# Infer the Whole from a Glimpse of a Part: Keypoint-Based Knowledge Graph for Vehicle Re-Identification

**Kai Lv**[1,2], **Yunlong Li**[1,2], **Zhuo Chen**[3,4,5], **Shuo Wang**[1,2], **Sheng Han**[1,2], **Youfang Lin**[1,2*]

[1]School of Computer Science & Technology, Beijing Jiaotong University
[2]Beijing Key Laboratory of Traffic Data Mining and Embodied Intelligence
[3]Zhejiang University
[4]CSSC Intelligent Innovation Research Institute
[5]CSSC Systems Engineering Research Institute
{lvkai, 21120379}@bjtu.edu.cn, huntercz@126.com, {19112029, shhan, yflin}@bjtu.edu.cn

## Abstract

Vehicle re-identification aims to match vehicles across non-overlapping camera views. Many existing methods extract features from one specific image, and these methods lack view-invariance when comparing vehicles of different orientations. As a result, discriminative parts obscured by viewpoint changes cannot contribute effectively to matching. This work presents a novel keypoint-based framework for vehicle Re-ID. We propose to explicitly model the intrinsic structural relationships between vehicle components via knowledge graph. By establishing connection between keypoints, our approach aims to leverage such prior to match vehicles even when some parts are not directly comparable due to orientation inconsistencies. Specifically, given query and gallery images, we first detect visible keypoints. Then, a transformer-based model infers features for non-overlapped keypoints by conditioning on visible correspondences defined in the knowledge graph. The final representation integrates visible and inferred features. Extensive experiments demonstrate our method outperforms state-of-the-arts on standard benchmarks under cross-view matching scenarios. To our knowledge, this is the first work introducing structural priors via keypoint knowledge graphs for view-invariant vehicle re-identification.

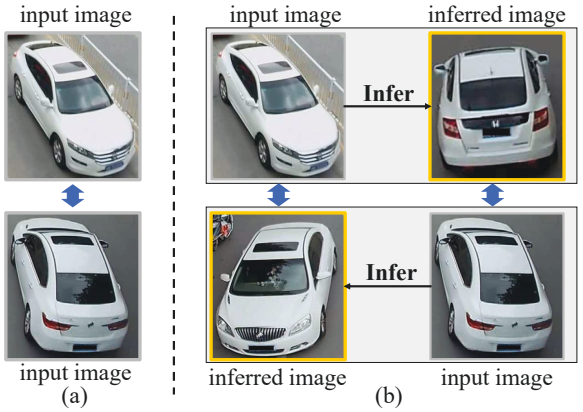


Figure 1: a) Most previous works directly calculate the similarity. b) Our brain generally associates vehicles between different orientations and compares their similarity when they have similar orientations. In this work, we intend to simulate this process to bridge the appearance between different orientations.

## Introduction

Vehicle re-identification is critical for intelligent transportation systems, where the primary goal is to accurately match a specific vehicle across various non-overlapping cameras. Traditional approaches in vehicle re-identification often focus on extracting global features from a single image of a vehicle. These features typically learn the overall shape, color, and structural characteristics of the vehicle, which are then used to search and match against a database of vehicle images taken from different angles and under varying lighting conditions.

However, these global features may not effectively capture the similarities between vehicle images in vehicle re-identification. As illustrated in Figure 1 (a), when comparing two vehicle images taken from different views, some discriminative parts of the vehicles, such as the front logo, may not be present in both images simultaneously, making it challenging to use these distinctive details to re-identify the vehicles. Therefore, relying solely on global features extracted from a single image could diminish the ability to re-identify a specific vehicle. Thus, it is crucial to deal with the problems caused by partial observations of vehicles.

Human's inherent deductive reasoning abilities can help solve the above problems. *Human brains can infer the whole from a glimpse*. For example, when we observe the front view of a vehicle as shown in Figure 1 (b), our brain leverages structural knowledge from past experience to implicitly infer the rear view. The main reason is that we understand the structural relationships and typical arrangements of different discriminative regions on the vehicle. This cognitive capability provides valuable insights for re-identification. On one hand, explicitly modeling these structural priors allows more effective utilization of the keypoint clues. When only partial views are observed, we can take hints from vis-

*Corresponding Author: Youfang Lin (yflin@bjtu.edu.cn)

ible regions to infer invisible parts. On the other hand, comparing images of similar orientations would be more intuitive for humans during the matching process.

Inspired by the human ability mentioned above, we propose a novel keypoint-based framework to infer complete vehicle representations from partial observations. Our key insight is that vehicles have intrinsic structural relationships between different parts, and these parts can be modeled as a knowledge graph. By leveraging such structure prior knowledge, we aim to infer the features of invisible regions based on the detected visible regions. Specifically, we first establish a vehicle knowledge graph to represent the structural relations between different vehicle keypoints. Then, given an input image containing only partial visible regions, we employ a transformer-based inference model to infer features for the invisible regions conditioned on the visible regions and their relations defined in the knowledge graph. In this way, we can achieve the features that contain all keypoint regions by combining both visible and inferred features, providing a more comprehensive representation of the vehicle.

In summary, the contributions can be summarised as:

- We introduce vehicle knowledge graphs to explicitly encode the structural relations between different vehicle keypoint regions. To our knowledge, this is the first work that establishes structural priors of vehicles with knowledge graphs for re-identification.
- We design a transformer-based inference model to generate features for invisible regions conditioned on visible regions and their relations.
- We propose a novel keypoint-based framework for vehicle re-identification that can infer a holistic vehicle representation from partial observations. This method is inspired by human cognitive habits, making it not only logical but also highly effective.

## Related Work

### Deep Learning in Vehicle Re-identification

In recent years, vehicle re-identification has attracted significant attention, due to advancements in deep learning. Initial methods rely on basic vehicle information like color and spatio-temporal data to augment global features (Liu et al. 2016b; Guo et al. 2018; Huang et al. 2020; Lv et al. 2019). Then, some papers (Luo et al. 2019; Lv et al. 2020) focus on innovating network architectures that more effectively extract a vehicle's global features. In addition, many methods (Hu, Shen, and Sun 2018; Liu et al. 2020; He et al. 2021; Vaswani et al. 2017; Sun et al. 2024) now incorporate attention mechanisms into vehicle re-identification tasks. For example, (Hu, Shen, and Sun 2018) proposes to learn channel attention and achieve good performance. (He et al. 2021) introduces transformer (Vaswani et al. 2017) into vehicle re-identification. This approach effectively combines the excellent performance of the transformer and achieves good results in vehicle re-identification. The above methods have confirmed that vehicle re-identification has gained increasing attention, and as a result, the proposed approaches have become more diverse.

### Local Region Features in Vehicle Re-identification

In vehicle re-identification, many methods (He et al. 2019; Yan et al. 2017; Zhang et al. 2020) utilize local vehicle features to mine detailed cues. There are also some methods (Wang et al. 2017; Khorramshahi et al. 2019) that utilize vehicle keypoints to make the model focus more closely on the local parts. (Wang et al. 2017) first introduces 20 vehicle keypoints and provides the coordinates of the keypoints in the VeRi-776 dataset (Liu et al. 2016c,b). However, the mentioned approaches primarily focus on individual local features without adequately considering the structural relationships and consistent integration of these features across different viewpoints.

### Knowledge Graph in Computer Vision

In knowledge representation and reasoning, a knowledge graph is a knowledge base that uses a graph-structured data model or topology to represent and operate on data. In computer vision, some methods (Lu et al. 2016; Wu et al. 2023; Ye et al. 2023; Tang et al. 2020; Xu et al. 2017) have introduced the concept of knowledge graph, using the graph-structured data model to assist visual tasks. For example, in the task of visual relationship detection, (Lu et al. 2016) studies the problem of visual relationship detection using knowledge graphs to model the relationships between objects in an image. Inspired by the successful application of knowledge graphs in various computer vision tasks, we aim to extend this concept to enhance our understanding and interpretation of images from multiple viewpoints. By employing a knowledge graph, we plan to build and analyze relationships between different keypoints of vehicle images, regardless of their views.

## Problem Statement and The Baseline

Vehicle re-identification is to match a query vehicle image with corresponding vehicle images from a pre-existing gallery across different cameras. In this task, there exist $N$ vehicle images, each associated with one of $K$ unique identities. Formally, the dataset can be represented as $D = (x_i, y_i)_{i=1}^{N}$, where $x_i$ denotes the vehicle image and $y_i$ is the associated identity label. The challenge is to sort the gallery (or the rank list) based on the similarities between the query image and the gallery images. The aim is to have images of the same vehicle identity as the query image appear higher in the rank list.

In this paper, the baseline model is based on the ID-discriminative Embedding (IDE) (Zheng et al. 2015), which is designed to learn a feature representation that maximizes the discriminability between different vehicle identities. The baseline first samples a vehicle image to the ResNet-50 (He et al. 2016) backbone and outputs the backbone features of the input image. With the backbone features, the baseline applies Global Average Pooling (GAP) to obtain the base features $F_{baseline}$, which are utilized to calculate the triplet loss. We then pass the base features through a Batch Normalization (BN) layer before calculating the cross-entropy loss. Finally, the baseline is jointly optimized by the above two losses. The optimization process iteratively adjusts the

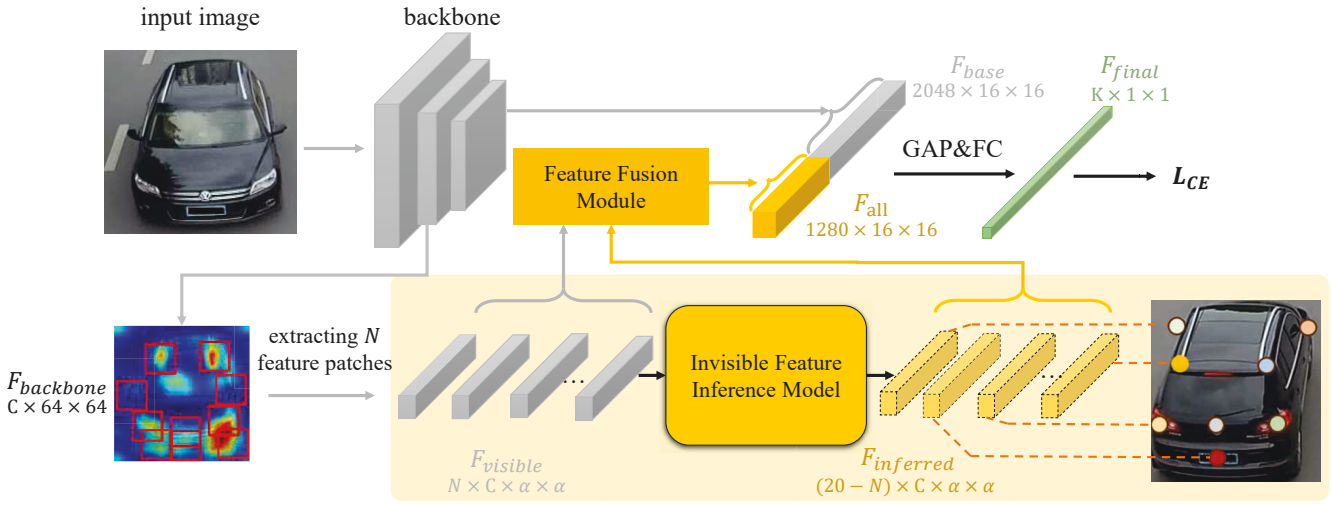


Figure 2: The overall framework of our method. Our method aims to generate representations involving all keypoints when only given one image. The final features $F_{final}$ derive from two features: the baseline features $F_{base}$ and the complete features $F_{all}$ that contains all keypoint information. $F_{base}$ is extracted by the baseline model, and $F_{all}$ is produced by a generative model named Invisible Feature Inference Model (IFIM).

model parameters to minimize the total baseline loss:

$$L_{baseline} = L_{CE} + L_{triplet}, \tag{1}$$

where $L_{CE}$ and $L_{triplet}$ are the cross-entropy loss and the triplet loss, respectively.

## The Proposed Method

### Overall Framework

The overall framework of our method has two main components: the baseline branch and the inference branch. The baseline branch employs pre-trained features, known as baseline features, which remain fixed during our methodology phase. In the inference branch, the Invisible Feature Inference Model (IFIM) leverages visible features $F_{visible}$ to infer the features of invisible keypoints $F_{inferred}$. $F_{visible}$ and $F_{inferred}$ are then combined to produce the features $F_{all}$ that contains all keypoint information using the feature fusion module. We utilize the final features, denoted as $F_{final}$, which integrate both baseline features and keypoint features $F_{all}$, as the definitive characteristics for comparing the similarity between vehicle images.

**$F_{base}$ is the baseline features.** Before we extract $F_{base}$, we initially train a baseline model, which thereafter remains fixed within our framework. The baseline model takes the vehicle image as input and processes it through the backbone to derive $F_{base} \in 2048 \times 16 \times 16$. As the baseline model is fixed to produce $F_{base}$, it can be replaced by other backbones. This flexibility allows our model to stay up-to-date and effective, even as newer and possibly better backbone networks are developed.

**$F_{all}$ contains all keypoint information.** According to (Wang et al. 2017), the vehicle has a total of 20 keypoints. Upon feeding an image into the backbone, we achieve $F_{backbone} \in C \times \alpha \times \alpha$ from the first layer of the backbone. We then obtain $N$ feature patches $F_{visible} \in N \times C \times \alpha \times \alpha$ of $N$ visible keypoints. The visible features $F_{visible}$ are input into the Invisible Feature Inference Model (IFIM) to infer the rest $20 - N$ invisible keypoint features $F_{inferred} \in (20 - N) \times C \times \alpha \times \alpha$. Note that IFIM is a transformer and is trained by the vehicle knowledge graph.

**The feature fusion module combines the visible features $F_{visible}$ and the inferred features $F_{inferred}$.** Initially, the module applies $1 \times 1$ convolution to both $F_{visible}$ and $F_{inferred}$, effectively reducing their channel dimensions $C$ to 64. This convolution operation standardizes the channel dimensions and helps in refining the feature maps. Following the convolution, we adopt an MLP layer to align the feature dimensions of $F_{visible}$ and $F_{inferred}$ and output a unified feature representation $F_{all} \in 1280 \times 16 \times 16$.

**$F_{final}$ is the final features that represent the vehicle.** After obtaining $F_{base}$ and the keypoint features $F_{all}$, we concatenate them to form the final features $F_{final}$. This process enriches the representational capacity of $F_{final}$. Then, the combination of two features is passed through a Global Average Pooling (GAP) layer and a Fully Connected (FC) layer to output $F_{final} \in K \times 1 \times 1$. During the training phase, $F_{final}$ is used with the cross-entropy loss for training, while in the testing phase, it serves as the feature representation for vehicles.

**The advantage of combining $F_{base}$ and $F_{all}$ lies in their complementary nature.** $F_{base}$, derived from the baseline model, is global and coarse-grained features extracted directly from the input vehicle images. These features inherently capture the primary characteristics of the vehicles, providing a strong foundation for the initial identification process. On the other hand, $F_{all}$, which integrates both visible and inferred features, adds critical details and nuances that might be missing from $F_{base}$ alone. This includes informa-

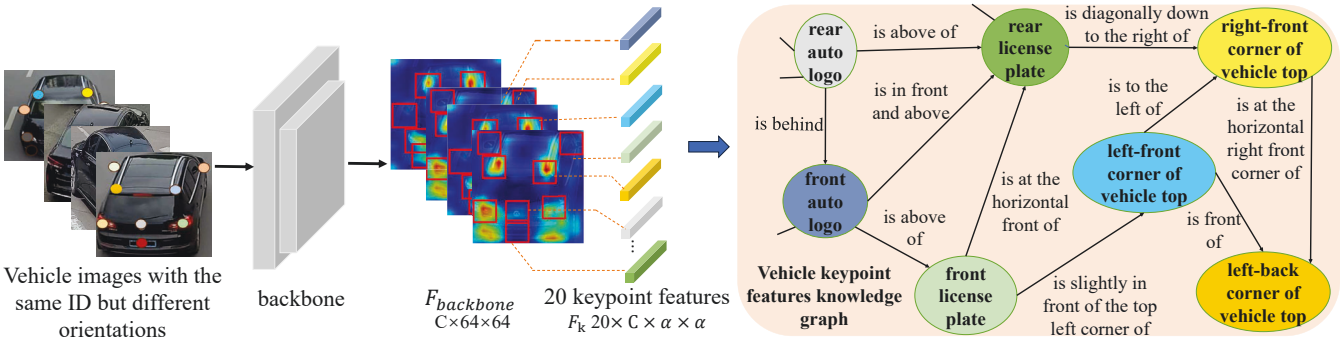


Figure 3: The process of building the vehicle knowledge graph for a specific vehicle. We input $N$ vehicle images of the same vehicle with different orientations into the backbone to extract the features $N \times F_{backbone}$. Then, we crop 20 keypoint features from $F_{backbone}$. By utilizing the structural topological relationships between these keypoint features, we form triples to constitute the vehicle knowledge graph in the form of a triplet structure.

tion about occluded or less conspicuous parts of the vehicle, which are crucial for distinguishing between very similar vehicles. To conclude, by combining the two features, $F_{final}$ leverages both the general features and the specificity of detailed features, enhancing the model's sensitivity to finer distinctions between vehicles.

## Vehicle Knowledge Graph

The vehicle knowledge graph is utilized in the inference branch and bridges different keypoints of a vehicle.

**Definition.** In this work, we define the vehicle knowledge graph $\mathcal{G}$ for a specific vehicle as a set of triplets $(h, r, t)$. $h$ denotes the keypoint features visible from the current view, $t$ represents the keypoint features extracted from another keypoint of the same vehicle, and $r$ is the physical and topological relationships between these keypoints.

In our method, we focus on leveraging a vehicle knowledge graph to infer the invisible parts of a vehicle from the visible ones. Specifically, our approach utilizes the visible features of one keypoint and designates the features as entity $h$. By utilizing the relationships denoted by $r$, we infer and synthesize the features of another invisible keypoint, represented as $t$.

**Vehicle Keypoint Detection.** As constructing the knowledge graph for a vehicle requires all 20 keypoint features, a vehicle keypoint detection model is essential. To this end, we fine-tune an existing model (Khorramshahi et al. 2019) to improve its precision in keypoint detection. Specifically, we incorporate data augmentation techniques such as random cropping and random erasing. We set the learning rate to 0.001 and epoch to 30 in the training stage. The detailed performance of this fine-tuned keypoint detection model is provided in the appendix.

**Building the Knowledge Graph.** As shown in Figure 3, we input multiple vehicle images with the same identity but from different views into the backbone of the trained baseline. Then, each image passes through the first layer of the backbone network to achieve the features denoted as $F_{backbone} \in C \times \alpha \times \alpha$. Next, we use the keypoint coordinates $(x^{(i)}, y^{(i)})$ of the input images to crop the keypoint features $F_k \in 20 \times C \times \alpha \times \alpha$. $\alpha$ denotes the length of this rectangular region, which is a hyperparameter in our model.

Upon acquiring all the keypoint features $F_K$ of a vehicle, we are equipped to create a detailed keypoint knowledge graph for that vehicle. This is achieved by organizing the keypoint features into structured triplets, as illustrated in Figure 3. Building the vehicle knowledge graph involves linking the 20 keypoint features in pairs to form triplets. Each triplet consists of a head entity $h$, a relation $r$, and a tail entity $t$. These triplets are systematically arranged based on the structural relationships among the keypoints, reflecting the vehicle's structural topology. This method allows for a comprehensive representation of the vehicle's geometric and structural attributes within the knowledge graph.

## The Invisible Feature Inference Model

With the vehicle knowledge graph mentioned above, as shown in Figure 2, the invisible feature inference model is to infer the invisible keypoint features $F_{inferred}$ from the visible features $F_{visible}$. Since the Invisible Feature Inference Model (IFIM) (see Figure 4) is based on the knowledge graph and a transformer module, We introduce the two parts separately.

**The role of the vehicle knowledge graph within our model is to provide the necessary data for training.** The training process of IMIF is determined by the number of visible keypoints $N$ in the input images. Specifically, when given $N$ visible keypoints, the knowledge graph automatically infers the remaining $20 - N$ invisible keypoints along with their corresponding relationships $r$. The features of these visible keypoints, denoted as $F_{visible}$, serve as the head entities $h$, and $r$ are used to derive the tail entities $t$. This allows us to utilize the knowledge graph to obtain the ground truth features $F_{gt}$ for the inferred keypoint features.

The relations $r$, generated by the knowledge graph, play a crucial role in the training of IFIM. The automatically inferred $(20 - N)$ relations for the invisible keypoints are also part of the input to the IFIM model. These relationships undergo specific processing to make them usable by the inference model. Specifically, we employ a Word2Vec model

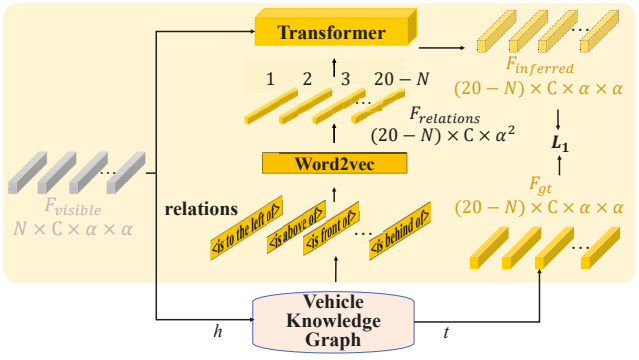


Figure 4: The Invisible Feature Inference Model (IFIM). IFIM has two main parts: the vehicle knowledge graph and the transformer module. The triplet components $(h, r, t)$ of the knowledge graph represent $F_{visible}$, $F_{relations}$, and $F_{gt}$, respectively. The transformer is to produce inferred features $F_{inferred}$.

(Mikolov et al. 2013) to transform the relationships $r$ into features that are compatible with deep learning models. The resulting feature representation, denoted as $F_{relations}$, has dimensions of $(20-N)\times C\times\alpha\times\alpha$.

**The transformer module in IFIM is specifically designed to infer the invisible features**. For this purpose, it utilizes the visible features, denoted as $F_{visible}\in(N)\times C\times\alpha\times\alpha$, along with the corresponding relational data, $F_{relation}$, as inputs. The transformer leverages this data to effectively generate the invisible features that are not directly observable in the provided images. The output of the transformer, the inferred invisible features, is represented as $F_{inferred}\in(20-N)\times C\times\alpha\times\alpha$. These features are then compared to the ground truth $F_{gt}\in(20-N)\times C\times\alpha\times\alpha$. The accuracy of the inferred features is quantified through the calculation of the $L_1$ loss between $F_{inferred}$ and $F_{gt}$. This metric helps in measuring the performance of the transformer in generating a precise and detailed representation of the vehicle's invisible features

**Here, we provide an example to illustrate how our inference model is trained with the vehicle knowledge graph.** When we can extract keypoint features of the rear auto logo and the left rear lamp, but are unable to capture those of the front auto logo, we can opt for one of two relational triplets to infer the missing features: either $(h_1, r_1, t_1)$ = (rear auto logo, is back of, front auto logo) or $(h_2, r_2, t_2)$ = (left rear lamp, is at the horizontal left rear corner of, front auto logo). We randomly select the triplet $(h_1, r_1, t_1)$, and the front auto logo $t_1$ features are then used as the ground truth $F_{gt}$ for the invisible features within our knowledge graph.

Then, we employ the relation $r_1$ = (is back of) alongside features from the rear auto logo $h_1$ as inputs to our model. Our approach integrates these extracted keypoint features $F_{visible}$, together with their corresponding relational data $F_{relations}$, feeding them into the transformer model to produce the inferred output features $F_{inferred}$. We compute the $L_1$ loss between the inferred keypoint features $F_{inferred}$ and the ground truth features $F_{gt}$. Our method not only refines the model's accuracy but also ensures a comprehensive understanding and reconstruction of vehicle components that are not visible.

Finally, IMIF combines $F_{visible}$ and $F_{inferred}$ to generate the final 20 keypoint features. As illustrated in Figure 2, after obtaining the inferred keypoint features $F_{infer}\in(20-N)\times C\times\alpha\times\alpha$ through the transformer, we concatenate them with the visible features $F_{visible}\in N\times C\times\alpha\times\alpha$. This process yields a complete set of 20 keypoint features $F_{all}\in 20\times C\times\alpha\times\alpha$. Finally, we concatenate these 20 keypoint features $F_{all}$ with global features to compute the final cross-entropy loss.

# Experimental Results

## Datasets and Evaluation Metrics

In this paper, we conduct our experiments on three widely used datasets: VeRi-776 (Liu et al. 2016c,b), VehicleID (Liu et al. 2016a), and VERI-Wild (Lou et al. 2019). In the field of vehicle re-identification, mean Average Precision (mAP) and rank-k (Liu et al. 2016b; Lou et al. 2019) are commonly used evaluation metrics. Therefore, in this paper, we also use mAP and rank-k as our evaluation metrics during the testing phase. The mAP measures the average precision of retrieval results, assessing the match between retrieved results and true labels at different thresholds. A higher mAP value indicates better model performance at different thresholds. The rank-$k$ metric indicates the proportion of true matches within the top $k$ most similar images.

## Comparison of State-of-the-art Methods

In this paper, we primarily conduct performance comparison experiments with other methods on three datasets: VeRi-776 (Liu et al. 2016c,b), VehicleID (Liu et al. 2016a), and VERI-Wild (Lou et al. 2019). For the VeRi-776 (Liu et al. 2016c,b) and VehicleID (Liu et al. 2016b) dataset, we mainly compare the mAP, rank-1, and rank-5 metrics. For the VERI-Wild (Lou et al. 2019) dataset, we use mAP and rank-1 as evaluation metrics.

**Evaluation on VehicleID (Liu et al. 2016a)** Table 1 also presents the comparison results of our method with other approaches on the VehicleID dataset. From Table 1, it is evident that our method consistently outperforms other methods in terms of the mAP metric. Moreover, our method achieves the highest results on both the Small and Large subsets. As for the Medium subset, although our method does not obtain the highest rank-5 result, the difference is not significant. This also indicates the effectiveness of our method on this dataset.

**Evaluation on VeRi-776 (Liu et al. 2016c,b)** In Table 1, we compare our method with recent approaches on the VeRi-776 dataset. It can be observed that our method performs the best in terms of mAP (83.3%), surpassing the second-best method by close to one percentage point. Additionally, our method achieves the highest rank-1 (97.6%) value among all methods. However, our method does not perform the best in the Rank-5 evaluation metric. The VARID (Li et al. 2020)

| Method | Year | VeRi-776 | | | VehicleID | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Small | | | Medium | | | Large | | |
| | | mAP | R-1 | R-5 | mAP | R-1 | R-5 | mAP | R-1 | R-5 | mAP | R-1 | R-5 |
| VAMI+STR | CVPR(2018) | 61.3 | 85.9 | 91.8 | - | 63.1 | 83.3 | - | 52.9 | 75.1 | - | 47.3 | 70.3 |
| RAM | ICME(2018) | 61.5 | 88.6 | 94.0 | - | 75.2 | 91.5 | - | 72.3 | 87.0 | - | 67.7 | 84.5 |
| EALN | TIP(2019) | 57.4 | 84.4 | 94.1 | 77.5 | 75.1 | 88.1 | 74.2 | 71.8 | 83.9 | 71.0 | 69.3 | 81.4 |
| AAVER | ICCV(2019) | 58.5 | 88.7 | 94.1 | - | 74.7 | 93.8 | - | 68.6 | 90.0 | - | 63.5 | 85.6 |
| PRN | CVPR(2019) | 74.3 | 94.3 | 98.7 | - | 78.4 | 92.3 | - | 75.0 | 88.3 | - | 74.2 | 86.4 |
| MAVN | TCSVT(2020) | 72.5 | 92.6 | 97.9 | - | 72.5 | 83.1 | - | - | - | - | - | - |
| DDM | TITS(2020) | 72.8 | 86.4 | 53.6 | 82.3 | 75.7 | 90.5 | 80.2 | 74.3 | 88.9 | 78.5 | 73.1 | 85.3 |
| PCRNet | ACM MM(2020) | 78.6 | 95.4 | 98.4 | - | 86.6 | 98.1 | - | 82.4 | 96.3 | - | 80.4 | 94.2 |
| VARID | TITS(2020) | 79.3 | 96.0 | **99.2** | 88.5 | 85.8 | 96.9 | 84.7 | 81.2 | 94.1 | 82.4 | 79.5 | 92.2 |
| PGAN | TITS(2020) | 79.3 | 96.5 | - | - | - | - | - | - | - | 83.9 | 77.8 | 92.1 |
| PVEN | CVPR(2020) | 79.5 | 95.6 | 98.4 | - | 84.7 | 97.0 | - | 80.6 | 94.5 | - | 77.8 | 92.2 |
| SAVER | ECCV(2020) | 79.6 | 96.4 | 98.6 | - | 79.9 | 95.2 | - | 77.6 | 91.1 | - | 75.3 | 88.3 |
| DFLNet | TPAMI(2021) | 81.0 | 97.1 | 99.0 | 82.8 | 78.8 | 95.0 | - | - | - | 75.4 | 69.7 | 90.5 |
| TransReID | ICCV(2021) | 82.0 | 97.1 | - | - | 82.3 | 96.1 | - | - | - | - | - | - |
| DFNet | TPAMI(2022) | 80.9 | 97.0 | 99.0 | 88.5 | 84.7 | 96.2 | 85.2 | 81.7 | 96.0 | 83.1 | 79.1 | 92.8 |
| PANet+PMNet | TCSVT(2022) | 81.6 | 96.5 | 98.6 | - | 85.3 | 97.3 | - | 80.5 | 94.5 | - | 77.6 | 92.2 |
| GiT | TIP(2023) | 80.3 | 96.9 | - | 90.1 | 84.7 | - | 86.8 | 80.5 | - | 84.3 | 77.9 | - |
| SSBVER | CVPR(2023) | 82.1 | 97.1 | 98.5 | 90.9 | 85.6 | 97.7 | 87.4 | 81.6 | 94.9 | 84.3 | 78.9 | 92.6 |
| SRF | TITS(2023) | 82.4 | 97.5 | 98.9 | 91.9 | 86.9 | 98.3 | 88.3 | 82.4 | **96.5** | 86.1 | 79.9 | 93.0 |
| SHCI | NN(2024) | 82.9 | 94.1 | 98.8 | - | 83.8 | 96.5 | - | 79.4 | **92.7** | - | 74.4 | 91.2 |
| baseline | | 81.7 | 96.5 | 98.0 | 90.8 | 86.2 | 97.5 | 87.9 | 82.3 | 95.7 | 85.2 | 80.2 | 93.5 |
| Ours | | **83.3** | **97.6** | 98.3 | **92.5** | **88.0** | **98.2** | **88.8** | **83.5** | 96.2 | **86.3** | **80.6** | **94.8** |

Table 1: Comparision of the state-of-the-art methods on VeRi-776(Liu et al. 2016b,c) and VehicleID(Liu et al. 2016a). Evaluation metrics include mAP, rank-1 (R-1), and rank-5 (R-5).

method achieves the best performance in the Rank-5 metric, but our method outperforms it significantly in terms of mAP and rank-1 metrics.

**Evaluation on VERI-Wild (Lou et al. 2019)** Table 2 displays the comparison results of our method with other approaches on the VERI-Wild dataset. As shown in Table 2, it is evident that our method has achieved favorable results compared to other methods. When compared to the baseline (He et al. 2020), our method demonstrates improvements in both the mAP and rank-1 metrics, further indicating the robustness of our approach on this dataset.

## Ablation Study

In this part, we mainly introduce the ablation study of our method on the VehicleID dataset (Liu et al. 2016b). As $F_{final}$ derives two parts: $F_{base}$ and $F_{all}$, we conduct comparative experiments to evaluate the effectiveness of the two features in Table 3.

**$F_{base}$ and $F_{all}$ are both important for our method.** Single $F_{all}$ refers to using only the 20 keypoint features to represent the vehicle, without utilizing $F_{base}$ when generating $F_{final}$. Conversely, Single $F_{base}$ does not use keypoint features and its performance is equivalent to the baseline model. On the VehicleID dataset, the Rank-1 and Rank-5 of the baseline reached (86.2%, 97.5%; 82.3%, 95.7%; 80.2%, 93.5%) on the three subsets, respectively. The results demonstrate that our baseline is a strong and effective baseline. Then, we can also observe that our method ($F_{base}$+$F_{all}$ (transformer))) outperforms the baseline, indicating that our method is effective.

**Single $F_{all}$ alone results in significantly poorer performance compared to other methods.** This underlines that $F_{all}$ and $F_{base}$ are complementary. Moreover, it demonstrates that relying solely on local information without considering the global context is inadequate. The reason for this is that the features in $F_{all}$ are cropped from a complete feature map, resulting in substantial information loss. We can conclude that integrating global information with keypoint-based local details is a highly effective approach.

**Regardless of whether the network of IFIM is used, our method is effective.** $F_{base}$ + $F_{all}$ (UNet) and $F_{base}$ + $F_{all}$ (transformer) represent the network of IFIM using UNet or transformer, respectively. Our test results on the VehicleID dataset are higher than the baseline. Moreover, the results also indicate that using the transformer as the architecture of IFIM is superior to using a UNet network.

## Parameter Analysis

In Table 4, we mainly discuss the hyperparameters on the VehicleID dataset (Liu et al. 2016a). The compared parameters involve the side length $\alpha$ of the keypoint feature patches, the attention layer number $K_a$ in IFIM, and the number of heads $K_h$ of the multi-head attention.

For the hyperparameter $\alpha$, we achieve the best results when $\alpha = 6$. This indicates that an appropriate cropping

| Method | Small | | Medium | | Large | |
|---|---|---|---|---|---|---|
| | mAP | R-1 | mAP | R-1 | mAP | R-1 |
| DRDL | 22.5 | 57.0 | 19.3 | 51.9 | 14.8 | 44.6 |
| FDA-Net | 35.1 | 64.0 | 29.8 | 57.8 | 22.8 | 49.4 |
| GSTE | 31.4 | 60.5 | 26.2 | 52.1 | 19.5 | 45.4 |
| AAVER | 62.2 | 75.8 | 53.6 | 92.7 | 41.6 | 58.6 |
| SAVER | 80.9 | 94.5 | 75.3 | 92.7 | 67.7 | 89.5 |
| PCRNet | 81.2 | 92.5 | 75.3 | 89.3 | 67.1 | 85.0 |
| VARID | 75.4 | 75.3 | 70.8 | 68.8 | 64.2 | 63.2 |
| DFLNet | 66.2 | - | 58.2 | - | 47.1 | - |
| IUID | 83.3 | 93.2 | 77.5 | 91.5 | 68.9 | 86.9 |
| DFNet | 83.0 | 94.7 | 77.2 | 93.2 | 69.8 | 89.3 |
| SSBVER | 82.7 | 95.1 | 77.5 | 93.4 | 70.1 | 90.1 |
| GiT | 81.8 | 92.7 | 75.7 | 89.2 | 67.6 | 85.4 |
| SRF | 82.7 | 95.1 | 77.9 | 93.8 | 70.2 | 90.6 |
| baseline | 85.4 | 95.4 | 82.2 | 94.1 | 76.3 | 92.5 |
| Ours | **86.94** | **96.51** | **83.65** | **95.42** | **78.41** | **94.21** |

Table 2: Performance on VERI-Wild (Lou et al. 2019), which contains three sub test set (Small, Meduim, Large). We utilize mAP, and rank-1 as evaluation metrics. R-1 represents rank-1.

| Method | Small | | Medium | | Large | |
|---|---|---|---|---|---|---|
| | R-1 | R-5 | R-1 | R-5 | R-1 | R-5 |
| single $F_{all}$ | 50.2 | 68.4 | 45.9 | 63.1 | 42.2 | 59.3 |
| single $F_{base}$ | 86.2 | 97.5 | 82.3 | 95.7 | 80.2 | 93.5 |
| $F_{base}$+$F_{visible}$ | 86.82 | 97.79 | 82.67 | 94.67 | 80.48 | 93.84 |
| $F_{base}$+$F_{all}$ (U) | 87.51 | 97.93 | 83.12 | 96.32 | 80.69 | 93.92 |
| $F_{base}$+$F_{all}$ (T) | **87.97** | **98.21** | **83.35** | **96.15** | **81.61** | **94.81** |

Table 3: Ablation study on VehicleID (Liu et al. 2016b). single $F_{all}$ means we only use the Invisible Feature Inference Model (IFIM), single $F_{base}$ means we only use the backbone of our framework, $F_{base}$ + $F_{all}$(U) means we use IFIM and backbone together and the network of IFIM is UNet, $F_{base}$ + $F_{all}$ (T) means the network of IFIM is transformer.

size is crucial for our model. This is because when the size is too large, it tends to introduce noise into the system. Conversely, it fails to capture sufficient keypoint information.

Both hyperparameters, $K_A$ and $K_h$, exhibit improved performance as their values increase, but this trend reverses beyond a certain point. The optimal settings for these parameters, determined through extensive testing, are $K_a = 2$ and $K_h = 4$. At these values, the model's performance is best.

## Computation Efficiency Analysis

In this section, we conduct performance comparison experiments using an A4000 GPU with 16 GB of memory and a Hygon C86 7151 CPU. As shown in Table 5, we compare the memory usage and processing time per batch on the VehicleID dataset during the testing phase between our method and other approaches. It is observed that our method consumes approximately 1.2 times the time of the baseline and uses nearly 5000M more GPU memory. While our method does not outperform the baseline in terms of efficiency, it does exceed the baseline in performance. Moreover, when comparing our method with TransReid (He et al. 2020) and SRF (Lv et al. 2023), it not only achieves superior re-identification accuracy but also shows better performance in terms of time and memory consumption.

| hyper parameter | seting | Small | | Medium | | Large | |
|---|---|---|---|---|---|---|---|
| | | R-1 | R-5 | R-1 | R-5 | R-1 | R-5 |
| $\alpha$ | 4 | 86.92 | 97.50 | 82.82 | 95.32 | 79.23 | 92.79 |
| | **6** | **87.97** | **98.21** | **83.35** | **96.15** | **80.61** | **94.81** |
| | 8 | 87.07 | 98.05 | 81.99 | 94.83 | 78.24 | 93.77 |
| $K_a$ | 1 | 86.44 | 97.42 | 82.26 | 95.05 | 79.24 | 93.38 |
| | **2** | **87.97** | **98.21** | **83.35** | **96.15** | **80.61** | **94.81** |
| | 4 | 86.07 | 97.72 | 81.67 | 95.48 | 78.29 | 93.78 |
| | 6 | 85.02 | 96.56 | 80.72 | 95.33 | 77.88 | 93.10 |
| $K_h$ | 2 | 86.42 | 97.89 | 82.51 | 95.92 | 79.18 | 93.20 |
| | **4** | **87.97** | **98.21** | **83.35** | **96.15** | **80.61** | **94.81** |
| | 6 | 86.51 | 97.60 | 82.59 | 95.83 | 78.52 | 93.34 |
| | 8 | 85.89 | 97.36 | 81.79 | 95.20 | 77.51 | 93.04 |

Table 4: Hyper parameter experiments on VehicleID (Liu et al. 2016a). $\alpha$, $K_a$, and $K_h$ represent the side length of the keypoint feature patches, the number of attention layers, and the number of head layers, respectively.

| Method | Small | | Medium | | Large | |
|---|---|---|---|---|---|---|
| | Time | Memory | Time | Memory | Time | Memory |
| TransReID | 0.83s | 13722M | 0.80s | 13910M | 0.80s | 14110M |
| SRF | 1.13s | 15230M | 1.099s | 15330M | 1.07s | 15458M |
| Ours | **0.73s** | **13182M** | **0.73s** | **13532M** | **0.72s** | **13350M** |
| Baseline | 0.56s | 8702M | 0.55s | 8420M | 0.55s | 8262M |

Table 5: Comparison of calculation time and GPU memory usage of each batch during testing.

## Conclusion

This paper addresses the challenges inherent in vehicle re-identification across non-overlapping camera views, particularly when vehicles are only partially visible. By recognizing the limitations of traditional methods that rely heavily on global features, we have introduced a novel keypoint-based framework that significantly enhances the accuracy and robustness of vehicle re-identification. This approach utilizes a vehicle knowledge graph to capture the structural relationships between various vehicle keypoints, enabling the inference of features for unseen vehicle parts based on visible regions. Our transformer-based inference model leverages these structural priors to generate a comprehensive representation of the vehicle, even from partial observations.

The results clearly demonstrate that our approach not only outperforms existing state-of-the-art methods but also provides a more robust solution in scenarios characterized by occlusions and varying camera angles. By mimicking human deductive reasoning in processing partial visual information, our model represents a significant advancement in the application of intelligent transportation systems, particularly in tasks like vehicle tracking and retrieval.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62206013).

## References

Guo, H.; Zhao, C.; Liu, Z.; Wang, J.; and Lu, H. 2018. Learning coarse-to-fine structured feature embedding for vehicle re-identification. In *AAAI Conference on Artificial Intelligence*, volume 32.

He, B.; Li, J.; Zhao, Y.; and Tian, Y. 2019. Part-regularized near-duplicate vehicle re-identification. In *IEEE Conference on Computer Vison and Pattern Recognition*, 3997–4005.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *IEEE Conference on Computer Vison and Pattern Recognition*, 770–778.

He, L.; Liao, X.; Liu, W.; Liu, X.; Cheng, P.; and Mei, T. 2020. Fastreid: A pytorch toolbox for general instance re-identification. *arXiv preprint arXiv:2006.02631*.

He, S.; Luo, H.; Wang, P.; Wang, F.; Li, H.; and Jiang, W. 2021. TransReID: Transformer-based Object Re-Identification. In *IEEE International Conference on Computer Vison*.

Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *IEEE Conference on Computer Vison and Pattern Recognition*, 7132–7141.

Huang, Y.; Liang, B.; Xie, W.; Liao, Y.; Kuang, Z.; Zhuang, Y.; and Ding, X. 2020. Dual domain multi-task model for vehicle re-identification. *IEEE Transactions on Intelligent Transportation Systems*, 23(4): 2991–2999.

Khorramshahi, P.; Kumar, A.; Peri, N.; Rambhatla, S. S.; Chen, J.-C.; and Chellappa, R. 2019. A dual-path model with adaptive attention for vehicle re-identification. In *IEEE International Conference on Computer Vison*, 6132–6141.

Li, Y.; Liu, K.; Jin, Y.; Wang, T.; and Lin, W. 2020. VARID: Viewpoint-aware re-identification of vehicle based on triplet loss. *IEEE Transactions on Intelligent Transportation Systems*, 23(2): 1381–1390.

Liu, H.; Tian, Y.; Yang, Y.; Pang, L.; and Huang, T. 2016a. Deep relative distance learning: Tell the difference between similar vehicles. In *IEEE Conference on Computer Vison and Pattern Recognition*, 2167–2175.

Liu, K.; Xu, Z.; Hou, Z.; Zhao, Z.; and Su, F. 2020. Further non-local and channel attention networks for vehicle re-identification. In *IEEE Conference on Computer Vison and Pattern Recognition Workshops*, 584–585.

Liu, X.; Liu, W.; Ma, H.; and Fu, H. 2016b. Large-scale vehicle re-identification in urban surveillance videos. In *IEEE International Conference on Multimedia and Expo*, 1–6. IEEE.

Liu, X.; Liu, W.; Mei, T.; and Ma, H. 2016c. A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *IEEE European Conference on Computer Vision*, 869–884. Springer.

Lou, Y.; Bai, Y.; Liu, J.; Wang, S.; and Duan, L. 2019. Veri-wild: A large dataset and a new method for vehicle re-identification in the wild. In *IEEE Conference on Computer Vison and Pattern Recognition*, 3235–3243.

Lu, C.; Krishna, R.; Bernstein, M.; and Fei-Fei, L. 2016. Visual relationship detection with language priors. In *IEEE European Conference on Computer Vision*, 852–869. Springer.

Luo, H.; Gu, Y.; Liao, X.; Lai, S.; and Jiang, W. 2019. Bag of tricks and a strong baseline for deep person re-identification. In *IEEE Conference on Computer Vison and Pattern Recognition Workshops*, 0–0.

Lv, K.; Du, H.; Hou, Y.; Deng, W.; Sheng, H.; Jiao, J.; and Zheng, L. 2019. Vehicle Re-Identification with Location and Time Stamps. In *IEEE Conference on Computer Vison and Pattern Recognition Workshops*, 399–406.

Lv, K.; Sheng, H.; Xiong, Z.; Li, W.; and Zheng, L. 2020. Pose-based view synthesis for vehicles: A perspective aware method. *IEEE Transactions on Image Processing*, 29: 5163–5174.

Lv, K.; Wang, S.; Han, S.; and Lin, Y. 2023. Spatially-Regularized Features for Vehicle Re-Identification: An Explanation of Where Deep Models Should Focus.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Sun, K.; Pang, X.; Zheng, M.; Nie, X.; Li, X.; Zhou, H.; and Yin, Y. 2024. Heterogeneous context interaction network for vehicle re-identification. *Neural Networks*, 169: 293–306.

Tang, K.; Niu, Y.; Huang, J.; Shi, J.; and Zhang, H. 2020. Unbiased scene graph generation from biased training. In *IEEE Conference on Computer Vison and Pattern Recognition*, 3716–3725.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Wang, Z.; Tang, L.; Liu, X.; Yao, Z.; Yi, S.; Shao, J.; Yan, J.; Wang, S.; Li, H.; and Wang, X. 2017. Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification. In *IEEE International Conference on Computer Vison*, 379–387.

Wu, Y.; Pan, C.; Wang, G.; Yang, Y.; Wei, J.; Li, C.; and Shen, H. T. 2023. Learning Semantic-Aware Knowledge Guidance for Low-Light Image Enhancement. In *IEEE Conference on Computer Vison and Pattern Recognition*, 1662–1671.

Xu, D.; Zhu, Y.; Choy, C. B.; and Fei-Fei, L. 2017. Scene graph generation by iterative message passing. In *IEEE Conference on Computer Vison and Pattern Recognition*, 5410–5419.

Yan, K.; Tian, Y.; Wang, Y.; Zeng, W.; and Huang, T. 2017. Exploiting multi-grain ranking constraints for precisely searching visually-similar vehicles. In *IEEE International Conference on Computer Vison*, 562–570.

Ye, S.; Xie, Y.; Chen, D.; Xu, Y.; Yuan, L.; Zhu, C.; and Liao, J. 2023. Improving Commonsense in Vision-Language Models via Knowledge Graph Riddles. In *IEEE Conference on Computer Vison and Pattern Recognition*, 2634–2645.

Zhang, X.; Zhang, R.; Cao, J.; Gong, D.; You, M.; and Shen, C. 2020. Part-guided attention learning for vehicle instance retrieval. *IEEE Transactions on Intelligent Transportation Systems*, 23(4): 3048–3060.

Zheng, L.; Shen, L.; Tian, L.; Wang, S.; Wang, J.; and Tian, Q. 2015. Scalable person re-identification: A benchmark. In *IEEE International Conference on Computer Vison*, 1116–1124.