1290

UNIVERSIDADE Đ COIMBRA

Timur Lomin

Transformers in Vehicle Re-Identification

Dissertação no âmbito do Mestrado em Engenharia Eletrotécnica e de Computadores, no ramo de Robótica, Controlo e Inteligência Artificial, orientada pelo Professor Doutor Jorge Manuel Moreira de Campos Pereira Batista e apresentada ao Departamento de Engenharia Eletrotécnica e de Computadores.

Setembro de 2022

· U C ·

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Transformers In Vehicle Re-Identification

Timur Lomin

Coimbra, Setembro de 2022

Transformers In Vehicle Re-Identification

Supervisor:

Prof. Dr. Jorge Manuel Moreira de Campos Pereira Batista

Jury:

Prof. Dr. João Pedro de Almeida Barreto

Prof. Dr. Paulo Jorge Carvalho Menezes

Prof. Dr. Jorge Manuel Moreira de Campos Pereira Batista

Dissertation submitted in partial fulfillment for the degree of Master of Science in Electrical and Computer Engineering.

Coimbra, Setembro de 2022

Agradecimentos

Em primeiro lugar, um obrigado especial e sincero ao meu orientador Professor Doutor Jorge Batista pela dedicação, orientação e conhecimentos partilhados que foram fundamentais para a concretização deste trabalho. Quero também agradecer a toda a equipa e colegas do laboratório pela constante disponibilidade, ajuda e partilha de ideias.

Um agradecimento muito especial a toda a minha família e também à minha namorada por todo o carinho, força e motivação ao longo de todo o meu percurso académico. Foi o vosso apoio incondicional que me deu forças para seguir em frente e a superar-me a mim próprio dia após dia. Sem o vosso apoio constante, nada disto teria sido possível. Estou eternamento grato.

Quero agradeçer também a todos os meus colegas de curso, pela amizade, pelo companheirismo e por todos os momentos vividos durante estes últimos 5 anos, que levo para sempre na minha memória.

Resta-me agradecer a todos os meus amigos de longa data e pessoas com as quais me cruzei ao longo da minha vida, que de certa forma tiveram influência na pessoa em que me tornei.

A todos, um muito obrigado!

Abstract

Vehicle Re-Identification (V-ReID) attempts to recognize the same vehicle across multiple cameras, and thus plays a major role in many applications such as video surveillance, criminal investigation, and traffic flows in Intelligent Transportation Systems (ITS).

Despite the substantial research that has been conducted in this field, V-ReID remains a difficult problem to solve due to large variations in appearance between intra-class pairs of vehicle images and small inter-class variability. In other words, the appearance of a vehicle varies significantly depending on different viewpoints, illumination changes, and other factors, yet there are different vehicle identities (IDs) that look extremely similar, increasing the difficulty of Re-Identification (ReID).

Convolution Neural Network (CNN)-based methods have shown impressive results so far, however, they suffer from detail information loss due to convolution and downsampling operators. Meanwhile, the Vision Transformer, a revolutionary architecture, has emerged and gained significant attention with outstanding results in image recognition problems as well as various ReID benchmarks. This novel architecture can extract more detailed information because it does not use convolution or downsampling operators.

Taking it all into account and based on the scaling success of Vision Transformers, the main purpose of this work is to evaluate Transformers ability to detect intrinsic characteristics of each vehicle that can distinguish it from others, i.e. a vehicle fingerprint based on appearance, in order to enhance V-ReID performance in well-known benchmarks.

Keywords: Vehicle Re-Identification, Transformer, Deep Learning

Resumo

A Re-Identificação de Veículos (V-ReID) consiste em reconhecer um veículo em diversas câmaras, desempenhando assim um papel importante em diversas aplicações, tais como videovigilância, investigação criminal e análise de fluxos de tráfego em Sistemas de Transporte Inteligentes (ITS).

Apesar da elevada investigação que tem sido conduzida neste campo, a V-ReID continua a ser um problema difícil de resolver devido às grandes variações de aparência entre pares intra-classe de imagens de veículos e à pequena variabilidade inter-classe. Por outras palavras, a aparência de um veículo varia significativamente com diferentes perspetivas, variações de iluminação, e outros factores, e além disso, diferentes veículos podem ser extremamente semelhantes, aumentando assim a dificuldade de Re-Identificação (ReID).

Os métodos baseados em Redes Neurais Convolucionais (CNNs) têm mostrado resultados impressionantes, no entanto, sofrem de perda de informação detalhada devido às convoluções e operadores de *downsampling*. Entretanto, surgiu uma nova arquitetura revolucionária designada de *Vision Transformer*, que recentemente ganhou muita atenção devido aos seus resultados surpreendentes em problemas de reconhecimento de imagem, bem como em várias *benchmarks* de ReID. Esta nova arquitectura pode extrair informação mais detalhada pois não utiliza convoluções nem opreadores de *downsampling*.

Tendo isto em conta e com base no crescente sucesso dos *Vision Transformers*, o principal objetivo deste trabalho é avaliar a capacidade dos *Transformers* para detetar características intrínsecas dos veículos que os possam distinguir entre eles, ou seja, uma espécie de impressão digital de cada veículo baseada na sua aparência, com o intuito de melhorar o desempenho da V-ReID em *benchmarks* conhecidas.

Palavras-chave: Re-Identificação de Veículos, Transformer, Aprendizagem Profunda

Contents

List of Acronyms

CNN	Convolutional Neural Network
ID	Identity
ISR	Institute of Systems and Robotics
ITS	Intelligent Transport Systems
mAP	Mean Average Precision
P-ReID	Person Re-Identification
ReID	Re-Identification
ViT	Vision Transformer
V-ReID	Vehicle Re-Identification

List of Figures

List of Tables

1 Introduction

1.1 Context and Motivation

Vehicle Re-Identification (V-ReID) attempts to recognize a specific vehicle across different cameras based on its appearance and has gained a lot of attention from the computer vision community in the past years.

Since the start of the twentieth century, the role of vehicles has become essential. They are used worldwide and have become the most popular mode of transport in many of the most developed countries. As a result, numerous applications where V-ReID plays a significant role have developed and are continually improving, including traffic flow in intelligent transportation systems (ITS), criminal investigation, and video surveillance.

Despite the extensive work developed in this area, there is still room for improvement in terms of robustness, as V-ReID remains a challenging problem due to large variations in appearance between intra-class pairs of vehicle images. These variations are caused by different viewpoints, illumination changes, occlusions, and noisy/blurred images, among others, as shown in figure 1.1. In addition, vehicles with different identities (IDs) can be of the same brand, model, and color, making it difficult to distinguish the visual differences between pairs of vehicle images, i.e. there is small inter-class variability. This issue is illustrated in Figure 1.2, where the samples in each row appear to be the same vehicle at first look, although they are not.

Most of the advancements in V-ReID have been accomplished using well-known Convolution Neural Network (CNN)-based methods, but recently Vision Transformer (ViT) that was introduced by Dosovitskiy et al. in 2020 has gained substantial traction in computer vision problems, including ReID. Different from CNNs that aggregate and transform features via local and dense convolutional kernels, Transformers treat an image as a serie of patch sequence and use the self-attention mechanism to directly model long-range dependencies of local patches (a.k.a. tokens), resulting in more detailed V-ReID features.

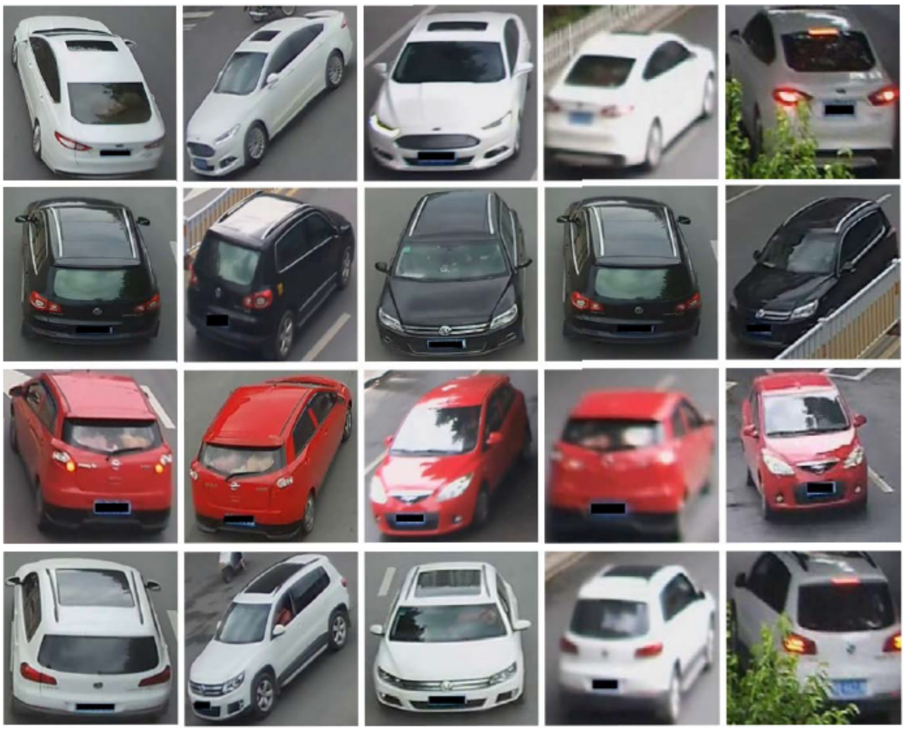


Figure 1.1: The problem of large variations between intra-class vehicle images. Each row denotes the same vehicle captured by cameras from different viewpoints. Taken from [1].

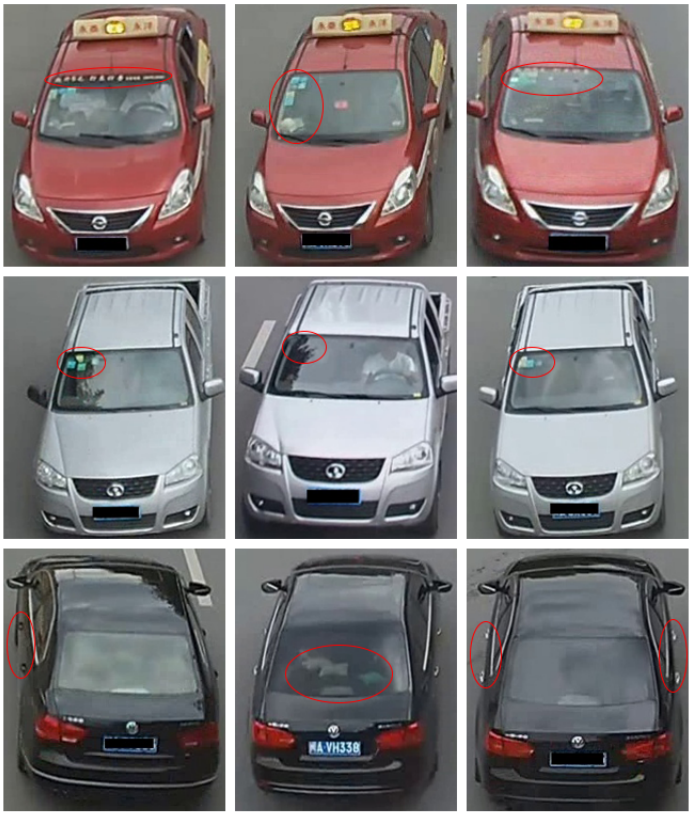


Figure 1.2: The problem of small variations between inter-class vehicle images. Each row denotes different vehicles with similar appearances.

1.2 Goals and Contributions

Given a query vehicle image taken from one camera, the objective of V-ReID is to identify the same vehicle from a large gallery set of images captured by other cameras within the surveillance system. To achieve this, it is required a sophisticated and efficient deep learning algorithm to generate the feature embedding of each image and then rank the similarity between query and gallery image vectors. This technology might be used to automatically locate a specific vehicle in a network of cameras and determine its route as illustrated in Figure 1.3.

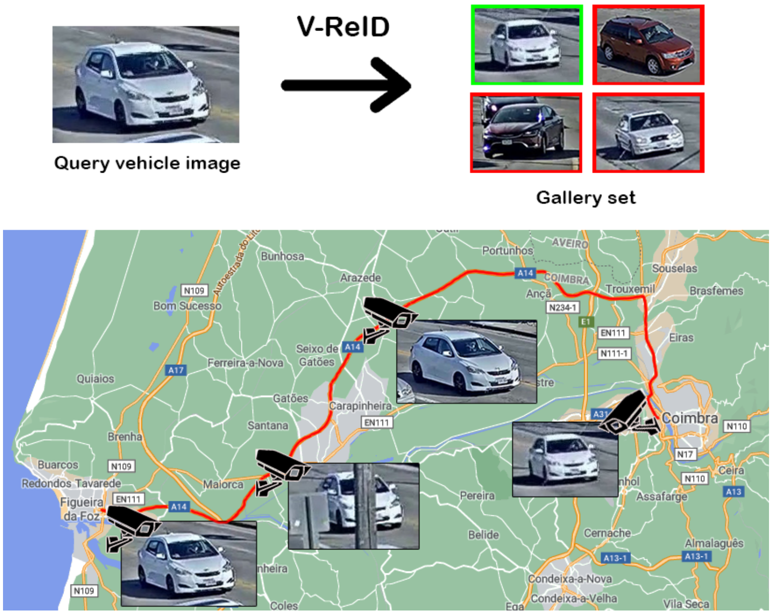


Figure 1.3: Simple illustration of V-ReID.

The main goal of this dissertation is to develop a robust V-ReID network based on the scaling success of Transformers to extract discriminative features of vehicles, which is crucial in V-ReID. The University of Coimbra's Institute of Systems and Robotics (ISR) has been working on many Research and Development (R&D) projects related to Brisa Highways, Portugal's largest private transport infrastructure company. One of the current projects in ISR is to identify inherent characteristics of each vehicle that can distinguish it from other identities, even if they are very similar (same brand, model, and color), i.e. to extract a vehicle fingerprint based on appearance without using vehicle registration plate information. Following that, one of the dissertation's intermediate tasks is to analyze and explore transformers' potential to extract strong features that can assign a unique identification to each vehicle.

1.3 Dissertation Structure

The course flow of this dissertation and content of each chapter are the following:

- **Chapter 2**: Review of the most relevant State-of-the-Art works on V-ReID and other types of methods, as well as the available datasets.
- **Chapter 3**: All of the approaches and developments that have been explored in this work are presented and clearly described.
- **Chapter 4**: Analysis of the experimental results obtained during the course of the work.
- **Chapter 5**: Conclusions are drawn, the possible future work is discussed.

2 State of Art

In this chapter, a survey of related works and the most important developments made in vehicle re-identification are presented.

2.1 Evolution of Re-Identification

Vehicle Re-Identification (V-ReID), as well as Person Re-Identification (P-ReID), have been widely researched and have made huge progress in the past 10 years, thanks to advances in deep learning and artificial intelligence. Although vehicle and person ReID appear to be the same problem, V-ReID is arguably more challenging due to high intra-class variability caused by the variety of shapes from different viewing angles, as well as small inter-class variability due to the limited shapes and colors of vehicle models made by different manufacturers. On the other hand, a person's appearance does not vary significantly as a result of different perspectives.

The traditional hand-crafted feature extractors such as LOMO and BOWCN were the dominant feature representation methods for ReID before the emergence of deep learning. Between 2012 and 2014, some well-known deep feature learning networks (AlexNet, VGGNet, and GoogLeNet) emerged, revolutionizing the entire concept of deep learning and significantly improving performance for image recognition tasks including V-ReID. Since then, many studies have been conducted using CNNs to determine the optimal strategy for extracting robust features, considering all of the V-ReID problems.

2.1.1 Convolutional Neural Networks

A Convolutional Neural Network (CNN) is a Deep Learning algorithm which can take an image as input, assign importance (learnable weights and biases) to various aspects in the image and be able to differentiate one from the other. CNNs are designed to automatically and adaptively learn spatial hierarchies of features through backpropagation by using multiple building blocks, such as convolution layers, pooling layers, and fully connected layers.

The most common CNN-based pipeline for ReID is to design appropriate loss functions to train a CNN backbone, which is used to extract features of vehicle images. In V-ReID, the cross-entropy loss and triplet loss are the most commonly employed metrics for optimal and faster convergence of training. Figure 2.1 is an illustration of this standard pipeline.

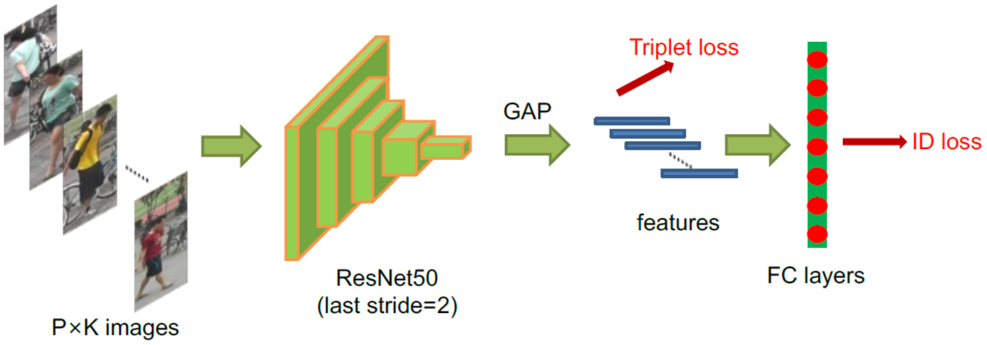


Figure 2.1: A common pipeline for object ReID. Taken from [2].

The Bag of Tricks and Strong Baseline for P-ReID (BoT) [2] purposed by Hao Luo et al. is an example of a CNN-based approach that uses a well-known CNN (Resnet-50) as the backbone. Unlike other works that design sophisticated network structures and concatenate multi-branch features, the main objective of this work was to design a simple yet effective pipeline for P-ReID by employing multiple tricks to optimize the training process.

The first trick mentioned in the article is Warmup Learning Rate. The Learning rate is a crucial parameter that highly impacts the performance of a network and the usual strategy is to use a constant base learning rate for the initial few steps. The Warmup Learning Rate prevents early over-fitting by setting the learning rate lower than the base learning rate and gradually increasing it to approach the base learning rate. Figure 2.2 shows the evolution of the learning rate with and without the Warmup Learning Rate.

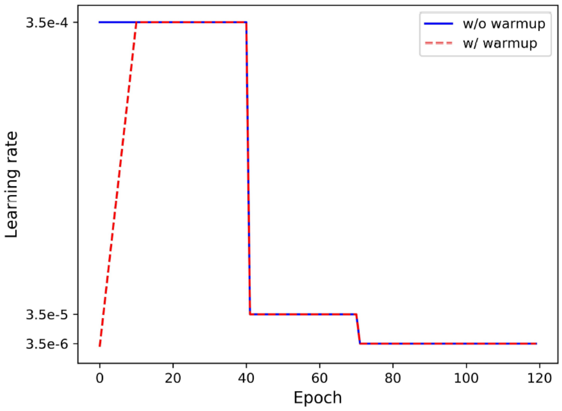


Figure 2.2: Comparison of learning rate schedules. Taken from [2].

Another trick used by BoT and many other works to overcome the occlusion problem and improve the generalization ability of ReID models is Random Erasing Augmentation. As the name implies, it is used to randomly erase a part of an image, typically a rectangular region of pixels.

Cross-Entropy loss optimizes the cosine distances for image pairs in the embedding space, whereas triplet loss optimizes the Euclidean distances. If these two losses are employed to optimize a feature vector at the same time, their goals may be inconsistent. One probable scenario during the training process is that one loss decreases while the other fluctuates or even increases. To address this issue, the authors created BNNeck, a structure that adds a batch normalization (BN) layer after features (and before classifier FC layers).

Since increasing spatial resolution always improves feature granularity, another trick used by BoT was removing the last spatial down-sampling operation in the backbone or reducing the stride size in the final layer, resulting in a feature map with higher spatial size, which enhanced performance.

BoT used the above-mentioned tricks, as well as others, to create a more efficient pipeline for P-ReID, as shown in Figure 2.3. Despite the fact that this method was developed specifically for P-ReID, the employed tricks are widely used in V-ReID works to enhance performance.

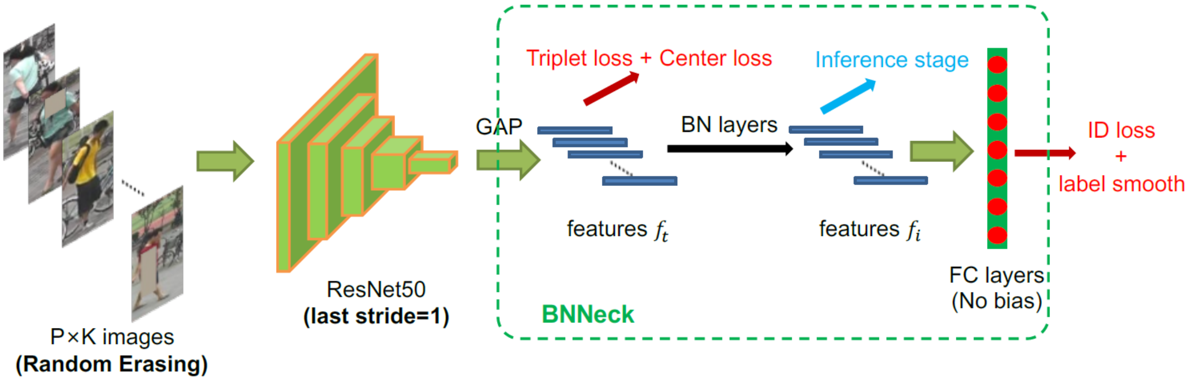


Figure 2.3: A more efficient pipeline proposed by BoT. Taken from [2].

2.1.2 Vision Transformers

One of the most important breakthroughs in Deep Learning research in the last decade is the attention mechanism. It has spawned the rise of so many recent developments in natural language processing (NLP) and Computer Vision, including the Transformer architecture which is the current state-of-the-art for V-ReID.

Attention mechanisms can be divided into several groups, however, in computer vision, the most popular type of attention is Self-Attention. Self-attention initially calculates the queries, keys, and values ($Q,\ K,\ V$) from a feature map. The attention function's goal is to map the set of key-value pairs and the query into an output. The output is computed as a weighted sum of the values, with each value's weight determined by a compatibility function of the query with the corresponding key. The output is computed as follows:

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{2.1}$$

By reviewing CNN-based methods, it is clear that there has been a significant improvement over the years, however, these methods mainly focus on small discriminative regions due to a Gaussian distribution of effective receptive field. Even with attention mechanisms to explore long-range dependencies, attention-based methods still prefer large continuous areas and make it difficult to extract multiple diversified discriminative parts as shown in figure 2.4. Besides that, the CNN downsampling operators (e.g. pooling and strided convolution) reduce the spatial resolution of output feature maps, which has a major impact on the discrimination ability of objects with similar appearances.

Transformers were introduced in NLP and have since become the new standard for a vast scope of NLP tasks. Transformers have recently been used in a wide range of vision tasks, including image classification, object detection, semantic segmentation, and visual tracking, and have demonstrated outstanding capabilities and potential for handling sequential data.

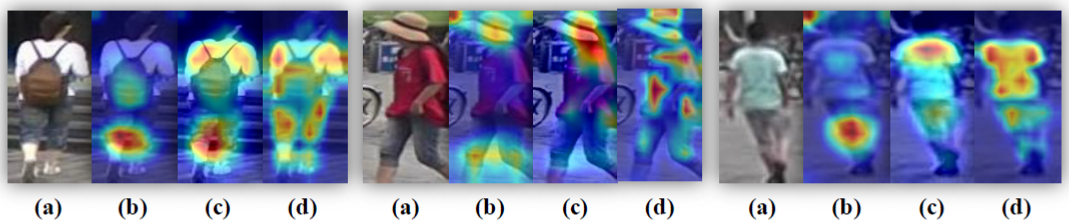


Figure 2.4: Grad-CAM [3] visualization of attention maps: (a) Original images, (b) CNN-based methods, (c) CNN+attention methods, (d) Transformer-based methods which captures global context information and more discriminative parts. Taken from [4].

Vision Transformer (ViT) [5] and Data-efficient image Transformers (DeiT) [6] have shown that pure transformers can be as effective as CNN-based methods on feature extraction for image recognition. The main problem of transformers is that they do not generalize well when dealing with small amounts of data. The authors of ViT, on the other hand, obtained impressive results when training on larger datasets (14M-300M images), achieving state-of-the-art results in the ImageNet benchmark, without using any convolution. DeiT also achieves competitive results on ImageNet without requiring a large training dataset and with less computing resources.

The functioning of a Vision Transformer can be stated as follows: the input image is split into fixed-size patches, each of them is linearly embedded, position embeddings are added, and the resulting sequence of vectors is fed to a standard Transformer encoder. The Transformer encoder consists of alternating layers of multiheaded self-attention (MSA) and MLP blocks. In order to perform classification, it is used the standard approach of adding an extra learnable "classification token" to the sequence, which output will serve as a global feature representation. Figure 2.5 shows an overview of the ViT model.

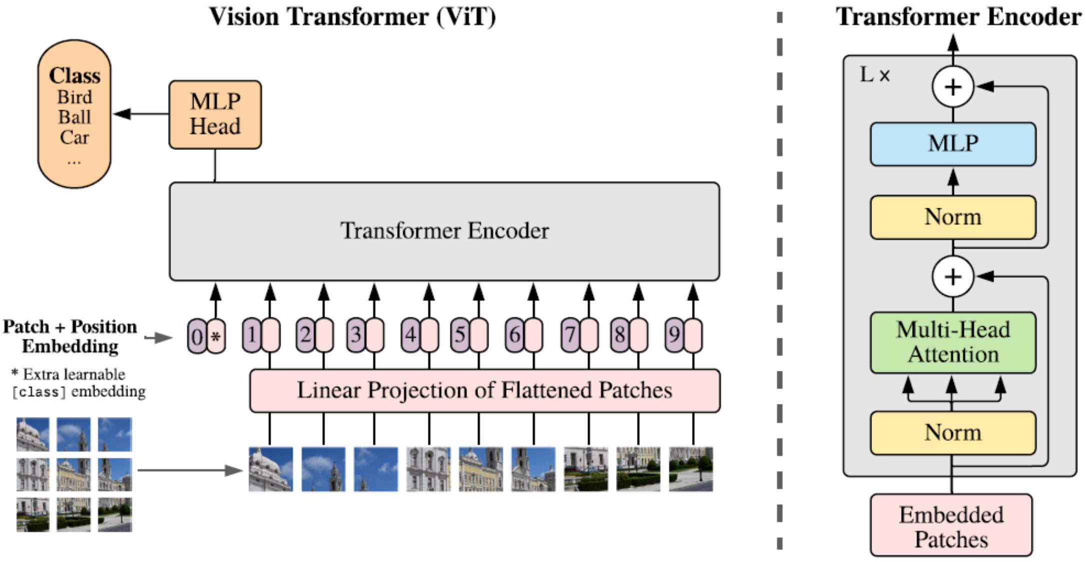


Figure 2.5: Vision Transformer model overview. Taken from [5].

2.2 Vehicle Re-Identification

Vehicle Re-Identification based on appearance is a difficult task since distinct vehicle identities (IDs) with similar appearances have little inter-instance variability, yet one vehicle has large intra-instance variability under different viewpoints and lighting conditions. The most difficult challenge in V-ReID is associating a vehicle with the same ID but from multiple perspectives, hence several studies focus on that problem, attempting to extract a highly discriminant feature from the vehicle so that it can be re-identified even from a different perspective.

Quadruple Directional Deep Learning Features (QD-DLF) [1] that was proposed by Jianqing Zhu et al. in 2018 is an example of work that focuses on the perspective problem. The author created four networks that share the same backbone, which is a shortly and densely connected CNN. To comprehensively describe vehicle images from different directions, quadruple directional (i.e., horizontal, vertical, diagonal, and anti-diagonal) average pooling layers are designed to extract different features after the backbone. Then, the resulting feature maps are concatenated together as a quadruple directional deep learning feature for V-ReID. This work is interesting in terms of methodology for employing multiple networks to extract different features, and it was state-of-the-art at the time it was published but due to the fast advancement in this field, was quickly outperformed by new methods that now perform significantly better.

2.2.1 Attribute extraction based methods

Many works explore attribute clues (for instance: color, type, model) to enhance V-ReID. The common pipeline for this approach is the Vanilla-Attribute Design Network (VAN) illustrated in figure 2.6. The problem with these methods is the lack of effective interaction between the attribute-based branches and the V-ReID branch, where the attribute modules learn features for attribute recognition but are not explicitly designed to serve V-ReID.

To address this problem, Rodolfo Quispe et al. developed a network called AttributeNet (ANet) [7] that jointly extracts identity-relevant features and discriminative feature attributes. ANet uses the VAN network as a backbone and then encourages interaction between the attribute features and the V-ReID feature by distilling attribute information and incorporating it into the global representation (from the backbone) to generate more robust features. ANet architecture is illustrated in Figure 2.7. The network, in particular, combines the feature maps of several attribute branches to create a single representation G of all the attributes. The helpful attribute feature from G is then distilled and compensated onto the global V-ReID feature F to produce the final feature map J, where the spatial average pooled feature of J is the final ReID feature. Furthermore, the authors introduced the Amelioration Constraint (AC), a novel supervision objective that pushes the compensated V-ReID feature J to be more discriminative than the V-ReID feature F.

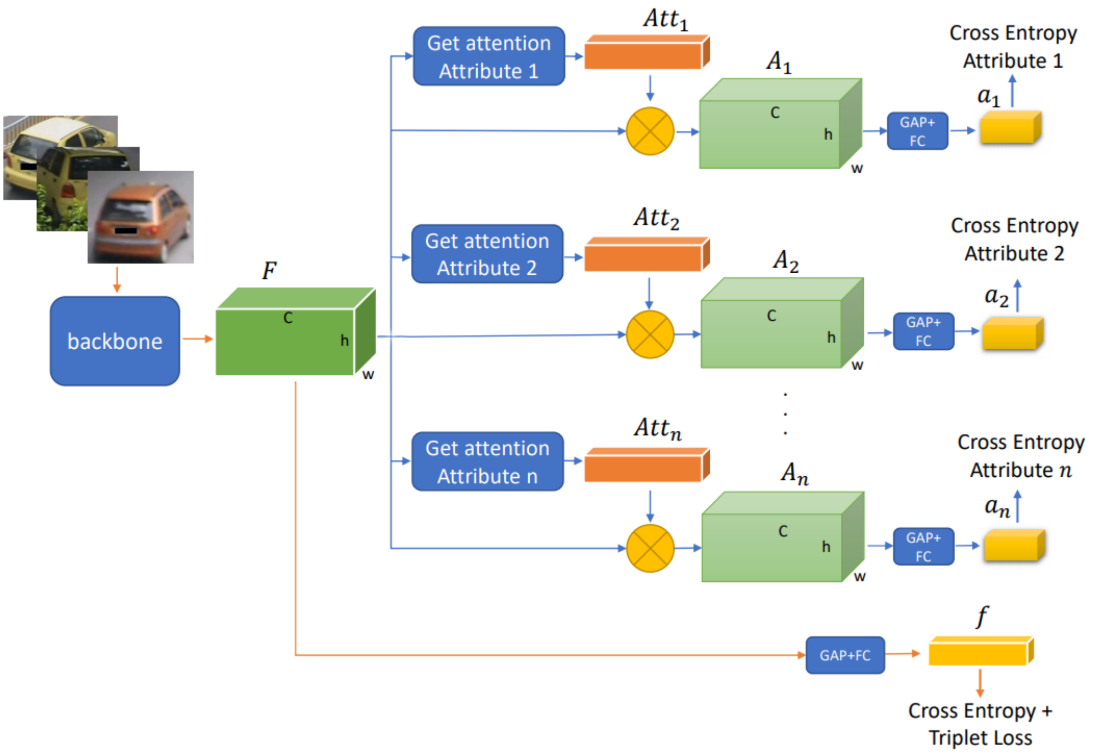

backbone
F
C
h
w
Get attention Attribute 1
Get attention Attribute 2
Get attention Attribute n
Att_1
Att_2
Att_n
A_1
A_2
A_n
GAP+ FC
a_1
a_2
a_n
Cross Entropy Attribute 1
Cross Entropy Attribute 2
Cross Entropy Attribute n
GAP+FC
f
Cross Entropy + Triplet Loss

Figure 2.6: Vanilla-Attribute Design Network (VAN). Adapted from [7].

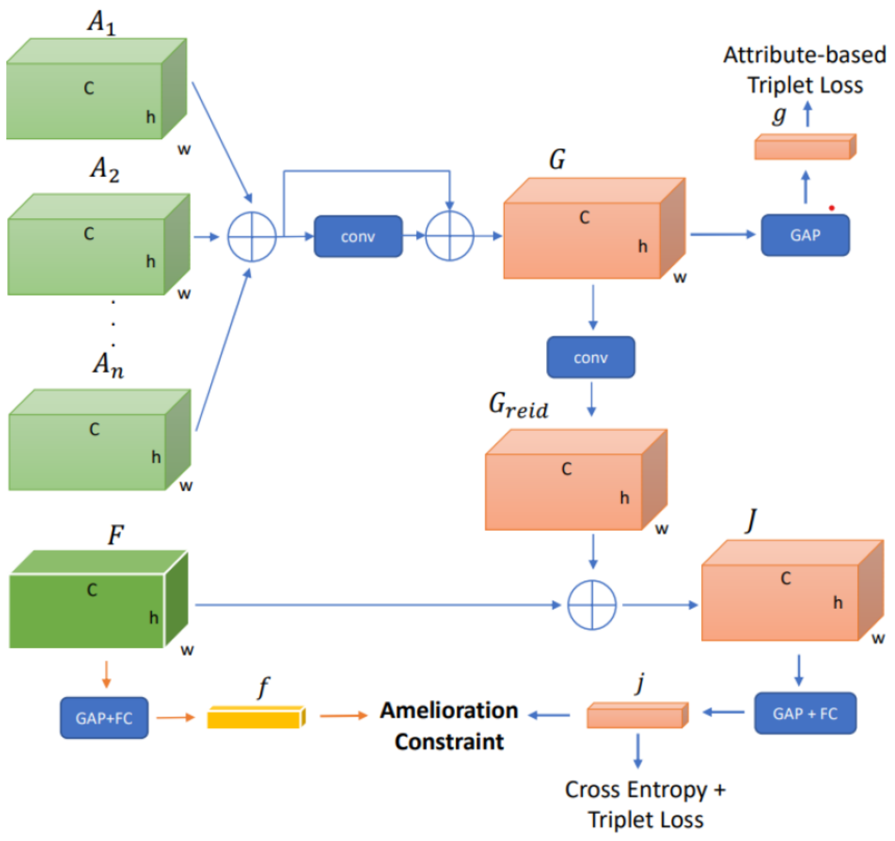


Figure 2.7: ANet Architecture. Taken from [7].

Stripe-based and Attribute-aware Network (SAN) [8] proposed by Jingjing Qian et al. is another work based on attributes to learn efficient embeddings for V-ReID. In this paper, a ResNet-50 is employed as the backbone network to generate a feature map, which is then used in two independent branches. SAN is a two-branch neural network that consists of a stripe-based branch and an attribute-aware branch as shown in Figure 2.8. Since the human body may be separated into multiple significant parts (e.g., head, thorax, legs, and feet), this feature division is a frequent method in P-ReID models. Inspired by this, the authors of this work also considered that a car body can be roughly divided into several meaningful parts along the vertical axis (e.g., ceiling, windshield, header panel, and wheels), so they used a stripe-based branch with horizontal average pooling and dimension-reduced convolutional layers to achieve part-level features. Meanwhile, the attribute-aware branch extracts the global feature under the supervision of vehicle attribute labels to separate similar vehicle identities with different attribute annotations. Finally, the part-level and global features are concatenated to generate the final descriptor for V-ReID.

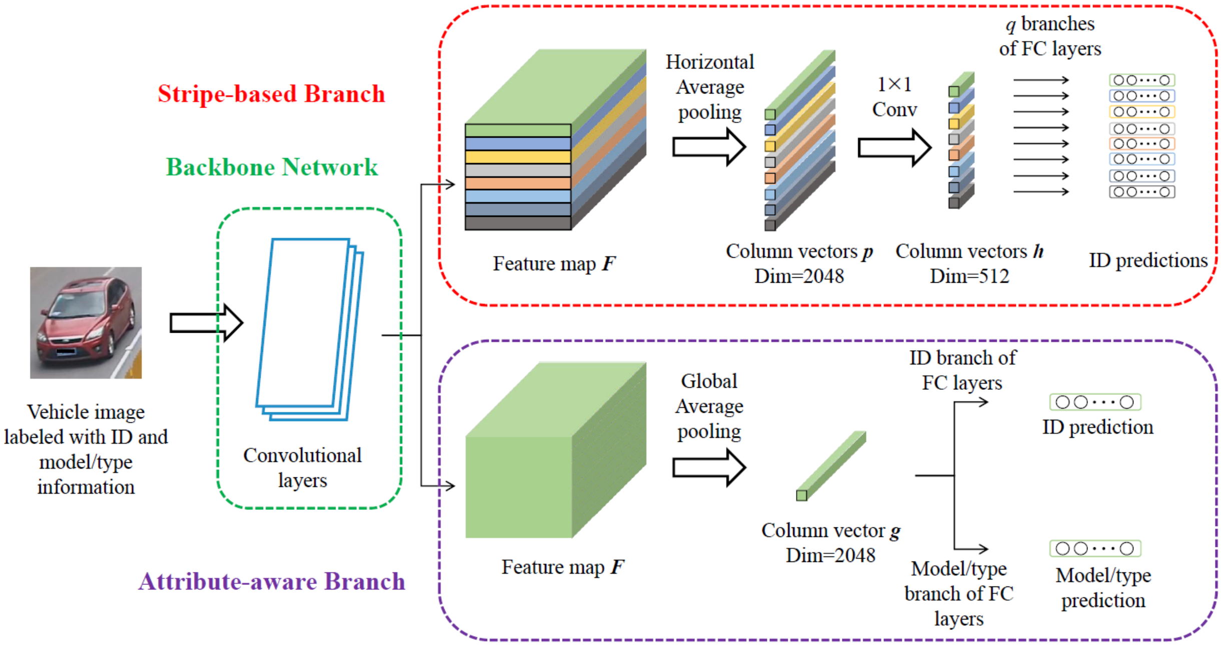


Figure 2.8: SAN Architecture. Taken from [8].

2.2.2 Attention based methods

Attention is one of the most essential mechanisms of human visual perception. When confronted with a complex scene, humans are able to select regions of interest and use attention to reduce the search and speed up recognition.

Attention mechanisms have proven to be particularly effective in fine-grained visual recognition tasks and become a key component in many state-of-the-art methods, improving the discriminative power of CNN features in the person and vehicle ReID problem by focusing on significant characteristics and suppressing irrelevant features.

Yongmin Rao et al. introduced a method called counterfactual attention learning (CAL) [9] to enhance attention learning based on causal inference. Despite the widespread use of attention mechanisms, the problem of learning efficient attention is still insufficiently understood. Most existing approaches learn visual attention in a weakly-supervised manner, i.e., the attention modules are merely supervised by the final loss function, without a powerful supervisory signal to guide the training process. This likelihood-based technique only explicitly supervises the final prediction but ignores the connection between prediction and attention. To address this issue, the authors designed a tool to analyze the effects of learned visual attention with counterfactual causality. The main idea is to measure the quality of attentions by comparing the learned attentions and the counterfactuals (i.e., uncorrected attentions) on the final prediction. Then, the goal is to maximize the difference to encourage the network to learn more effective visual attentions and reduce the effects of biased training set. The results of this method were not particularly outstanding in V-ReID, but they were interesting in P-ReID.

A Strong Baseline for V-ReID was proposed by Su V. Huynh et al. specifically targeting the Track 2 dataset of the 5th AI City Challenge [10]. Since the dataset contains synthetic data, they adopted MixStyle Transfer as a regularization method to reduce the gap between the real and synthetic data. To help the model learn more detailed features, a multi-head with an attention mechanism was employed attached to the backbone as seen in Figure 2.9. Using multiple heads instead of only one encourages the ReID model to learn more diverse features from different vehicle characteristics. The 2048-dim feature of the backbone is fed into many parallel fully connected (FC) layers (heads), and the attention mechanism selects which head's features contribute more to the final encoding feature.

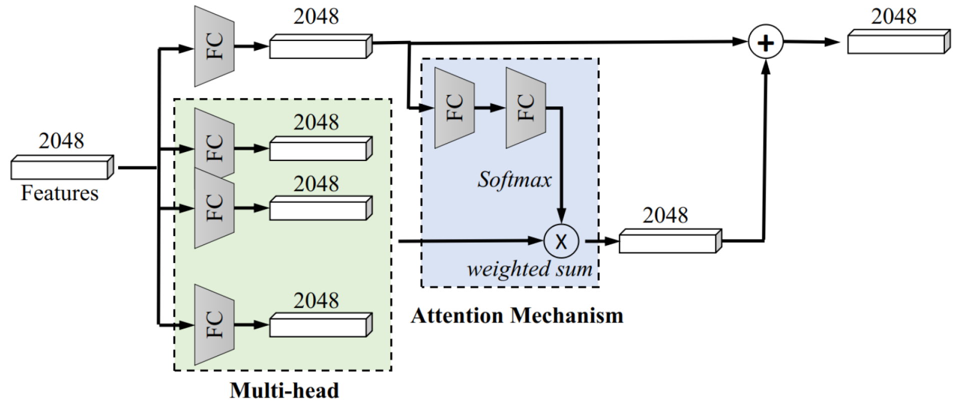


Figure 2.9: Multi-Head with attention mechanism. Taken from [11].

Additionally, the typically used Triplet Loss was replaced by the Supervised Contrastive Loss [12], which helped the network to learn more effectively. In practice, ID Loss is substantially larger than Metric Loss, causing an imbalance and affecting training performance. The loss weights are generally set equally, i.e. 1:1 ratio, therefore the authors developed a simple module called Momentum Adaptive Loss Weight (MALW). This module increase training stability by automatically updating loss weights according to the statistical characteristics of loss values between the Supervised Contrastive Loss and the Cross-Entropy Loss, which significantly enhanced performance.

2.2.3 Variational Auto-Encoders based methods

Self-supervised Attention for V-ReID (SAVER) [13] was proposed by Pirazh Khorramshahi et al. that developed a Variational Auto-Encoder (VAE) to generate a vehicle image template without its details. Following that, the authors create a residual image by calculating the pixel-wise difference between the reconstructed template image and the original one. This residual includes key ReID details and serves as a pseudo-saliency or pseudo-attention map, highlighting discriminative and salient regions in a vehicle image. These vehicle-specific salient regions carry critical details that are essential for distinguishing two visually similar vehicles. Several methods in the literature rely on costly key-point labeling, part annotations, and other characteristics such as vehicle brand, model, and color. Strongly-supervised algorithms are unable to scale across domains due to a large number of V-ReID datasets with various levels of annotations. Self-supervised Attention, on the other hand, stands out for its ability to effectively learn vehicle-specific discriminative features without the use of additional annotations, attributes, spatio-temporal, or multi-modal information. Figure 2.10 shows the proposed SAVER pipeline.

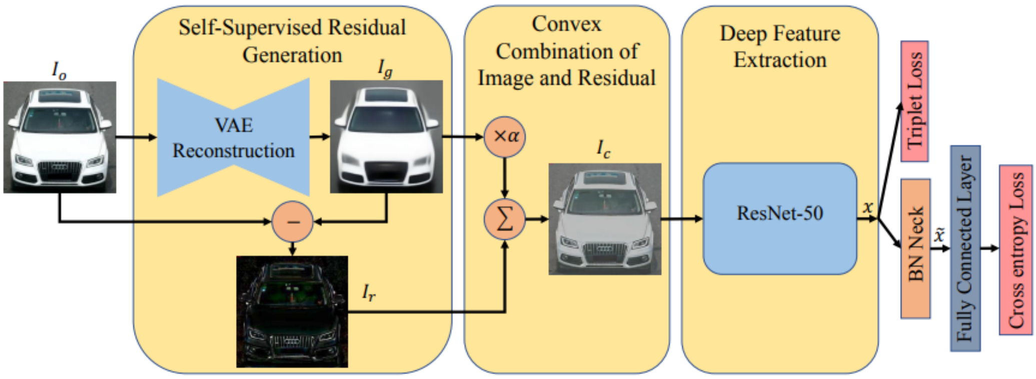


Figure 2.10: SAVER pipeline. Taken from [13].

2.2.4 Data Augmentation based methods

Alternatively to the abovementioned studies, several other works use additional data for more robust training and better results. The extra data can be obtained from generative adversarial networks (GANs), which create vehicle images with different orientations, appearance variations, and other attributes. Since labeling data is extremely costly and time-consuming, using synthetic data provided by GANs is increasingly being used by many works. However, there is always a domain gap between the real and the synthetic data, which leads to feature distribution shifting.

Pose-Aware Multi-Task Learning for Vehicle Re-Identification Using Highly Randomized Synthetic Data (PAMTRI) [14] and Joint Semi-supervised Learning and Re-ranking for Vehicle Re-identification [15] are two examples of works that used GANs to generate large-scale synthetic data to enrich the training set, demonstrating that using synthetic images is beneficial to improve the results.

CNN's are known as data hungry and most of the V-ReID datasets are small-scale, causing the networks to easily over-fit. To overcome this issue, Zhedong Zheng et al. created a unique large-scale vehicle dataset (called VehicleNet [16]) by combining four public vehicle datasets and designed a simple yet successful two-stage progressive technique to learn more robust visual representations from VehicleNet. The purpose of the first stage is to train with the traditional classification loss to learn the generic representation for all domains (i.e., the datasets present in VehicleNet). The second stage involves fine-tuning the trained model purely on the target vehicle set by decreasing the distribution gap between VehicleNet and any target domain.

2.2.5 Transformers based methods

To overcome some of the CNN problems, a pure transformer-based object ReID framework named TransReID [4] was introduced by Shuting He et al. This architecture use multi-head self-attention to capture long-range dependencies for spatial and sequential data and does not use downsampling operators, keeping more detailed information. Moreover, the authors designed a Jigsaw Patch Module (JPM), consisting of shift and patch shuffle operation, which facilitates perturbation-invariant and robust feature representation of objects. They also introduced a Side Information Embeddings (SIE) that encodes non-visual information, such as camera IDs, viewpoints, or other types of information into embedding representations to learn invariant features and is shown to effectively mitigate the bias of learned features. The results obtained in both person and vehicle ReID surpassed some of the most advanced CNN-based algorithms, demonstrating the capability of Transformers. Figure 2.11 shows the TransReID architecture.

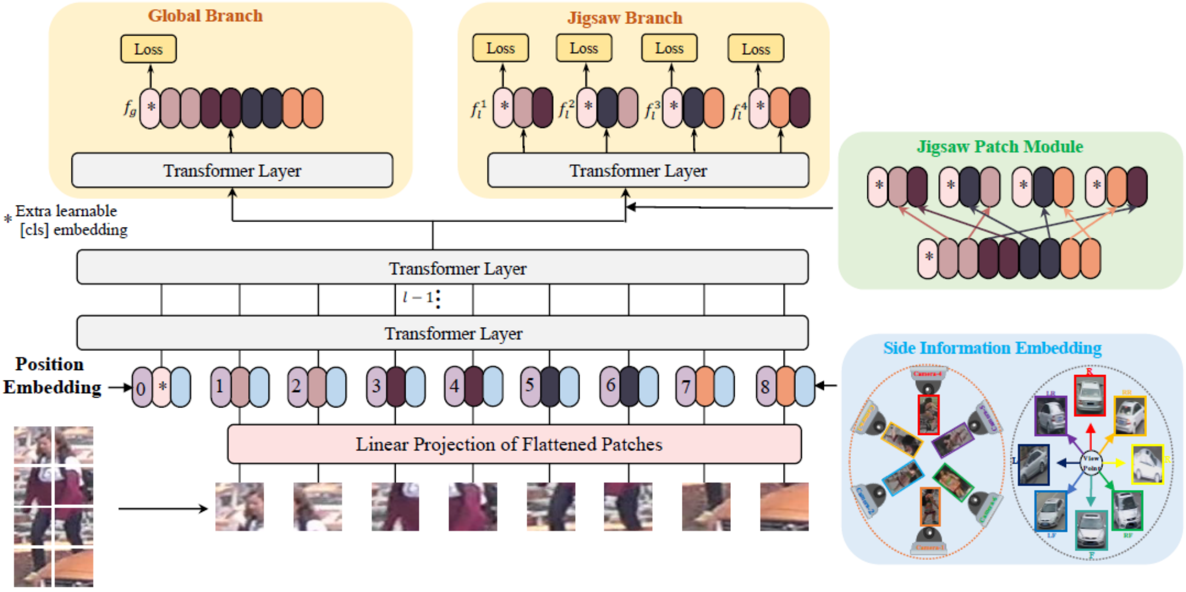
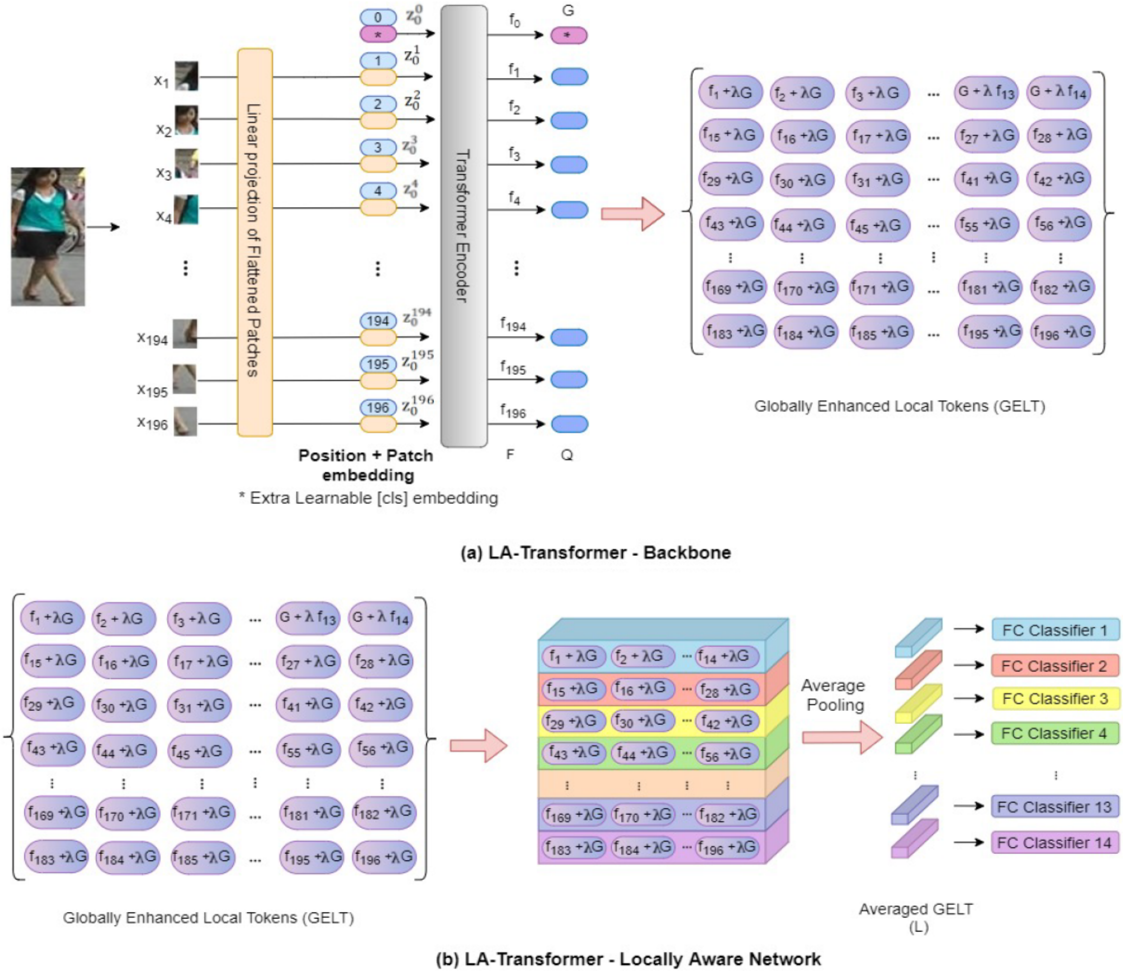


Figure 2.11: TransReID architecture. Taken from [4].

In P-ReID, the rising use of transformers was also noticeable, with high success rates such as the novel Locally Aware Transformer (LA-Transformer) [17] developed by Charu Sharma et al. The primary output of a vision transformer is a global classification token but vision transformers also yield local tokens which contain additional information about local regions of the image, which are not used in ViT. LA-Transformer employs a Parts-based Convolution Baseline (PCB)-inspired strategy for aggregating globally enhanced local classification tokens into an ensemble of $\sqrt{N}$ classifiers, where N is the number of patches. PCB is a strong convolutional baseline for P-ReID that divides the feature vector obtained from the backbone network into six vertical regions and uses a voting strategy to form an ensemble of regional classifiers to determine the predicted class label. One of PCB's weaknesses is that each regional classifier ignores global information, which is crucial for recognition and identification. Despite this limitation, PCB has had a lot of success, hence LA-Transformer architecture incorporates a PCB-like approach for combining globally enhanced local tokens. Another innovative aspect of this approach is the use of blockwise fine-tuning, which improves the classification accuracy of the LA-Transformer for P-ReID. Blockwise fine-tuning is viable as a form of regularization when training models with a large number of parameters over relatively small in-domain datasets. It entails freezing all the transformer layers except for the bottleneck model at the start, then unfreezing one layer at a time from the last to the first every t epochs. Figure 2.12 shows the LA-Transformer architecture.



(a) LA-Transformer - Backbone

(b) LA-Transformer - Locally Aware Network

Figure 2.12: LA-Transformer architecture. Taken from [17].

A more recent study entitled Multi-attribute adaptive aggregation transformer for vehicle re-identification [18] developed by Zhi Yu et al. is an attribute-based Transformer network that considers image feature and the attribute feature simultaneously. The authors consider that the vehicle's color and model are the most intuitive attributes, as well as the most stable and distinctive. Another additional attribute is the viewpoint of the vehicle images which differs from image to image. Different attributes are supposed to have different importance, therefore, a multi-attribute adaptive aggregation network was designed to compare different attributes and assign different weights to the corresponding features, which is a benefit for generating more robust and discriminative features. In addition, a multi-sample dispersion triplet (MDT) loss was proposed to improve the suggested transformer network. This loss includes not only the hardest positive sample and hardest negative sample based on the hard mining strategy but also some extra positive and negative samples. The loss is dynamically adjusted via multi-sample dispersion, which can guide better feature space division for V-ReID. The architecture is illustrated in Figure 2.13.

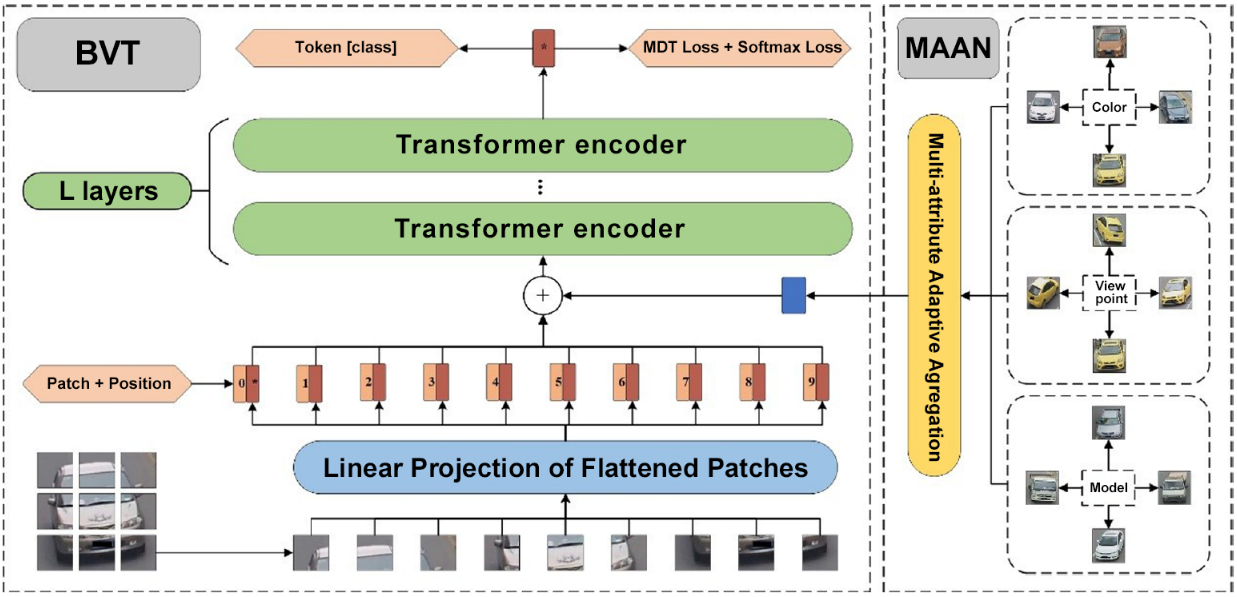


Figure 2.13: Multi-attribute adaptive aggregation transformer. Taken from [18].

2.3 AI City Challenge

The AI City Challenge has the purpose of improving the efficiency of operations in city environments by pushing the boundaries of research and development in intelligent video analysis for smarter cities use cases, and assessing tasks where the level of performance is enough to cause real-world adoption. One of the challenges purposed is a V-ReID Track with both real-world and synthetic data in the 5th AI City Challenge 2021 [19].

The winning team of this challenge [19] used the above-mentioned Bag of Tricks (BoT) as the CNN-based baseline and mainly focus on four points. First, they used synthetic data because there was a lack of real data. They observed that the real data had inaccurate bounding boxes, which introduced noise into the images, so they re-detected the bounding boxes of real-world images based on their heatmaps and added them to training data. The second stage was to use an unsupervised domain-adaptive model (UDA) since a new scenario appeared in the test set, resulting in domain bias between the training and test sets. This method is essentially a clustering algorithm that generates pseudo labels on testing data before fine-tuning baseline models to reduce current domain bias between the training and test sets. The third point was applying a variety of post-processing techniques, such as re-ranking, image-to-track retrieval, inter-camera fusion, and so on, which considerably enhanced final performance. Lastly, to boost performance, they made an ensemble of several CNN-based models with a transformer-based model (TransReID) and discovered that the Transformer provided representation diversity different from CNN models.

2.4 Loss Functions

The loss functions have an important role in the optimization of the network training and the main goal is to pull samples with the same ID together, while pushing those with different IDs far apart. Generally, in almost all V-ReID works, the final cost function is composed of two losses: Cross-Entropy and Triplet Loss. The Cross-Entropy Loss (CE Loss) is also known as Identity Loss and its main goal is to minimize the Identity Classification Error. CE Loss is:

$$L_{CE} = \sum_{i=1}^{N} -q_i log(p_i) \qquad \begin{cases} q_i = 0, & y \neq i \\ q_i = 1, & y = i \end{cases} \tag{2.2}$$

where, q_i is the truth label and p_i is the Softmax probability for i^{th} class.

The Triplet Loss (Tri Loss) is also known as Metric Loss, and it has the ability to enhance the intra-class compactness and inter-class separability in the Euclidean space. Because of these two factors, triplet loss is more suitable for a V-ReID network than identity loss. A triplet loss is defined in terms of three image instances: Anchor (a randomly chosen instance); Positive (an instance that has a common ID with the anchor); and Negative (an instance that does not share a common ID with the anchor). Denoting these instances a, p and n, respectively, the triplet loss can be written as:

$$L(a_i, p_i, n_i) = max(d(a_i, p_i) - d(a_i, n_i) + \alpha, 0) \tag{2.3}$$

where α is the desired margin separation between the positive and negative instance, and $d(a_i, p_i)$ and $d(a_i, n_i)$ are the feature distances of positive pair and negative pair, respectively.

The final triplet-cost is computed by summing the individual triplet losses:

$$L_{tri} = \sum_{j=1}^{T} L(a_{i,j}, p_{i,j}, n_{i,j}) \tag{2.4}$$

where T is the total number of triplets.

Despite the fact that triplet loss is better than identity loss, it is not used alone. Instead, combining the two works out better, which leads to the final cost function:

$$L = \lambda_{CE} L_{CE} + \lambda_{tri} L_{tri} \tag{2.5}$$

where λ_{CE} and λ_{tri} are the loss weights.

Incorporating every possible triplet into the triplet-cost in a naive manner frequently produces unsatisfactory results. Therefore many studies opt for triplet-mining, a technique that aims to incorporate only the most relevant triplets into the triplet-cost. The most commonly used sample strategy heuristic is the hard mining Triplet Loss [20], which selects the hardest positive sample and the hardest negative sample, within the batch. In other words, the hardest positive is an embedding of the same class as the anchor, and the euclidean distance between them is maximized. The hardest negative is an embedding that does not belong to the same class as the anchor and the euclidean distance between them is minimized. Compared with the traditional Triplet Loss, hard mining Triplet Loss can increase the model's training speed and accuracy.

Recently, Adhiraj Ghosh et al. developed a Relation Preserving Triplet Mining (RPTM) for Stabilizing the Triplet Loss in Vehicle Re-identification [21]. The authors of this paper suggest that severe appearance changes due to pose variations are indicators that an object ID is made up of numerous natural groups and that forcing instances from different groups to a shared location is counter-productive. The idea is to choose a semi-hard positive that shares a natural group with the anchor, rather than a hard positive, which is the most challenging positive to the anchor and usually has a completely distinct pose and appearance.

To select the triplet samples, a modern feature matcher GMS [22] is used between anchor-positive pair images as a relational indicator, using the following criterion: high number of matches between anchor-positive pair images means they are similar and belong to the same natural group; otherwise, they do not. A positive that is too similar to the anchor does not provide significant value to the triplet-cost, thus a threshold τ is set as the average number of GMS matches in the set of non-zero pairwise GMS matches between the anchor and all other images. The anchor-positive pair that has the closest number of matches to τ is then selected for the triplet. According to the authors, relational triplets give more attention to the problem of intra-class separability than alternative triplet mining methods.

Many additional studies in the literature focus on different aspects of loss; for example, some use not only the hardest positive and negative sample based on the hard mining technique but also some extra positive and negative samples. Others, on the other hand, explore losses such as Circle Loss, Center Loss, Large Margin Cosine Loss, and so on.

2.5 Grad-CAM

Deep learning algorithms are typically used as a "black box" and it can be challenging to comprehend both how they function and, more importantly, why they do so. For the purpose of facilitating and improving the interpretation of the results, Gradient-weighted Class Activation Mapping (Grad-CAM) [3] was created to produce "visual explanations" for Deep Learning Algorithms' decisions.

Grad-CAM uses the gradients of any target concept flowing into the final layers to create a coarse localization map highlighting the key regions in the image for predicting the concept. Grad-CAM can map explanatory visuals at any layer of the model, however as is well known, higher-lever visual representations are obtained at deeper layers. Because the neurons will be searching for class-specific information in the last stages, the interest is typically in the final layers. One example of this representation has been shown previously in Figure 2.4.

2.6 Post-Processing Techniques

Lot of studies started to adopt Post-Processing Techniques to improve the final results. The most common is Re-Ranking methodology proposed in [23]. Re-ranking is a k-reciprocal encoding method to refine the ReID results which takes the high-confidence candidate images into consideration. Given the initial ranking list, if a gallery image is similar to the probe in the k-reciprocal nearest neighbors, it is more likely to be a true match as seen in Figure 2.14. Specifically, given an image, a k-reciprocal feature is calculated by encoding its k-reciprocal nearest neighbors into a single vector, which is used for re-ranking under the Jaccard distance. The final distance between two images is computed as the combination of the original distance and the Jaccard distance. This method is widely used because it does not require any human interaction or any labeled data.

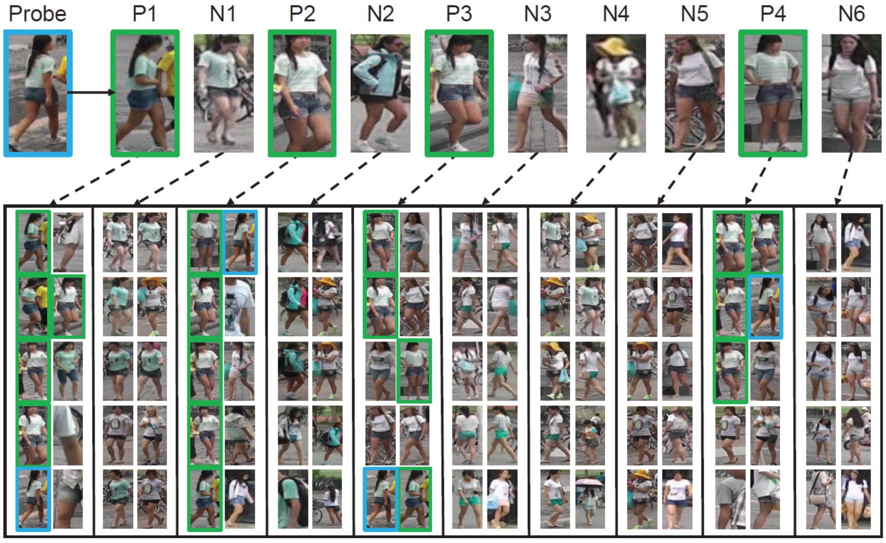


Figure 2.14: Re-Ranking. Taken from [23].

Camera Verification is another technique used in Post-Processing. Given the assumption that the query image and the gallery images were captured with different cameras, candidate images with the same camera ID are removed for each query image.

Model Ensemble is another popular approach to enhance final performance. It requires to train multiple networks with different backbones and then combine all them together by simply taking the averaged distance of each query image to gallery images or concatenating features.

2.7 Datasets

During the development and evaluation of ReID algorithms it is essential to rely on a properly annotated datasets. Therefore, a review of the most relevant V-ReID benchmark datasets is done below.

Veri-776 dataset [24], proposed by Li et al., contains a large number of vehicles captured by non overlapping cameras with different perspectives, scales and illuminations in real-world traffic surveillance environment. VeRi-776 contains 49,357 images of 776 vehicles from 20 cameras. The training subset has 37,781 images of 576 vehicles and for the evaluation, Veri776 dataset provides a query set with 1,678 images of 200 vehicles and a gallery with 11,579 images of 200 vehicles. It includes attribute labels for color and type. The author also provides the meta data, e.g., the collected time and the location.

Vehicle ID dataset [25] is captured in daytime by multiple real-world surveillance cameras distributed in a small city of China. There are 221,763 images of 26,267 subjects in the entire database. The vehicle images are collected in two views, i.e., frontal and rear views, and the vehicle brand and model information are marked. The colors are divided into 7 categories. A total of 110,178 images from 13,134 vehicles make up the training subset. VehicleID also provides 3 testing subsets for evaluating performance on different data scales: Test800, Test1600, and Test2400. Test800, in particular, has 800 gallery images and 6,532 probe images of 800 vehicles. Test1600 contains 1600 gallery images and 11,395 probe images of 1,600 vehicles. Test2400 contains 2400 gallery images and 17,638 probe images of 2,400 vehicles.

Veri-Wild dataset [26] is one of the largest V-ReID datasets, composed of 416,314 vehicle images of 40,671 identities acquired from 174 surveillance cameras over the course of a month, under unconstrained scenarios. Poor weather conditions, such as rainy, foggy, and so on, are also included in Veri-Wild, which are not included in other datasets. The training set contains 277797 images of 30671 vehicles and the testing set is divided into three sets with 3,000 (small), 5,000 (medium) and 10,000 (large) vehicle IDs. It includes attribute labels for vehicle model, color and type.

CityFlow-V2 dataset [27] is a real-world dataset captured by 46 cameras in a real-world traffic scenario. It has 85,058 images of 880 vehicles in total. For training, 52,717 images of 440 vehicles were used. The remaining 31,238 images of 440 vehicles is used for testing. It's worth noting that the training set has been recorded by 40 cameras. In the test set, part of images were captured by 6 new cameras that do not exist in the training set.

VehicleX dataset [28] is a synthetic dataset that only provides training set that contains 192,150 images of 1,362 vehicles in total. In addition, the attribute labels, such as car colors, car types, orientation labels are also annotated.

2.8 State-of-the-art Results

In the following tables 2.1 and 2.2 are shown the State-of-the-art results for the datasets Veri-776 and Veri-Wild, respectively.

Method	mAP (%)	Rank-1 (%)	Rank-5 (%)
QD-DLF [1]	61.8	88.5	94.5
PAMTRI [14]	71.9	92.9	97.0
PVEN [29]	79.5	95.6	98.4
SAVER [13]	79.6	96.4	98.6
GLAMOR [30]	80.3	96.5	98.6
VAT [18]	80.4	97.5	98.7
SGFD [31]	81	96.7	98.6
Transreid [4]	82.1	97.4	98.4
VehicleNet* [16]	83.4	96.8	-

Table 2.1: State-of-the-art on the Veri-776 dataset.

*This method uses large additional data from other datasets (extra data).

	Test Size = 3000			Test Size = 5000			Test Size = 10000		
Method	mAP (%)	Rank-1 (%)	Rank-5 (%)	mAP (%)	Rank-1 (%)	Rank-5 (%)	mAP (%)	Rank-1 (%)	Rank-5 (%)
Strong Baseline [11]	76.6	90.8	97.3	70.1	87.5	95.2	61.3	82.6	92.7
PVEN [29]	79.8	94.0	98.1	73.9	92.0	97.2	66.2	88.6	95.3
SAVER [13]	80.9	93.8	97.9	75.3	92.7	97.5	67.7	89.5	95.8
DFNet [32]	83.1	94.8	98.1	77.3	93.2	97.5	69.9	89.4	96.0

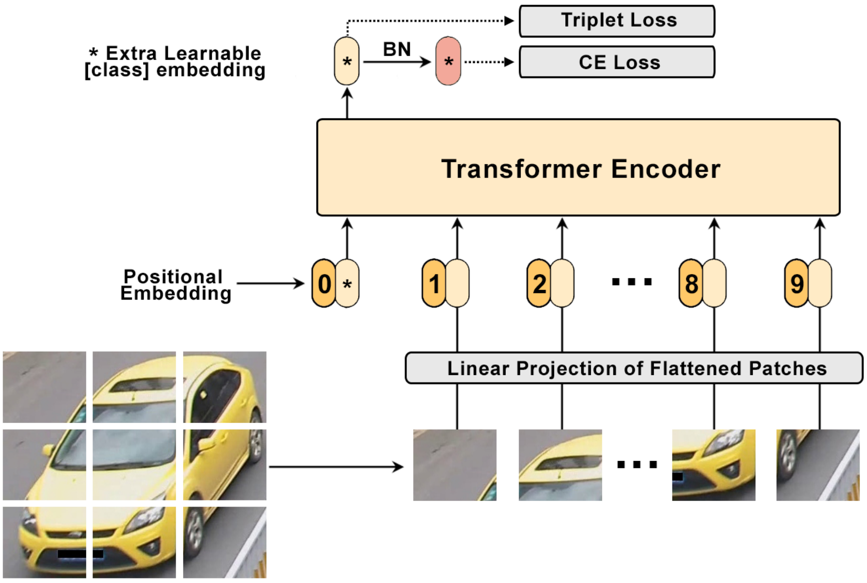
Table 2.2: State-of-the-art on the Veri-Wild dataset.

3 Methodology

This chapter presents the solutions and methods developed during the thesis towards the improvement of current V-ReID approaches. In order to accomplish the objective of this dissertation, the methodology that will be followed is greatly influenced by TransReID [4], which is the current V-ReID state-of-the-art approach.

3.1 Baseline Network

In the first place, a simple V-ReID Baseline network was designed using the Vision Transformer ViT [5] as a backbone to extract features of vehicle images. Figure 3.1 illustrates the Baseline pipeline.



Triplet Loss
* Extra Learnable [class] embedding
BN
*
CE Loss
Transformer Encoder
Positional Embedding
0
*
1
2
...
8
9
Linear Projection of Flattened Patches
...

Figure 3.1: Baseline pipeline. Adapted from [5].

Given an input image $x \in \mathbb{R}^{H \times W \times C}$, where $H,\ W,\ C$ denote its height, width, and number of channels, respectively, it is decomposed into a batch of N fixed-sized patches x_p^i, $i = 1,..,N$ as seen in figure 3.2.

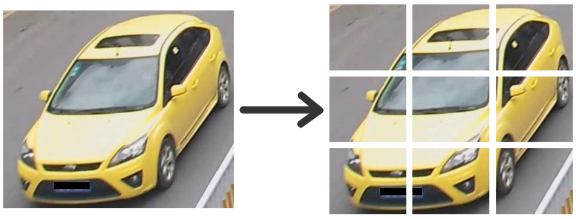


Figure 3.2: Image x is split in N patches.

Each patch x_p^i is linearly projected and flattened into a vector of D dimensions using a Patch Embedding Function $E(x_p^i)$, which is obtained using a convolution layer with a kernel size of $P \times P$ and stride size of P for non-overlapping patches. D is the number of channels and is set to 768 which represents the size of the embedding and *(P,P)* is the patch resolution. As seen in Figure 3.3, the outcome of this operation is a sequence of vectors (embedded patches) that form a $D \times N$ matrix.

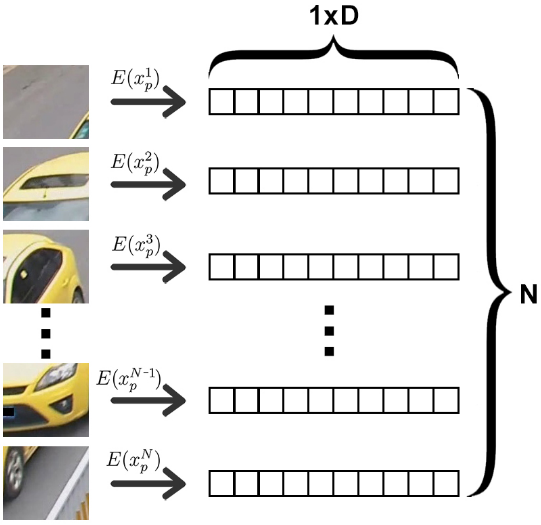


Figure 3.3: Sequence of embedded patches.

Afterward, a learnable class embedding x_{class} is prepended at the first position of the sequence of embedded patches as depicted in Figure 3.4. The output state of x_{class} keeps the information of the entire image and serves as the global feature to predict the class. To preserve positional information for each patch, learnable 1-D position embeddings P_e are added to the embedded patches. Then, the resulting sequence of embedding vectors Z_0 (Eq. 3.1) serves as input to the transformer encoder to generate $N+1$ feature vectors where N is the number of patches plus the class embedding.

$$Z_0 = [x_{class}; E(x_p^1); E(x_p^2); \cdots; E(x_p^N)] + P_e \tag{3.1}$$

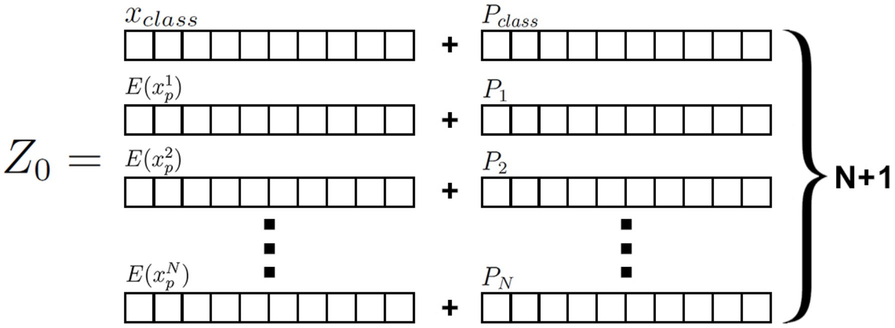


Figure 3.4: The input of the Transformer Encoder.

3.1.1 Transformer Encoder

The transformer encoder is composed of $L = 12$ blocks in series. Each block contains alternating MSA (Multiheaded Self-Attention) and MLP (Multi-layer Perceptron) blocks. The MLP block is composed of two linear layers separated by a GeLu activation. The first linear layer expands the dimension from D to $4D$, and the second layer reduces the dimension from $4D$ back to D.

Before MSA and MLP blocks, a Layernorm (Norm) is applied, as well as residual connections before each Norm block to allow better information flow through the network and prevent data loss. LayerNorm is a technique to normalize the distributions of intermediate layers. It enables smoother gradients, faster training, and better generalization accuracy. In Figure 3.5 is presented a schematic of a single Transformer Encoder block.

The following equations describe the functioning of a Transformer Encoder. The first Transformer Encoder block takes the Embedded Patches Z_0 as input, and after passing through all of the L blocks (eq. 3.2 and 3.3), the final output of the Transformer Encoder y (eq. 3.4) is obtained. This output y is composed of a global feature and N local features associated with each patch, however, only the global feature will be used for V-ReID in the Baseline network.

$$Z_l' = MSA(LN(Z_{l-1})) + z_{l-1}, \qquad l = 1...L \tag{3.2}$$

$$Z_l = MLP(LN(Z_l')) + Z_l', \qquad l = 1...L \tag{3.3}$$

$$y = LN(Z_L) \tag{3.4}$$

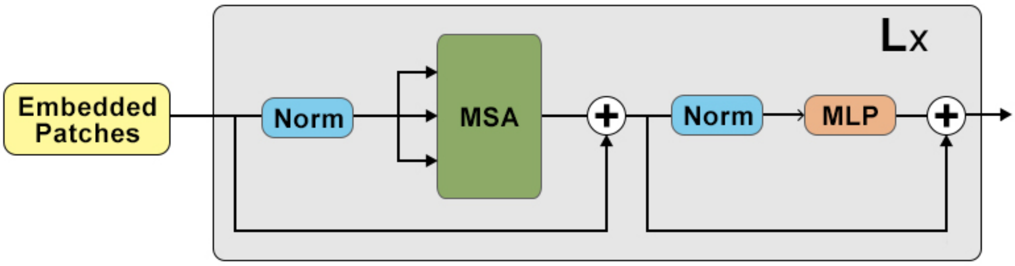


Figure 3.5: Transformer Encoder block.

Multihead self-attention (MSA) was proposed by Vaswani et al. [33] and it is an expansion of standard QKV self-attention (SA). The Attention block takes its input in the form of three parameters, known as the Query, Key, and Value. The logic behind the attention mechanism can be described as mapping a query vector and a set of key-value vector pairs to an output. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. Each query vector is matched against all the keys using inner products. These inner products are then scaled and normalized with a softmax function to obtain the weights on the values. In practice, sets of vector Queries, Keys and Values are packed together into matrices Q, K and V, resulting in a output matrix defined as:

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{3.5}$$

Finally, the Multi-head self-attention (MSA) is defined by taking into account h attention "heads," i.e., h self-attention functions applied in parallel to the input and project their concatenated outputs as illustrated in Figure 3.6

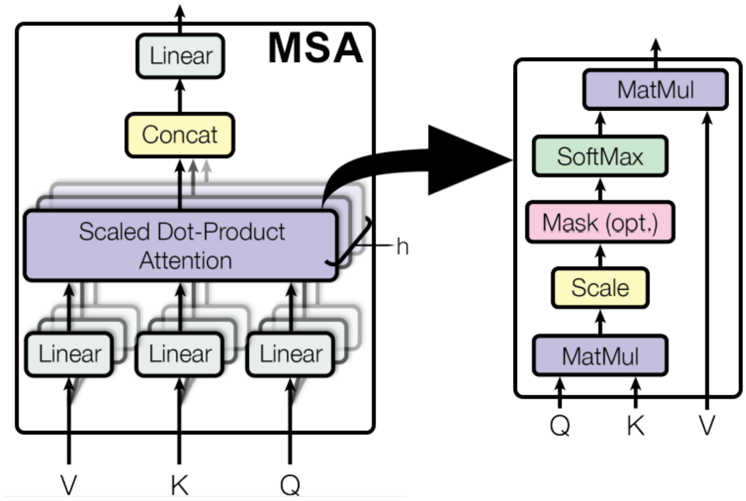


Figure 3.6: Multihead self-attention block. Adapted from [33].

3.1.2 Loss Functions

After obtaining the global feature, ID Loss and Metric Loss are employed to optimize the network. The ID Loss is the typical Cross-Entropy loss (eq. 3.6) and the Metric Loss is the Triplet Loss with soft-margin (eq. 3.7). When compared to the standard Triplet, the advantage of this loss is that it removes the need to determine the margin, which in the usual triplet must be carefully selected for the best results.

$$L_{CE} = \sum_{i=1}^{N} -q_i log(p_i) \qquad \begin{cases} q_i = 0, & y \neq i \\ q_i = 1, & y = i \end{cases} \tag{3.6}$$

Given a triplet set $\{a,p,n\}$, where a is the anchor, p is the positive and n is the negative, $d(a,p)$ and $d(a,n)$ are feature distances of positive pair and negative pair.

$$L(a,p,n) = log[1 + exp(d(a,p) - d(a,n))] \tag{3.7}$$

A batch normalization (BN) layer is applied before passing the global feature through the ID Loss as shown in Figure 3.7, resulting in a better feature compactness and loss convergence as mentioned in section 2.4.

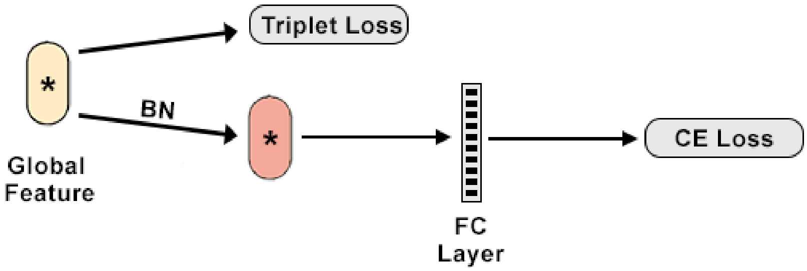


Figure 3.7: Loss Functions of the Baseline network.

The final loss is given by a weighted sum of Triplet Loss (L_T) and Cross-Entropy Loss (L_{CE}) as follows:

$$L = \lambda_{CE} L_{CE}(f_g) + \lambda_T L_T(f_g) \tag{3.8}$$

where f_g is the final global feature vector and λ_{CE}, λ_T are the loss weights which are typically set to a ratio of 1:1.

3.2 TransReID Network

The Baseline network only uses the global feature, however, the authors of TransReID added two modules to extract more robust features for V-ReID improvement. Furthermore, the TransReID uses a sliding window to generate patches with overlapping pixels, contrary to pure transformer-based networks such as ViT, which split images into non-overlapping patches, resulting in the loss of local neighboring structures around the patches. Denoting the step size (stride) as S and size of the patch as P, an input image with a resolution $H \times W$ will be split into N patches. N can be easily calculated using the eq. 3.9.

$$N = \left\lfloor \frac{H + S - P}{S} \right\rfloor \times \left\lfloor \frac{W + S - P}{S} \right\rfloor \tag{3.9}$$

The Jigsaw Patch Module (JPM) and Side Information Embedding (SIE) are the two novel modules designed for TransReID as shown in Figure 3.8.

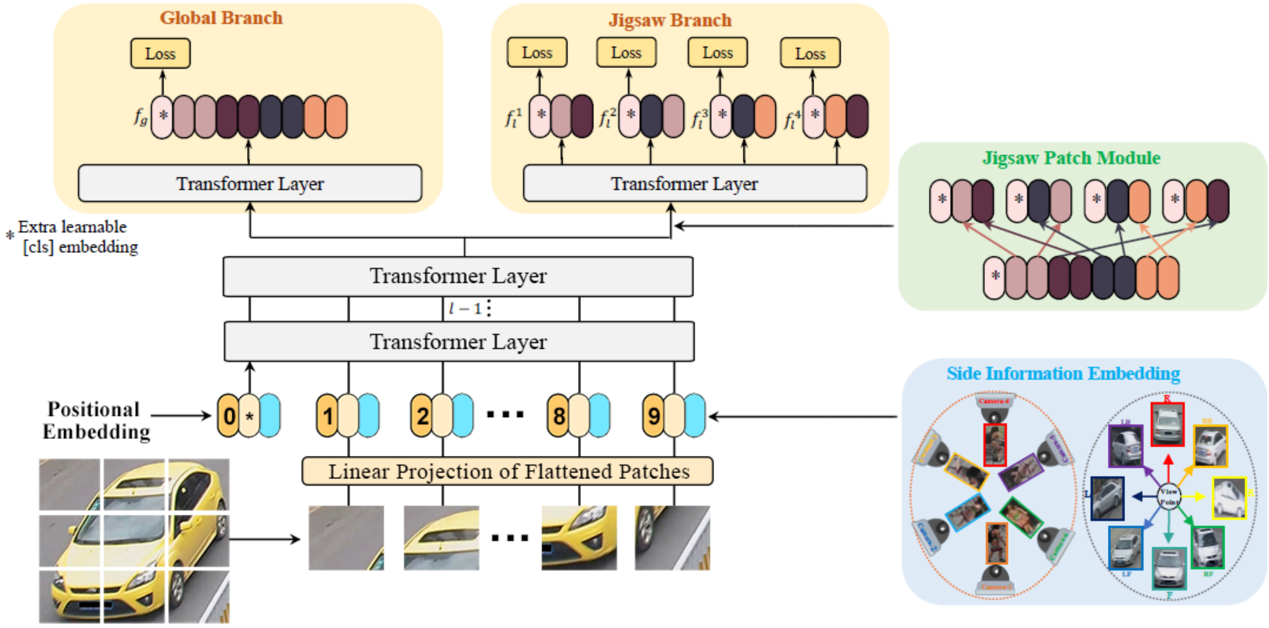


Figure 3.8: TransReID pipeline. Adapted from [4].

SIE was inspired by position embeddings, which use learnable embeddings to encode positional information. The fundamental purpose of SIE is to include non-visual information, such as camera IDs or viewpoints, into learnable 1-D embedding representations to learn invariant features. This will improve the network in reducing scene-bias caused by multiple cameras and viewpoints, helping the same vehicle to be recognized in diverse scenarios. In particular, as shown in Figure 3.8, SIE is placed before the transformer encoder, added to the patch embeddings, and position embeddings. In specific, if there are total N_C camera

IDs, the learnable side information embeddings are set to $S_C \in \mathbb{R}^{N_C \times D}$. Camera embeddings S_C are the same for all patches of an image, unlike position embeddings, which change between patches. Furthermore, if the vehicle's viewpoint is known, either through a viewpoint estimate algorithm or through human annotations, it is possible to encode the viewpoint label as $S_V \in \mathbb{R}^{N_V \times D}$ for all patches of an image, where N_V is the number of viewpoint IDs. In order to integrate two different types of information, the authors proposed to encode the camera and viewpoint together as $S_{(C,V)} \in \mathbb{R}^{(N_C \times N_V) \times D}$.

Finally, the input sequences with camera ID r and viewpoint ID q are fed into transformer layers as follows:

$$Z_0^{'} = Z_0 + \lambda S_{(C,V)}[r \times N_V + q] \tag{3.10}$$

where Z_0 is the sequence of patch embeddings from Eq. 3.1 and λ is a hyperparameter to balance the weight of SIE.

JPM was created to cope with the problem of occlusions and misalignments. The global feature of the Baseline network uses information from the entire image for V-ReID, however ReID is more difficult with only a partial observation of the vehicle. Therefore, JPM shuffles the patch embeddings and then re-group them into different parts, each containing several random patch embeddings of an entire image. This way, local features can cover patches from diverse body or vehicle parts, implying that local features hold global discriminative capability.

First, the patch embeddings (except the class token) $[z_{l-1}^1, z_{l-1}^2, ..., z_{l-1}^N]$ obtained before the last layer are shifted in m steps to become $[z_{l-1}^{m+1}, z_{l-1}^{m+2}, ..., z_{l-1}^N, z_{l-1}^1, z_{l-1}^2, ..., z_{l-1}^m]$. The shifted patches are further shuffled by the patch shuffle operation and divided in k groups. In this case, k=4, which will result in 4 local features $[f_l^1, f_l^2, f_l^3, f_l^4]$.

The concatenation of the global feature f_g with the local features results in the final output feature, which is represented by $Z_L = [f_g, f_l^1, f_l^2, f_l^3, f_l^4]$.

Finally, L_{ID} and L_T are used to train the global feature and local features. The overall multi-branch loss is computed as follow:

$$L = L_{ID}(f_g) + L_T(f_g) + \frac{1}{k}\sum_{j=1}^{k}(L_{ID}(f_l^k) + L_T(f_l^k)) \tag{3.11}$$

During inference, the global feature and local features are concatenated $[f_g, f_l^1, f_l^2, ..., f_l^k]$ as the final feature representation.

3.3 Viewpoint Estimation Network

One of the biggest challenges of V-ReID is the varying appearances of the same vehicle ID in different viewpoints. As a result, the TransReID network incorporates viewpoint information into the embeddings to improve the results; however, not all datasets provide that type of information. Zhongdao Wang et al. [34] provided vehicle orientation annotations for all of the images in the VeRi-776 Dataset, which are categorized as illustrated in Figure 3.9.

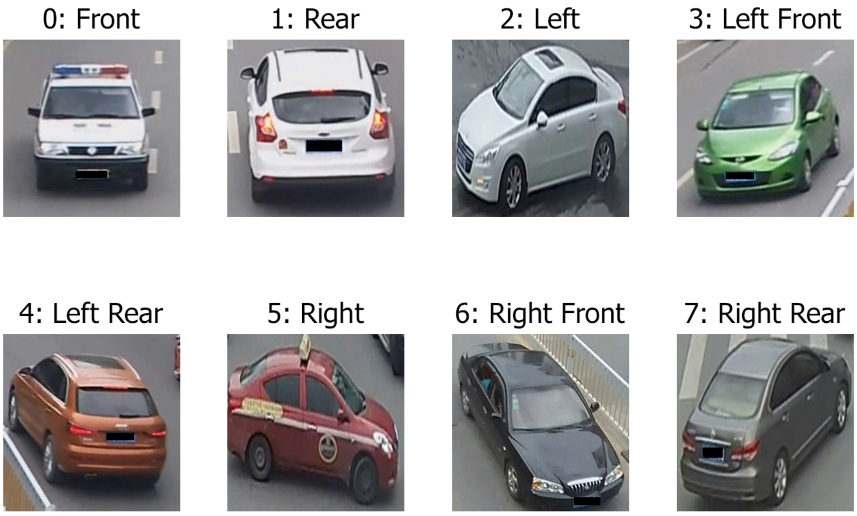


Figure 3.9: VeRi-776 Dataset Viewpoint Labels.

Since VeRi-776 Dataset have a large amount of images, a classification network was trained offline to predict the vehicle orientation using a pre-trained ResNet-18 as the backbone as seen in Figure 3.10.

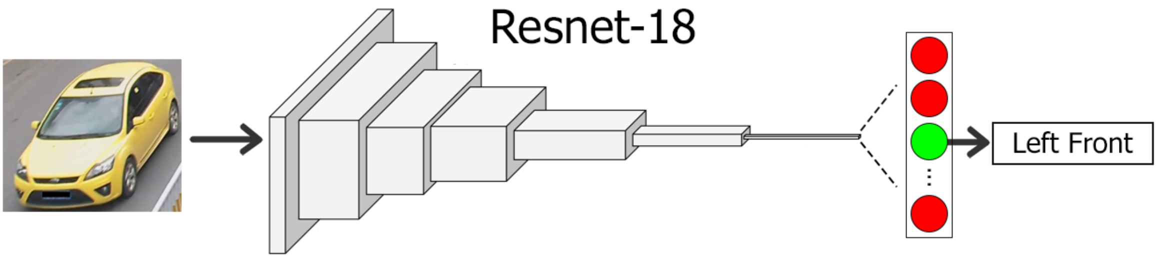


Figure 3.10: Viewpoint Estimation Network Diagram.

A brief analysis of the Veri-776 dataset's images and orientation labels showed up some annotation errors as well as some extremely similar vehicle images with different viewpoint labels, which introduced additional error into the classification network training. The labels that potentially confuse the network the most were combined together based on similarity in order to minimize errors. Therefore, the original 8 viewpoints were reduced to only 3, grouping together the following label's groups: (front, left front, right front), (rear, left rear, right rear), and (left, right). The proposed rearrangement of labels is illustrated in Figure 3.11.

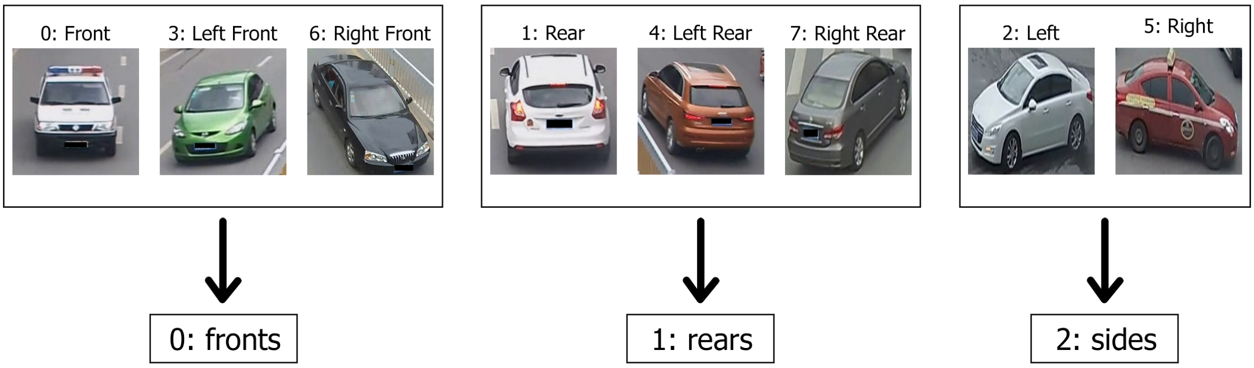


Figure 3.11: Rearrangement of viewpoint labels.

After training the classification network, the produced model was incorporated into the TransReID pipeline to automatically determine the viewpoint for each input vehicle image. To determine which model produced the best results, tests were conducted using both models with 8 and 3 viewpoint predictions.

3.4 Locally Aware Transformer Network

Despite TransReID's outstanding performance and state-of-the-art results in V-ReID, multiple efforts to improve the current network have been made. In this case, a Person ReID-specific Locally Aware Block from [17] was used to investigate its performance in V-ReID. First, the ViT Baseline network is used as backbone to extract $N+1$ features $F = [f_g, f_1, f_2, ..., f_N]$, where N is the number of patches. Then, as shown in Figure 3.12, the Locally Aware Block is added. The SIE module from TransReID is also used in this network.

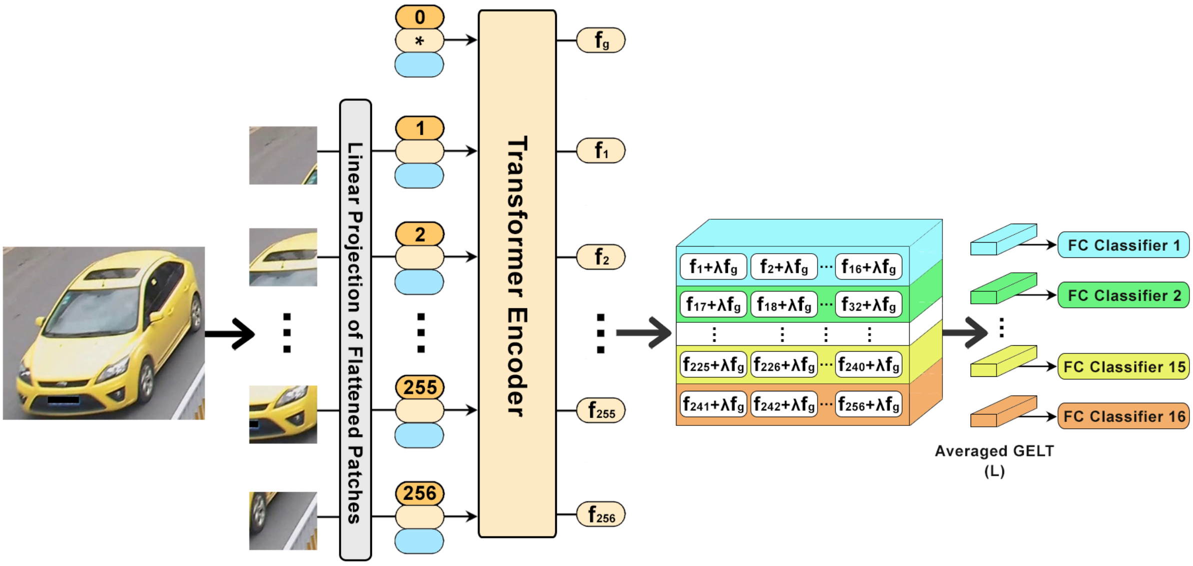


Figure 3.12: Locally Aware Transformer Network. Adapted from [17].

The transformer encoder generates $N+1$ feature vectors, with f_g representing the global token and the rest tokens ranging from f_1 to f_N representing the local tokens, which are then combined via weighted averaging and placed into a 2D spatial grid, resulting in Globally Enhanced Local Tokens (GELT). The total number of patches per row is defined as N_R, and the total number of patches per column is defined as N_C. In this case, vehicle images have a fixed input size of 256×256, therefore N_R=N_C=$\sqrt{N}$=16. Then L is defined as the averaged GELT obtained after average pooling of f_g and f as follows:

$$L_i = \frac{1}{N_R} \sum_{j=i\times N_R+1}^{(i+1)\times N_R} \frac{f_j + \lambda f_g}{1+\lambda} \qquad i = 0...N_C - 1 \tag{3.12}$$

The FC classifiers are composed of two fully connected layers separated by a ReLU and Batch Normalization. The final output y is defined as follows:

$$y_i = FC_i(L_i) \qquad i = 1...N_C \tag{3.13}$$

3.5 Momentum Adaptive Loss Weight

Most ReID studies combine an ID loss with a Metric Loss, however the ID Loss is much larger than the Metric Loss, causing an imbalance and affecting training performance, and the relationship between the two is rarely explored.

The loss weights are generally set equally, i.e. 1:1 ratio, therefore a simple module called Momentum Adaptive Loss Weight (MALW) was proposed in [11] to increase training stability by automatically updating loss weights according to the statistical characteristics of loss values between ID Loss and Metric Loss as illustrated in Figure 3.13.

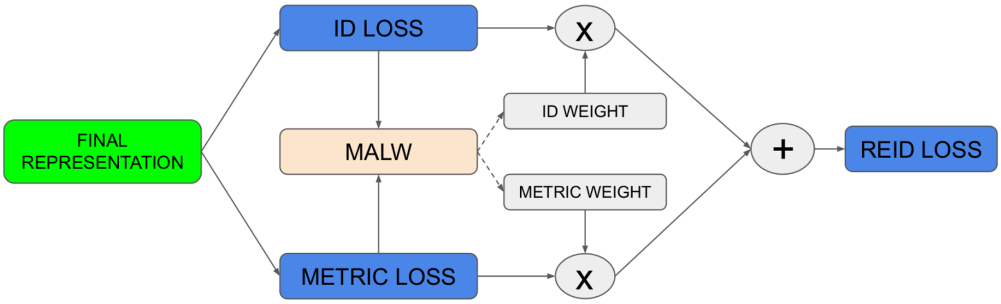


Figure 3.13: Momentum Adaptive Loss Weight (MALW). Taken from [11]

Assuming that λ_{ID} and λ_{METRIC} are the ID Loss and Metric Loss weights, respectively, the ratio between λ_{ID} and λ_{METRIC} is initially set to 1:1. After K iterations of training, the ID loss weight λ_{ID} is adjusted based on the standard deviation of the recorded ID Loss σ_{ID} and Metric Loss σ_{METRIC} with a momentum factor α, using the following equations 3.14 and 3.15.

$$\lambda_{ID_{NEW}} = 1 - \frac{\sigma_{ID} - \sigma_{METRIC}}{\sigma_{ID}} \tag{3.14}$$

$$\lambda_{ID} = \alpha\lambda_{ID} + \lambda_{ID_{NEW}} \tag{3.15}$$

The authors reported significant improvements in V-ReID when employing the MALW module and different fixed ratios between the two losses, which are addressed in the results section.

3.6 Relation Preserving Triplet Mining Loss

Inspired in [21], a Relation Preserving Triplet Mining Loss was implemented to compare its performance with the commonly used Triplet Hard-Mining Loss. According to the authors of this work, significant appearance changes caused by pose variations indicate that an object ID is made up of many natural groups, and that forcing instances from distinct groups to a single location is counter-productive. Instead of choosing a hard positive, which is the most difficult positive for the anchor and usually have an entirely different appearance, the idea is to pick a semi-hard positive that shares a natural group with the anchor. The above provides a soft positive mining, that ensures anchor-positive pairs meet the natural grouping while also assuring that the positive is significantly different from the anchor. Figure 3.14 illustrates the difference between Positive Hard-mining and Positive Soft-mining.

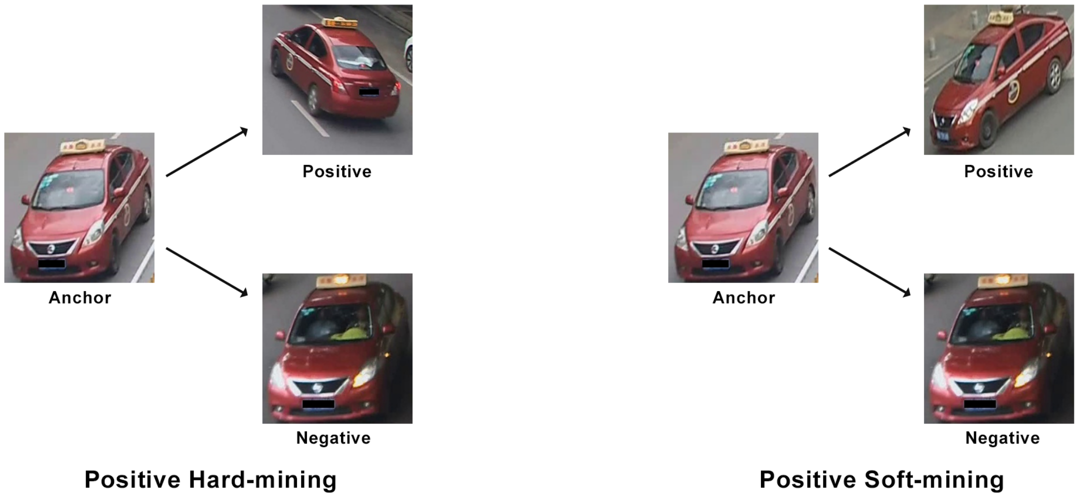


Figure 3.14: Difference between Positive Hard-mining and Positive Soft-mining.

In order to select the triplet samples, orb keypoints are extracted and a modern feature matcher GMS [22] is used between anchor-positive pair images as a relational indicator, using the following criterion: high number of matches between anchor-positive pair images means they are similar and belong to the same natural group; otherwise, they do not. However, a positive that is too similar to the anchor does not add considerable value to the triplet-cost, therefore the decision is made based on the number of matches between anchor-positive pair images.

Figure 3.15 shows the number of matches for the same vehicle ID in different situations: on the left, there are two images from different natural groups, so there are no matches; on the middle, the vehicle is shown in similar scenarios, not exactly the same but not too dissimilar, so there are 250 matches. On the right, both images are identical, resulting in approximately 5000 matches; this is only an example to demonstrate that if the images are extremely similar, the number of matches will be incredibly high.

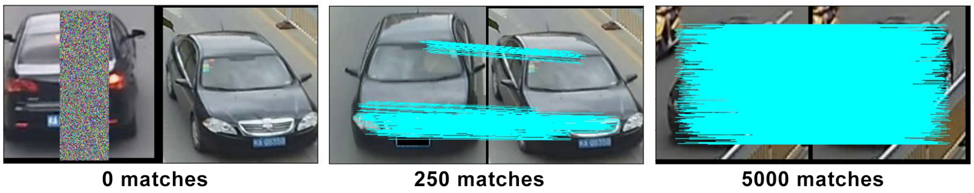


Figure 3.15: Number of matches between different image pairs.

After analyzing a large number of cases, the choice criterion for selecting the anchor-positive pair is the example whose number of matches is closest to the threshold $\tau = 200$.

3.7 Cross-ViT Network

Cross-Attention Multi-Scale Vision Transformer (Cross-ViT) for Image Classification [35] developed by Chun-Fu Chen et al. is a novel dual-branch vision transformer that combines image patches of different sizes. The popularity of Transformers as well as multi-scale feature representation, which has a long history in computer vision, served as inspiration for this study. This method processes small-patch and large-patch tokens using two separate branches of different computational complexity. These tokens are then fused using a novel token fusing module based on cross attention, which uses a single token for each branch as a query to exchange information with other branches. The Cross-ViT pipeline is illustrated in figure 3.16.

In this work, the ViT backbone was replaced by the Cross-ViT network and the V-ReID pipeline was trained on the Veri-776 dataset. In previous results, the ViT network was pre-trained on ImageNet-21k, which has around 14.2 million images. The Cross-ViT network, on the other hand, only provides a model that has been pre-trained on ImageNet-1k, which has approximately 1.2 million images. As a result, the Baseline network was trained using ViT pre-trained on ImageNet-1k to provide a base for comparison with Cross-ViT results pre-trained on the same dataset.

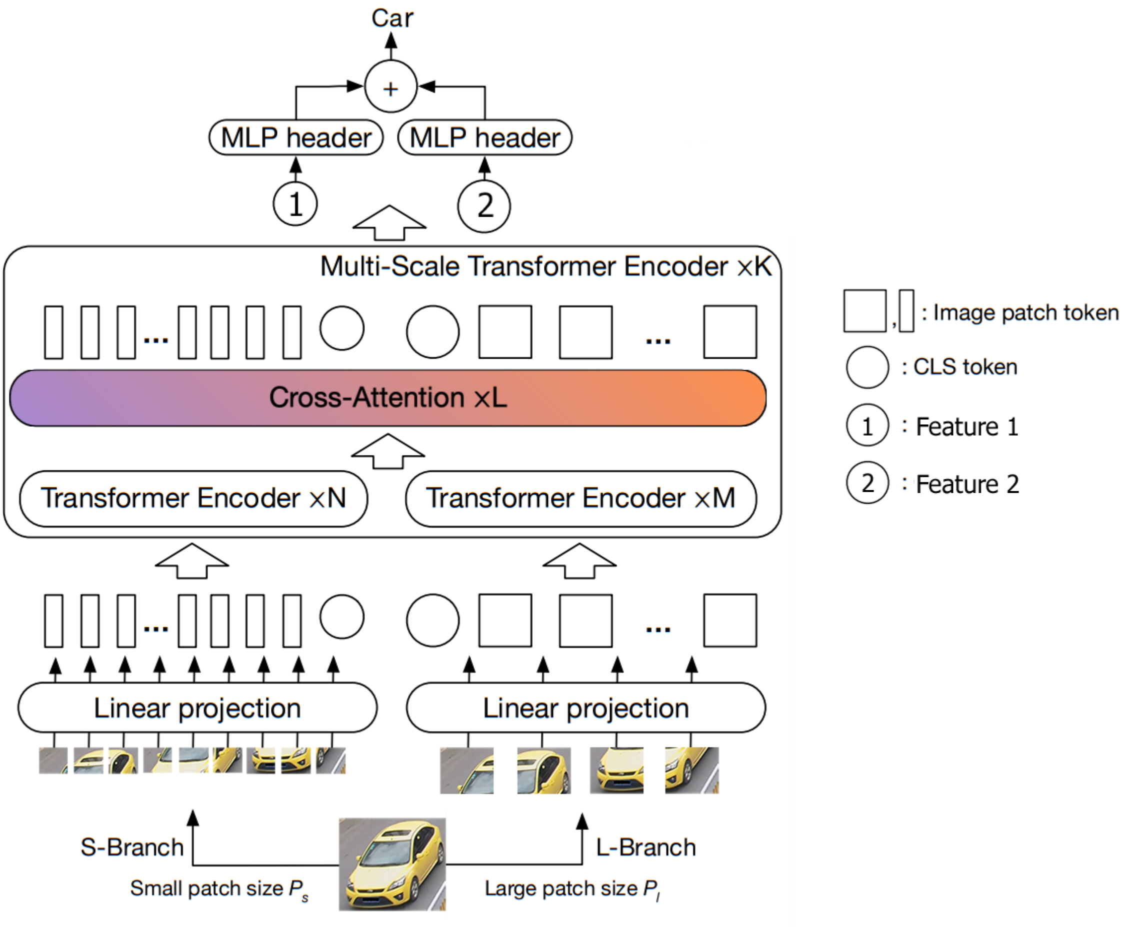


Figure 3.16: Cross-ViT pipeline. Adapted from [35].

The Cross-Attention Module is similar to Self-Attention, but two feature maps are used instead of one. The main goal is to make the CLS token from the large branch interact with the patch tokens from the small branch and the CLS token from the small branch interact with the patch tokens from the large branch. Figure 3.17 shows the Cross-Attention Module for the large branch. The small branch follows the same procedure but swaps CLS and patch tokens from another branch. $f^l()$ and $g^l()$ are projection functions to align dimensions.

Specifically, for L-Branch, it first collects the patch tokens from the S-Branch and concatenates them with its own CLS tokens to produce x'^l. The module then conducts cross-attention between x^l_{cls} and x'^l, where CLS token is the only query as the information of patch tokens are fused into CLS token. Cross-Attention (CA) can be expressed as follows:

$$\begin{gathered} q = x'^l_{cls} W_q, \qquad k = x'^l W_k, \qquad v = x'^l W_v \\ CA(Q, K, V) = Softmax\left(\frac{qk^T}{\sqrt{C/h}}\right) v \end{gathered} \tag{3.16}$$

where C is the embedding dimension, h is the number of heads, and W_q, W_k, and W_v are learnable weights.

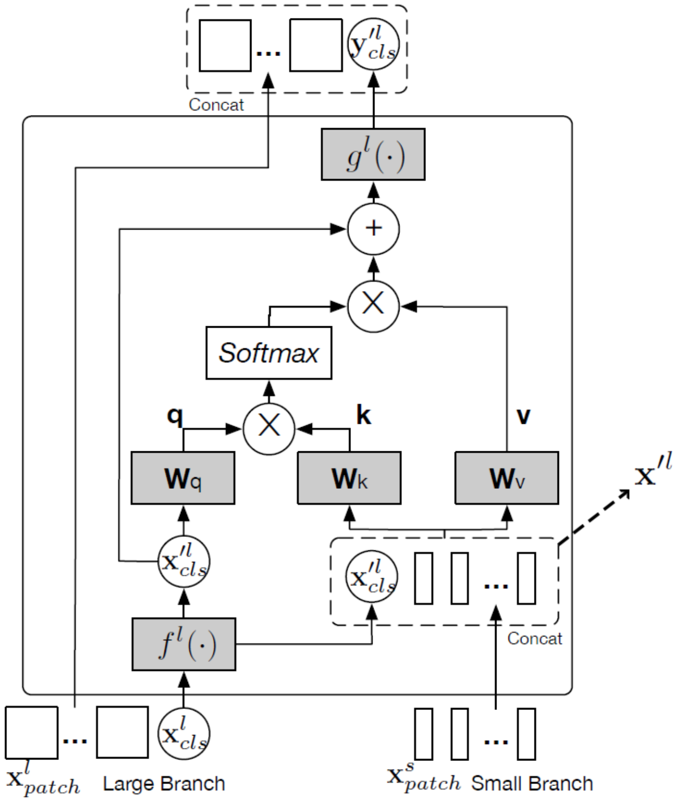


Figure 3.17: Cross-Attention Module for Large Branch. Taken from [35].

Cross-ViT produces two features, whereas ViT produces only one, therefore different fusing strategies were also explored. The first branch processes 12×12 patches, producing feature 1 with a dimension of 224. The second branch works with 16×16 patches to produce feature 2 with a dimension of 448. The following feature combinations were tested:

- A - Using only feature 1 with dimension 224.
- B - Using only feature 2 with dimension 448.
- C - Concatenating features 1 and 2 with default dimensions.
- D - Reducing the dimension of feature 2 to 224 and concatenating the features.
- E - Reducing the dimension of feature 2 to 224 and average the features.
- F - Reducing the dimension of feature 2 to 224, stacking the features and using a convolutional layer to automatically learn weights and decide which feature is more significant for V-ReID. The result of this operation is a feature with dimension 224.

Figure 3.18 shows a visual representation of the fusion strategies above-mentioned.

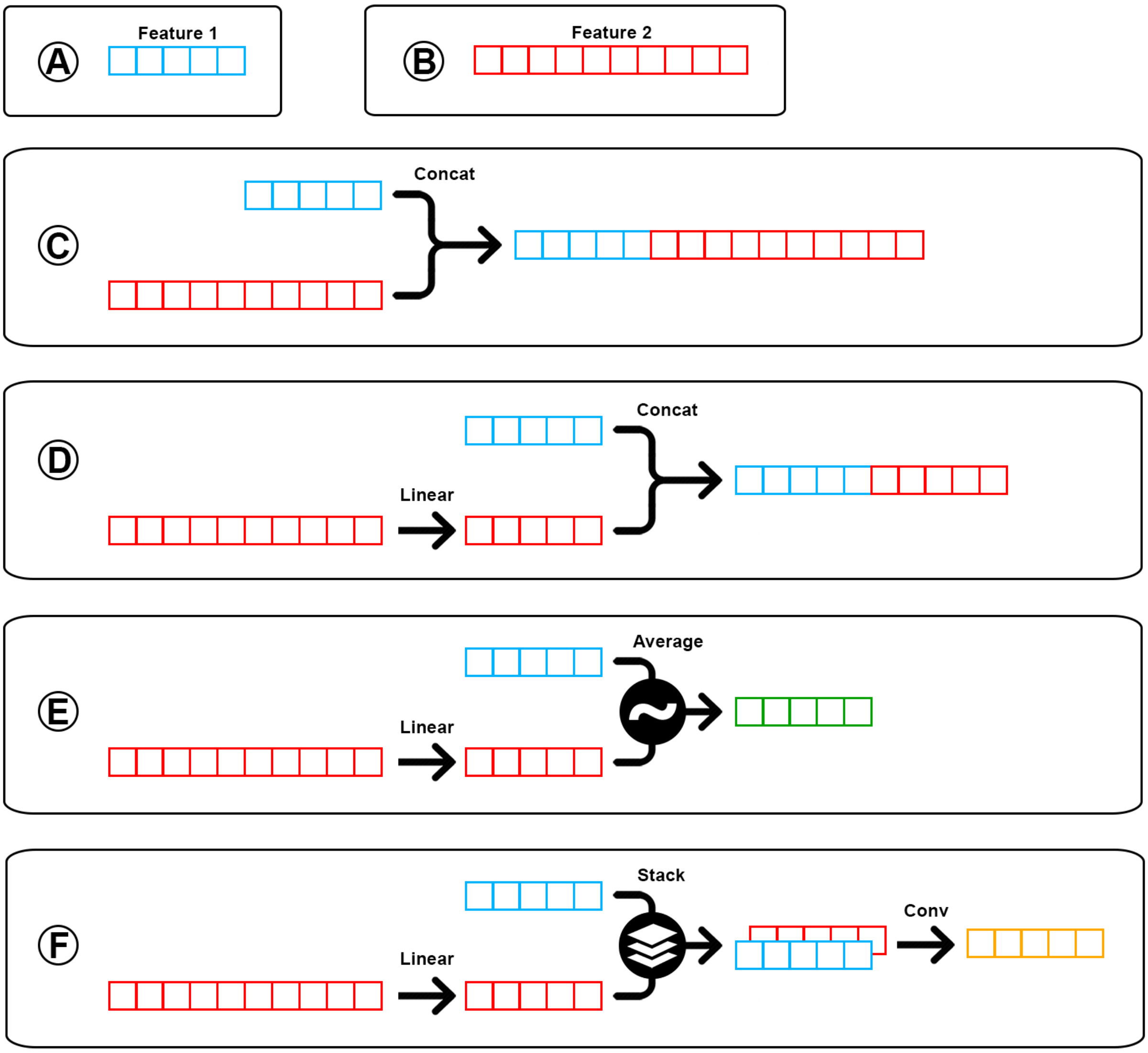


Figure 3.18: Fusion strategies used to produce the final feature.

4 Results and Discussion

This chapter presents all the relevant outcomes obtained during the development of this dissertation. Section 4.1 begins with a brief overview of the evaluation metrics that were used. Then, in section 4.2, all of the important training and testing implementation details are specified. Finally, the performance results of the proposed methodologies and more experiments are shown in section 4.3.

4.1 Evaluation Metrics

Mean Average Precision (mAP) and Cumulative Matching Characteristic (CMC) are the standard metrics for ReID evaluation that will be used to evaluate the proposed methodologies.

The final goal of V-ReID is to compute a ranking similarity list for each query image, i.e. the feature of each query will be compared with all the images in a gallery set using an euclidean distance, generating a list sorted from most similar to least similar. A larger distance means less similarity between features.

The mean average precision of the model can be computed if there are multiple ground-truths in the gallery for each query. The average precision (AP) for each query can be determined as follows:

$$AP = \frac{\sum_{k}^{n} P(k) \times R(k)}{GTP} \tag{4.1}$$

where GTP refers to the total number of ground truth positives, n refers to the total number of images in gallery set, *P(k)* refers to the precision and *R(k)* is a relevance function. The relevance function is an indicator function that returns a value of 1 if the k^{th} image belongs to the same class as the query and a value of 0 otherwise.

Finally, the mAP is simply the mean of all the queries. Denoting Q as the total number of queries, mAP is defined as follow:

$$mAP = \frac{1}{Q} \sum_{q=1}^{Q} AP(q) \tag{4.2}$$

CMC-k or Rank-k metric shows how often, on average, the correct vehicle ID is included in the best k matches against the gallery set for each query image. In case of a single query, if there is just one correct image in the 5th position in the top 10 matches, the Rank-1 is equivalent to 0%, Rank-5 is equal to 100%, and Rank-10 is equal to 100%. In the end, each Rank-k result will be the average for all the queries.

4.2 Implementation Details

The backbone network for all the proposed methodologies is ViT and its initial weights are pre-trained on ImageNet-21K and then finetuned on ImageNet-1K. All the images are normalized, resized to 256×256, and the training images are augmented with random horizontal flipping, padding, random cropping and random erasing. The networks are trained over 120 epochs with a batch size of 32. A batch consists of 8 identities, each containing 4 samples. The optimizer used is SGD with a momentum of 0.9, weight decay of 1e-4 and an initial learning rate of 0.008 with cosine learning rate decay. These are the parameters used unless otherwise mentioned.

Despite the fact that all networks are trained for 120 epochs, the best model that achieves a superior result in the test set can be obtained before the last epoch. Therefore, after epoch 100 an inference is performed every epoch and all models are saved to ensure that the best model is always obtained.

The exact same training and testing sets are used in every run due to a random seed that is always set to the same value. This is useful for comparing different models always under the same conditions and reproducing the results.

All the networks were trained using PyTorch and one of the following GPUs: Nvidia TITAN X or Nvidia RTX 3090.

4.3 Validation Results

4.3.1 Baseline Network Results

Initially, the Baseline network was trained on Veri-776 and Veri-Wild datasets in order to establish a Baseline for comparison with upcoming results. The Baseline network is a simple network in which only the global feature output generated by the ViT backbone is used for V-ReID. Since it serves as the foundation for the subsequent most sophisticated approaches, the initial stage also included certain parameter adjustments.

The Veri-776 dataset is relatively small, with about 38 000 training images, while Veri-Wild has nearly 280 000 training images. Due to the longer training times for the Veri-Wild dataset, it was required to create smaller training sets, specifically with 10% and 20% of the total images. However, most of the parameter adjustments were performed on the Veri-776 dataset for more reliable outcomes, since the entire dataset was trained.

After training on Veri-776 and Veri-Wild datasets over 120 epochs, the models achieved 99.8% and 99.9% accuracy in the training set, respectively. The following figures 4.1 and 4.2 show the loss and accuracy plots in training. Although it seems that the model trained on Veri-776 (figure 4.1) stopped learning around epoch 40, the validation results and the up-scaled graph from epoch 40 to 120 revealed the opposite. An inference is made every 20 epochs until epoch 100, at which point the inference is made every epoch until epoch 120, and here is when the best model with the best validation results is always obtained, showing the importance of training across all of the 120 epochs. The loss of the model trained on Veri-Wild (figure 4.2), on the other hand, takes significantly longer to converge because of the larger size of the dataset.

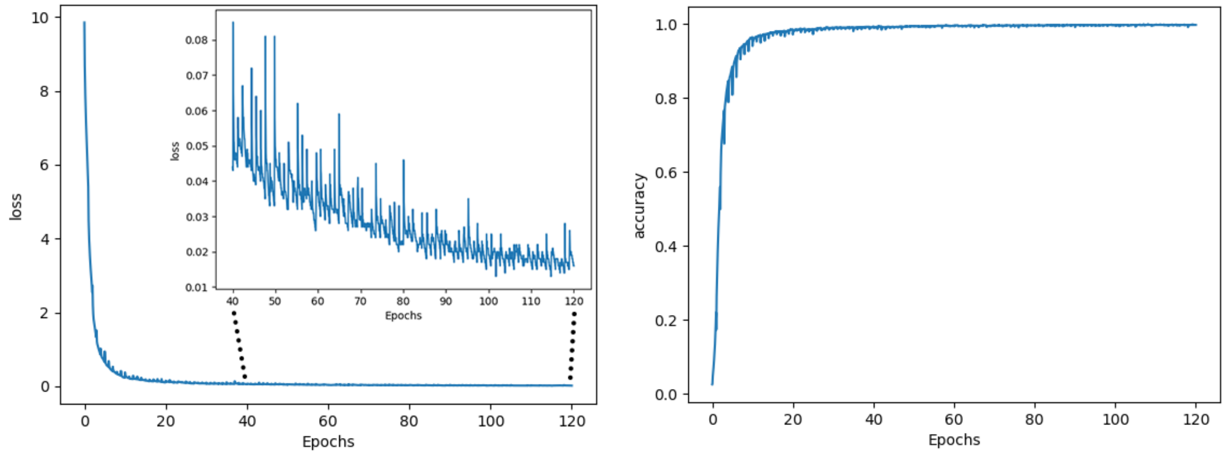


Figure 4.1: Baseline network Loss and Accuracy over 120 epochs on the Veri-776 dataset.

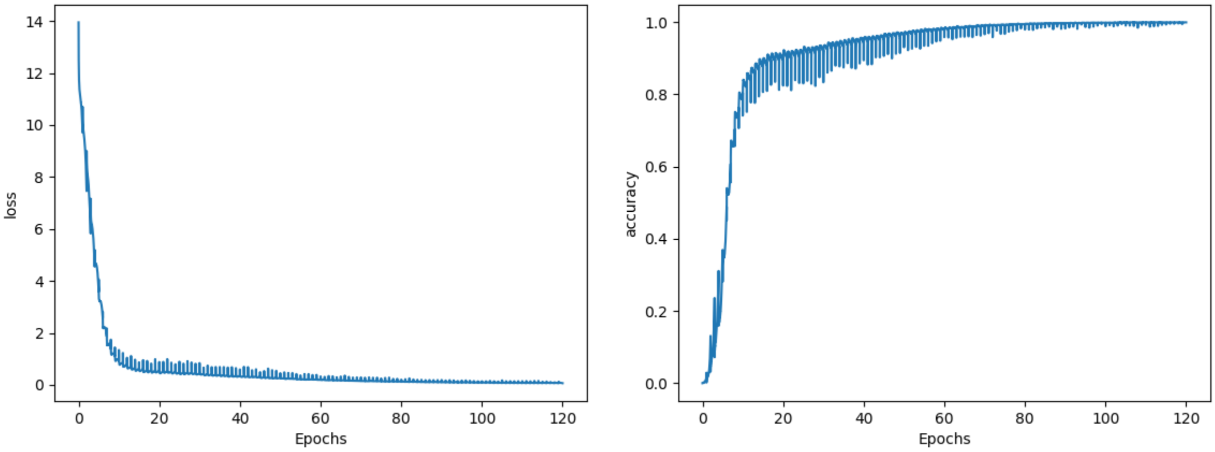


Figure 4.2: Baseline network Loss and Accuracy over 120 epochs on the Veri-Wild dataset.

The achieved results with the Baseline network for the Veri-776 and Veri-Wild datasets are presented in table 4.1 and table 4.2, respectively.

Method	mAP (%)	Rank-1 (%)	Rank-5 (%)
Baseline	77.8	96.2	98.5
Baseline + RR	88.0	97.1	97.8

Table 4.1: Performance results of the Baseline network on the Veri-776 dataset.

	Test Size = 3000			Test Size = 5000			Test Size = 10000		
Method	mAP (%)	Rank-1 (%)	Rank-5 (%)	mAP (%)	Rank-1 (%)	Rank-5 (%)	mAP (%)	Rank-1 (%)	Rank-5 (%)
Baseline	71.4	86.4	95.9	64.1	81.8	93.6	54.2	76.0	90.1
Baseline + RR	72.8	82.6	93.5	64.9	77.8	81.0	-	-	-

Table 4.2: Performance results of the Baseline network on the Veri-Wild dataset.

The reported results are a good starting point and even superior to many existing approaches, given that the state-of-the-art mAP scores are 82.1% and 83.1%, respectively, for the Veri-776 and Veri-Wild datasets. Given that all parameter adjustments were made on the Veri-776 dataset and because of the dataset's increased complexity, it was previously expected that the performance on the Veri-Wild dataset would be slightly worse.

As can be seen, the re-ranking (RR) post-processing technique mentioned in section 2.6 significantly enhances the results on Veri-776. However, if the initial ranking list is poor, this technique may not perform well and might potentially make the outcomes worse. Thus, it must be used with caution. Another disadvantage is that it is time-consuming, but for applications where time is not a crucial factor, it is beneficial in terms of performance. This technique is also highly computationally demanding, hence it was not possible to assess it on the largest Veri-Wild test set. However, because the initial list was poor, no good results with re-ranking were expected.

Figure 4.3 shows a few examples of Grad-CAM attention maps that were generated in order to better understand what the network is capturing from each vehicle to perform the V-ReID task. The network is focusing more on the areas highlighted in red. Two images are displayed side by side for each vehicle example; the left image is the original, and the right image is the resized version that has been fed into the network with the attention map superimposed. In the first row of images, it is possible to see that the network is focusing on distinctive car features like wheels, rearview mirrors, headlights, and even a text at the top front of the truck. However, it is also possible to notice some less explicit examples in the second row, where the network concentrates on areas outside of the vehicle region.

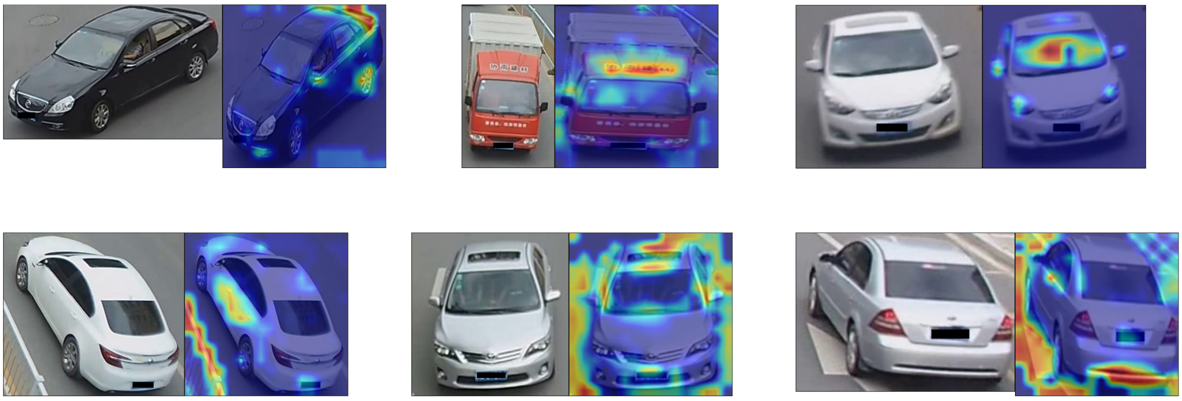


Figure 4.3: Examples of some Grad-CAM attention maps obtained with the Baseline network on the Veri-776 dataset.

For each query image, a ranking list is created with all the images of the gallery set. The top 10 ranking results for some query images can be seen in figure 4.4. In general, the top 10 gallery images retrieved are true positives, but in some situations, such as the last row, the network got most of the images wrong. However, the false positives are very similar to the query, and the query image is partially occluded and has some reflections on the windows, making the ReID more difficult.

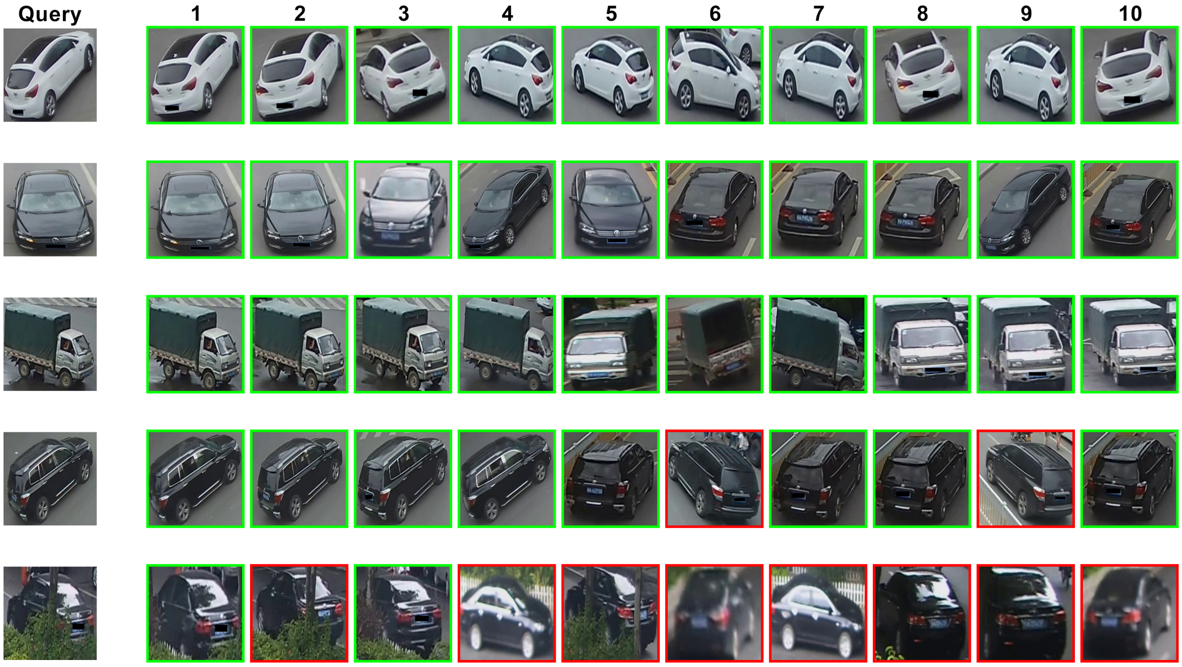


Figure 4.4: Top 10 ranking results using the Baseline network. Each row presents the query images and retrieved top 10 gallery images. Green and red boxes denote true positive and false positive samples, respectively.

4.3.2 TransReID Network Results

The second stage of this work was to enhance the results obtained with the Baseline network in the previous section. With this in mind, the TransReID Network was created by adding two more modules to the Baseline network in order to improve the performance of V-ReID.

To summarize, the primary changes between this network and the Baseline network are the use of Overlapping patches (OP), the Jigsaw Patch Module (JPM), and the Side Information Embeddings (SIE) module. Several tests were performed to analyze the influence of each module independently, as shown in table 4.3.

Method	OP	JPM	SIE	mAP (%)	Rank-1 (%)	Rank-5 (%)
Baseline				77.8	96.2	98.5
	✓			79.4	96.9	98.9
		✓		79.5	96.5	98.3
			✓	79.8	97.2	99.0

Table 4.3: The effects of different TransReID modules.

Since the SIE module considers information about the camera and the vehicle's perspective, an ablation experiment was conducted to examine the impact of these elements separately. These results are presented in table 4.4.

Method	Camera	Viewpoint	mAP (%)	Rank-1 (%)	Rank-5 (%)
Baseline			77.8	96.2	98.5
	✓		78.9	96.2	98.5
		✓	77.8	96.5	98.5

Table 4.4: Ablation study of SIE module.

Finally, the achieved results with the TransReID network for the Veri-776 and Veri-Wild datasets are presented in table 4.5 and table 4.6, respectively.

Method	mAP (%)	Rank-1 (%)	Rank-5 (%)
Baseline	77.8	96.2	98.5
TransReID	82.1	97.3	98.8
TransReID + RR	91.1	97.7	98.5

Table 4.5: Performance results of the TransReID network on the Veri-776 dataset.

	Test Size = 3000			Test Size = 5000			Test Size = 10000		
Method	mAP (%)	Rank-1 (%)	Rank-5 (%)	mAP (%)	Rank-1 (%)	Rank-5 (%)	mAP (%)	Rank-1 (%)	Rank-5 (%)
Baseline	71.4	86.4	95.9	64.1	81.8	93.6	54.2	76.0	90.1
TransReID	81.1	92.5	97.4	75.3	89.9	96.7	66.4	85.1	94.2
TransReID + RR	81.3	89.7	95.4	75.0	85.5	94.0	-	-	-

Table 4.6: Performance results of the TransReID network on the Veri-Wild dataset.

As seen in table 4.3, all three additional modules improved the Baseline outcome when used separately. The SIE module was then analyzed more deeply to discover which parameter had the largest effect, and it was observed that the camera information had the greatest impact on the outcome. Although viewpoint information had minimal influence when employed alone, it produced the best performance when combined with camera information.

The increase in complexity and the use of overlapping patches which, resulted in more patches and hence bigger features, increased the training time by approximately 30% when compared to the Baseline. However, the TransReID network significantly outperforms the Baseline results in both datasets, demonstrating the efficacy of the employed modules. The Veri-Wild dataset does not provide orientation labels, therefore the network was trained only using camera embeddings information in the SIE module.

In practice, SIE can be extended to encode more types of information, such as categorical and numerical data. The color of vehicles is information that is available in most of the datasets and could be used in SIE; however, the Transformer network must already consider this information. To verify this, the Veri-776 test set images were converted to grayscale with three channels, yielding 59.6% mAP, which is more than 20% lower than the best result, proving that color is a crucial attribute for extracting discriminative features in Transformer networks. As a result, there is no need to include this information in the SIE module.

Figure 4.5 depicts the Grad-CAM attention maps for the same vehicle examples as in figure 4.3 but using the TransReID network. The differences between the Grad-CAM attention maps produced by the Baseline and TransReID networks can be more easily seen in Figure 4.6. There is definitely an improvement in the activation maps, as well as more discriminating attention on particular vehicle elements, proving the efficacy of the JPM and SIE modules once again.

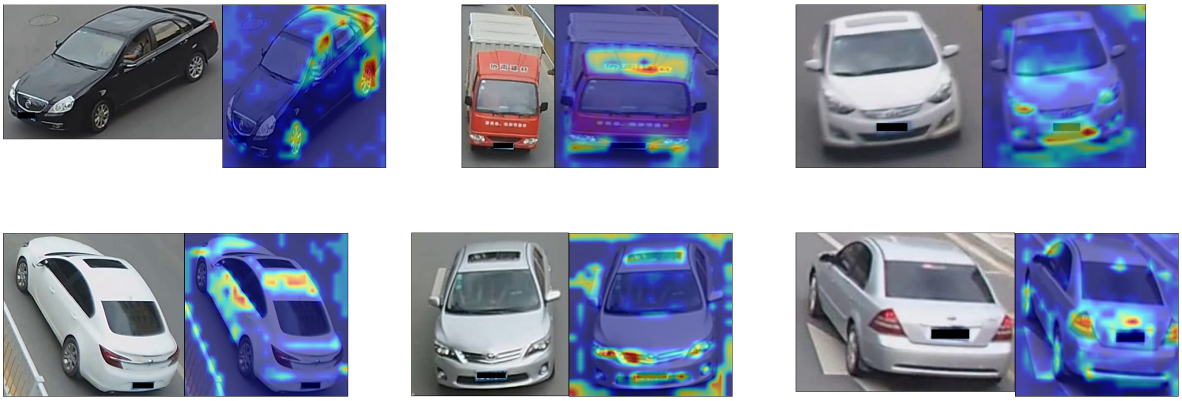


Figure 4.5: Examples of some Grad-CAM attention maps obtained in the validation with TransReID network in Veri-776 dataset.

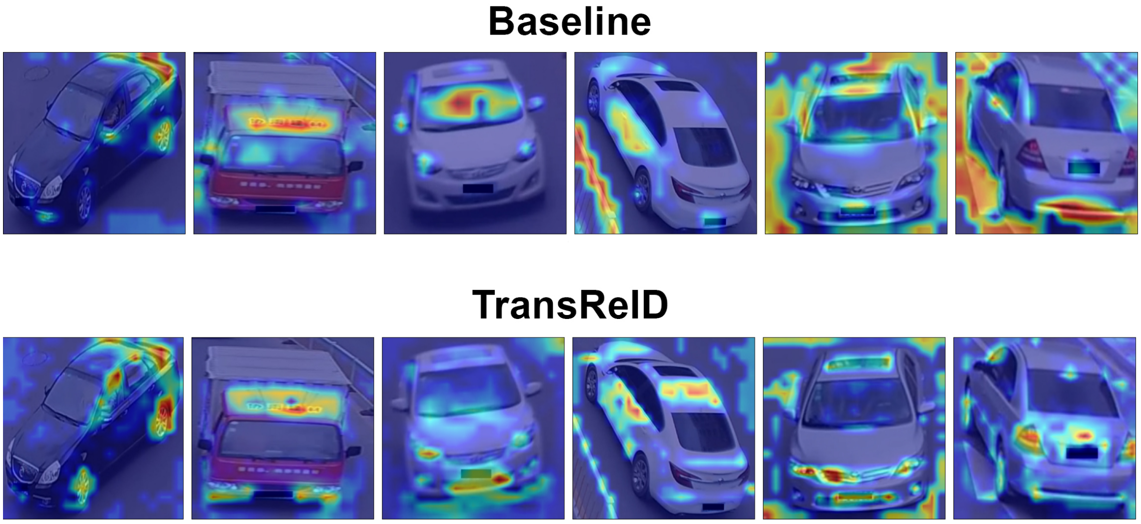


Figure 4.6: Examples of Grad-CAM attention maps obtained on the Veri-776 dataset using the Baseline and TransReID networks.

Figure 4.7 depicts the top 10 ranking results for some query images. The same vehicle queries as in Figure 4.4 were used to compare the results to the Baseline network. As can be observed, the TransReID network considerably improved the number of true positives in the last two rows.

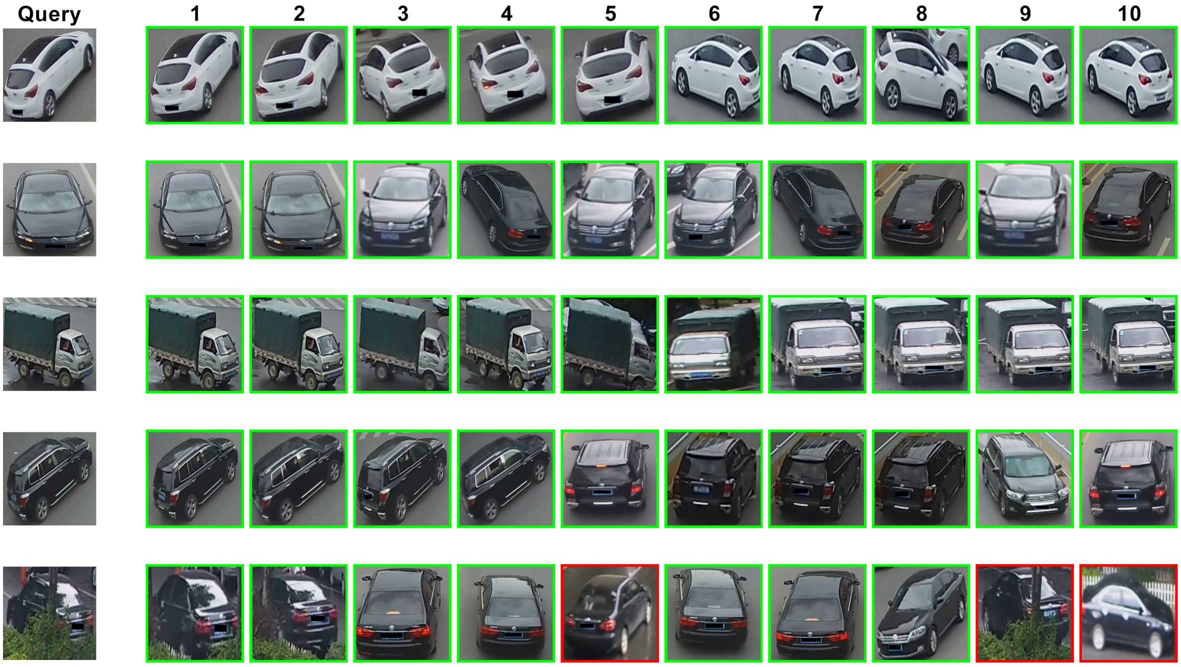


Figure 4.7: Top 10 ranking results using the TransReID network. Each row presents the query images and retrieved top 10 gallery images. Green and red boxes denote true positive and false positive samples, respectively.

4.3.3 Viewpoint Estimation Network Results

The Veri-776 dataset has vehicle viewpoint annotations that were annotated by Zhongdao Wang et al. for the proposed work [34]. However, the Veri-Wild dataset, like many other well-known datasets, lacks information regarding the vehicle's orientation.

Since the vehicle's viewpoint proved to be a benefit for V-ReID in the TransReID network when used jointly with camera information, it motivated the development of a classification network that receives vehicle images and predicts their orientation.

To accomplish that goal, a pre-trained Resnet-18 network was trained using the Veri-776 training set. Padding, random cropping, and random erasing were applied to the training images, which were resized to 256×256. The network was trained over 10 epochs with a batch size of 32. Adam was used as the optimizer, with an initial learning rate of 0.001 and a weight decay of 0.001. A ReduceLROnPlateu scheduler with patience of 1 was used to reduce the learning rate throughout the epochs.

The initial strategy was to use the Veri-776 dataset to train the network to predict 8-class orientations. In the test set, the first approach produced an 87% mAP, which is not optimal. The detailed results for each class is shown in table 4.7 and figure 4.8.

	AP (%)	Nº Images
Class 0 (front)	73	1262
Class 1 (rear)	79	826
Class 2 (left)	67	65
Class 3 (left front)	90	3021
Class 4 (left rear)	95	3202
Class 5 (right)	81	256
Class 6 (right front)	77	1172
Class 7 (right rear)	85	1176
mAP (%)	87	

Table 4.7: Classification results for 8-classes.

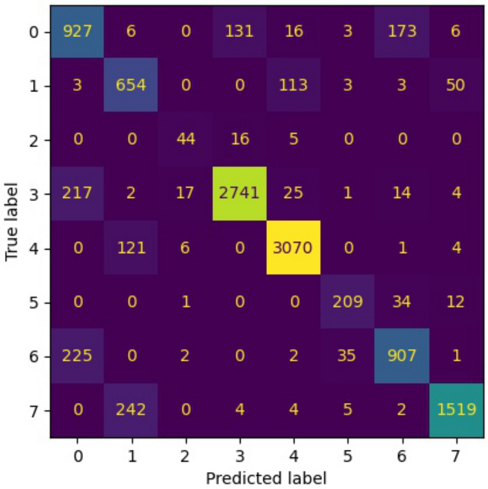


Figure 4.8: Confusion Matrix for 8-classes.

Many efforts were made to increase the classification mAP result, such as several parameter adjustments and the use of a Resnet-50, but there were no major changes. Following a brief inspection of the annotations, it was discovered that some of them are inaccurate or may confuse the networks to learn properly. Figure 4.8 shows the confusion matrix for the obtained results, which shows that the network usually confuses the label groups: (front, left front, right front) and (rear, left rear, right rear), which are very similar in most cases. In addition, there is a significant imbalance in the number of images available for each class.

The solution to this issue was to group several labels that, due to their similarities, could induce confusion in the network. The proposed rearrangement of labels is illustrated in Figure 3.11.

After rearranging the viewpoint labels, a Resnet-18 was trained under the same conditions as before, yielding a result of 98.1% mAP, which is considerably better than the result with 8-classes. The detailed results for each class is shown in table 4.8 and figure 4.9.

	AP (%)	Nº Images
Class 0 (fronts)	97	5455
Class 1 (rears)	99	5804
Class 2 (sides)	82	321
mAP (%)	98.1	

Table 4.8: Classification results for 3-classes.

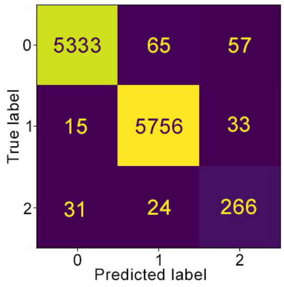


Figure 4.9: Confusion Matrix for 3-classes.

These classification networks were then used to predict vehicle viewpoints for the V-ReID network. The Veri-Wild dataset was trained for the first time using viewpoints information for the SIE module, and the Veri-776 dataset, despite having its own annotations, was also trained with the predicted viewpoint labels to see if it was still better than without. Tables 4.9 and 4.10 show the results for Veri-776 and Veri-Wild datasets, respectively.

Method	mAP (%)	Rank-1 (%)	Rank-5 (%)
Transreid	82.1	97.3	98.8
TransReID + 8V	81.7	97.3	98.6
TransReID + 3V	82.1	97.0	98.7

Table 4.9: Performance results of the TransReID model on the Veri-776 dataset using the 8-class (8V) and 3-class (3V) models to predict the viewpoints.

Method	Test Size = 3000 mAP (%)	 Rank-1 (%)	 Rank-5 (%)	Test Size = 5000 mAP (%)	 Rank-1 (%)	 Rank-5 (%)	Test Size = 10000 mAP (%)	 Rank-1 (%)	 Rank-5 (%)
TransReID	81.1	92.5	97.4	75.3	89.9	96.7	66.4	85.1	94.2
TransReID + 8V	80.5	92.6	97.8	74.8	90.0	96.4	65.9	84.9	94.3
TransReID + 3V	81.4	92.7	97.9	75.7	90.2	96.7	66.8	85.7	94.5

Table 4.10: Performance results of the TransReID model on the Veri-Wild dataset using the 8-class (8V) and 3-class (3V) models to predict the viewpoints.

As can be seen, using the model to predict the 8 views yields unsatisfactory results; nonetheless, this was to be expected given the model's low classification accuracy. However, when using the 3-class model, a better performance was obtained on the Veri-Wild dataset that had previously been evaluated without the viewpoint information, and the results in the Veri-766 dataset remained nearly identical to the results with the given viewpoints, indicating that using the predicted viewpoints is beneficial to using no viewpoint information.

4.3.4 Locally Aware Transformer Network Results

The Locally Aware Transformer network is utilized in Person ReID and also employs the ViT as a backbone but uses the output tokens differently from TransReID. The results obtained with the Locally Aware Block can be observed in table 4.11.

Method	mAP (%)	Rank-1 (%)	Rank-5 (%)
Baseline	77.8	96.2	98.5
LA-Transformer without OP	78.6	96.6	98.4
*LA-Transformer without OP	78.8	96.4	98.2
*LA-Transformer with OP	80.6	96.5	98.6

Table 4.11: Performance results of the LA-Transformer network on the Veri-776 dataset.

The star * in the superscript indicates that the classifiers employed differ from those in the original paper. Instead of using two linear layers separated by a ReLU, only one linear layer was employed for classification, which enhanced the network's performance.

As can be seen, the obtained results were better than the Baseline but significantly lower than TransReID. This result suggests that networks created specifically for Person ReID do not perform well in V-ReID. In fact, Person ReID is an easier challenge because, in most cases, the perspective has little effect on a person's appearance. With that in mind, this network was evaluated in a simpler scenario, in which the network was trained and tested using vehicle images with a single perspective. In this case, the Veri-776 dataset was used with only front-oriented vehicle images according to the labels. The results of this study are presented in table 4.12.

Method	mAP (%)	Rank-1 (%)	Rank-5 (%)
TransReID	93.1	96.8	99.2
LA-Transformer	93.8	97.6	99.2

Table 4.12: Performance results of the TransReID network versus the LA-Transformer network in a subset of the Veri-776 dataset using only front-oriented vehicle images.

The LA-Transformer network performed poorly in the multiple-views scenario; but, when evaluated simply in one orientation scenario, the LA-Transformer network outperformed the TransReID network.

Figure 4.10 shows a few examples of Grad-CAM attention maps using the LA-Transformer network. As can be seen, the network focuses on the most distinguishing features of each vehicle.

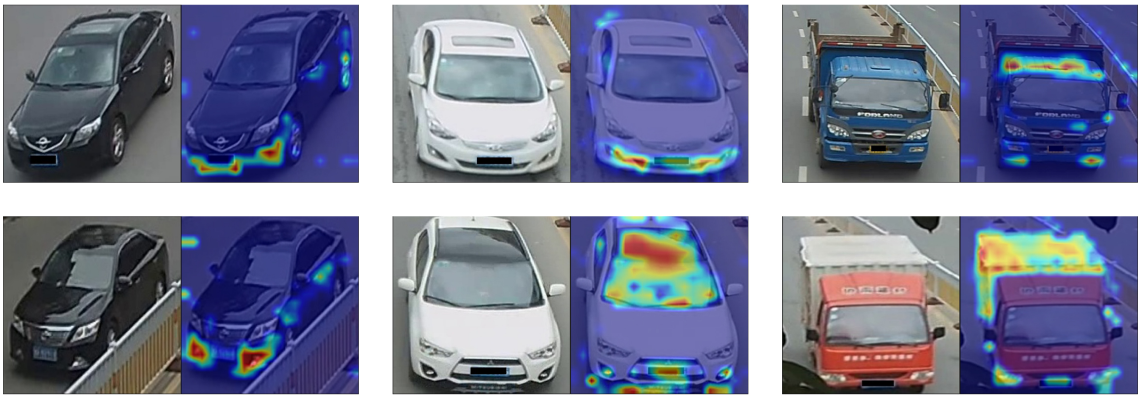


Figure 4.10: Examples of Grad-CAM attention maps obtained on the Veri-776 dataset using the LA-Transformer network.

The top 10 ranking results for some query images using the TransReID and LA-Transformer networks trained only with front-oriented vehicle images can be seen in figure 4.11.

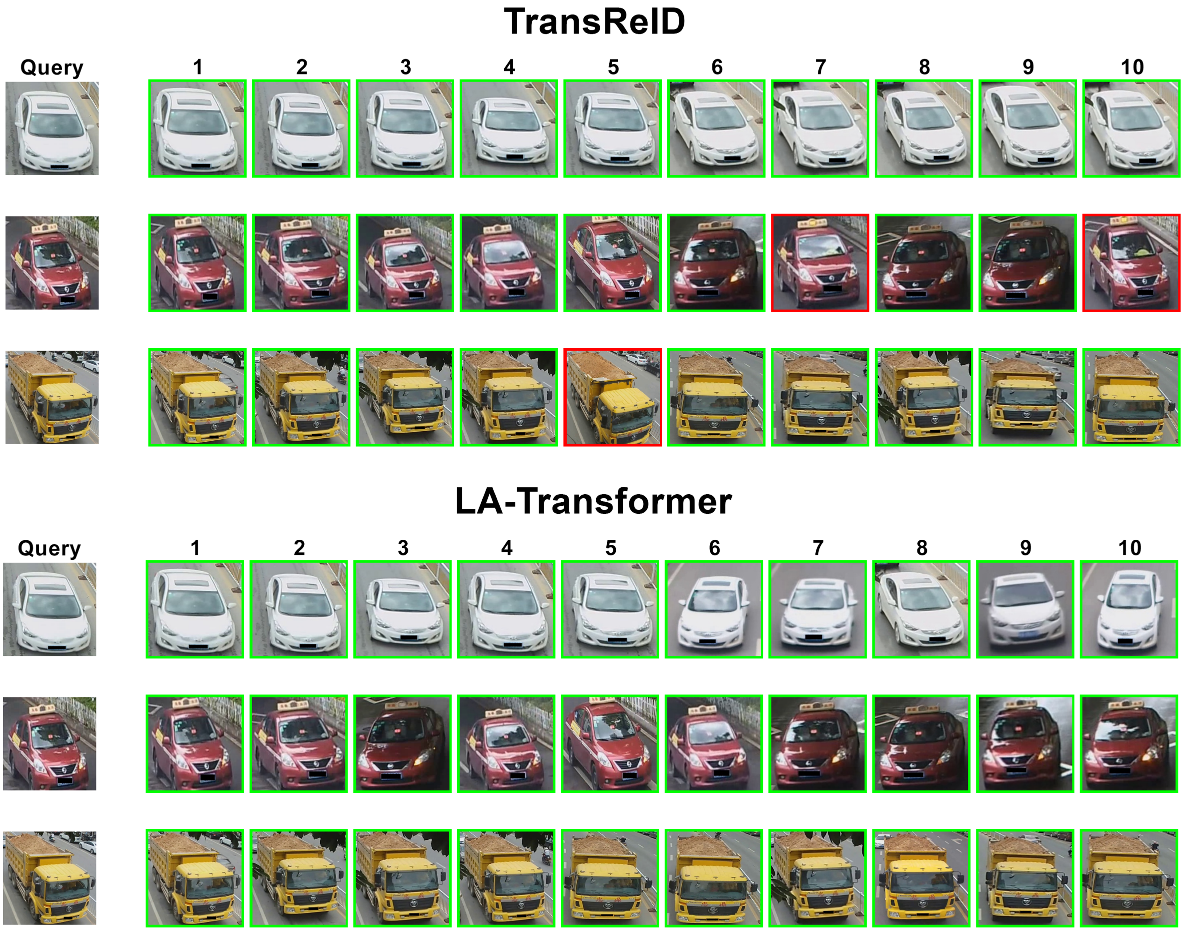


Figure 4.11: Top 10 ranking results using the TransReID and LA-Transformer networks. Each row presents the query images and retrieved top 10 gallery images. Green and red boxes denote true positive and false positive samples, respectively.

Since this is a simpler problem, both networks perform well, and the top 10 retrieved gallery images are usually true positives. However, as seen in the second and third query examples, the LA-Transformer network outperforms the TransReID network.

4.3.5 Loss Weight Ratios Results

The loss functions play an important role in network training optimization, and it is well known that Cross Entropy Loss is significantly larger than Triplet Loss. Nevertheless, the majority of V-ReID studies use a loss weight ratio of 1:1, as do all of the experiments below. Following the literature study [11], multiple experiments were conducted to identify the effect of loss weights on the final V-ReID outcomes.

First, various fixed ratios were tested, including 1:2 and 0.5:0.5, which produced better results than a 1:1 ratio, according to the paper. Next, the Momentum Adaptive Loss Weight (MALW) module was incorporated into the TransReID network. Figure 4.12 displays the loss weights behavior while using the MALW module and Table 4.13 presents the obtained results with different loss weight ratios.

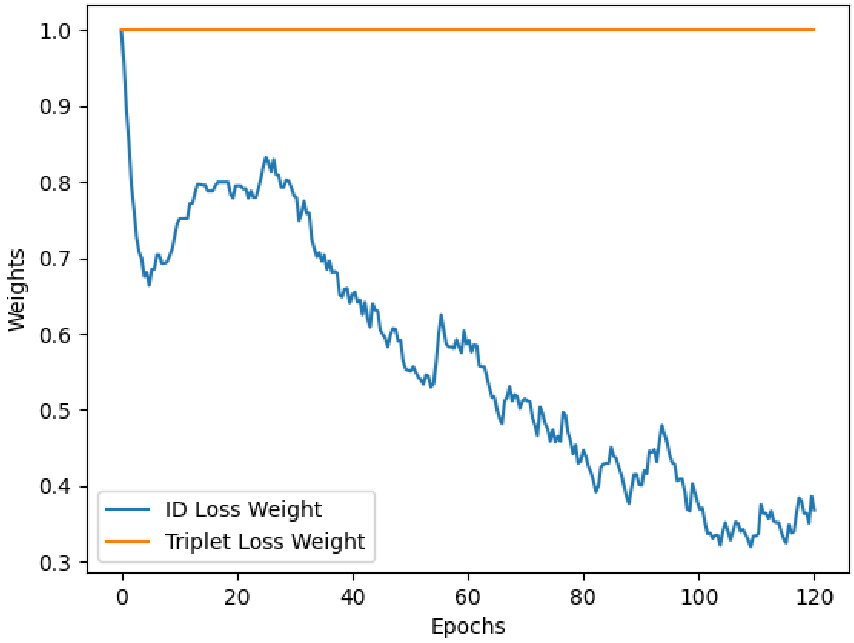


Figure 4.12: Loss weights over the epochs with the MALW module.

The loss weights were both set to 1 initially, and the MALW module had a significant impact on the ID loss weight, which decreased over epochs. The Triplet Loss weight, on the other hand, remained constant with the initial value over all epochs. The evolution of the weights over the epochs makes sense; since the ID Loss is superior, the goal is to give it a lower weight so that both losses contribute to the final loss in a more equal proportion.

Ratio	mAP (%)	Rank-1 (%)	Rank-5 (%)
1:1	82.1	97.3	98.8
1:2	82.1	97.1	98.6
0.5:0.5	82.0	97.2	98.7
MALW	82.0	96.6	98.7

Table 4.13: Performance results with different loss weights strategies on Veri-776 dataset.

As can be seen, all of the experiments with different ratios and the MALW module produced slightly worse results, which contradict those reported in the article. This also shows that the weight given to losses has a minimal impact on the final results. Therefore, the 1:1 ratio, which was used in all previous experiments, produced the best outcomes.

4.3.6 Triplet Soft-Mining Loss Results

Triplet Loss Function is essential for V-ReID, therefore one of the experiments was to replace the Triplet Hard-Mining Loss with the Triplet Soft-Mining Loss. Figure 4.13 shows the visual difference between Hard-Positive Mining and Soft-Positive Mining.

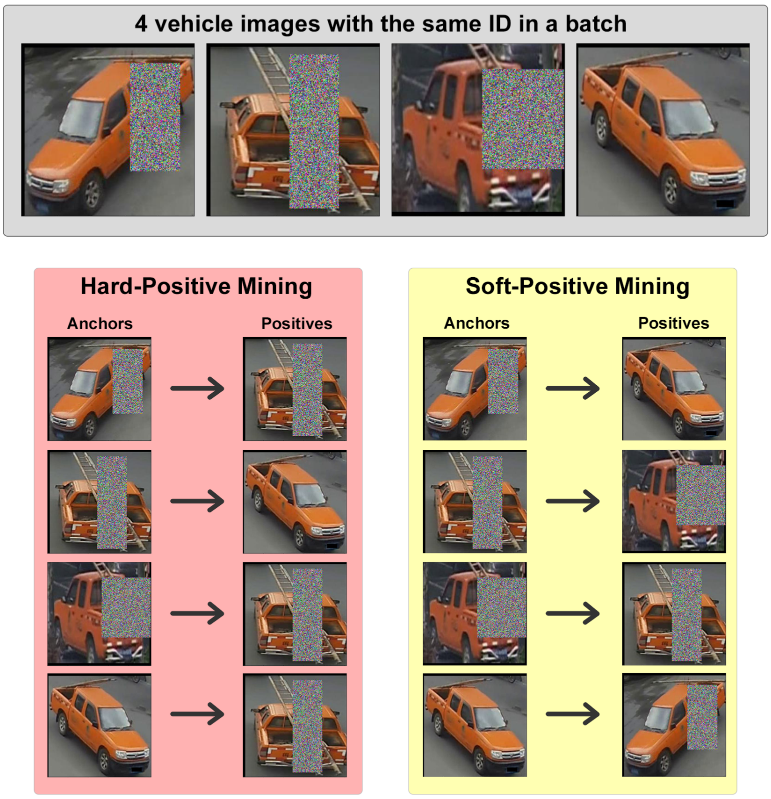


Figure 4.13: Difference between the Hard-Positive Mining and Soft-Positive Mining.

The last figure shows the anchor-positive choices with both losses during training on the Veri-776 for a specific vehicle ID within a batch. As can be seen, while Hard-Positive Mining always selects the hardest positive for the anchor, which usually has a completely different perspective, Soft-Positive Mining selects a positive that is more similar but not completely equal. This shows that the implemented loss is working correctly.

After incorporating this loss into the TransReID network, the training loss started lower than usual, as expected given the lower distances between anchor-positive pairs. The loss, however, converged to the same as with Hard-Positive Mining, and the final result remained exactly the same as in table 4.5. In conclusion, the Triplet Soft-Mining Loss had no effect on the current results.

4.3.7 Cross-ViT Network Results

Cross-ViT is a network that generates two features: the first branch produces feature 1 with a dimension of 224, while the second branch produces feature 2 with a size of 448. The triplet loss requires only one feature to conduct V-ReID, therefore the following feature fusion strategies were tested to achieve the best outcome. The final results are shown in table 4.14.

- A - Using only feature 1 with dimension 224.
- B - Using only feature 2 with dimension 448.
- C - Concatenating features 1 and 2 with default dimensions.
- D - Reducing the dimension of feature 2 to 224 and concatenating the features.
- E - Reducing the dimension of feature 2 to 224 and average the features.
- F - Reducing the dimension of feature 2 to 224, stacking the features and using a convolutional layer to automatically learn weights and decide which feature is more significant for V-ReID. The result of this operation is a feature with dimension 224.

Method	mAP (%)	Rank-1 (%)	Rank-5 (%)
A	69.9	94.1	97.2
B	70.7	93.9	97.8
C	72.4	93.3	97.4
D	71.4	93.7	97.6
E	71.9	93.6	97.2
F	72.8	95.1	97.6

Table 4.14: Performance results using Cross-ViT network with different feature fusing strategies on the Veri-776 dataset.

As observed, when only one feature is used, feature 2 outperforms feature 1, but using both features is always beneficial because the results are higher. When both features are used, the optimal fusion method for generating a more discriminating final feature for V-ReID is to use a convolutional layer that learns the best weights for each feature vector during training. This method yielded 72.8% mAP in the Veri-776 dataset, which was lower than all previous results; however, the Cross-ViT network was pre-trained on ImageNet-1K, whereas all previous results were obtained with ViT pre-trained on ImageNet-21K. As a result, the Baseline network was trained using weights pre-trained on ImageNet-1K to provide a basis for comparison. Table 4.15 shows the results achieved with the two different backbones.

Method	mAP (%)	Rank-1 (%)	Rank-5 (%)
Baseline (ViT)	71.2	93.6	97.4
Cross-ViT	72.8	95.1	97.6

Table 4.15: Performance results using different backbones pre-trained on ImageNet-1k on the Veri-776 dataset.

When both methods are tested under the same pre-training conditions, the network with Cross-ViT as the backbone outperforms the Baseline network that uses ViT by 1.6% in mAP. This is a great result that implies Cross-ViT has the potential to outperform the TransReID network when trained on ImageNet-21K; however, this cannot be inferred due to hardware limitations.

5 Conclusions and Future Work

Re-Identification (ReID) has many challenging issues that result from the high variability of the vehicle's appearance in the camera images due to different illumination, different camera perspectives, occlusions, noisy images, and other factors. Furthermore, a vehicle identity can be very similar to other identities, and since ReID in this work is purely based on appearance and does not use license plate information, a network capable of extracting strong features is required.

Vision Transformers are novel networks known for extracting robust fine-grained features and are state-of-the-art for many computer vision tasks; thus, this was the choice for extracting the vehicle's features. Alike the work on [4], the TransReID network proved to operate extremely well achieving very good ReID results. This network makes use of external information such as vehicle perspective, and because this information is not available in all datasets, a classification network was trained to predict the vehicle viewpoint, which helped to improve the results on the Veri-Wild dataset.

Person Re-Identification (P-ReID) is a much more researched topic than Vehicle Re-Identification (V-ReID), so a network called Locally Aware Transformer from a top-performing P-ReID work was integrated into this work. When trained with the original datasets with multiple viewpoints, this network did not produce good results; however, when trained with a single viewpoint, such as only the fronts of the vehicles, this network outperformed the TransReID network. The P-ReID problem is typically simpler than V-ReID because the perspective problem is not as significant. Therefore, this result is interesting since it reveals that this network performs well when the problem is simplified. This implies that networks designed for P-ReID will typically perform poorly on V-ReID.

The loss functions play an important role in network training optimization, and it is well known that Cross Entropy Loss is substantially larger than Triplet Loss, therefore alternative weight ratios were tested, as well as a module called Momentum Adaptive Loss Weight, which automatically adjusts the weights during training. Furthermore, the Triplet Hard-Mining

Loss was replaced with a Soft-Mining Loss, however, none of these experiments produced better results.

Finally, the ViT backbone was replaced by a novel network known as Cross-ViT. Because this network outputs two features, multiple feature fusion strategies were tested, and Cross-ViT outperformed ViT when both were pre-trained on ImageNet-1K. Cross-ViT has shown promising results that should surpass TransReID when pre-trained on ImageNet-21K, but due to hardware limitations, that result could not be inferred.

Considering the work developed and the results obtained, there are several possibilities for continuing the project to improve the current work, such as:

- The sampler used to generate each batch is completely random, but when different seeds are used, the results can vary considerably, indicating that some batches may have better examples to optimize the triplet loss than others. Given the importance of image batches in V-ReID results, one possible future work would be to analyze which batches perform best and develop an algorithm based on data information to generate batches with the best samples to optimize the triplet loss.

- A classification network was used to predict the vehicle's perspective label, however, this could be improved by using different methods such as clustering, regression, or even using more advanced methods using specific vehicle keypoints to extract the exact orientation.

- The Cross-ViT network could not be trained on ImageNet-21K due to hardware limitations. According to the results, the pre-trained model has a significant impact on V-ReID performance. Since Cross-ViT outperformed ViT when pre-trained on ImageNet-1K, it is expected to outperform ViT when pre-trained on ImageNet-21K. As a result, one potential future work would be to train Cross-ViT on ImageNet-21K and then infer the results on V-ReID.

6 Bibliography

[1] Jianqing Zhu, Huanqiang Zeng, Jingchang Huang, Shengcai Liao, Zhen Lei, Canhui Cai, and LiXin Zheng. Vehicle re-identification using quadruple directional deep learning features. 11 2018.

[2] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. 3 2019.

[3] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. 10 2016.

[4] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification. 2 2021.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. 10 2020.

[6] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers distillation through attention. 12 2020.

[7] Rodolfo Quispe, Cuiling Lan, Wenjun Zeng, and Helio Pedrini. Attributenet: Attribute enhanced vehicle re-identification, 2021.

[8] Jingjing Qian, Wei Jiang, Hao Luo, and Hongyan Yu. Stripe-based and attribute-aware network: A two-branch deep model for vehicle re-identification. 10 2019.

[9] Yongming Rao, Guangyi Chen, Jiwen Lu, and Jie Zhou. Counterfactual attention learning for fine-grained visual categorization and re-identification. 8 2021.

[10] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Xiaodong Yang, Yue Yao, Liang Zheng, Pranamesh Chakraborty, Christian E. Lopez, Anuj Sharma, Qi Feng, Vitaly Ablavsky, and Stan Sclaroff. The 5th ai city challenge. 4 2021.

[11] Su V. Huynh, Nam H. Nguyen, Ngoc T. Nguyen, Vinh TQ. Nguyen, Chau Huynh, and Chuong Nguyen. A strong baseline for vehicle re-identification. 4 2021.

[12] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. 4 2020.

[13] Pirazh Khorramshahi, Neehar Peri, Jun cheng Chen, and Rama Chellappa. The devil is in the details: Self-supervised attention for vehicle re-identification. 4 2020.

[14] Zheng Tang and Milind Naphade Stan Birchfield Jonathan Tremblay William Hodge Ratnesh Kumar Shuo Wang Xiaodong Yang NVIDIA. Pamtri: Pose-aware multi-task learning for vehicle re-identification using highly randomized synthetic data.

[15] Fangyu Wu, Shiyang Yan, Jeremy S Smith, and Bailing Zhang. Joint semi-supervised learning and re-ranking for vehicle re-identication.

[16] Zhedong Zheng, Tao Ruan, Yunchao Wei, Yi Yang, and Tao Mei. Vehiclenet: Learning robust visual representation for vehicle re-identification. *IEEE Transactions on Multimedia*, 23:2683–2693, 2021.

[17] Charu Sharma, Siddhant R. Kapil, and David Chapman. Person re-identification with a locally aware transformer. 6 2021.

[18] Zhi Yu, Jiaming Pei, Mingpeng Zhu, Jiwei Zhang, and Jinhai Li. Multi-attribute adaptive aggregation transformer for vehicle re-identification. *Information Processing and Management*, 59, 3 2022.

[19] Hao Luo, Weihua Chen, Xianzhe Xu, Jianyang Gu, Yuqi Zhang, Chong Liu, Yiqi Jiang, Shuting He, Fan Wang, and Hao Li. An empirical study of vehicle re-identification on the ai city challenge, 2021.

[20] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. 3 2017.

[21] Adhiraj Ghosh, Kuruparan Shanmugalingam, and Wen-Yan Lin. Relation preserving triplet mining for stabilizing the triplet loss in vehicle re-identification. 10 2021.

[22] Jiawang Bian, Wen-Yan Lin, Yasuyuki Matsushita, Tan-Dat Nguyen, and Ming-Ming Cheng. Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence.

[23] Zhun Zhong, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding, 2017.

[24] Xinchen Liu, Wu Liu, Huadong Ma, and Huiyuan Fu. Large-scale vehicle re-identification in urban surveillance videos. volume 2016-August. IEEE Computer Society, 8 2016.

[25] Hongye Liu, Yonghong Tian, Yaowei Wang, Lu Pang, and Tiejun Huang. Deep relative distance learning: Tell the difference between similar vehicles.

[26] Yihang Lou, Yan Bai, Jun Liu, Shiqi Wang, and Ling-Yu Duan. Veri-wild: A large dataset and a new method for vehicle re-identification in the wild.

[27] Zheng Tang, Milind Naphade, Ming-Yu Liu, Xiaodong Yang, Stan Birchfield, Shuo Wang, Ratnesh Kumar, David Anastasiu, and Jenq-Neng Hwang. Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. 3 2019.

[28] Yue Yao, Liang Zheng, Xiaodong Yang, Milind Naphade, and Tom Gedeon. Simulating content consistent vehicle datasets with attribute descent. 12 2019.

[29] Dechao Meng, Liang Li, Xuejing Liu, Yadong Li, Shijie Yang, Zhengjun Zha, Xingyu Gao, Shuhui Wang, and Qingming Huang. Parsing-based view-aware embedding network for vehicle re-identification. 4 2020.

[30] Abhijit Suprem and Calton Pu. Looking glamorous: Vehicle re-id in heterogeneous cameras networks with global and local attention. 2 2020.

[31] Ming Li, Xinming Huang, and Ziming Zhang. Self-supervised geometric features discovery via interpretable attention for vehicle re-identification and beyond.

[32] Yan Bai, Jun Liu, Yihang Lou, Ce Wang, and Lingyu Duan. Disentangled feature learning network and a comprehensive benchmark for vehicle re-identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. 6 2017.

[34] Zhongdao Wang, Luming Tang, Xihui Liu, Zhuliang Yao, Shuai Yi, Jing Shao, Junjie Yan, Shengjin Wang, Hongsheng Li, and Xiaogang Wang. Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification.

[35] Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification.