Artificial Intelligence Evolution
http://ojs.wiserpub.com/index.php/AIE/

UNIVERSAL WISER PUBLISHER

Research Article

ALSA: Adaptive Low-light Correction and Self-Attention Module for Vehicle Re-Identification

Shao Liu[1*], **Sos S. Agaian**[2]

[1]Graduate Center, City University of New York, New York, NY, USA
[2]College of Staten Island and Graduate Center, City University of New York, New York, NY, USA
E-mail: usaliushao@gmail.com

Received: 23 April 2023; **Revised:** 7 June 2023; **Accepted:** 16 June 2023

Abstract: Multi-Camera Vehicle Re-identification and Tracking (MCVRT) is a challenging task that involves identifying and tracking vehicles across multiple camera views in a surveillance network. Multi-Target Multi-Camera Tracking (MTMCT) and vehicle Re-Identification (Re-ID) are two major technologies applied to MCVRT tasks. Variations in aspect ratio, occlusion, orientation, and lighting conditions make vehicle re-identification and multi-camera tracking challenging. While some existing methods address these problems, it remains a significant challenge in the field. Additionally, most Re-ID datasets only include images captured in well-lit environments, and the impact of dark images on the performance of existing models needs to be explored further. This paper presents a new approach to address the challenge of low-light images in vehicle re-identification and achieves state-of-the-art results on public datasets. Our approach is based on two key components: (i) an Adaptive Low-light correction and Self-Attention module (ALSA) for image pre-processing in Vehicle Re-ID networks, and (ii) a new loss function called Log Triplet Loss (LT-Loss). We evaluated the presented approach through computer simulations on the VeRi-776 dataset, and the results showed that our model achieved a Rank@1 accuracy of 98.99%, and also outperformed commonly used models on dark images. Our study highlights the importance of considering lighting conditions in vehicle re-identification and provides a new approach to address this challenge.

Keywords: vehicle re-identification, attention mechanism, image processing, deep learning

1. Introduction

The rapid rise and continuous escalation of the urbanization process in modern society has resulted in an exponential surge in the number and flow of vehicles. The implementation of all-around monitoring and identification of vehicles is imperative to guarantee the welfare and security of road traffic while also optimizing urban management efficiency. This requires integrating Multi-Camera Vehicle Re-identification and Tracking (MCVRT) technology. MCVRT involves Multi-Target Multi-Camera Tracking (MTMCT) and vehicle Re-Identification (Re-ID) technology. It facilitates the identification and tracking of cars across multiple camera views in a surveillance network, thereby enabling efficient automated monitoring and management of vehicles.

MCVRT technology is beneficial for traffic management since it can improve real-time traffic monitoring and

DOI: https://doi.org/10.37256/aie.4220232901

management by tracking vehicles through the road network. As a result, optimized traffic flow, reduced congestion, improved road safety, richer data for traffic signal timing, automatic monitoring of traffic flow, and traffic flow prediction and simulation can be achieved. Moreover, MCVRT enhances security monitoring by identifying and tracking stolen, suspicious, and dangerous vehicles, while also facilitating the timely detection and resolution of traffic accidents and violations. In summary, MCVRT technology offers smarter and more efficient solutions for public traffic analysis and urban traffic management that can be implemented in various fields, such as intelligent parking and driverless driving. MCVRT can generally be divided into three systems, a vehicle detection system in a single camera, a vehicle Re-Identification (Re-ID) system, and a multi-camera tracks matching system.

In recent years, many vehicle detection models have been proposed based on object detection methods such as Faster R-CNN [1], DeepSort [2], and YOLO series [3]. The multi-camera tracks matching system includes multiple cameras with overlapping and non-overlapping Fields of View (FOV) and relies on vehicle detection and re-identification. In overlapping camera settings, some researchers combine temporal information from GPS locations to match tracks from different cameras [4, 5]. Vehicle Re-Identification (Re-ID) is a crucial intermediate process of MCVRT and Intelligent Transportation Systems (ITS), which are vital components of a Smart City. Therefore, our research will focus on the vehicle re-identification system on non-overlapping cameras in MCVRT, and we will provide more details about it.

In recent times, vehicle Re-ID has utilized computer vision and image processing to identify and track vehicles across different cameras in a surveillance network. The primary objective of vehicle re-identification is to match the same vehicle as it moves from one camera's field of view to another within the network. Extracting features from images captured by the cameras and utilizing algorithms to match those characteristics across different cameras is the typical process of vehicle Re-ID. The features utilized for vehicle re-identification can include vehicle color, make, model, license plate number, along with other distinguishing identifiers. Different machine learning methods, such as deep learning, can be used by the algorithms to learn from the characteristics and improve the precision of vehicle Re-ID.

Nevertheless, vehicle Re-ID remains a challenging task due to its intra-class variability, whereby images of the same car captured in different environments might look different, and inter-class similarity, whereby two different cars might appear similar due to comparable models and manufacturing techniques. Figure 1 demonstrates these challenges in vehicle Re-ID.

(a) (b)

Figure 1. Demonstration of two main challenges in vehicle Re-ID. (a) Intra-class variance: two images are from the same car; (b) Inter-class similarity: two images are from two different cars. (All images are collected from VeRi-776 [6])

One of the significant challenges for Re-ID is recognizing images of the same vehicle captured from various angles and different lighting conditions, including low-light environments. Identifying vehicles captured in low-light environments is challenging, and this issue is exacerbated because the global vehicle survey [7] reveals that 22% of cars are black. Convolutional Neural Networks (CNNs) and transfer learning have been widely implemented in resolving Re-ID challenges. Moreover, CNNs are utilized in image enhancement, including low-light image correction [8]. Researchers [9] utilize pre-trained models on pooled big data to construct a robust model capable of handling intra-class and inter-class problems in different conditions. Nevertheless, applying image enhancement approaches to re-identification networks to complete the model for extreme conditions (low light) has received little attention. There are several challenges for this task, such as effectively identifying low-light images from one batch for correction (intra-class

variability). Since most pictures are taken in well-lit conditions, running image enhancement on all samples in a single batch may add noise to standard images, which could potentially impact the model's accuracy. Secondly, inter-class similarity necessitates precise and efficient evaluation metrics to optimize the model thoroughly. Additionally, there is a need to explore methods of integrating the enhancement network and harnessing other powerful modules to improve the overall performance of the model. To tackle the earlier issues, the following measures were taken:

• We proposed an easily implemented Adaptive Low-light correction and Self-Attention Module (ALSA) for image pre-processing in Vehicle Re-ID networks (Figure 2) and achieved SOTA results on public datasets.

• We have proposed a new loss function, namely Log Triplet Loss (LT-loss), and provided a simplified mathematical explanation to clarify its relationship with the triplet loss.

• We have demonstrated through computer simulations how the lighting environment affects the intra-class variability and the performance of the Re-ID model.

Our study aims to enhance the performance of existing models for challenging scenarios, such as low light conditions, by utilizing image enhancement methods. This paper is structured as follows: Section 1 presents the motivation and structure of this work. In Section 2, previous studies related to Vehicle Re-identification tasks, public datasets, and loss functions are discussed. Section 3 outlines our proposed methods. We present the results of our experiments in Section 4. Finally, in Section 5, we conclude this paper and provide directions for further research.

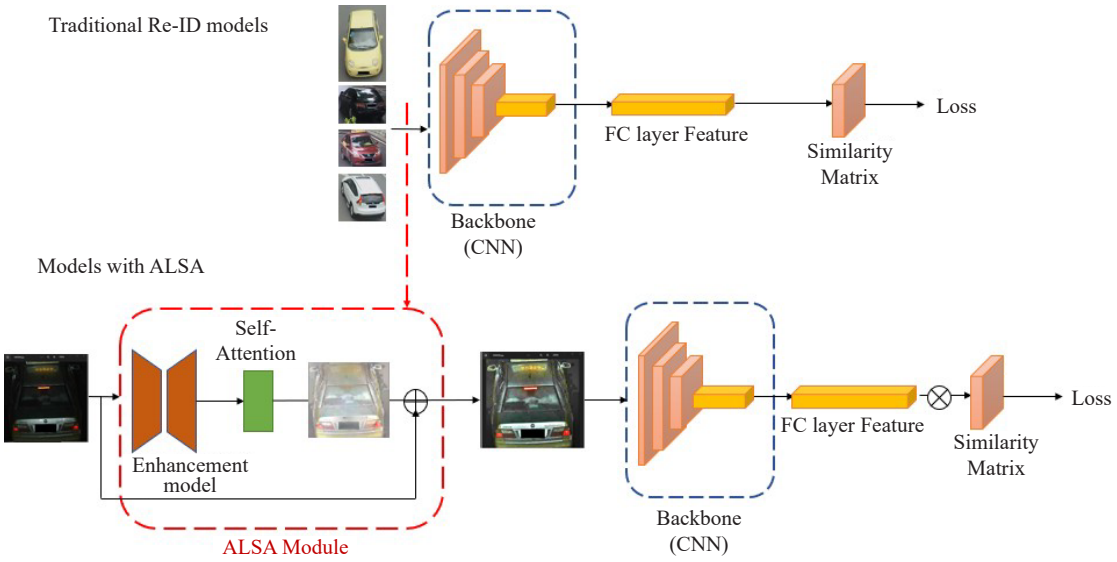


Figure 2. Overview of the architecture of the ALSA module. The red dash arrow shows the location where ALSA is added in a traditional Re-ID model. In the red dash rectangle, the ALSA module consists of a pre-trained enhancement model and self-attention blocks to generate corrected images for the following feature-extracting (CNN) model

2. Related works

2.1 *Public datasets*

The vehicle Re-ID task holds significant importance in the areas of surveillance, traffic monitoring, and law enforcement. To aid researchers in developing and evaluating algorithms for vehicle re-identification, several datasets have been created. Typically, these datasets consist of images or videos of vehicles captured by various cameras, along with vehicle identity annotations, such as unique IDs or license plate numbers. VeRi-776 [6], PKU VehicleID [10], Cars196 [11], CompCars [12], VRIC [13], and VERIWild [14] are some of the popular vehicle re-identification datasets mentioned in Table 1. These datasets provide valuable resources for algorithm development and evaluation. Comparing the performance of different algorithms on these datasets can help researchers advance the state of the art in this field.

Table 1. Popular datasets for Vehicle Re-ID

Dataset	Images	Vehicle models	Vehicles	Description
VeRi-776 [6]	49,357	10	776	Each captured vehicle images have 2 to 18 viewpoints with different resolutions, occlusion, and illumination.
PKU VehicleID [10]	221,763	250	26,267	All the images are captured during the daytime with multiple surveillance cameras. It has Small (800), Medium (1,600), and Large (2,400) test sets.
Cars196 [11]	16,185	-	196	The data is split into 8,144 training images and 8,041 testing images, where each class has been split roughly in a 50-50 split.
CompCars [12]	136,726	1,716	-	CompCars dataset consists of two types of image nature (1) Web-nature images and (2) Surveillance-nature images.
VRIC [13]	60,430	-	5,622	VRIC contains images captured day and night. Images with different angles, viewpoints, and occlusions.
VERIWild [14]	416,314	-	40,671	It consists of 174 cameras across, recorded for one month (30 × 24 h). VERIWild dataset contains images with viewpoint changes, illumination variations, occlusion, and background variations.

2.2 *Vehicle Re-ID methods*

Deep learning-based (CNN) models. Various popular deep learning-based feature representation models, including GoogLeNet [15], VGGNet [16], AlexNet [17], and ResNet [18], are commonly employed in vehicle Re-ID tasks. The authors of this paper also utilize these models as baselines in their vehicle Re-ID techniques. For example, Zakria et al. [19] proposed a deep learning-based approach for vehicle detection and classification using Convolutional Neural Network and license plate recognition. Additionally, Zhang et al. [9] used a large-scale dataset with diverse variations, including changes in lighting, viewpoints, and occlusion, to train a Convolutional Neural Network (ResNet) and introduced a hard example mining strategy during model training. Patel et al. [20] proposed the R@k-SML (Recall@k) loss function, which measures the number of relevant examples correctly identified among the top-k model predictions. They also proposed a similarity mixup technique to encourage feature learning in the model, promoting robustness against varying data input and noise.

Attention Mechanism-based models. Huynh et al. [21] proposed a modified approach to the backbone CNN model by stacking multi-heads using an attention mechanism to improve performance. He et al. [22] introduced TransReID, a transformer-based architecture that helps to extract unique features from object images that enable cross-camera object matching. In addition, the authors suggested a multi-level attention mechanism to enhance the visibility of the extracted features. Tan et al. [23] proposed the use of visual features such as color, texture, and shape for vehicle identification, alongside spatial-temporal features like trajectory and speed.

2.3 *Image enhancement methods*

Low-light image enhancement models aim to improve the visibility and quality of images captured in low-light conditions. There are many different approaches and techniques for enhancing low-light images. In recent years, deep learning-based methods have become increasingly popular for low-light image enhancement. These methods typically involve training a neural network on a dataset of low-light images and their corresponding ground-truth high-quality images and then using the trained model to enhance new low-light images. Some of the popular deep learning-based

low-light image enhancement models include Retinex [24], IAT [25], and ASV [8]. Jiang et al. proposed EnlightenGAN [26] based on a Generative Adversarial Network (GAN) and achieved significant results on public low-light datasets.

2.4 *Loss functions*

The Re-Identification (Re-ID) task aims to match identical objects and distinguish them from different objects, and efficient loss function families have been developed to address the problem, such as Contrastive Loss [27, 28] and Triplet Loss [29-33] (see Table 2). Triplet Loss, widely used and improved in recent years, was introduced in [29] and measures the distance difference between anchor-positive and anchor-negative samples to bring similar objects closer. In the Re-ID problem, the anchor-positive samples belong to the same class while anchor-negative samples belong to different classes. Given one anchor input X, $f(X)$ outputs the embedding vector, and $S_c(\,)$ calculates the cosine similarity between two embedding vectors. In Triplet Loss, we pick the hardest negative sample in negative samples (X_-) with the most significant similarity to the anchor in the mini-batch (Eq. 1) to calculate the hardest anchor-negative similarity (ag_h). Meanwhile, we pick the hardest positive sample in positive samples (X_+) with the lowest similarity to the anchor (Eq. 2) to calculate the hardest anchor-positive similarity (ap_h).

$$ag_h = max\left(S_c(f(X), f(X_-))\right) \tag{1}$$

$$ap_h = min\left(S_c(f(X), f(X_+))\right) \tag{2}$$

The triplet loss learns to make the ap_h larger than ag_h in order to minimize the similarity between X and X_- and maximize the similarity between X and X_+ with the following equation:

$$L_{triplet} = max\left(0, ag_h - ap_h\right) \tag{3}$$

However, this straightforward approach has several challenges. Firstly, only three samples are used in calculating the loss in the mini-batch, neglecting the global information. To overcome this, researchers propose mining negative samples and the hardest positive sample in the mini-batch to address the problem. Therefore,

$$L_{triplet} = \sum_j max(0, ag_j - ap_h) \tag{4}$$

In Eq. 4, ag_j refers to the similarity of anchor-negative pairs where j is the index. Another major drawback of the triplet loss is its instability during training, leading to slow convergence. [32] proposed optimizing over a smooth upper bound of the triplet loss to solve this problem,

$$L_{upper} = log\left(1 + \sum_j exp(0, ag_j - ap_h)\right) \tag{5}$$

We can use Eq. 6 which is the definition of cross entropy loss to comprehend Eq. 5 by defining ap_h as a positive class and all ag_j as negative classes. This approach enables us to differentiate and correctly classify every class.

$$L_{CE} = -log\left(\frac{exp(ap_h)}{exp(ap_h) + \sum_j exp(ag_j)}\right) \tag{6}$$

This methodology is only applicable when selecting one toughest positive example since the cross-entropy loss is restricted to one-label multi-class tasks. Consequently, the expression in Eq. 5 is mathematically interpretable only when utilizing one positive sample in a single batch, which means we miss the data on the other positive samples in the batch.

To include all the positive samples in the batch, [34] proposed the UniMoCon Loss. Then, Eq. 5 was changed to Eq. 7 by integrating multi-positive samples into the loss function.

$$L_{unicon} = log\left(1 + \sum_{j} exp(ag_j) \sum_{i} exp(-ap_i)\right) \tag{7}$$

In Eq. 7, i, j are indexes of positive samples (X_+) and negative samples (X_-) of the anchor sample (X). Integrating multiple positive samples into Eq. 7 without any theoretical support is questionable and not well demonstrated in [34]. As stated in the preceding paragraph, the precondition for Eq. 5 to hold is the presence of a single positive sample. Eq. 7 ignored this condition by integrating multiple positive samples, without any mathematical justification. To address this problem, We aim to propose a novel loss function that includes theoretical justification and can use multi-positive samples. We proposed a modification of the triplet loss definition (without temperature τ), resulting in Eq. 8, which simplifies Eq. 7 by removing the constant 1 from the *log* function with mathematical justification. More details will be explained in section 3.3.

$$L_{ours} = log\left(\sum_{j} exp(ag_j) \sum_{i} exp(-ap_i)\right) \tag{8}$$

Table 2. Review of Re-ID Loss function

Dataset	Purpose
Contrastive Loss [27]	calculates a contrastive loss function that aims to obtain a higher value for pairs of dissimilar objects and aims to obtain a lower value for pairs of a similar object
Supervised Contrastive Loss [28]	instead of using only one positive and one negative pair for each anchor, the SupCon considers many positive and negative pairs
Triplet Loss [29]	calculates the distance difference between anchor-positive samples and anchor-negative samples and aims to bring similar objects closer
Hierarchical triplet loss [30]	aims to collect informative samples and capture global data context with an online class-level tree update
Angular Loss [31]	focuses on limiting the angle in the negative sample of triplet triangles
N-Pair Loss [32]	aims to develop triplet loss focusing on pushing a positive sample away from multiple negative samples at each training stage
Multi-Similarity Loss [35]	aims to collect informative paired samples and weights these pairs as both their own and relative similarities
Recall@k Surrogate Loss [20]	aims to create a differentiable surrogate loss for the recall
Mixed Loss [36]	aims to feed multiple positive and negative samples to the neural network per time
Relation Preserving Triplet Mining [33]	a feature matching guided triplet mining scheme, that ensures triplets will respect the natural sub-groupings within an object ID
Part Loss [37]	aims to reduce empirical classification risks for training and representation learning risks for the test by dividing images into K parts
Clustering Loss [38]	aims for a new metric learning approach based on the structural prediction that takes the global structure of the embedding space into account by a clustering quality metric

3. Methods

In this section, we introduce our algorithm and neural network module Adaptive Low-light Self-Attention (ALSA) module. In section 3.1, we introduce the pipeline of ALSA. In section 3.2, The architecture of the model will be introduced. Last, the new combined loss function will be introduced in section 3.3.

3.1 *Overview of Adaptive Low-light Self-Attention (ALSA) module*

To mitigate issues with illumination and exposure that we introduced in the introduction section, we propose an image pre-processing module for Vehicle Re-ID networks. Leveraging a method introduced in [26], we modified the network to enhance the brightness of low-light images. However, running the low-light correction algorithm on every image in the Re-ID dataset risks adding noise to bright images and surplus calculations. To focus on images in low-light environments and ignore those well-lit images collected during the day, we introduce the Adaptive Low-light Self-Attention module (ALSA). Its main elements comprise a low-light image enhancement network, a Global Channel Attention (GCA) block, and a Spatial Self-Attention (SSA) block. Figure 3 illustrates how the enhancement network generates a raw enhanced image with noise, biased color, and incorrect luminance. We then proceed to apply a Spatial Self-Attention (SSA) block, which uses weights β (Eq. 10) to adjust the color and luminance of color channels (RGB). Finally, the GCA improves model efficiency and accuracy by disregarding bright images and focusing on dark ones with a parameter α (Eq. 9) based on the original input image.

The Global Channel Attention (GCA) mechanism generates an α value between 0 and 1 based on the global brightness information of the image. During the learning process, the module will generate $\alpha = 1$ for normal or bright images when the value of α is larger than a threshold scalar t while ignoring the enhanced images by multiplying by $(1 - \alpha)$ based on Eq. 11. On the other hand, if the input image is a low-light image, the generated α will increase the weight of the enhanced image in the final output. Generally, the darker the image is, the lower the α is, and more weight will be put on the enhanced output compared to the Input.

The Enhancement Network (EN) is collected from EnlightenGAN [26] which is a Generative Adversarial Network (GAN) for low-light image enhancement. Our team experimented with multiple methods such as [8] and [25] but found that EnlightenGAN outperformed the alternatives. The SSA block modifies the output of the EN using the color channel weights β, which is generated through the use of spatial self-attention layers. In Algorithm 3 line 1, each color channel (RGB) in X_{in} will be multiplied by β to be adjusted, and added by itself as the modified enhanced images $output_m$. The modified output is combined with the original image, using the equation described in Eq. 11 associated with α generated from Eq. 9 by GCA.

$$\alpha = Threshold(GCA(Input), t) \tag{9}$$

$$\beta = SSA(EN(Input)) \tag{10}$$

$$Output = \alpha * Input + (1 - \alpha) * (\beta * EN(Input) + EN(Input)) \tag{11}$$

3.2 *Architecture of ALSA*

Enhancement Network: We adapted the Enhanced Network (EN) from EnlightenGAN [26], which is a deep-learning model designed specifically for enhancing low-light images by automatically adjusting their brightness and contrast. The system's architecture is based on a Generative Adversarial Network (GAN), which comprises two neural networks: the generator and the discriminator. The responsibility of the generator network is to enhance low-light images, while the discriminator network distinguishes between genuine and generated images. EnlightenGAN's generator network includes convolutional layers that can learn to map input low-light images to produce high-quality, well-lit images. In addition to the convolutional layers, the generator network also employs residual blocks, which are used to extract and refine image features. The discriminator network is also designed to identify fake (generated) images through several convolutional layers. After extensive training, the generator network can produce images of outstanding quality which are indistinguishable from real images. For further details on the structures, please refer to [26].

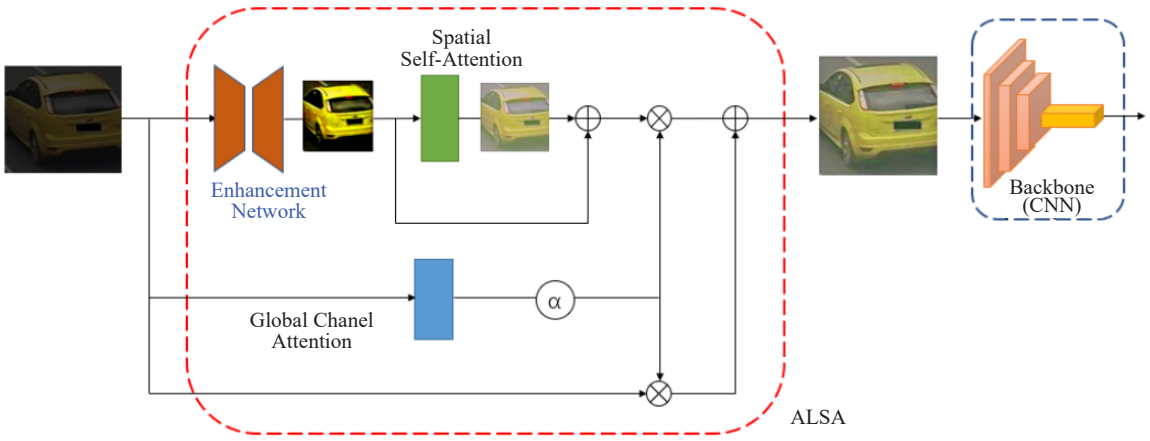


Figure 3. ALSA module

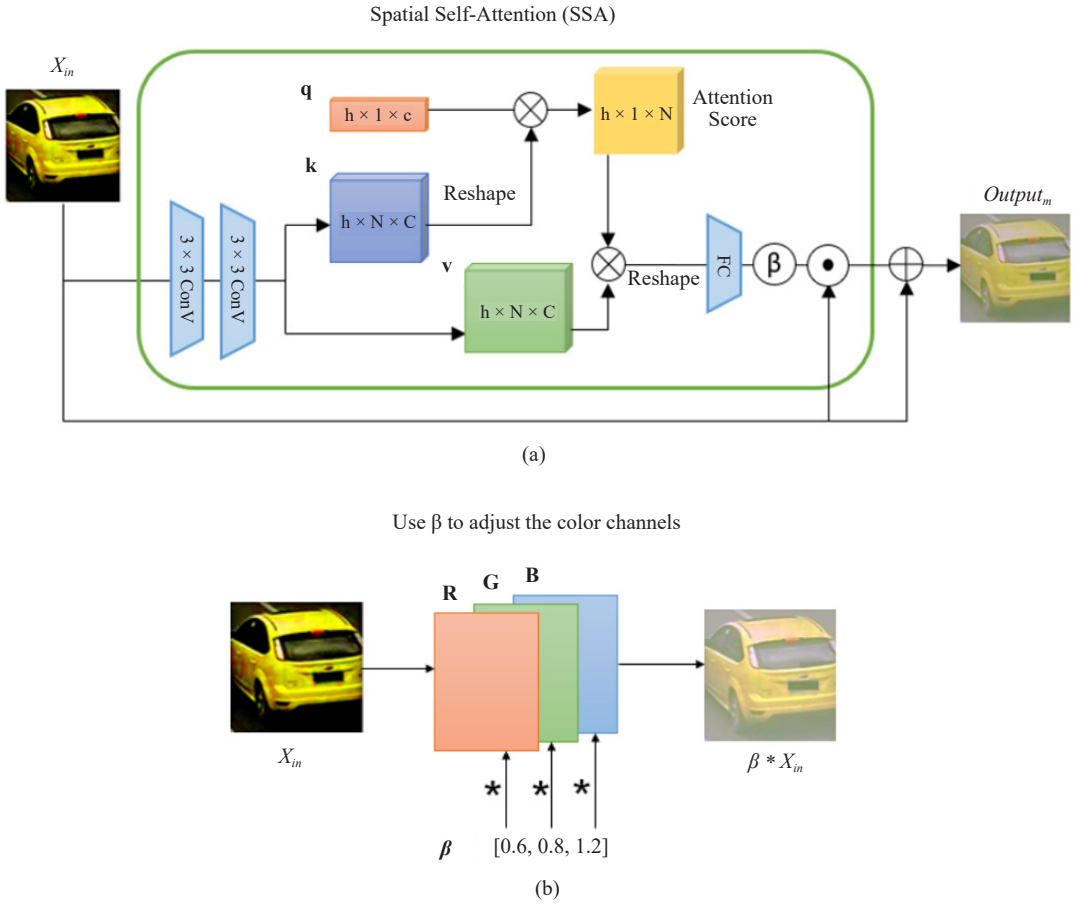


Figure 4. (a) The Spatial Self-Attention (SSA) block is a self-attention network similar to Transformer Mechanism. The dimension of batch size (B) is removed in the diagram for generalization. C is the number of channels and h is the number of multi-heads. q is the query matrix with B × h × 1 × C dimension. The dimension of K is B × h × N × C, where N is the flattened feature map of the encoded image. The matrix product of the attention matrix and v is B × h × 1 × C. Then, it is reshaped to B × 1 × (C · h) and passed to a fully connected layer, outputs β which is B × 3 × 1 × 1. (b) Each color channel (RGB) in X_{in} will be multiplied by β to be adjusted, and added by itself as the modified enhanced images $output_m$

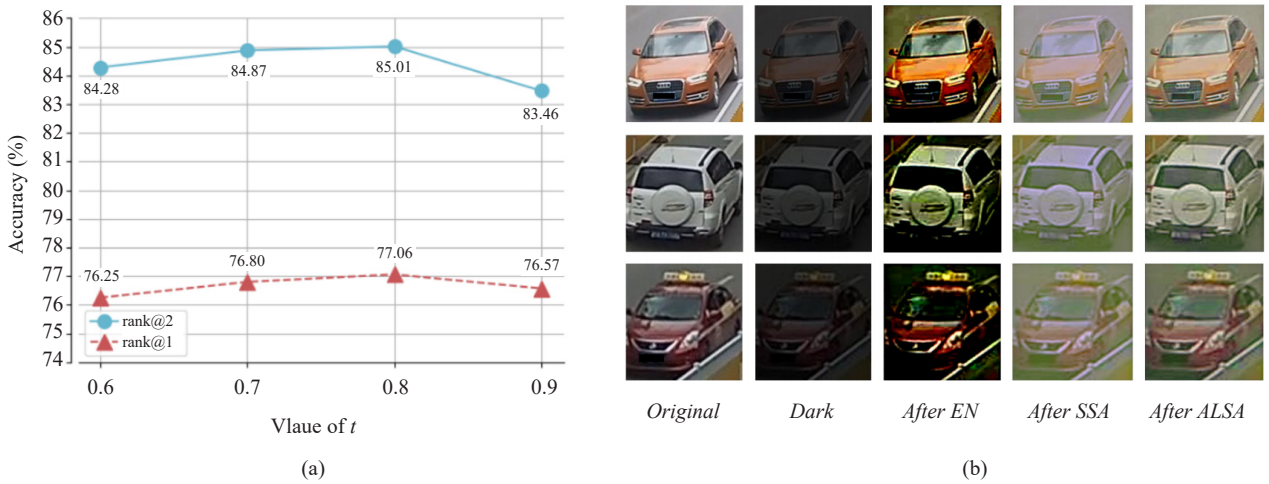
SSA: We introduce Spatial Self-Attention (SSA) to modify the color and brightness of the enhanced image and achieve natural and accurate images. We drew inspiration from several attention-based models including [8, 22], and

[25] for the design of queries that allow us to capture the global spatial relationship between pixels and generate matrix β for creating a new RGB image. Figure 4 (a) illustrates how the SSA structure incorporates two convolutional layers (the *embedding*( ) function in Algorithm 1) as an encoder to capture extensive information with high-dimension and global-level features. The encoded features generate Keys k and Values v in a transformer mechanism, with Queries q initialized as ones for generalization. Together with k, this generates an attention matrix, which is used to calculate the matrix product of the attention matrix and v. These are passed to a fully connected layer to generate a weight matrix β, which modifies the color channels of the enhanced image. (Algorithm 1) The SSA technique significantly reduces noise while improving image quality, as seen in Figure 4 (a) and Figure 5 (b).

Algorithm 1 Forward Algorithm of SSA
Require: mini-batch input X_{in}, X_{in} is the output of Enhancement Network ▷ B × 3 × W × H
Require: Query matrix q, initialized with value 1. ▷ B × h × 1 × C
1: X = *embedding*(X_{in}) ▷ B × C·h × W × H
2: X = *reshape*(X) ▷ B × N × C·h
3: # following codes (lines 4-7) use a typical self-attention mechanism to generate a feature map
4: k = *k*(X).*reshape*( ) ▷ B × h × N × C
5: v = *v*(X).*reshape*( ) ▷ B × h × N × C
6: Attention = *softmax*(q × k.*reshape*(B, h, C, N)) ▷ B × h × 1 × N
7: X = (Attention × v).*reshape*(B, 1, C·h) ▷ B × 1 × C·h
8: # following codes pass the feature map X to a fully connected layer *Linear*( ) and the output will be 3 scalers (β). Each of them will be used to adjust the RGB color channel
9: β = *Linear*(X).*reshape*(B, 3, 1, 1) ▷ B × 3 × 1 × 1
10: **return** β

GCA: Inspired by CBAM [39], the global channel attention block (GCA) contains two pooling layers: the average pooling layer and the maximum pooling layer. Each pooling layer performs separate calculations on the color channels. We take an average of the pooling layer outputs and pass this through the threshold function. The threshold function (*threshold*( ) in Algorithm 2) we use in the algorithm is a built-in threshold function of Pytorch [40], which uses a scalar t between 0 and 1 to determine the α value. Values in α that are greater than t are replaced with 1, whereas all other values remain the same. In Algorithm 3 line 2, if α is 1, the output will be the original image X_O, otherwise, the output will be a combination of X_O and adjusted enhanced image $output_m$ (Figure 6). In our experiments, we selected the threshold scalar to be 0.8, which gave us the best results. Further details on experimental results can be found in section 4.

Algorithm 2 Forward Algorithm of GCA
Require: mini-batch input X_O, X_O is the original input from dataset ▷ B × 3 × W × H
Require: The threshold t initialized as 0.8.
1: avg = *avgpooling*(X_O) ▷ B × 3 × 1 × 1
2: max = *maxpooling*(X_O) ▷ B × 3 × 1 × 1
3: # *sigmoid*( ) helps to keep α in range [0, 1]
4: α = *sigmoid*(avg + max) ▷ B × 3 × 1 × 1
5: # *threshold*( ) function keep the values that are less than t and change values to 1 if they are larger than t in α
6: α = *threshold*(α, t) ▷ B×3×1×1
7: **return** α



(a) (b)

Figure 5. (a) Different t values and models' accuracy; (b) Visualized images after different methods in ALSA (All images are collected from VeRi-776 [6])

Algorithm 3 Final stage of ALSA

Require: mini-batch of original input X_O, enhanced input X_{in} ▷ B × 3 × W × H

Require: The α generated by GCA ▷ B × 3 × 1 × 1

Require: The β generated by SSA ▷ B × 3 × 1 × 1

1: $output_m = \beta \cdot X_{in} + X_{in}$ ▷ B × 3 × W × H

2: output $= \alpha \cdot X_O + (1 - \alpha) \cdot output_m$ ▷ B × 3 × W × H

3: **return** output

3.3 *Log-triplet loss function*

Our Loss Function is defined as Eq. 8 and the one with temperature τ is as follows:

$$L_{ours} = \tau \cdot log\left(\sum_j exp\left(\frac{ag_j}{\tau}\right)\sum_i exp\left(\frac{-ap_i}{\tau}\right)\right) \tag{12}$$

Our loss function is generated from the definition of traditional triplet loss Eq. 3. The similarities (ap and ag) are calculated by the cosine distance between embedding vectors of the anchor sample and vectors of the positive and negative samples.

$$log\left(\sum_i exp(X_i)\right) > \approx max(X_i)$$

The hardest anchor-positive similarity ap_h can be regarded as $-max(-ap_i)$ and the hardest anchor-negative similarity ag_h is equal to $max(ag_j)$ where i and j denote the index of positive and negative samples in one mini-batch. Based on Eq. 13, the triplet loss (Eq. 3) can be transferred into:

$$L_{log-triplet} = log\left(\sum_j exp\left(ag_j\right) - \left(-log\sum_i exp\left(-ap_i\right)\right)\right) \tag{13}$$

$$= log\left(\sum_j exp\left(ag_j\right) + log \sum_i exp\left(-ap_i\right) \right)$$

$$= log\left(\sum_j exp\left(ag_j\right) \sum_i exp\left(-ap_i\right) \right) \tag{14}$$

This is the same as Eq. 8. There are three advantages of using $L_{log\text{-}triplet}$ compared to traditional modified triplet loss functions. First, we include all positive samples in a mini-batch to calculate the loss. Second, we simplified the UniMoCon [34] loss by removing the constant scalar 1 in *log* function, which makes it more concise and mathematically reasonable. Moreover, this loss function can be further modified by methods or parameters that benefit traditional triplet loss such as temperature τ and margin m [29] (Eq. 15).

$$L_{ours} = \tau \cdot log\left(\sum_j exp\left(\frac{ag_j}{\tau} \right) \sum_i exp\left(\frac{-ap_i}{\tau} \right) \right) + m \tag{15}$$

The value τ serves to increase ag_j and ap_i. Making use of Eq. 13, it can be deduced that the left side tends to get closer to the right-hand side in the equation as the value of $max(X_i)$ increases. Eq. 15 utilizes the value of τ outside the equation to effectively scale back the value to the right range. To prevent the triplet loss from dropping to zero when anchor-negative similarity is the same as anchor-positive similarity, margin: M, a small value is introduced. In triplet loss, the margin is leveraged to specify the minimum similarity between the anchor sample and the positive sample. The positive sample ought to be more proximate to the anchor than any negative sample. Additionally, the maximum similarity between the anchor sample and all negative samples is defined using the margin m. The introduction of this value, thereby enforcing a significant margin in the embedding space, contributes to enhancing the embedding model's proficiency towards distinguishing positive and negative samples more efficiently.

4. Experiments

In this section, we evaluate the proposed ALSA module and Log Triplet Loss (LT-Loss) through several experiments. First, implementation details are explained in Section 4.1. Second, we introduce three public datasets and the modified darker datasets in Section 4.2. Details of the effectiveness of our loss function and different components of our model will be discussed in sections 4.3 and 4.4. Then we demonstrate an analysis to describe the impact of dark images in datasets and the effectiveness of our method in section 4.5. In the last section 4.6, we evaluate the proposed methods on two benchmarks and compare the obtained results with the state-of-the-art methods, and discussed the improvements of our methods.

4.1 *Implementation details*

Major Architecture. This study employs two backbones for generating image embeddings of inputs. The first is Resnet50 [18], which is pre-trained on ImageNet [41]. Similar to previous works such as [20] and [42], we apply Generalized Mean Pooling [43] and Layer Normalization [44] following the fully connected layer of the backbone. To reduce the dimensionality of the features for inference to 512, we attach another fully connected layer. The second backbone we evaluate is VehicleNet [9], which is pre-trained on four publicly available vehicle re-identification datasets. We utilize the weights of the pre-trained VehicleNet as the initial weights for the image embedding network. The structure of the enhancement network is obtained from EnlightenGAN [26]. We have set the GCA Threshold to 0.8. In the SSA component, the medium channels are configured as 64 with 4 heads. The query inputs in the attention mechanism constitute a matrix of ones.

Training hyper-parameters. Adam optimizer [45] is used in all training processes. This paper follows the standard

class balanced sampling [6, 35, 56] with 4 samples per class for all the datasets. For VeRi-776 [6] and VehicleID [10], the batch size for training is set to 48. Cars196 [11] is set to 64. For each batch, a similarity matrix will be calculated. Each sample in the batch will have 3 positive pairs and the others are negative samples. The learning rate has been set to 0.0001, the Learning rate decays after the 10, 20, and 30 epochs, the decay size is set to 0.3, and the total number of epochs to 40. Training is conducted on the entire training set and evaluated on the test set split by each dataset.

Evaluation Metrics. To validate the effectiveness of our proposed method, we adopt Cumulative Matching Characteristics (CMC) accuracy as the previous work does. In Re-ID tasks, CMC@K is often adopted, where K represents the hit accuracy of the top K positions. CMC@k accuracy is a metric used to evaluate the performance of a ranking model in a retrieval task, where the goal is to retrieve the correct item from a set of possible items. CMC@K is defined as:

$$CMC@k = \frac{\sum_{q=1}^{Q} gt(q,k)}{Q}, \tag{16}$$

where q refers to the qth queried image and Q refers to the total number of query images. When the correct image is in the top-K of the predicted rank list, $gt(q, k) = 1$, and otherwise, $gt(q, k) = 0$.

4.2 *Datasets*

VeRi-776 [6] is a public benchmark dataset widely used for evaluating vehicle re-identification models. The dataset comprises 49,357 images of 776 unique vehicles captured by 20 cameras over a three-month span on a campus. In addition to detailed information regarding the make, model, and color of the vehicles, the dataset provides spatial-temporal information, such as the distance between various cameras. Specifically, the dataset includes 37,781 images of 576 identities for training, while the remaining 11,579 images of 200 vehicles are set aside for testing purposes.

VehicleID [10] is one of the most popular datasets in vehicle Re-ID tasks. It is collected by the National Engineering Laboratory for Video Technology (NELVT) of Peking University in 2016. All the images are captured from multiple angles and under different lighting and environmental conditions by a combination of fixed and moving cameras, including surveillance cameras and cameras mounted on drones. Also known as the PKU VehicleID dataset, it contains a total of 221,763 images from 26,267 vehicles. VehicleID includes three test subsets: VehicleID-800, VehicleID-1600, and VehicleID-2400. VehicleID-800 has 6,493 images from 800 vehicles, VehicleID-1600 has 13,777 images from 1,600 vehicles, and VehicleID-2400 has 19,777 images from 2,400 vehicles. All of our experiments are tested on VehicleID-800 testing set for simplicity.

Cars196 [11] is a collection of 16,185 images of cars belonging to 196 different classes, created by Stanford University researchers. The data is split into 8,144 training images and 8,041 testing images. The dataset was designed for use in fine-grained image recognition tasks, which require distinguishing between highly similar object categories. The images were collected from online sources and covered a wide range of car models.

Darker version of data. To test the performance of different models on darker images and how it can impact the results of existing models, we modified the VeRi-776 [6] and VehicleID [10] by randomly picking 10% of the images from both training and testing sets and multiplying the pixels by 0.3 to darken the picked images (Figure 7).

4.3 *Effectiveness of loss function*

We evaluated the effectiveness of our Log-Triplet Loss (LT-loss) function by comparing its results to those of other loss functions on the VehicleID dataset. To ensure a fair comparison, we used VehicleNet [9] as our baseline model and applied ALSA while following the settings outlined in Section 4.1 of our paper. The loss functions chosen for comparison were Recall@k [20], SupCon [28], Unimoco [34], and Triplet [29]. In our experiments, we used the LT-loss function modified with τ and margin (Eq. 15), with τ set to 0.07 and margin set to 0.1 as determined by our prior experiments. Additionally, we assigned a weight of either 1 or 0 to each loss based on whether it was used in the combination.

Our results, presented in Table 3, demonstrate the various combinations and utility of loss functions. When

tested on VehicleID [9] using a single loss function, the performance of our LT-Loss exhibited results similar to the best Recall@k [20] (96.77 on Rank@1 accuracy) and outperformed the other loss functions with 96.52 on Rank@1 accuracy. Notably, the combination of LT-Loss and Recall@k loss [20] achieved the best scores in both Rank@1 and Rank@5 accuracy. We attribute this success to the fact that our loss function incorporates all positive samples in a batch, allowing for the optimization of the model based on global information from that batch.

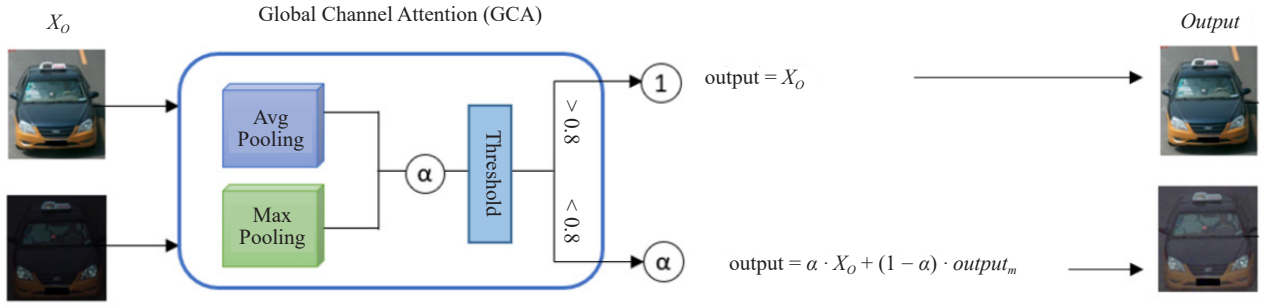


Figure 6. GCA structure. Average pooling is a pooling layer that outputs the average value for each color channel whereas max pooling is to output the maximum value of each channel

Table 3. Log-triplet loss function comparison on the VehicleID dataset

Loss					Dataset	
Recall@k [20]	SupCon [28]	Unimocon [34]	Triplet [29]	Modified Triplet (Ours)	VehicleID [10]	
Weight	Weight	Weight	Weight	Weight	Rank@1	Rank@5
1	0	0	0	0	96.77	98.46
0	1	0	0	0	96.00	98.04
0	0	1	0	0	96.23	98.18
0	0	0	1	0	94.45	96.74
0	0	0	0	1	96.52	98.43
1	1	0	0	0	96.1	98.24
1	0	1	0	0	96.35	98.41
1	0	0	1	0	95.52	97.84
1	0	0	0	1	**96.8**	**98.49**

4.4 *Ablation studies*

Our proposed method, ALSA, is composed of three parts: the Enhancement Network (EN), the Global Channel Attention (GCA) block, and the Spatial Self-Attention (SSA) block. The EN serves as the low-light enhancement network, and we tested three pre-trained low-light correction networks. To assess the effectiveness of each branch in our proposed method, we conducted several ablation experiments on the Cars196 dataset. In these experiments, we employed the LT-loss function and the same hyper-parameters as Section 4.1. Our baseline model was VehicleNet [9], which shares the ResNet50 [18] structure and was pre-trained on a combination of various vehicle Re-ID datasets. We

evaluated the performance of our method on both the VehicleID [10] and Cars196 [11] datasets.

Effectiveness of Enhancement Network (EN): The results in Table 4 show the effectiveness of different EN methods in our proposed ALSA. We can see that adding image enhancement networks before image embedding networks can improve the performance of the model. However, the improvement is very small (around 0.1%). This proves our assumption that the traditional enhancement method would bring noise and unbalanced color and luminance to the normal and bright images. Although low-light images would be corrected, as a trade-off, the accuracy of normal and bright images would be decreased. Among the three methods, EnlightenGAN [26] gives us the best results. If not specialized, EnlightenGAN will be the EN model we use in ALSA. We also tested EnlightenGAN on dataset cars196. (Table 5) The problem is more obvious in cars196. After applying EN, the rank@1 accuracy stays the same. This problem can be solved by our GCA block. Another observation in our experiments is that freezing the weights of pre-trained EN or not has little impact on the performance of the model (less than 0.5% in most cases).

Table 4. Evaluation of different EN methods on VehicleID dataset

Methods	Freeze-EN	VehicleID	
		Rank@1	Rank@5
VehicleNet [9]	N/A	96.37	98.06
+ASV [8]	Y	96.44	98.01
	N	96.41	98.23
+IAT [25]	Y	96.32	97.92
	N	95.83	97.88
+EnlightenGAN [26]	Y	**96.52**	**98.43**
	N	96.41	98.23

Effectiveness of Global Channel Attention (GCA): As we noted in the preceding paragraph, simply applying the Enhancement Network (EN) results in only marginal accuracy improvements. However, when used in conjunction with GCA ($t = 0.8$), normal and bright images ($\alpha > 0.8$) are left unmodified by the EN, while the quality of dark images ($\alpha < 0.8$) is enhanced. By selectively targeting dark images for enhancement, we achieved a 1% increase in rank@1 accuracy after applying GCA.

Effectiveness of Spatial Self-Attention (SSA): To evaluate our proposed method SSA against other denoising and white balance techniques applied to enhanced images, we tested the attention-based method CBAM [39] and the CNN-based method Deep Retinex [24] and compared their performance. In all tests, the image was first enhanced using our EN and GCA methods. We employed VehicleID for our experiments and recorded the results in Table 6. The data shows that attention-based methods outperformed CNN-based methods, where Deep Retinex yielded no improvement and potentially introduced more noise or low-quality images, as evidenced by the 2% drop in accuracy after applying it to the baseline model. Conversely, SSA improved model performance by more than 4% on the Cars196 dataset, as shown in Table 5. Our method captures spatial information to adjust color channels and effectively enhances the quality of the enhanced image, ultimately resulting in the best performance.

Impact of Threshold *t* for GCA: To determine the optimal value for threshold t in Eq. 9, we tested a range of values from 0.6 to 0.9 in increments of 0.1. The results, summarized in Table 7, show that the addition of GCA consistently improves the performance of the model, regardless of the value of t. However, when t is set to 0.8, the model achieves its best results, while setting t to 0.9 decreases performance (see Figure 5 (a)). These findings suggest that excluding a certain percentage of normal images and exclusively enhancing dark images can enhance the overall performance of the model.

Table 5. Effectiveness of methods in ALSA on Cars196

Methods	Freeze-EN	Cars196	
		Rank@1	Rank@2
VehicleNet [9]	N/A	76.10	84.11
+ Enhancement Network (EN)	N	76.12	84.16
+ EN + GCA	N	77.06	85.01
+ EN + GCA + SSA (ALSA)	N	**81.16**	**87.76**
+ EN + GCA + SSA (ALSA)	Y	80.43	87.14

Table 6. Evaluation of different attention methods on VehicleID dataset

Methods	Freeze-EN	VehicleID	
		Rank@1	Rank@5
VehicleNet [9]	N/A	96.37	98.06
+CBAM [39]	Y	96.45	98.11
+Retinex [24]	Y	94.52	96.76
+SSA (**Ours**)	Y	**96.52**	**98.43**

Table 7. Evaluation of different t values in GCA on Cars196 dataset

Methods	t	Cars196	
		Rank@1	Rank@2
Resnet50	N/A	76.10	84.11
+ Enhancement Network (EN)	N/A	76.12	84.16
+ EN + GCA	0.9	76.57	83.46
+ EN + GCA	**0.8**	**77.06**	**85.01**
+ EN + GCA	0.7	76.80	84.87
+ EN + GCA	0.6	76.25	84.28

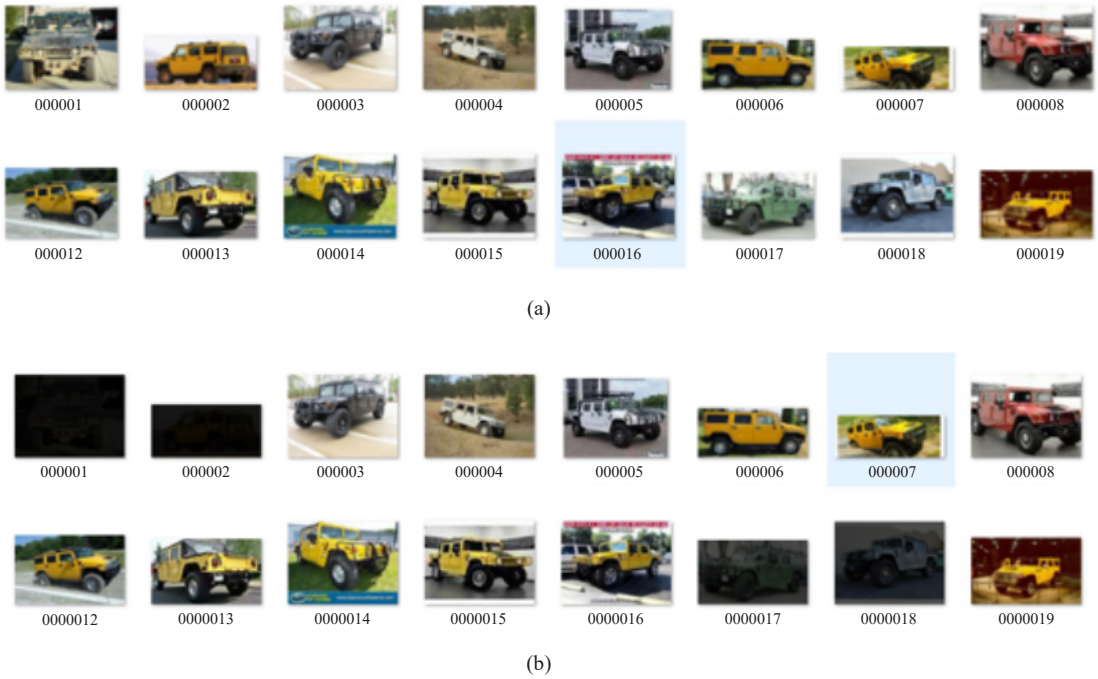


Figure 7. Demonstration of two versions of the dataset. (a) Original dataset; (b) Darkened dataset. (All images are collected from cars196 [11])

4.5 *Evaluations and analysis*

Finally, we select the optimal settings and parameters for our method and perform an additional experiment. Table 5 demonstrates that the addition of EN and GCA increases the model's performance by approximately 1%. Adding SSA to modify the enhanced image for denoising and balancing the luminance further improves the accuracy by over 3%, confirming the effectiveness of SSA. ALSA improves the baseline accuracy by over 4%. This demonstrates the effectiveness of our method. After undergoing EN and SSA, the final output image in the last column restores the original image in high quality, as shown in Figure 5 (b).

Impact of Darker Dataset. Table 8 shows that dark images significantly impact the performance of existing Re-ID methods. For example, when applied to the darker dataset, the performance of [20] on VeRi-776 [6] decreases by 0.71%. A similar decrease can be observed on Cars196 [11] darker version, which drops by 1.01% and 1.12% on rank@1 and rank@2 accuracies, respectively, due to the similarity of vehicle categories. Nevertheless, our proposed method demonstrates its effectiveness and robustness on dark images. Specifically, adding ALSA to the model reduces the score of the darker dataset by less than 0.5% on VeRi-776 and by 0.7% on Cars196, which shows the efficiency of ALSA on dark images. Although our method (ALSA + L-Loss) is not as good as Recall@k [20], the performance of our model on the Original and Darker datasets is similar. Therefore, our model shows better robustness to brightness variation. Additionally, adding the ALSA module to [20] improves accuracy by 0.2% on the original and 1% on the darker dataset, demonstrating that the ALSA module has better performance on darker images that could be implemented in more realistic scenarios.

4.6 *Comparison with the state-of-the-art*

VeRi-776. Table 9 shows that our model achieved the best performance in terms of the rank@1 and rank@5 accuracies, with 98.39% and 99.23%, respectively, when trained with Resnet50. Our method surpasses the state-of-the-art accuracy by more than 1%. We also experimented with different combinations of existing models. By using [9] as the backbone model, our method further improves the accuracy to 98.99% on rank@1 accuracy. Additionally, our module improves the baseline Resnet50's performance by more than 10% and improved the baseline [9] by more than 2%.

Table 8. Comparison of the original dataset and darker dataset

Methods	Backbone	F-ALSA	VeRi-776				Cars196			
			Original		Darker		Original		Darker	
			Rank@1	Rank@5	Rank@1	Rank@5	Rank@1	Rank@2	Rank@1	Rank@2
Recall@k [20]	Res50	-	97.48	98.70	96.77 (**-0.71**)	98.23 (**-0.47**)	78.6	86.3	77.59 (**-1.01**)	85.18 (**-1.12**)
Ours (L-Loss + ALSA)	Res50	Y	98.39	99.23	98.03 (-0.36)	99.34 (+0.11)	76.03	84	76.03 (0)	83.95 (-0.05)
ALSA + [20]	Res50	Y	98.99	99.46	98.51 (-0.48)	99.4 (-0.06)	78.81	86.57	78.5 (-0.31)	86.1 (-0.47)

Table 9. Comparison with existing methods

Methods	Backbone	VeRi-776		VehicleID	
		Rank@1	Rank@5	Rank@1	Rank@5
Res50 [18]	-	88.3	-	84.0	-
EALV [46]	-	84.39	-	75.11	-
RAM [47]	-	88.60	-	75.20	-
MADSTR [48]	-	89.27	-	-	-
BS [49]	MobileNet [50]	90.23	-	78.80	-
PR [51]	-	94.30	-	78.40	-
CAL [27]	Res50	95.4	97.9	82.5	-
UMTS [52]	-	95.80	-	80.90	-
VehicleNet [9]	Res50	96.78	-	83.64	96.86
ASB [21]	Res101	97	-	-	-
TransReID [22]	ViT-B/16	97.1	-	85.2	97.5
Anet [53]	Res50	97.1	98.6	87.9	97.8
RPTM [33]	Res101	97.3	98.4	95.5	97.4
Recall@k [20]	Res50	97.48	98.70	95.7	97.9
SupCon [28]	[9]	96.83	98.25	96.0	98.04
Unimocon [34]	[9]	96.36	98.03	96.23	98.18
Ours (L-Loss + ALSA)	Res50	**98.39**	**99.23**	93.92	97.4
Ours	[9]	**98.99**	**99.52**	**96.52**	**98.43**

VehicleID. Using the ALSA module with freeze weights for the enhancement network, we achieved the best performance by training on the baseline VehicleNet [9] with 96.52% at rank@1 and 98.43% at rank@5. Although unable

to surpass the state-of-the-art accuracy (95.7%) when using Resnet50 as the backbone model, we still rank within the top 3 of all methods known to us. Furthermore, our method significantly improves the rank@1 accuracy of VehicleNet by more than 10% and improved Resnet50 by 10% on the VehicleID testing set.

In summary, our methods achieve the best results on both VehicleID and VeRi-776 datasets when trained on VehicleNet. Our loss function and ALSA module significantly improve the performance of the baseline by up to 10%. Our module improves existing methods by more than 1% on original datasets and by an even greater margin on darker datasets.

5. Conclusion

This study proposes several simple yet effective methods to improve the performance of Vehicle Re-Identification (Re-ID) networks. The fast and efficient Adaptive Low-light Correction Module (ALSA) is proposed for image pre-processing. The module improves the performance of pre-trained ResNet50 by up to 10% on the testing set of VeRi-776 and VehicleID, while also outperforming existing models on darker vehicle images. The Log Triplet Loss is introduced to utilize multiple positive and negative samples in one mini-batch, resulting in a fused model that achieves state-of-the-art results on the VeRi-776 testing set and VehicleID testing set with 98.99% and 96.52% Rank@1 accuracy respectively. The proposed model surpasses previous models by 1% to 2% on original datasets and achieves better results on darker datasets for VeRi-776 and Cars196. This study has demonstrated through computer simulations how the lighting environment affects the intra-class variability and the performance of the Re-ID model. Future research should focus on exploring the relationship between the performance of ALCM and different attention-based mechanisms. Additionally, the impact of the attention-based methods on testing the proposed model on additional datasets should be explored to improve the generalization and robustness of the proposed model.

Acknowledgments

This work was supported by the U.S. Department of Transportation, Federal Highway Administration (FHWA), under Contract 693JJ320C000023.

Conflict of interest

The authors declare no competing financial interest.

References

[1] Tan MX, Pang RM, Le QV. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Seattle, WA, USA: IEEE; 2020. p. 10778-10787.

[2] Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*. Beijing, China: IEEE; 2017. p. 3645-3649. Available from: doi: 10.1109/ICIP.2017.8296962.

[3] Redmon J, Farhadi A. *Yolov3: An incremental improvement*. arXiv [Preprint]. 2018. Available from: doi: 10.48550/arXiv.1804.02767.

[4] Hsu HM, Huang TW, Wang GA, Cai JR, Lei ZC, Hwang JN. Multi-camera tracking of vehicles based on deep features re-id and trajectory-based camera link models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. Long Beach, CA, USA: IEEE; 2019. p. 416-424.

[5] Li PL, Li GZ, Yan ZX, Li YZ, Lu MQ, Xu PF, et al. In *Spatio-temporal consistency and hierarchical matching for multi-target multi-camera vehicle tracking*. CVPR Workshops. Long Beach, CA, USA: IEEE; 2019. p. 222-230.

[6] Liu XC, Liu W, Mei T, Ma HD. Provid: Progressive and multimodal vehicle reidentification for large-scale urban surveillance. *IEEE Transactions on Multimedia*. 2018; 20(3): 645-658. Available from: doi: 10.1109/

TMM.2017.2751966.
[7] Blackley J. *Most popular car colors*. 2018. Available from: https://www.iseecars.com/most-popular-car-colors-study [Accessed 22nd July 2022].
[8] Jiang ZQ, Li HT, Liu LJ, Men AD, Wang HY. A switched view of retinex: Deep self-regularized low-light image enhancement. *Neurocomputing*. 2021; 454: 361-372. Available from: doi: 10.1016/j.neucom.2021.05.025.
[9] Zheng ZD, Ruan T, Wei YC, Yang Y, Mei T. Vehiclenet: Learning robust visual representation for vehicle re-identification. *IEEE Transactions on Multimedia*. 2020; 23: 2683-2693. Available from: doi: 10.1109/TMM.2020.3014488.
[10] Liu HY, Tian YH, Wang YW, Pang L, Huang TJ. Deep relative distance learning: Tell the difference between similar vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Las Vegas, NV, USA: IEEE; 2016. p. 2167-2175.
[11] Krause J, Stark M, Deng J, Li FF. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*. Sydney, Australia: IEEE; 2013. p. 554-561.
[12] Yang LJ, Luo P, Change LC, Tang XO. A large-scale car dataset for fine-grained categorization and verification. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Boston, MA, USA: IEEE; 2015. p. 3973-3981.
[13] Kanaci A, Zhu XT, Gong SG. Vehicle re-identification in context. In *Pattern Recognition - 40th German Conference, GCPR 2018, September 10-12, 2018, Proceedings*. Stuttgart, Germany: IEEE; 2019. p. 377-390.
[14] Bai Y, Liu J, Lou YH, Wang C, Duan LY. Disentangled feature learning network and a comprehensive benchmark for vehicle re-identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2021; 44(10): 6854-6871.
[15] Szegedy C, Liu W, Jia YQ, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Boston, MA, USA: IEEE; 2015. p. 1-9.
[16] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations*. San Diego, CA, USA: ICLR; 2015. p. 1-14.
[17] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*. 2017; 60(6): 84-90. Available from: doi: 10.1145/3065386.
[18] He KM, Zhang XY, Ren SQ, Sun J. *Deep residual learning for image recognition*. 2016; 770-778. Available from: doi: 10.1109/CVPR.2016.90.
[19] Zakria, Cai JY, Deng JH, Aftab MU, Khokhar MS, Kumar R. Efficient and deep vehicle re-identification using multi-level feature extraction. *Applied Sciences*. 2019; 9(7): 1291. Available from: doi: 10.3390/app9071291.
[20] Patel Y, Tolias G, Matas J. Recall@k surrogate loss with large batches and similarity mixup. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA: IEEE; 2022. p. 7492-7501.
[21] Huynh SV, Nguyen NH, Nguyen NT, Nguyen VTQ, Huynh C, Nguyen C. A strong baseline for vehicle re-identification. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Nashville, TN, USA: IEEE; 2021. p. 4142-4149.
[22] He ST, Luo H, Wang PC, Wang F, Li H, Jiang W. Transreid: Transformer-based object re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Montreal, QC, Canada: IEEE; 2021. p. 15013-15022.
[23] Tan X, Wang ZG, Jiang MY, Yang XP, Wang J, Gao Y, et al. Multi-camera vehicle tracking and re-identification based on visual and spatial-temporal features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. Long Beach, CA, USA: IEEE; 2019. p. 275-284.
[24] Wei C, Wang WJ, Yang WH, Liu JY. *Deep retinex decomposition for low-light enhancement*. arXiv [Preprint]. 2018. Available from: doi: 10.48550/arXiv.1808.04560.
[25] Cui ZT, Li KC, Gu L, Su S, Gao P, Jiang Z, et al. You only need 90k parameters to adapt light: a light weight transformer for image enhancement and exposure correction. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24*. British: BMVA Press; 2022. p. 1-16.
[26] Jiang YF, Gong XY, Liu D, Cheng Y, Fang C, Shen XH, et al. Enlightengan: Deep light enhancement without paired supervision. *IEEE Transactions on Image Processing*. 2021; 30: 2340-2349.
[27] Hadsell R, Chopra S, LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Las Vegas, NV, USA: IEEE; 2006. p. 1735-1742.
[28] Khosla P, Teterwak P, Wang C, Sarna A, Tian YL, Isola P, et al. Supervised contrastive learning. *Advances in*

Neural Information Processing Systems. 2020; 33: 18661-18673.
[29] Hoffer E, Ailon N. Deep metric learning using triplet network. In *Similarity-Based Pattern Recognition: Third International Workshop, SIMBAD 2015, Copenhagen, Denmark, October 12-14, 2015*. Proceedings 3. Switzerland: Springer, Cham; 2015. p. 84-92.
[30] Ge WF, Huang WL, Dong DK, Scott MR. Deep metric learning with hierarchical triplet loss. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Switzerland: Springer, Cham; 2018. p. 269-285.
[31] Wang J, Zhou F, Wen S, Liu X, Lin Y. Deep metric learning with angular loss. In *Proceedings of the IEEE International Conference on Computer Vision*. Venice, Italy: IEEE; 2017. p. 2593-2601.
[32] Sohn K. Improved deep metric learning with multi-class n-pair loss objective. In: Lee D, Sugiyama M, Luxburg U, Guyon I, Garnett R. (eds.) *Advances in Neural Information Processing Systems*. Curran Associates, Inc.; 2016. vol. 29. Available from: https://proceedings.neurips.cc/paper_files/paper/2016/file/6b180037abbebea991d8b1232f8a8ca9-Paper.pdf [Accessed 3rd November 2022].
[33] Ghosh A, Shanmugalingam K, Lin WY. *Relation preserving triplet mining for stabilizing the triplet loss in vehicle re-identification*. arXiv [Preprint]. 2022. Available from: doi: 10.48550/arXiv.2110.07933.
[34] ZG Dai, Cai BL, Chen JY. Unimoco: Unsupervised, semi-supervised and fully-supervised visual representation learning. In *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. Prague, Czech Republic: IEEE; 2022. p. 3099-3106.
[35] Wang X, Han XT, Huang WL, Dong DK, Scott MR. Multi-similarity loss with general pair weighting for deep metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Long Beach, CA, USA: IEEE; 2019. p. 5022-5030.
[36] Chen L, He YH. Dress fashionably: Learn fashion collocation with deep mixed-category metric learning. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2018; 32(1): 2103-2110.
[37] Yao HT, Zhang SL, Hong RC, Zhang YD, Xu CS, Tian Q. Deep representation learning with part loss for person re-identification. *IEEE Transactions on Image Processing*. 2019; 28(6): 2860-2871. Available from: doi: 10.1109/TIP.2019.2891888.
[38] Song HO, Jegelka S, Rathod V, Murphy K. *Learnable structured clustering framework for deep metric learning*. arXiv [Preprint]. 2016; 1(2): 8. Available from: doi: 10.48550/arXiv.1612.01213.
[39] Woo S, Park J, Lee JY, IS Kweon. CBAM: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Switzerland: Springer, Cham; 2018. p. 3-19.
[40] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc.; 2019. p. 8024-8035. Available from: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf [Accessed 5th July 2022].
[41] Deng J, Dong W, Socher R, Li LJ, Li K, Li FF. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Miami, FL, USA: IEEE; 2009. p. 248-255. Available from: doi: 10.1109/CVPR.2009.5206848.
[42] Teh EW, DeVries T, Taylor GW. ProxyNCA++: Revisiting and revitalizing proxy neighborhood component analysis. In: Vedaldi A, Bischof H, Brox T, Frahm JM. (eds.) In *Computer Vision - ECCV 2020. ECCV 2020*. Part of the Lecture Notes in Computer Science, vol. 12369. Switzerland: Springer, Cham; 2020. p. 448-464. Available from: doi: 10.1007/978-3-030-58586-0_27.
[43] Radenović F, Tolias G, Chum O. Fine-tuning cnn image retrieval with no human annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2019; 41(7): 1655-1668. Available from: doi: 10.1109/TPAMI.2018.2846566.
[44] Jimmy LB, Jamie RK, Geoffrey EH. *Layer normalization*. 2016. Available from: doi: 10.48550/arXiv.1607.06450.
[45] Kingma DP, Ba J. *Adam: A method for stochastic optimization*. 2015. arXiv [Preprint]. Available from: doi: 10.48550/arXiv.1412.6980.
[46] Lou YH, Bai Y, Liu J, Wang SQ, Duan LY. Embedding adversarial learning for vehicle re-identification. *IEEE Transactions on Image Processing*. 2019; 28(8): 3794-3807. Available from: doi: 10.1109/TIP.2019.2902112.
[47] Liu XB, Zhang SL, Huang QM, Gao W. Ram: A region-aware deep model for vehicle re-identification. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*. San Diego, CA, USA: IEEE; 2018. p. 1-6.
[48] Jiang N, Xu Y, Zhou Z, Wu W. Multi-attribute driven vehicle re-identification with spatial-temporal re-ranking. In *2018 25th IEEE international conference on image processing (ICIP)*. Budapest, Hungary: IEEE; 2018. p. 858-862.
[49] Kuma R, Weill E, Aghdasi F, Sriram P. Vehicle re-identification: an efficient baseline using triplet embedding. In *2019 International Joint Conference on Neural Networks (IJCNN)*. Budapest, Hungary: IEEE; 2019. p. 1-9.

[50] Howard AG, Zhu MG, Chen B, Kalenichenko D, Wang WJ, Weyand T, et al. *Mobilenets: Efficient convolutional neural networks for mobile vision applications*. arXiv [Preprint]. 2017. Available from: doi: 10.48550/arXiv.1704.04861.
[51] He B, Li J, Zhao YF, Tian YH. Part-regularized near-duplicate vehicle re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Long Beach, CA, USA: IEEE; 2019. p. 3997-4005.
[52] Jin X, Lan CL, Zeng WJ, Chen ZB. Uncertainty-aware multi-shot knowledge distillation for image-based object re-identification. *AAAI Conference on Artificial Intelligence*. 2020; 34(07): 11165-1172. Available from: doi: 10.1609/aaai.v34i07.6774.
[53] Quispe R, Lan CL, Zeng WJ, Pedrini H. Attributenet: Attribute enhanced vehicle re-identification. *Neurocomputing*. 2021; 465: 84-92. Available from: doi: 10.1016/j.neucom.2021.08.126.