VR-PROUD: Vehicle Re-identification using PROgressive Unsupervised Deep architecture

R.M.S. Bashir a, M. Shahzad a,b,∗, M.M. Fraz a,b,c,d

a School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Islamabad, Pakistan
b Deep Learning Laboratory, National Center of Artificial Intelligence (NCAI), Islamabad, Pakistan
c Department of Computer Science, University of Warwick, Coventry, CV47AL, United Kingdom
d The Alan Turing Institute, British Library, London WC1V 2DB, United Kingdom

A R T I C L E   I N F O
Article history:
Received 1 June 2018
Revised 14 December 2018
Accepted 4 January 2019
Available online 15 January 2019

Keywords:
Vehicle re-id
Deep learning
Unsupervised
Clustering
Visual surveillance
Progressive learning
Self pace

A B S T R A C T
Vehicle re-identification (Re-ID) is one of the primary components of an automated visual surveillance system. It aims to automatically identify/search vehicles in a multi-camera network usually having non-overlapping field-of-views. Majority of the approaches dealing with the re-ID problem tackle it in a supervised manner which have certain limitations that pose challenges of generalization e.g., large amount of annotated data is required for training and is often limited to the dynamic growth of the data. Unsupervised learning techniques can potentially cope with such issues by drawing inference directly from the unlabeled input data and have been effectively employed in the context of person re-ID. To this end, this paper presents an approach that essentially formulates the whole vehicle re-ID problem into an unsupervised learning paradigm using a progressive two step cascaded framework. It combines a CNN architecture for feature extraction and an unsupervised technique to enable self-paced progressive learning. It also incorporates the contextual information into the proposed progressive framework that significantly improves the convergence of the learned algorithm. Moreover, the approach is generic and has been the first attempt to tackle the vehicle re-ID problem in an unsupervised manner. The performance of the proposed algorithm has been thoroughly analyzed over two large publically available benchmark datasets VeRi and VehicleID for vehicle re-ID using image-to-image and cross-camera search strategies and achieved better performance in comparison to current state-of-the-art approaches using standard evaluation metrics.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

Recent advancements of computer vision and machine learning algorithms coupled with the breakthrough internet of things (IoT) technologies have enabled us in gathering and processing the data acquired from different sensors for effective and efficient resource management leading towards the actual realization of the smart city concepts [1]. Automated surveillance is one of the most integral and important aspect to ensure safety and security of any smart environment. For this purpose, vehicle re-identification (Re-ID) serves as the basis and is one of the primary components of such an intelligent visual monitoring environment [2]. Furthermore, it has vital role in the applications such as live monitoring or multi-view vehicles tracking for urban surveillance or performing forensic examination on the stored data for different tasks like analyzing traffic flow conditions [3], finding patterns of different vehicles etc.

Automatic vehicle re-ID is a procedure to identify/search vehicles in a multiple camera network usually having non-overlapping field-of-views. It does so by assigning unique IDs to individually detected vehicles for the first time in a multiple camera environment and keeping track of them if identified at other camera locations as depicted in Fig. 1.

An obvious solution to vehicle re-ID is via license plate recognition [4] as it’s a unique ID of the vehicle. This makes the problem trivial, however, there are some issues e.g., in unconstrained real world scenarios the videos captured are often of low resolution that are affected primarily by environmental factors (e.g. fog, dust, rain, storm etc.), side viewpoints (i.e., neither frontal nor backward) and poor/improper illumination owing to the change in viewpoint or temporal light variations e.g., shadows etc. Moreover, there may exist issues pertaining to difficulties in reading the license plate especially in scenarios where the license plates having

∗ Corresponding author.
E-mail addresses: rbashir.msc16@seecs.edu.pk (R.M.S. Bashir), muhammad.shahzad@seecs.edu.pk (M. Shahzad), moazam.fraz@seecs.edu.pk (M.M. Fraz).
nonstandard templates (e.g., in developing countries), are partially broken, improperly localized, fainted and/or occluded etc. These issues thus make the problem of reading license plate nontrivial and consequently limits them for the problem of vehicle re-ID. Hence, other alternatives taking into account the appearance of vehicles are essential.

Appearance based techniques are usually based on geometric and structural attributes of the vehicles such as color, texture, shape [2,5,6] etc. The extracted attributes are then fed as input to a conventional classifier e.g., Support Vector Machine (SVM) [6] etc. Although these approaches perform fairly well but are limited in terms of accuracy since these handcrafted attributes e.g., colors are strongly affected by varying illumination and the shape for the same model/make of vehicles is merely different. With recent advancements in neural network architectures, the deep learning techniques allows better generalization by incorporating high number of hidden layers to learn high level features. Among several deep neural network architectures, the Convolutional Neural Networks (CNN) and Recurrent Neural Network (RNNs) are the most prominent approaches and have significantly outperformed other when it comes to process images for diverse computer vision related applications. In the re-ID context, the CNNs are usually employed for representation [7,8] or similarity learning [9] and RNNs are subsequently utilized to model temporal paths [9]. In similarity learning we feed the model with either a pair of similar/dissimilar images or a set of triplet (similar/dissimilar and neutral images) and train a similarity function e.g. Siamese NN [10] can output the similarity score between the two images after learning the distinct features. Representation learning on the other hand, extracts the features explicitly using a trained model and then a distance function (e.g., Euclidian distance) is used find the distance between the images. In comparison to similarity learning, it is more robust and efficient to solve large-scale re-ID problems and has therefore gained more attraction [11].

Majority of approaches dealing with the re-ID problem tackle it in a supervised manner which have certain limitations that pose challenges of generalization e.g., large number of data annotation is required for training and is often restricted to the dynamic growth of the available data. Techniques employing unsupervised learning methods can potentially cope with such issues by drawing inference directly from the unlabeled input data and have been effectively employed in the context of person re-ID [11,12]. In this paper, we propose an unsupervised approach to re-identify vehicles by training a base deep neural network with iterative and self-paced progressive unsupervised learning technique to allow transferring the deeply learned supervised representation to the unlabeled data. The proposed approach is based on appearance base features and extends the initial idea presented in [13] by incorporating the contextual information into the progressive framework that significantly improves the convergence of the learned algorithm. Moreover, additional experiments have been carried out to thoroughly analyze the algorithm’s performance in terms of scalability and practicality. In the following, we highlight the significant contributions proposed in this work:

- We have modeled the problem of vehicle re-ID as an unsupervised learning paradigm by proposing a progressive two-step cascaded framework. At first instance, a CNN is used for extracting vehicle features, afterwards an unsupervised clustering algorithm is utilized for appearance-based grouping of similar vehicles by enforcing certain heuristic constraints to obtain accurate and stable clusters. These robust clusters (representing vehicles) are then utilized to fine-tune another CNN, where the training sample set grows till convergence with progressively refined clusters to enable unsupervised self-learning. To the best of our knowledge, such a cascaded formation has not been adopted in the domain of vehicle re-ID.
We have incorporated color information in the cascaded network architecture that yields fast and reliable selection of clusters (i.e., vehicles) resulting in fast convergence with improved accuracy. For this purpose, a color CNN is trained to reliably filter the clustering results for robustly populating the training pool of images in subsequent iterations.

The proposed methodology has been thoroughly evaluated over a couple of large publicly available vehicle re-ID benchmark VeRi [7] and VehicleID [14] datasets using image-to-image and cross-camera search strategies; outperforming the counterpart state-of-the-art supervised algorithms (PROVID [14], DRDL [15]) in Rank-1 accuracy.

2. Related work

Object re-identification solves the re-ID problem of same object across the network of cameras if it has been observed before in the network. It plays an important role in tracking and identifying objects, e.g., persons and vehicles etc., in a multi-network environment. To this end, there exist few techniques aiming to solve the generic object re-ID problem. E.g., the techniques proposed by Ayedi et al. [16] and Kink et al. [17] utilize multi-scale covariance based image descriptor with quadtree based structure and spatially distributed binary classifiers for generic object re-ID respectively. Nevertheless, owing to wide range of potential applications, e.g., in security/surveillance and law enforcement, many existing approaches have posed the re-identification problem in the domain of person re-ID. In this regard, majority of the techniques make use of conventional machine learning methods to solve the problem of re-id. For instance, Camps et al. [18] developed an end-to-end person re-ID system for the real busy airport environment which performs person detection via mixture of Gaussian, tracking via Kanade-Lucas-Tomasi tracker and person re-ID by employing a metric learning technique to find similarities using spatial, color and texture features. Similarly, Perwaz et al. [19] employed a hybrid feature representation that is reinforced by metric learning. Lui et al. [20] proposed a generic metric learning based technique which incorporates the extracted multimodal illumination features in the person re-id pipeline to boost the performance by discovering the shift invariance property in the log-chromaticity space. In addition to these, several other algorithms to extract and describe hand crafted features specifically for person re-ID have been proposed e.g., Histogram Oriented Gradients (HOG) [21], Local Maximal Occurrence (LOMO) [22,23], Weighted Histogram of Overlapping Stripes (WHOS) [23,24], Bag of Words (BoW) [25] etc. Subsequently, these extracted features are utilized to train typical classifiers such as Support Vector Machines (SVM) [26], AdaBoost [27] etc. For a comprehensive overview on advancements in feature extraction and metric learning in the domain of person re-ID, the readers are referred to a systematic evaluation and benchmark study, carried out in [28], including 11 feature extraction algorithms together with 22 metric learning and ranking techniques evaluated over 16 publicly available datasets.

Apart from the supervised learning a few researchers have also attempted to solve the person re-ID using the unsupervised learning techniques. In this regard, Kodirov et al. [29] presented a novel approach where they proposed graph based regularized dictionary learning with L1 norm instead of L2 which makes it robust to outliers, pose and background changes. Yang et al. [30] also proposed an unsupervised learning approach using weighted linear coding to learn the multi-level descriptor from the raw image pixels. Zheng et al. [25] proposed a dataset and an unsupervised bag of words technique for person re-ID. Liao et al. [22] and Mumtaz et al. [23] performed metric learning using features including Gaussian of Gaussian and LOMO via Cross-view Quadratic Discriminant Analysis (XQDA) etc. In the case of person re-ID in video sequences, Liu et al. [31] focused on eliminating the labelling errors, often introduced as a result of inaccurate tracking, by incorporating the semantic similarity confidence scores estimated using k nearest neighbors in the multiple instance metric learning framework.

The aforementioned approaches relying on conventional supervised and unsupervised machine learning performs fairly well and accurate but lacks the generalization capabilities for different datasets. Recent advancements in artificial neural networks (i.e., deep neural networks DNNs) allows to obtain such generalization capability by incorporating high number of hidden layers. Among several deep neural network architectures, the Convolutional Neural Network (CNN) is the most prominent approach and has significantly outperformed other techniques when it comes to process images for diverse computer vision related applications. It has been found highly effective and have significantly outperformed other methods and became state-of-the-art particularly in image recognition. In the following, we recall the major DNN architectures through which various deep learning strategies have been derived. An earlier and worth mentioning object classification architecture is AlexNet presented by Krizhevsky et al. [32] which is a large deep CNN that won the ILSVRC-2012 (ImageNet Large-Scale Visual Recognition Challenge) competition comprising of 5 convolutional layers followed by 3 fully connected layers with ReLU (Rectified Linear Unit) activation function. The success of AlexNet ignited the use of deep learning models in image recognition. The next model which really pushed forward the CNN architectures to get more deep (i.e., to include more hidden layers) is the VGG or OxfordNet model proposed by Simonyan and Zisserman [33] in 2014. The developed deep VGG architecture consisted of 19-hidden layers and smaller 3 × 3 kernels with stride and padding of 1 together with 2 × 2 max-pooling layers with stride of 2. Later, Szegedy et al. [34] proposed an architecture consisting of a 22 layer deep CNN called GoogleLeNet (or Inception) which reduced the number of parameters from 60M (AlexNet) to 4M and used average pooling instead of fully connected layers in the last layer of the network. The idea of going deep is further exploited by He et al. [35] who proposed a 152-layer deep learning architecture, known as ResNet, that won the ILSVRC 2015 with a remarkable low error rate of 3.6% setting up a new record in image recognition and localization via a single network architecture. The network, consisting of the so-called ResNet-blocks, essentially solves the vanishing gradient problem by adding shortcut connections.

Owing to the success of these architectures especially in problems related to classification, recognition and detection, several researchers have recently tried to solve the person re-ID using such deep architectures. In this regard, Yuan et al. [36] presented a deep multi instance learning framework for person re-ID. They built an end-to-end system using the deep multiple instance learning where they used L2 norm for finding the distance between two persons using the fused feature representation of upper, body and leg. Franco et al. [37] introduced convolutional covariance features to retrieve noise-invariant information which is embedded with deep features for improved hybrid feature representation. Oh et al. [38] presented a deep learning architecture for person re-ID for crowded environment. They designed a technique of learning features and distance jointly in unit hyperspace embedding to decrease the complexity and then used the minimum average distance matching for robustness. Zheng et al. [12] proposed a semi-supervised technique to use generative adversarial network (GAN) to generate unlabeled samples. Deng et al. [39] also presented an unsupervised approach for image to image cross domain adaption using the self-similarity and domain-dissimilarity in the training. They used the similarity preserving GANs consisting of the Siamese Neural Networks using the contrastive loss for the re-ID purpose. Fan et al. [11] proposed an approach using deep learning for unsupervised person re-ID. They used self-paced learning algorithm
for the extraction, clustering and selection of the images for fine tuning. Zhou et al. [40] also employed a self-paced constraint together with a symmetric soft polynomial regularizer for discriminative and robust feature extraction. Similarly, Wang et al. [41] presented a joint attribute-identity learning based approach to simultaneously learn both semantic and attributes in source domain and transferring it to target domain to enable unsupervised learning. Lv et al. [42] proposed an unsupervised incremental learning based approach which combines the visual features and learned spatio-temporal features using a Bayesian fusion algorithm to achieve the improved performance.

The problem of vehicle re-ID is somewhat relevant to the problem of person re-ID. However, the proposed person re-ID techniques are not directly suitable to re-identify vehicles since the widely employed feature descriptors such as LOMO, WHOS, GOG etc. are particularly tailored for persons and therefore may not work in case of vehicles due to monotone texture/appearance. Few approaches have been proposed in the domain of vehicle re-ID. Among them, Kuo and Nevatia [43] proposed a method based on linear embeddings and boosted cascaded classifier to detect and classify vehicles in non overlapping multi-view camera configurations. Similarly, Feris et al. [44] trained an AdaBoost classifier to re-identify vehicles primarily using shape/motion features and augmented the training data by synthetically generating images using Poisson distribution to over occlusion problem. Zapletal and Herout [45] proposed an approach to tackle the vehicle re-ID problem via linear regression in multiple camera environment. The features used include (color) histogram of oriented gradients and apart from re-ID their method was also able to estimate vehicle flow parameters with application to automated traffic surveillance. Yang et al. [46] presented a deep learning based solution to vehicle re-ID where the CNNs [32] are used for extracting features over the proposed COMPARIS dataset and later employed conventional machine learning classifiers for the purpose of re-ID. In similar context, Liu et al. [7] also presented a large scale benchmark VeRi-776 dataset for vehicle re-ID and depicted results obtained using hybrid deep and handcrafted features (i.e., BOWs and SIFT) with six different appearance based vehicle re-ID methods. The images contained in VeRi are specifically captured to meet the actual re-ID scenarios including cross/multipe non overlapping camera IDs (2 to 18 distinct views) of massive vehicles in real traffic situations. Liu et al. [8] employed a staged (top-down) framework to re-identify vehicles where in first stage the vehicles are primarily grouped together using color/textures information and later exact vehicles are matched based on license plate information extracted by exploiting a trained siamese [10] network architecture. Liu et al. [15] presented a deep CNN based relative distance based similarity learning method to essentially place same/similar vehicles close in feature space. Zhang et al. [47] also formulated the problem as a ranking problem and proposed a triplet loss based similarity estimation method for better vehicle re-ID performance. Their method essentially improves the traditional triplet loss in enabling stronger classification oriented constraint and improved triplet sampling strategy. Shen et al. [9] employed a two staged modular method in which the first method used a markup random field to translate the vehicle paths as sequential pattern that is exploited by the later module employing a Long Short Term Memory (LSTM) model to compute similarity score for the purpose of vehicle re-ID. Liu et al. [14] introduced a metric learning based approach which they call Null-space-based FACT (NuFACT) that basically projects the extracted features onto a null space prior to using Euclidean distance for similarity computation between the vehicles.

All of these approaches tackle the vehicle re-ID problem in a supervised manner and need huge amount of annotated training data and are therefore limited in terms of scalability. The potential use of unsupervised learning algorithms may help in this regard. To address this, inspired from the unsupervised techniques mentioned above, we have proposed a progressive unsupervised formulation that takes as input the pre-trained model weights and then progressively learns new weights using unlabelled data. The details of the proposed approach are presented in the subsequent section.

3. Methodology

Fig. 2 presents the block diagram highlighting the working procedure of the proposed algorithm. It essentially formulates the whole vehicle re-ID problem into an unsupervised learning paradigm using a progressive two step approach: In the first phase, a base deep pre-trained CNN model is employed which serves as an initializer for the subsequent step where it is initially used to extract image features. These extracted features in the second phase are fed to an unsupervised clustering algorithm to group similar vehicles on the basis of their appearance. The whole idea of progressive model training is to arbitrarily replace the actual labels with the cluster ID sequence numbers to enable latent space representation of the original labels. To elaborate, when we progressively train the network in unsupervised manner, the input images are fed without labels to the base network which performs the feature extraction using pretrained weights. These extracted features are clustered and the respective cluster IDs are then assigned as “pseudo” labels to the clustered images for the next subsequent iteration. The clustering results are further refined to obtain accurate and stable clusters by enforcing certain heuristic constraints. These robust clusters (representing vehicles) are then utilized to fine-tune another CNN network having the same architecture as the base CNN. The process is iteratively performed where the training sample set grows in every iteration with increasingly robust clusters to enable unsupervised self-progressive learning till convergence. Next, we present the detailed explanation of the working procedure of the proposed scheme in the dedicated subsections as follows.

3.1. Base model training

The whole idea of using a base model is to transfer the learned information (i.e., features or weights fine tuned over an existing labelled dataset) to a deep unsupervised vehicle re-ID scheme which is then trained over new (unlabelled/unseen) dataset in a progressive manner. Thus, assuming that the labelled dataset is given for training, the idea of base model training is to make use of these existing labels to fine tune any generic deep feature extractor (e.g., ResNet50 [35]) where the last layer of the model is replaced with a fully connected layer having SoftMax activations and neurons equal to the number of vehicles M. However, prior to training, the existing labeled dataset is preprocesssed together with data augmentation using standard techniques (e.g., rotation, flipping and shifting etc.). The optimization of categorical cross entropy loss function is achieved using stochastic gradient descent optimizer having the learning rate of 0.001 and momentum of 0.9. With these settings the problem now becomes a classification problem with number of classes equals to M for classification. The model is initialized with pre-trained weights using well-known ImageNet dataset to avoid the training from scratch. Fine tuning is done till convergence to train the base model, which is further iteratively used for initialization purposes (e.g., extraction of the image features for clustering) and initializing the model for fine tuning on each iteration.
3.2. Progressive model training

Once the base model is trained, it is utilized to initialize and fine-tune the progressive network module including the feature extraction, clustering & reliable selection to be explained in the subsequent subsections.

3.2.1. Feature extraction

This step is initialized by extracting the feature vectors using the base model trained in the last step. The output of the average pooling layer (prior to the classification layer in ResNet50) is considered as features of the input image. To elaborate, the features are obtained by removing the last layer of classification from the model and instead use the average pooling layer as output i.e.,
\[ f_i = \theta(x_i, w_i) \]  
(1)
where \( x_i \) denotes the \( i \)th input image, \( w_i \) are the weights and the \( \theta \) represents the learned model which outputs the feature vector \( f_i \). During the next iteration features are extracted using the newly trained model which is considered more accurate and would yield more suitable features.

3.2.2. Feature clustering

Once the features of the input images have been extracted out, the idea now is to somehow group the similar vehicles together so that a relatively more refined training sample set could be generated. For this purpose, we fed the extracted features \( F = [f_1, f_2, \ldots, f_n] \) to \( k \)-means algorithm that clusters the similar features (i.e., the corresponding images) together. This enables us to formulate the basis for progressive training in a sense that the clustered group of images are assigned the same labels which when iteratively used for training improves the model for feature extraction. Using \( k \)-means allows us to obtain a set \( C = \{c_1, c_2, \ldots, c_k\} \) of \( k \) (\( =M \)) cluster centroids by minimizing the following optimization function:
\[ C \leftarrow \arg \min_{c_j} \sum_{i=1}^{n} \sum_{j=1}^{k} \|f_i - c_j\|^2 \]  
(2)
where \( c_{j=1..M} \) denotes the obtained \( M \) cluster centroids. Although \( k \)-means cluster similar images together but it may happen that due to noise (i.e., similarity of background appearance, texture, shape etc.) together with variations of size/density of vehicles in the data, different vehicles may get wrong assignments and consequently get grouped into the same cluster. These wrong assignments may potentially make the proposed latent variable formulation more susceptible to oscillations or to get stuck in bad local optimum while fine tuning the CNN model. To this end, it is necessary to refine the obtained clustering results to improve the model accuracy and convergence.

3.2.3. Reliable feature selection

A naive solution to refine the clustering results may include computing the Euclidean distances of all points belonging to a single cluster with respect to its corresponding centroid and retaining only the near ones. However, this may only tackle the weak assignments (i.e., far distant features) but may not be appropriate in cases of wrong cluster assignments where a feature whose distance is near to the center but is not similar to it. A better solution to this would be to find the similarity of all the features with respect to their respective centroids and then discard all those whose similarity is less than a certain defined threshold, e.g., using dot product (cosine similarity) [11] as follows:
\[ \frac{f_i}{\|f_i\|} \cdot \frac{c_j}{\|c_j\|} > \lambda \]  
(3)
where \( \cdot \) denotes the dot product of the normalized feature and center vectors. Such a similarity function is useful in handling both the weak and wrong assignments, but it does not take into account the magnitude (i.e., suffer with the collinearity problem [48]). Other similarity measures e.g. Manhattan (\( L_1 \)), weighted mean variance, Euclidean (\( L_2 \)), Chebyshev (\( L_{\infty} \)), Mahalanobis, Canberra, Bray-Curtis, squared chord and squared chi-squared distances [49] may be employed but all have their own limitations. To this end, in this paper, we have refined the clustering results to obtain more accurate and stable clusters by enforcing certain heuristic constraints. For this purpose, since the color provides a strong
hint particularly in case of vehicles (i.e., the dominant color stays the same), we have trained a color CNN model to incorporate this as prior knowledge in order to predict the colors of vehicles and retaining the vehicles having same color in an individual cluster. For inference purpose, it is essential to determine the dominant color of the cluster which is obtained via majority voting, i.e., by taking the mode of the vehicles in a particular cluster as follows:

$$\beta_j = \text{mode}\left(\{b_m\}_{m=1}^{\vert V_j \vert}\right) \quad \text{with} \quad b_m = \psi(v_m, w_c)$$

(4)

where $\psi(v_m, w_c)$ denotes the color CNN which returns the color $b_m$ of $m$th image $v_m$ in the $j$th cluster using trained (or fine-tuned) weights $w_c$. If $V_j$ represents the set of images contained in $j$th cluster, then $\vert V_j \vert$ denotes the set cardinality (i.e., number of images in any particular $j$th cluster) while $\beta_j$ denotes the dominant color of the vehicles in $j$th cluster obtained by taking the statistical mode (majority voting) of vehicle colors $\{b_m\}_{m=1}^{\vert V_j \vert}$ corresponding to individual cluster images retrieved by the employed color CNN. The estimated dominant clusters color $\{\beta_j\}_{j=1}^J$ are then used to discard dissimilar color images in every cluster.

The benefits of using color CNN model is two-fold i.e., it enables the reliable refinement of the obtained clusters by forcing the constraint such that each cluster has vehicles of one color only. Besides this, it also improves the overall convergence of training the whole network. Fig. 3 conceptually visualizes the scheme for reliable image selection during training where we ensure that the red diamonds are discarded if belonging to different color than clusters dominant color.

3.2.4. Model training and optimization

The color CNN is used for multinomial (or multi-class) classification of distinct vehicle colors where the training of the model has been done using the color information for each (training) vehicle image. The training set comprises of available vehicle images $x_i$ together with the color information as labels $y_i$ provided in the two employed datasets. The ResNet50 [35] architecture pre-trained on ImageNet has been employed as the color CNN model with the addition of average pooling, 50% dropout and final classification layer with SoftMax activation. Categorical cross entropy is used for loss calculation of the training data with stochastic gradient descent as an optimizer, with the learning rate of 0.001, momentum of 0.9 and batch size of 16. The model is fine-tuned for 20 epochs.

The trained color CNN is used in the progressive part to select the reliable cluster images (representing vehicles). These filtered images are subsequently utilized to fine tune another CNN model having the same base CNN architecture. To elaborate, after the selection of reliable images $R$ from the clusters, the cluster centroids are used as labels $Y$ to prepare the training dataset. Similar to color CNN, the ResNet50 [35] network architecture is used with slight modifications including fully connected layer with an additional dropout layer using SoftMax activation. The base model weights $\theta$ are used for initialization. The categorical cross entropy loss function is loss function $L$ is employed together with stochastic gradient descent for training with the learning rate and momentum having values of 0.001 and 0.9 respectively. In each iteration, the fine-tuning of the model is achieved using reliable training set of images with the following optimization function:

$$\min_{w_i} \sum_{p=1}^{R_j} L(y_p, \theta (f_p, w_i), w_i)$$

(5)

where $w_i$ represents the fine-tuned weights and $f_p$ denotes a reliable training sample/image. The procedure is iteratively performed where in each iteration, the training sample set grows with increasingly robust (and refined) clusters enabling unsupervised self-progressive learning till convergence as depicted in Fig. 4. The convergence of the algorithm is directly related to the saturation in the number of reliable images. The reason why such saturation in reliable training samples leads to convergence is essentially because in each progressive iteration, the number of reliable images increases which consequently helps in better feature representation learning that in turn improves the model. Algorithm 1 depicts the whole pseudo code algorithm employed to train the proposed neural network architecture.

4. Experimental evaluation

There does not exist any notable architecture which tackles the problem of vehicle re-ID in an unsupervised manner. Nevertheless, to evaluate and compare the performance of the proposed network architecture, we analyzed the obtained results by comparing with the state-of-the-art supervised deep learning based vehicle re-ID approaches including PROVID [14] and DRDL [15]. Next, we give an overview of the employed datasets.
Algorithm 1 Unsupervised Vehicle Re-Identification.

**Input:** Unlabeled images \( \{x_i\}_{i=1}^n \);
- Number of clusters \( k \);
- Base model \( \theta(\cdot, \mathbf{w}_b) \);
- Color model \( \psi(\cdot, \mathbf{w}_c) \);

**Output:** Model \( \psi(\cdot, \mathbf{w}_c) \);
1. **Initialization:** \( \mathbf{w}_c \leftarrow \mathbf{w}_c^0 \);
2. **while maximum iterations not reached do**
3.  **Initialization:** \( \mathbf{w}_c \leftarrow \mathbf{w}_c^0 \);
4.  **while maximum iterations not reached do**
5.  **Cluster colors:** \( \beta_j \) with \( \beta_j = \text{mode} \left( [b_m]_{m=1}^{\lvert V_i \rvert} \right) \) & \( b_m = \psi(\mathbf{v}_m, \mathbf{w}_c) \)
6.  \( R, Y = \Phi \);
7.  **for** \( j = 1 \) to \( k \) **do**
8.  \( l, L = \Phi \);
9.  **for** \( m = 1 \) to \( \lvert V_i \rvert \) **do**
10.  **if** \( \psi(\mathbf{v}_m, \mathbf{w}_c) = \beta_j \) **then**
11.  \( l \leftarrow [l, \mathbf{v}_m] \); // reliable image
12.  \( L \leftarrow j \); // labels
13.  **end**
14.  **end**
15.  \( R \leftarrow [R, l] \);
16.  \( Y \leftarrow [Y, l] \);
17.  **end**
18.  **Fine tuning:** \( \min_{\mathbf{w}_c} \sum_{x_i} L(y_{ip}, \theta(\mathbf{f}_p, \mathbf{w}_c), \mathbf{w}_c) \)
19.  **end while**

4.1. Datasets

To evaluate the algorithm performance, two different datasets have been employed. The first dataset has images covering the area of interest in a cross nonoverlapping multi-view multi-camera scenario while the images of the other dataset are acquired in a single-view multi-camera scenario. The details of both these datasets are provided in the following sub-sections.

4.1.1. Vehicle re-identification (VeRi)

VeRi [7] is a comprehensive dataset designed especially for the sake of vehicle re-ID. It covered the area of 1 km² with 20 cameras installed arbitrary at different locations, covering the traffic while ensuring the quality of the data. It includes different viewpoints like road crossings, two and four lane roads, various lighting conditions and backgrounds. The videos are acquired with 1920 x 1080 image resolution having frame rate of 25 frames per second. The videos are collected for 24 h from the 20 cameras with time-slots between 4:00 PM to 5:00 PM after compression and transcoding. They collected one frame out of every five frames in 25 frames and end up in 360,000 frames for one hour. Manual annotation is done on the selected frames which resulted in 50,000 images with 776 distinct vehicles and 9000 tracks. Each vehicle is captured in at least 2 cameras to ensure that it works for the purpose of re-ID. The dataset is further divided into the train and test sets. The training set contains 576 individual vehicles with 37,781 images while the test set have 200 distinct vehicles having total of 11,579 images. Query set is created by selecting one image from each car and camera which counts to 1678 images. Additionally, information related to vehicle color and model is also available. There are 10 colors selected for the vehicles i.e. green, gray, black, white, yellow, red, orange, golden, blue, brown. There are 9 types of vehicles in the dataset i.e. SUV, MPV, sedan, hatchback, truck, estate car, bus, pickup and van. Dataset also includes the information about the vehicle brand and covers about 30 brands including BMW, Toyota, Audi etc. Although this dataset has few demerits i.e. this dataset is not normalized in terms of the image distribution between vehicles as there are some vehicles having 11 images and then some having 289 images. Number plates are mostly marked with black rectangle so one cant read it.

4.1.2. VehicleId

In comparison to VeRi, VehicleID [15] is a large dataset with total number of 26,000 distinct vehicles with over 200,000 images of those vehicles. Vehicles are assigned IDs based on their license plate numbers and are attached with the images. Moreover, the color and model information is also available for 10,319 vehicles and 9000 images. The total of 250 vehicle models are covered in this dataset including Mercedes-Benz R-class 2010, Audi A6L–2012, Toyota Prado 2004 etc. The vehicles have 6 dominant colors i.e.,
black, blue, gray, red, silver, white, yellow. The training set consists of 13,164 vehicles while the test set comprises of multiple test sets having different difficulty levels ranging from small set having 800 vehicles, medium set having 1600 vehicles and large set having 3200 and even larger sets having 6000 and 13,164 vehicles respectively. Although the dataset is huge but lacks the multi view information as there is only one camera being used in the creation of dataset which means mostly vehicles are captured from two views i.e. front and back. The vehicle distribution is non-uniform dataset meaning few vehicles have very less number of images per vehicles and some have relatively higher number of images with the range of around 2 to 142 images per vehicle.

### 4.2. Performance analysis

The performance analysis of the proposed architecture for vehicle re-ID has been carried out using different testing subset of images as depicted in Table 1. These subsets of images are carefully chosen to ensure the generalization capabilities of the proposed approach. The VehicleID dataset does not contain any query set. In order to cope with it, the query set is created from the test set by randomly selecting a query candidate for each vehicle and making it to have at least two images per vehicle in test set. Moreover, the vehicle distribution is not uniform in both the datasets and therefore we carefully designed the new subsets to avoid possible under-fitting and over-fitting scenarios along with the originals. In the subsequent subsections, we detail the evaluation strategy and the employed metrics to evaluate the performance of the proposed approach.

### 4.2.1. Evaluation strategy & metrics

Finally, the progressively trained network is evaluated under two scenarios/strategies, i.e., either using a cross-camera search strategy in which multiple view information exists (contained only in VeRi dataset) or via an image-to-image search strategy for which there is no multi-view information available (i.e., when vehicle data is provided only). In the former, the search (or test) query vehicle is searched across the other cameras whereas in the latter case, the search for query vehicles is carried out in all the images irrespective of the camera (or viewpoint) information. Using these two strategies, following are the evaluation metrics which have been computed:

- **CMC Curve:** Cumulative match curve is the measure the performance of 1:m ranking capabilities of an ID system.
- **Rank:** Rank measures the similarity of a test to its class e.g. if test1 corresponds to class1 and found in top1 results then its called rank@1, if found in top5 results then its called rank@5 and so on.
- **mAP:** Mean average precision is the average of all the average precision as can be seen below.

\[
AP = \frac{\sum_{k=1}^{n} p(k)g(k)}{N_q}
\]

where \(n\) is the number of test and \(N_q\) is the number of reference (ground truth) images while the precision is \(p(k)\) at the \(k\)th position. \(g(k)\) represents the indicator function where the value is 1 if match is found at \(k\)th else 0. The mean Average Precision (mAP) is formulated as follows:

\[
mAP = \frac{\sum_{q=1}^{Q} AP(q)}{Q}
\]

where \(Q\) is the total number queries.

#### 4.2.2. Hardware & experimental settings

The convergence of the algorithm is directly related to the saturation in the number of reliable images. The reason why such saturation in reliable training samples leads to convergence is essentially because in each progressive iteration, the number of reliable images increases which consequently helps in better feature representation learning that in turn improves the model. The convergence is achieved when the number of iterations crosses the maximum set limit (30 in our case), or the number of reliable images become stable, i.e., when there is no change in the number of reliable images in consecutive iterations for more a certain defined delta \(\Delta\). In our case, the values for these parameters are empirically set as \(n = 3\) and \(\Delta = 0.01\%\) of the training data set.

Additionally, within each iteration, we primarily used the stability of loss and accuracy as the convergence criterion. Empirically, we have seen that the loss gets stable (and minimum) in around 30–40 epochs in all our experiments with the following hardware specifications: Intel core i7 processor, Nvidia Pascal X 12 GB GPU and 64 GB of RAM. In terms of software, we employed the batch size in the range of 16–32 using both Keras and Tensorflow backend. Specifically, Keras provides us with the loss and accuracy metrics which can be easily monitored from the console logs and tensorboard. Alternatively, an early stopping built-in criterion in Keras can be employed.

#### 4.2.3. Quantitative and qualitative results

For VehicleID, the training images are split into multiple suitable (uniform) sets containing 16,000 to 27,000 images while for testing, we used the given test sets containing 800 (easy), 1600 (medium), 2400 (difficult) vehicles. The query sets are formed on the basis of random selection from the test sets. For VeRi dataset, the training and testing is performed using the all the given images Table 2 provides the quantitative evaluation results over both the utilized datasets.

---

**Table 1**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>VehicleID</th>
<th>VeRi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>9 images per vehicle</td>
<td>65 images per vehicle</td>
</tr>
<tr>
<td>Minimum</td>
<td>2 images per vehicle</td>
<td>11 images per vehicle</td>
</tr>
<tr>
<td>Maximum</td>
<td>142 images per vehicle</td>
<td>289 images per vehicle</td>
</tr>
<tr>
<td>Train Set (Full)</td>
<td>113,346 images of 13,164 vehicles</td>
<td>images of 576 vehicles</td>
</tr>
<tr>
<td>Test Set (Small)</td>
<td>6493 images of 800 vehicles</td>
<td>-</td>
</tr>
<tr>
<td>Test Set (Medium)</td>
<td>13,377 images of 1,600 vehicles</td>
<td>-</td>
</tr>
<tr>
<td>Test Set (Large)</td>
<td>19,777 images of 2,400 vehicles</td>
<td>-</td>
</tr>
<tr>
<td>Test Set (X Large)</td>
<td>26,353 images of 3,200 vehicles</td>
<td>-</td>
</tr>
<tr>
<td>Test Set (XX Large)</td>
<td>48,922 images of 6,000 vehicles</td>
<td>-</td>
</tr>
<tr>
<td>Test Set (Full)</td>
<td>108,221 images of 13,164 vehicles</td>
<td>11,579 images of 200 vehicles</td>
</tr>
<tr>
<td>Query Set (Full)</td>
<td>-</td>
<td>1678 images of 200 vehicles</td>
</tr>
</tbody>
</table>
As evident, the performance of the proposed scheme exceeds the state-of-the-art supervised deep learning based approaches including DRDL and PROVID in Rank@1 accuracy in both image-to-image and cross-camera strategies. In Rank@5, the algorithm also outperforms in image-to-image strategy and also provides competitive results in cross-camera strategy. Fig. 5 depicts the CMC curve in image-to-image search scenario obtained by the proposed VR-PROUD algorithm using both VehicleID and VeRi datasets. In case of VeRi dataset, the orange line depicts the extra-ordinary performance of the proposed architecture primarily due to the fact that many images of the same vehicles from single view point are available at temporally close scales. In contrast, the blue curve shows the performance of the image-to-image search over VehicleID dataset which contains images of the same vehicles at relatively different time scales and viewpoints. Fig. 6 represents the CMC curve achieved in cross-camera search strategy which shows that the curve reaches 90% at Rank@5 characterizing that the presented approach correctly determines the vehicle ID in 5 top matches. On VeRi we performed tests using the proposed methodology and we got 83.19% as Rank@1 which is higher than the PROVID which is 81.56% and 40.05% mAP which is less than the PROVID 53.42%. The reason behind the low average mean precision is that in PROVID [14] they are using three step progressive learning to filter out the results and adding them to improve the accuracy, i.e., NuFACT having mAP = 48.47%, Rank@1 = 76.76%, Rank@5 = 91.42% which rely on the fusion of different attributes, then using the number plate information with NuFACT + Plate-SNN to get mAP = 50.87%, Rank@1 = 81.11%, Rank@5 = 92.79% and finally using the spatio temporal reasoning they get PROVID mAP = 53.42%, Rank@1 = 81.56%, Rank@5 = 95.11%.

Figs. 7 and 8 shows the qualitative results obtained using both the VeRi and VehicleID datasets via image-to-image and cross-camera search strategy respectively. In Fig. 8 (first row), we can see that the top-1 image is correctly searched while the top-2, top-3 and top-4 shows incorrect results with vehicles having different colors and viewpoints. Similarly, in the 2nd row, the obtained results have similar background texture/appearance (here we refer to the green bushes in top-1, top-2, top-3 and top-5). In third and fourth rows, the taxi and bus are easily recognized owing to the existence of distinct textures both in case of taxi and bus respectively. The image-to-image search results obtained over VehicleID are shown in Fig. 7 where majority of the vehicles (first five and the seventh row) are correctly identified with rank-1 accuracy. For cases, where the model fails to retrieve the vehicle in rank-1 is shown in 6th row where it identifies the vehicle with rank-5 accuracy. The overall rank-1 and rank-5 accuracies obtained over VehicleID dataset are 71.45% and 81.69% respectively. On similar note, the rank-1 and rank-5 accuracies obtained using VeRi dataset are 83.19% and 91.12% respectively as reported in Table 2. Moreover, Table 3 shows the results in cross-domain adaptation scenario where we used VehicleID for training the base-model while tested the performance of the trained base-model using other VeRi test dataset. Later, the (VehicleID trained) base-model is fine-tuned in an unsupervised manner using VeRi training dataset and subsequently evaluated using the same test data. Table 4 depicts the quantitative results achieved in this experiment which shows the performance improvement when trained in an unsupervised manner as proposed in this paper.

In terms of scalability and practicality, there may arise situations when we either do not have any prior information or only have rough idea about the number of distinct vehicles in advance. As a consequence, this makes it difficult to select the precise value for $k$. In this regard, to see the performance of the proposed scheme, we have explicitly done experiments with rough number of clusters. We ran the proposed algorithm with three different $k$ values and report the achieved quantitative results in Table 4.

<table>
<thead>
<tr>
<th>Dataset/Search</th>
<th>Image-to-image</th>
<th>Cross-camera</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rank@1</td>
<td>Rank@5</td>
</tr>
<tr>
<td>VehicleID [15]</td>
<td>DRDL</td>
<td>45.41</td>
</tr>
<tr>
<td></td>
<td>NuFACT [14]</td>
<td>43.72</td>
</tr>
<tr>
<td></td>
<td>VR-PROUD</td>
<td>71.45</td>
</tr>
<tr>
<td>VeRi [7]</td>
<td>PROVID [14]</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>VR-PROUD</td>
<td>100</td>
</tr>
</tbody>
</table>
For 7 query images, the achieved Top 5 results under image-to-image evaluation search shows the effectiveness of the approach. The first, second, third, fourth and fifth row contains the correct results of the car searched. In the sixth we can see that the van searched is found in the top-4 and top-5 and in the top-1 to top-3 the first image is similar but not the same and top-2 looks similar but is the back view of the van. However, in the sixth query image the results are not that bad as the only difference is the vehicle model (i.e. the lights design is changed).

Table 3
Quantitative results showing performance improvement when trained in unsupervised manner in cross-domain adaptation scenario. The iteration 0 indicates the result of base-model trained in supervised manner using VehicleID dataset. Iterations 1 to 5 shows the performance improvement when the base-model is iteratively trained using VeRi dataset via proposed unsupervised scheme.

<table>
<thead>
<tr>
<th>Model</th>
<th>Iterations</th>
<th>Rank@1</th>
<th>Rank@5</th>
<th>Rank@10</th>
<th>Rank@20</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base (supervised)</td>
<td>0</td>
<td>51.54</td>
<td>65.31</td>
<td>73.18</td>
<td>79.73</td>
<td>16.44</td>
</tr>
<tr>
<td>Progressive (unsupervised)</td>
<td>1</td>
<td>54.52</td>
<td>68.95</td>
<td>75.32</td>
<td>81.46</td>
<td>20.96</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>57.50</td>
<td>72.94</td>
<td>77.94</td>
<td>84.08</td>
<td>23.11</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>54.58</td>
<td>70.50</td>
<td>77.83</td>
<td>83.07</td>
<td>22.64</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>57.51</td>
<td>70.79</td>
<td>77.23</td>
<td>83.49</td>
<td>22.73</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>55.78</td>
<td>70.02</td>
<td>76.51</td>
<td>83.25</td>
<td>22.75</td>
</tr>
</tbody>
</table>
Fig. 8. Top 5 results of the query images for VeRi dataset using cross-camera search where the vehicles are queried in other camera images (i.e., not in the same camera images). As in the first query we can see that the first image is correct and then the second third and fourth are incorrect but have the same viewpoint with different color. Similarly, we can see that the algorithm tries to retrieve the same images having similar visual features (e.g., bushes in 2nd row images, taxi roof signs in the 3rd row images).

Table 4
Quantitative results achieved using experiments performed with rough number of clusters k. The test dataset set comprises of 10,000 images of VeRi dataset on cross-camera search with a total number of 143 distinct vehicles. Thus, the “optimal” value of k denotes 143 while the other two “optimal − 10” and “optimal + 10” selected values of k represent 133 and 153 respectively.

<table>
<thead>
<tr>
<th>Number of clusters k</th>
<th>Rank@1</th>
<th>Rank@5</th>
<th>Rank@10</th>
<th>Rank@20</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>optimal − 10</td>
<td>85.67</td>
<td>92.61</td>
<td>94.38</td>
<td>96.55</td>
<td>50.23</td>
</tr>
<tr>
<td>optimal</td>
<td>87.23</td>
<td>94.56</td>
<td>96.82</td>
<td>97.34</td>
<td>53.11</td>
</tr>
<tr>
<td>optimal + 10</td>
<td>84.41</td>
<td>92.04</td>
<td>94.65</td>
<td>96.52</td>
<td>50.41</td>
</tr>
</tbody>
</table>

Moreover, we have also done initial experiments with the mean-shift clustering algorithm over the same dataset. The preliminary results have shown that in terms of accuracy, the performance of the progressive scheme with the mean-shift is in par with the k-means clustering algorithm (see the quantitative results depicted in Table 5) which indeed confirms the efficacy of the proposed scheme.

5. Discussion

The algorithm adopts a progressive unsupervised formulation to tackle the vehicle re-ID problem. Following are few design parameters that are worth to mention here:

Choice of base architecture. In this paper, the chosen base architecture is well known ResNet50 [35]. Other architectures may also be employed for this purpose. However, the reason behind its selection compared to other implementations of deep CNN for recogni-
tion is the trade-off between the accuracy and memory efficiency. For instance, architectures such as VGG [33] and Inception [34] are memory efficient, but their top accuracies are less compared to ResNet. DenseNet [50] on the other hand have higher accuracy but is not memory efficient, i.e., the number of parameters is in the order of 2 to 3 in comparison to ResNet50.

**Clustering algorithm.** To adopt the unsupervised progressive formulation, k-means algorithm has been adopted due to its suitability to the problem at hand. Specifically, when the number of training classes are known, it is straightforward to initialize the value of k equal to them (e.g., the number of distinct vehicles in our case). In case, such a prior information is not available or cannot be exploited, the applicability of k-means is somewhat limited. Use of other clustering algorithms, e.g., mean-shift [51] (as shown in the previous section) or unsupervised deep embedding [52] etc., may be more appropriate in formulating the progressive scheme for improved model training.

**Reliable selection.** The clustering algorithms may produce wrong/weak assignments as a result of varying illumination, appearance and size/density of vehicles. These wrong/weak assignments can potentially lead to bad optimum or oscillations while fine tuning the CNN architecture. We proposed to tackle this problem by incorporating color as prior heuristic. Although this practically improves the model convergence and accuracy but theoretically cannot handle a particular case e.g., when two different vehicles having same color gets clustered in a single group. To overcome this, a fine-grained classification networks with attribute based prediction model, e.g., [46], may be employed to reliably filter cluster assignments.

**Fine versus scratch training.** The use of base network architecture is to kick-off the process of self-pace progressive learning as it serves as the initializer to reduce the overall training times. In the proposed approach, the incorporation of strong prior (color) heuristic allows us to significantly reduce the training time of the progressive step (as discussed earlier). Thus, the use of such strong prior particularly applicable to the vehicles (i.e., usually the color, make and models of the vehicle stays the same), may allow removal of the base network (i.e., fully unsupervised training) which in other object re-ID scenarios without such strong prior (e.g., persons) may not be possible.

**Need of benchmarking datasets/evaluation techniques.** In general, to solve the vehicle re-ID problem, the benchmarking datasets are required that are both complete in terms of size as well as covering all the necessary re-ID information such as multi-view points, spatio-temporal, make/model and color information. The existing primary datasets, e.g., Vehicle ID and VeRi, partially contain such information and therefore the models trained using them may not be directly applicable in practical scenarios. Moreover, the developed vehicle re-ID techniques with limited available information limits the evaluation strategies e.g., without multi-viewpoints, cross camera searches cannot be performed and consequently the evaluation only reduces to image-to-image search while without spatio-temporal information, no sequential model could be ap. Furthermore, the cross camera search allows to compute the rank information and mAP due to the availability of ground truth but in case of image-to-image search this is not possible since no true/false negatives could be computed, e.g., in case the retrieved vehicle has same id as the query image (vehicle), it is taken as true positive but in the other case, it is taken as false positive and there is no way to compute the true/false negatives that are required in computing the standard mAP.

### 6. Conclusion and outlook

In this paper, we have presented an unsupervised deep learning based approach to solve the vehicle re-ID problem. The whole idea is to initialize and use a base model to transfer the learned information to a deep unsupervised vehicle re-ID scheme which is then trained over new (unlabelled/unseen) dataset in a progressive manner. The convergence of the learned algorithm is significantly improved by incorporating heuristic color information into the progressive framework. The approach is generic and has been the first attempt to tackle the vehicle re-id problem in an unsupervised manner. Both the qualitative and quantitative results show that our system out-performs existing state-of-the-art supervised techniques in rank-1 accuracy over two standard publicly available datasets VeRi and VehicleID using cross-camera and image-to-image search strategies. Although the achieved accuracy is high, there are several aspects to improve/extend the proposed approach which are discussed earlier. For instance, the module of unsupervised clustering and reliable selection may be merged by replacing k-means with an enhanced algorithm capable of incorporating the vehicular semantics for robust clustering. This would not only help in improved convergence but may also help to enable completely self-paced progressive learning without the aid of base network. Moreover, for feature extraction the proposed approach only relies on CNN network trained on single latent variable. However, the training on multiple latent variables can lead to extraction of strong features representing more distinctive and descriptive vehicle attributes. This will allow to particularly focus more on foreground (i.e., vehicle in our case) which would consequently help to reduce the biasness introduced in the results due to background similarity. In future, we aim to explore these possibilities to enhance the overall vehicle re-identification pipeline.
References


Dr. Muhammad Shahzad received B.E degree in electrical engineering from the National University of Sciences and Technology, Islamabad, Pakistan, M.Sc. degree in autonomous systems (robotics) from the Bonn Rhein Sieg University of Applied Sciences, Sankt Augustin, Germany, and his PhD degree on radar remote sensing & image analysis at the department of Signal Processing in Earth Observation (SiPEO), Technische Universität München (TUM), Munich, Germany in 2004, 2011 and 2016 respectively. His PhD topic was automatic 3-D reconstruction of objects from point clouds retrieved from spaceborne synthetic-aperture-radar (SAR) image stacks. During his PhD, he proposed several novel and sophisticated methodologies and achieved several “firsts” in the international community, e.g. the first facade models from interferometric SAR (InSAR) data, its first reconstruction of an entire city, the first 2D building footprint models from InSAR and the first reconstruction of even individual trees. His work was closely linked to the TerraSAR-X and TanDEM-X satellite missions, the biggest German Earth observation endeavors ever, with both scientific as well as commercial applications. Besides, Dr. Shahzad also worked as visiting research scholar at the Institute for Computer Graphics and Vision (ICG), Technical University of Graz, Austria and attended twice two weeks professional thermography training course at Infrared Training Center (ITC), North Billerica, Massachusetts, USA in 2005 and 2007. Since 2016, he is working as Assistant Professor at the School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Islamabad, Pakistan. His research interests include deep learning, object detection, image classification, processing both unstructured/structured 3D point clouds, optical RGBD data and very high-resolution radar images.

Dr. Muhammad Moazam Fraz completed his PhD (Computer Science) from Faculty of Science Engineering and Computing, Kingston University London in 2013. His research area is the application of machine learning/computer vision techniques for diagnostic retinal image analysis. His PhD thesis was nominated for IET excellence awards 2013. After completing his PhD, he has worked as post doc research fellow at Kingston University in collaboration with St Georges University of London and UK BioBank on development of an automated software tool for epidemiologists to quantify and measure retinal vessels morphology and size; determine the width ratio of arteries and veins as well as the vessel tortuosity index on very large datasets, to enable them to link systemic and cardiovascular disease to the retinal vessel characteristics. Dr. Fraz received his MS and BS degrees in Software Engineering in 2008 and 2003 respectively. He was the recipient of two Gold Medals for ‘Best Graduate Award’ and ‘Securing Top Position in the batch’. He has started his career in 2003 as a Software Development Engineer at Elixir Technologies Corporation, a California based software Company. He has served with Elixir until 2010 at various roles and capacities including software developer, development manager and program manager. He is a PMI (www.pmi.org) certified Project Management Professional (PMP). Since June 2014, he is working as Assistant Professor at School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Islamabad, Pakistan. Besides, he is also working as Rutherford Visiting Fellow at The Alan Turing Institute, United Kingdom.