginton, and N. Jungman, "Detection measures in real-life criminal guilty knowledge tests," *Journal of Applied Psychology*, vol. 77, no. 5, p. 757, 1992.
- [7] J. B. Kim and H. J. Kim, "Efficient region-based motion segmentation for a video monitoring system," *Pattern recognition letters*, vol. 24, no. 1–3, pp. 113–128, 2003.
- [8] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 12, pp. 2341–2353, 2010.
- [9] R. Gonzalez, R. Woods, and S. Eddins, "Digital image processing using matlab" gatesmark publishing," 2009.
- [10] P. Chen, X. Bai, and W. Liu, "Vehicle color recognition on urban road by feature context," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2340–2346, 2014.
- [11] B. Su, J. Shao, J. Zhou, X. Zhang, and L. Mei, "Vehicle color recognition in the surveillance with deep convolutional neural networks," in *Joint International Mechanical, Electronic and Information Technology Conference (JIMET 2015)*, 2015.
- [12] K. Pasupa, P. Kittiworapanya, N. Hongnorn, and K. Woraratpanya, "Evaluation of deep learning algorithms for semantic segmentation of car parts," *Complex & Intelligent Systems*, pp. 1–13, 2021.
- [13] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [14] R. Nock and F. Nielsen, "Statistical region merging," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 26, no. 11, pp. 1452–1458, 2004.
- [15] N. Dhanachandra, K. Mangle, and Y. J. Chanu, "Image segmentation using k-means clustering algorithm and subtractive clustering algorithm," *Procedia Computer Science*, vol. 54, pp. 764–771, 2015.
- [16] L. Najman and M. Schmitt, "Watershed of a continuous function," *Signal Processing*, vol. 38, no. 1, pp. 99–112, 1994.
- [17] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. C. Loy, D. Lin, and J. Jia, "Psanet: Point-wise spatial attention network for scene parsing," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 267–283.
- [18] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi, "Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.
- [19] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.
- [20] K. Fu, Q. Zhao, and I. Y.-H. Gu, "Refinet: A deep segmentation assisted refinement network for salient object detection," *IEEE Transactions on Multimedia*, vol. 21, no. 2, pp. 457–469, 2018.
- [21] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Ccnnet: Criss-cross attention for semantic segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 603–612.
- [22] C. Yu, J. Wang, C. Gao, G. Yu, C. Shen, and N. Sang, "Context prior for scene segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12416–12425.
- [23] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.
- [24] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:1412.7062*, 2014.
- [25] Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang, "Semantic image segmentation via deep parsing network," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1377–1385.
- [26] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1529–1537.
- [27] A. G. Schwing and R. Urtasun, "Fully connected deep structured networks," *arXiv preprint arXiv:1503.02351*, 2015.
- [28] G. Lin, C. Shen, A. Van Den Hengel, and I. Reid, "Efficient piecewise training of deep structured models for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3194–3203.
- [29] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.
- [30] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.
- [31] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.
- [32] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.
- [33] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on*

Medical image computing and computer-assisted intervention. Springer, 2015, pp. 234–241.

- [34] G. Ghiasi and C. C. Fowlkes, “Laplacian pyramid reconstruction and refinement for semantic segmentation,” in *European conference on computer vision*. Springer, 2016, pp. 519–534.
- [35] P. Burt and E. Adelson, “The laplacian pyramid as a compact image code,” *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983.
- [36] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, “Learning a discriminative feature network for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1857–1866.
- [37] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, “Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation,” *International Journal of Computer Vision*, pp. 1–18, 2021.
- [38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS’12. Red Hook, NY, USA: Curran Associates Inc., 2012, p. 1097–1105.
- [39] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” 2014.
- [40] C. Hu, X. Bai, L. Qi, P. Chen, G. Xue, and L. Mei, “Vehicle color recognition with spatial pyramid deep learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2925–2934, 2015.
- [41] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*, vol. 2. IEEE, 2006, pp. 2169–2178.
- [42] R. F. Rachmadi and I. Purnama, “Vehicle color recognition using convolutional neural network,” *arXiv preprint arXiv:1510.07391*, 2015.
- [43] W. Maungmai and C. Nuthong, “Vehicle classification with deep learning,” *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, pp. 294–298, 2019.
- [44] Q. Zhang, L. Zhuo, J. Li, J. Zhang, H. Zhang, and X. Li, “Vehicle color recognition using multiple-layer feature representations of lightweight convolutional neural network,” *Signal Processing*, vol. 147, pp. 146–153, 2018.
- [45] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” 2015.
- [46] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015, software available from tensorflow.org. [Online]. Available: <https://www.tensorflow.org/>
- [47] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, “Microsoft coco: Common objects in context,” 2015.
- [48] J. MacQueen *et al.*, “Some methods for classification and analysis of multivariate observations,” in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Oakland, CA, USA, 1967, pp. 281–297.
- [49] L. Rokach and O. Maimon, “Clustering methods,” in *Data mining and knowledge discovery handbook*. Springer, 2005, pp. 321–352.
- [50] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *kdd*, 1996, pp. 226–231.
- [51] T. Zhang, R. Ramakrishnan, and M. Livny, “Birch: A new data clustering algorithm and its applications,” *Data Mining and Knowledge Discovery*, vol. 1, no. 2, pp. 141–182, 1997.
- [52] Y. Cheng, “Mean shift, mode seeking, and clustering,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 17, no. 8, pp. 790–799, 1995.
- [53] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
- [54] P. Jaccard, “Distribution de la flore alpine dans le bassin des dranses et dans quelques régions voisines,” *Bull Soc Vaudoise Sci Nat*, vol. 37, pp. 241–272, 1901.