Symmetric Network with Dual-vehicle Attributes Augmentation for Natural Language Vehicle Retrieval

Quang Minh Dinh[1], Hung Phong Tran[2], Quoc Huy Pham[2], Minh Khoi Ho[2]
[1]Simon Fraser University,Canada
[2]Hanoi University of Science and Technology, Vietnam
qmd@sfu.ca
{phong.tnh200465,huy.pq200283,khoi.hm204917}@sis.hust.edu.vn

Abstract

Natural Language-based tracked-vehicle retrieval problem has received much attention in recent years, as it provides practical utilities in urban systems. Several existing works that aim to solve it commonly encounter some major problems, including the lack of information in a query, the similar attributes shared by vehicles with different identities, and the lack of annotated data for tracks and vehicles. To address these problems, we introduce the Symmetric Network with Dual-vehicle Attributes Augmentation for Natural Language Vehicle Retrieval (SNDA). Our solution is able to capture the local and global information of the language queries and the track images, at the same time learning the cross-modal representations between them, both locally and globally. We propose an attributes enhancement system that makes use of a dual-vehicle paradigm to enhance our system with both the two vehicles' attributes and their relationship. Our solution produces a 35.44% MRR score, achieving rank 7^{th} in Track 2 of the 7^{th} AI City Challenge.

1. Introduction

In recent years, vehicle retrieval has emerged as an interesting and important approach to address urban city planning problems. Various researchers and city planners have shown attention towards this field, with image-based approaches being commonly used. Most recent works focus on solving the image-to-image matching task, also known as vehicle re-identification (Vehicle ReID). They achieved impressive results [6, 18, 24, 25] as their models are used as the backbones for several vision-based vehicle-related tasks [20]. However, there are a lot of limitations to deploying this type of architecture for urban systems, as it is most often difficult and time-consuming to obtain image queries in practice. As there are several advances in multimodal machine learning recently [2, 12, 15, 19], another approach that has gained lots of interests is using cross-model representation learning to jointly train the text encoder and the visual encoder, and use natural language text queries as an alternative to image queries. The 7^{th} AI City Challenge has further elevated the problem of text-based vehicle retrieval with its Tracked-Vehicle Retrieval by Natural Language Descriptions track, which adapts video understanding into this field.

As achieving a high result in the natural language tracked-vehicle retrieval task can be especially challenging due to (1) the ambiguity and lack of information in a single query, (2) vehicles with different identities showing having identical appearance attributes or motion patterns, and (3) the lack of data needed to train a robust retrieval model [20], we propose a symmetric network with an attributes augmentation system with the aim to tackle the mentioned challenges, at the same time provide a method to extract the attributes from the language queries and utilize them to augment the text inputs. The main part of our system consists of four branches that can capture the local and global representations of both text queries and track images. As our augmentation system makes use of a dual-vehicle paradigm, it can learn both the attributes of two nearby vehicles and their relationship at the same time. We evaluate our solution on a variation of the CityFlow-NL dataset [13] and report the Mean Reciprocal (MRR), Recall@5 and Recall@10 scores as the performance metrics. We have achieved a 35.44% MRR accuracy on the public test set of the track, finishing at 7_{th} place on the leaderboard.

2. Related Work

2.1. Video Retrieval using Natural Language Queries

Vehicle video retrieval using natural-language inquiries refers to a process of searching and retrieving video content related to vehicles by using natural language queries.

This approach often utilizes a pretrained language model such as DistillBERT [16] or RoBERTa [10] to encode textual features and visual backbone network such as Vision Transformer [7] or EfficientNet [17] to extract visual feature from input videos or from the frames sampled from the videos.

The emergence of latest works on large-scale pretrained models for video understanding has significantly improved the performance for tasks such as video question-answering, video text-retrieval, video captioning, etc. CenterClip [22] utilizes two clustering algorithms to mitigate the effect of non-essential tokens while leveraging the impact of important tokens. CLIP-hitchhiking [3] aims at a similar goal but applies query-scoring, self-attention scoring and joint-attention scoring. In addition, Frozen in Time [2] combines the works of TimeSFormer [4] and Vision Transformer [7], snapshots the frames from the videos to gradually increase attention to temporal context.

2.2. Re-identification (ReID) and vehicle ReID

Object re-identification (ReID) refers to the task of recognizing and tracking objects across different camera views. Many approaches have been applied for ReID, including Siamese networks [23] and spatial-temporal modules [4].

In particular, vehicle ReID involves identifying the same vehicle from multiple cameras or frames. It is a challenging problem due to variations in lighting conditions, camera angles, and occlusions. One of the earliest works in Vehicle ReID was proposed by Liu *et al*. [9] who used a deep learning-based approach to extract features from vehicle images and perform ReID. They proposed a spatially aligned pooling method to handle variations in vehicle pose and scale. More recently, Zhang *et al*. [24] proposed an appoach that leveraged the advantages of data augmentation, task-specific and model ensembling to yield further impressive performance. Some of the latest works such as MsKAT [8] includes State elimination Transformer (SeT) to eliminate impacts of the viewpoint and camera variation and Attribute aggregation Transformer (AaT) to learn specific attributes of the vehicles.

We proposed a simpler approach, using a symmetric architecture with a dual-vehicle attributes enhancement system aims to address the tracked-vehicle retrieval by natural language descriptions task, using local and global representations of natural language descriptions and track images fused together and applying the InfoNCE loss for cross-modal learning, along with text augmentation methods and motion modeling using motion maps generated from track videos.

3. Methodology

We adopt the idea of Zhao *et al*. [21] and propose a symmetric architecture with a dual-vehicle attributes enhancement system as a solution to the tracked-vehicle retrieval by natural language descriptions task. As illustrated in Fig. 2, our system consists of four main branches similar to SSM [21], which are used to capture the local and global representations of both the natural language descriptions and the track images. The local and global representations of each modality are fused together to get the natural language and visual representations, and we apply the InfoNCE loss [14] to each of the corresponding text-visual pairs to learn the representations between the two modalities. To enhance the cross-model learning process, we introduce a dual-vehicle attributes enhancement system that effectively captures the meaningful appearance and motion characteristics of two adjacent vehicles from the track images. We also provide a text augmentation method to amplify the corresponding characteristics of the main vehicle, which shows promising results as the system can obtain more information about the vehicle's appearance and motion attributes. We use the Mean Reciprocal Rank (MRR) as the primary evaluation metric, along with Recall@5 and Recall@10, to evaluate the performance of our method on the tracked-vehicle retrieval by natural language descriptions task.

3.1. Data Augmentation

3.1.1 Image Augmentation and Motion Modelling

Since our analysis of the dataset shows that the cameras are static, and the background of each track video remains stable throughout the video, we generate the motion maps using the method introduced by Bai *et al*. [1]. For each track, we calculate the mean of every frame in the video and paste the cropped vehicles in the bounding boxes from the same track to get the motion image for that track. The background image generation process can be formulated as follow:

$$B = \frac{1}{N}\sum_{i}^{N} F_i, \tag{1}$$

where N is the number of frames in the video, F_i is the i_{th} frame of the video, and B is the background image. As the cropped vehicles images from consecutive frames can occlude with each other, we follow the adjustment made by Zhao *et al*. [21] and calculate the Intersection over Union (IoU) of consecutive bounding boxes and ignore the latter if its IoU with the previous bounding box is greater than a threshold T, which we set to 0.05.

Both the motion images and the cropped vehicle images in the training process are put through a vanilla random crop and random rotation augmentation. The local image of each sample in the training process is obtained by randomly se-

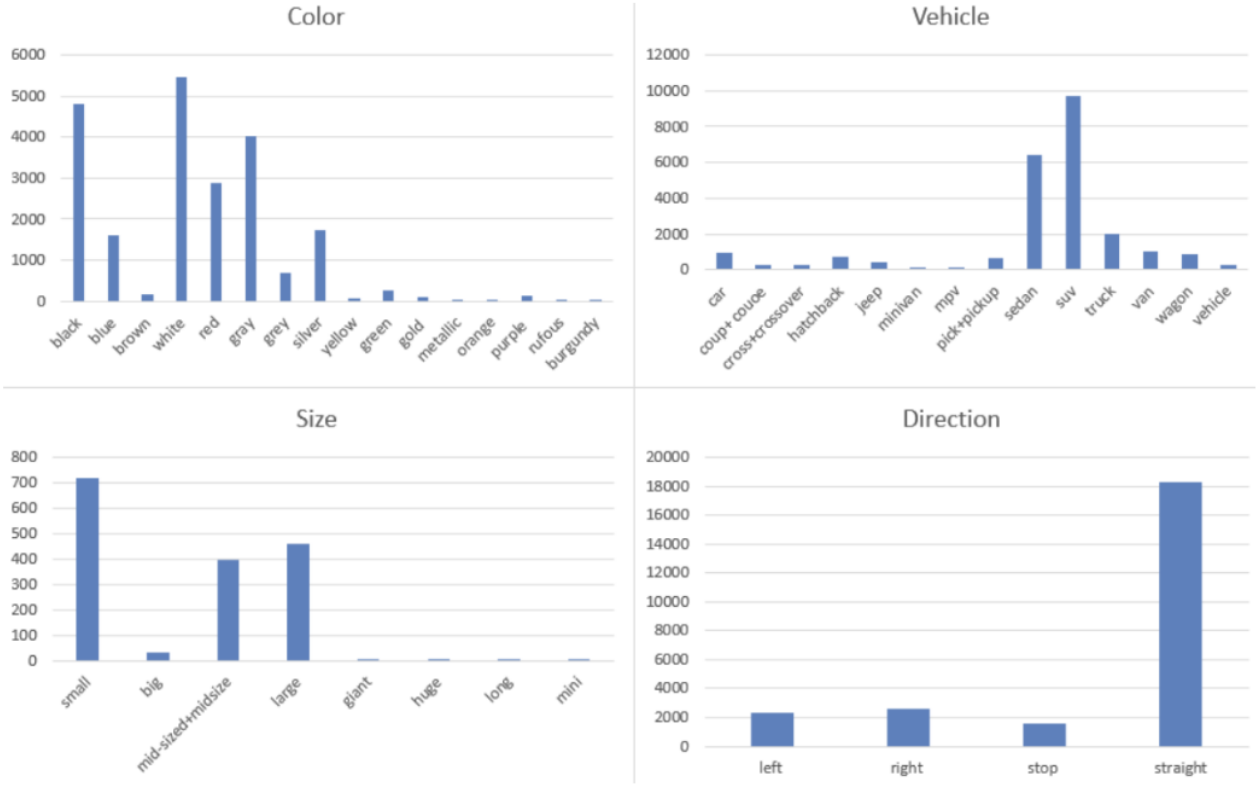

Figure 1. The frequencies of some common words describing the vehicle's type, motion, size, and color appeared in the dataset.

lecting one cropped vehicle image from the images corresponding to the bounding boxes provided for each track.

3.1.2 Dual-vehicle Attributes Natural Language Extraction

As most of the natural language descriptions contain rich information about the appearance and motion attributes of the vehicles, we extract all the words describing the type, motion, size and color of the vehicles and analyze their frequencies, which are shown in Fig 1. We select a subset of the most appeared words for each attribute and place them into different categories. For vehicle type, the categories are *"suv"*, *"sedan"*, *"truck"*, and *"van"*, which also contains *"wagon"*, as they often show up together in the queries. We choose *"left"*, *"right"* and *"stop"*, whereas the following word is not *"lane"* for the motion categories and assume that the vehicle motion for all other queries is *"go straight"*. The color categories are *"red"*, *"blue"*, *"black"*, *"white"*, and three colors *"gray"*, *"grey"*, *"silver"* belong to the same category. As for the vehicle size, we notice that all description words can be placed in 3 groups, which are *"small"*, *"mid-sized"*, and *"large"*.

For each sample of the training set and the test queries set, the dataset provides three natural language descriptions

Descriptions	A gray SUV turns left from intersection.
	A gray SUV turns left through a busy intersection.
	A gray minivan takes a left at an intersection.
Other Views	A big SUV making a left turn at the intersection.
	A gray SUV turn left followed by another pickup truck.
	A gray van turns left.

Table 1. An example of the information extracted from the primary descriptions and the descriptions of the same scene from different angles.

of the frame and optional descriptions of the same scene from different cameras, whose number is different for each sample. Since the only difference between the primary descriptions and the descriptions from other angles is the motion of the vehicle, we merge both description sets into a single combined set to search for appearance attributes effectively, and shuffle it for a robust extraction process. As the appearance of the word "intersection" in the queries is a good representation of the long-distance relationships in the track [21], we also search for intersection in the entire combined set and name the attribute *inter*. Throughout the dataset, a substantial amount of queries contain both the type and color descriptions of two nearby vehicles. To en-

hance our model using the additional information about the second vehicle, for both the type and color categories, we prioritize looking for two words in the same category that appear together in the same query of the shuffled combined set, for which we call *type1*, *type2*, *color1*, and *color2*. *color2* is discarded with the absence of *type2*. As for the vehicle's motion, we only search in the primary description set for a word in the motion category and name it *motion1*. We search for the word describing the vehicle's size in the combined set and call it *size1*. An example of the extraction process is shown in table 1.

3.1.3 Text Augmentation

Local	big gray suv. A gray minivan takes a left at an intersection.
Global	left. A gray SUV turns left from intersection. intersection.

Table 2. An example of text augmentations for the local and the global inputs.

Similar to Zhao *et al*. [21], we randomly select two queries from the primary description set of each sample and apply subject augmentation on one query, and motion augmentation and location augmentation on the other. Subject augmentation is performed by combining *size1*, *color1*, and *type1* extracted in the previous section and appending them to the beginning of a query to construct the local input. The global input is created by appending *motion1* to the beginning of the other query (subject augmentation) and appending "intersection" to the end of the same query if that word is found in the combined description set (location augmentation). As illustrated in Tab. 2, since each attribute word is extracted once from several queries, the appended words might differ from those in the original query, providing more attribute details to the inputs.

3.2. Symmetric Network with Dual-vehicle Attributes Augmentation

3.2.1 Cross-modal Representation Learning

We adopt a similar symmetric architecture to SSM [21] to learn the cross-modal representations of the vehicle tracks and the corresponding natural language queries.

Natural Language Representation. Two of the main branches of our system, which are on the left of Fig. 2, are used to learn the natural language representations provided by the queries. Both use the same pre-trained backbone model to encode the input text and global text generated using the method mentioned in Sec. 3.1.3 and output the text embeddings of the input queries. Both text embeddings then go through two different projection heads with the same formulation:

$$f_t = g_t(h_t) = W_2\sigma(LN(W_1 h_t)), \tag{2}$$

where h_t is the text embedding, LN is a Layer Normalization layer (LN), σ is a ReLU layer, and f_t is the local or global text feature encoded by the branch. We fuse the two features f_t^l and f_t^g using concatenation to obtain the natural language representation E_t of the track.

Visual Representation. Two branches on the right side of Fig. 2 are the visual branches used to learn the representations of the track videos. Similar to the two natural language branches, both of the visual branches use the same pre-trained backbone visual model to obtain the visual embeddings from the local image and the motion image created by the method mentioned in Sec. 3.1.1. The visual embeddings also go through two different projection heads with a similar formulation:

$$f_i = g_i(h_i) = W_2\sigma(BN(W_1 h_i)), \tag{3}$$

where h_i is the image embedding, BN is a Batch Normalization layer (BN), σ is a ReLU layer, and f_i is the branch's local or global visual feature output. The two visual features f_i^l and f_i^g are then concatenated to get the fused visual representation E_i of the track.

3.2.2 Dual-vehicle Attributes Augmentation

As one of the major challenges for cross-modal representation learning is that vehicles with different attributes often show small interclass variations visually [20], we enhance the training process by enhancing the vehicles' local and global attributes using a method similar to Zhang *et al*. [20].

Natural Language Attribute Features. For each attribute $attr \in \{type1, type2, color1, color2, size1, motion1, inter\}$ extracted using the method mentioned in Sec. 3.1.2 with n_{attr} categories, we created a $(n_{attr} + 1) \times 1$ one-hot vector L_{attr} with entry $L_{attr}^j = 1, j <= n_{attr}$ if the extracted attribute belongs to category j_{th}, any other entry $L_{attr}^{k!=j} = 0$. If the extracted attribute is empty, indicating a lack of information in the queries, or the attribute word in the queries belongs to an insignificant category that is not in the subset we chose for that attribute, we set $L_{attr}^{n_{attr}+1} = 1$, and any other entry $L_{attr}^{k<n_{attr}+1} = 0$.

Visual Attribute Feature. To make the system learn additional attribute signals on the visual side, we forward the two visual features f_i^l and f_i^g through a total of 7 projection heads with the same formulation similar to Eq. (3), each having the same size as a label one-hot vector on the natural language side. Specifically, we generate f_{type1}^a, f_{size1}^a, and f_{color1}^a using f_i^l and f_{type1}^a, f_{color2}^a,

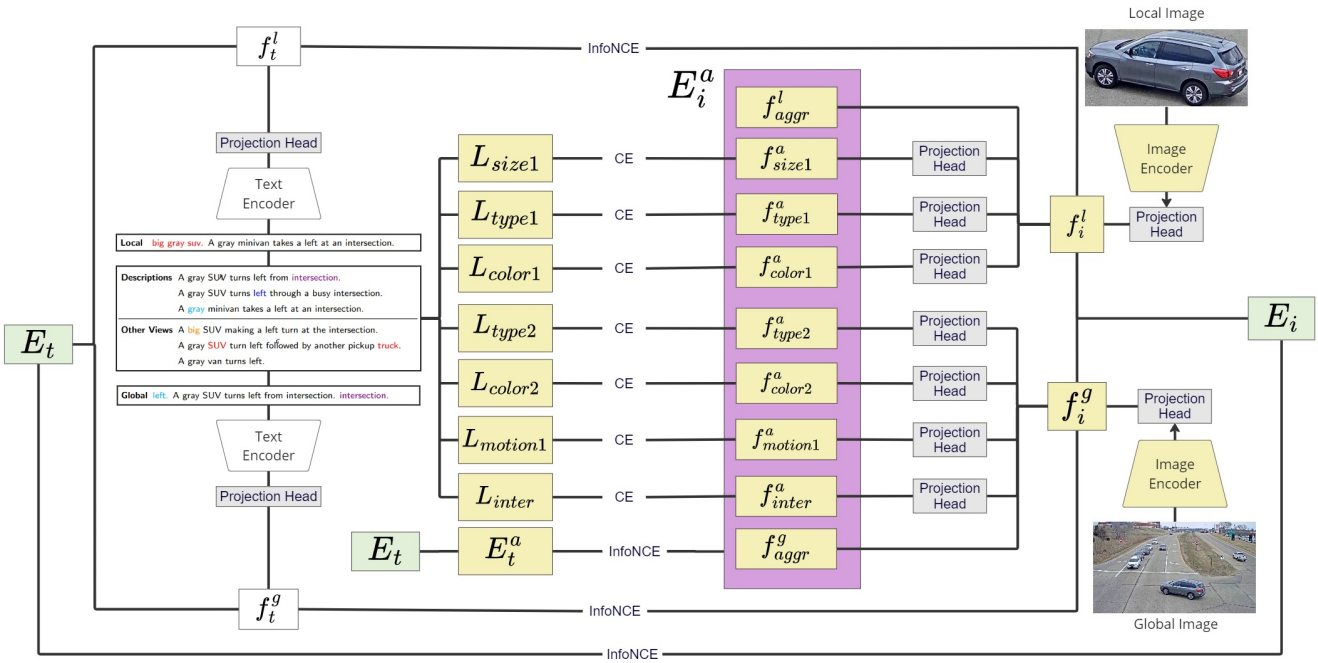

Figure 2. Overview of our method.

$f^a_{motion1}$, and f^a_{inter} using f^g_i.

Cross-modal Augmentation Representation Learning. We create a new visual-language representation pair with pre-specified attributes augmented for cross-modal representation learning. To embed the comprehensive vehicle and track features, we forward both visual features through a linear layer to generate two vectors with the same dimension $aggr\_size \times 1$. All visual attribute vectors and feature vectors are fused using concatenation to generate the visual augmentation vector E^a_i with size $fuse\_dim = (2aggr\_size + 7 + \sum_{attr} n_{attr}) \times 1$. On the natural language side, we forward the fused natural language representation vector E_t through a linear layer to create the natural language augmentation vector E^a_t with the same size $fuse\_dim$.

3.3. Loss Functions

Similar to other works on the tracked-vehicle natural language retrieval task [1,5,13,20,21], we apply the symmetric InfoNCE loss [14] to each of the four visual-language pairs $\langle f^l_i, f^l_t \rangle$, $\langle f^g_i, f^g_t \rangle$, $\langle E_i, E_t \rangle$, and $\langle E^a_i, E^a_t \rangle$. For a batch of N image-text pairs, the system produces $4N$ cross-modal representation pairs. The text-to-image and the image-to-text losses are defined as follow:

$$\mathcal{L}^z_{t2i} = \frac{1}{N}\sum_{i=1}^{N} -\log \frac{\exp\left(\cos\left(z^i_{text}, z^i_{img}\right)/\tau\right)}{\sum_{j=1}^{N}\exp\left(\cos\left(z^j_{text}, z^j_{img}\right)/\tau\right)} \tag{4}$$

$$\mathcal{L}^z_{i2t} = \frac{1}{N}\sum_{i=1}^{N} -\log \frac{\exp\left(\cos\left(z^i_{img}, z^i_{text}\right)/\tau\right)}{\sum_{j=1}^{N}\exp\left(\cos\left(z^j_{img}, z^j_{text}\right)/\tau\right)} \tag{5}$$

where $z = \langle z_{img}, z_{text} \rangle$ is one of the four visual-language pairs. The symmetric InfoNCE loss is formulated as:

$$\mathcal{L}^z_{SNCE} = \frac{\mathcal{L}^z_{t2i} + \mathcal{L}^z_{i2t}}{2} \tag{6}$$

Our representation loss is the sum of the four losses:

$$\mathcal{L}_{rep} = \mathcal{L}^{\langle f^l_i, f^l_t \rangle}_{SNCE} + \mathcal{L}^{\langle f^g_i, f^g_t \rangle}_{SNCE} + \mathcal{L}^{\langle E_i, E_t \rangle}_{SNCE} + \mathcal{L}^{\langle E^a_i, E^a_t \rangle}_{SNCE} \tag{7}$$

To make the model learn the additional signals, we apply the cross entropy loss (CE) to each of the $\langle f^a_{attr}, L_{attr} \rangle$ pairs, denoted as $\mathcal{L}^a_{attr}$. We calculate our second loss by taking the mean of the 7 CE losses:

$$\mathcal{L}_{aug} = \frac{1}{7}\sum_{attr} \mathcal{L}^a_{attr} \tag{8}$$

The total loss is defined as follow:

$$\mathcal{L} = \mathcal{L}_{rep} + \mathcal{L}_{aug} \tag{9}$$

3.4. Post-processing

We adopt the same post-processing strategy as SSM [21] to enhance our final similarity matrix using Long-distance Relationship Modeling and Short-distance Relationship modeling.

Long-distance Relationship Modeling. For each visual track or language query, intersection detection can be implemented to capture the position relationships between two modalities. For the natural language query, intersection can be detected using the strategy we discussed in Sec. 3.1.3. Since each camera stays fixed, intersection in each visual track can be captured by tracking the location of a vehicle in consecutive n frames. If the vehicle's location does not change much throughout n frames, we conclude that the location that the camera captures is an intersection. Each of the visual and natural language location embeddings is represented using a one-hot vector, and the location similarity matrix S_l between all tracks and queries can be calculated using the cosine similarity.

Short-distance Relationship Modeling. Similar to our dual-vehicle attributes augmentation, SSM [21] has a post-processing step to enhance the similarity matrix. For each track, several frames are randomly selected, and the local visual branch is used to extract all the local visual features of every vehicles in each frame, by using the provided bounding boxes. For each query, if the relationship between two vehicles is mentioned, the similarity score can be calculated by taking the maximum value of the cosine similarity between each vehicle feature and all other vehicle features in the track. The final similarity matrix is denoted as S_r.

Our final similarity matrix is formulated as:

$$S = \sum_z S_z + \alpha S_r + \beta S_l, \tag{10}$$

where $z \in \left\{\left\langle f_i^l, f_t^l \right\rangle, \left\langle f_i^g, f_t^g \right\rangle, \left\langle E_i, E_t \right\rangle, \left\langle E_i^a, E_t^a \right\rangle\right\}$, $\alpha = 1$ and $\beta = 0.2$ are two hyper-parameters.

4. Experiments

4.1. Dataset

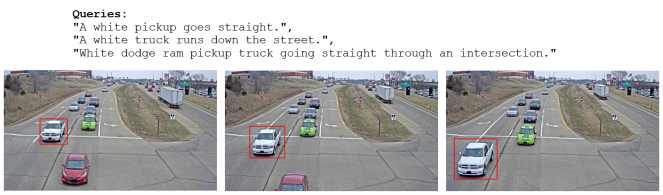

Figure 3. Example of queries - track pair in the dataset.

CityFlow-NL. The CityFlow-NL dataset [13] comprises of 666 targeted vehicles in 3,598 single-view tracks tracked from 46 calibrated cameras, each annotated with a query set of three distinct NL descriptions for the total of 6,784. The NL descriptions give details about vehicle color, vehicle maneuver, traffic scene, and relations with other vehicles.

In the Tracked-Vehicle Retrieval by Natural Language Descriptions task of the 7^{th} AI City Challenge, the dataset that is used to train and evaluate our models is a variation of the CityFlow-NL [13], including 2,155 vehicle tracks and additional 184 vehicle tracks and 184 query sets used for testing.

4.2. Evaluation Metrics

Following to the requirements of the AI City Challenge Track 2, we adopted the Mean Reciprocal Rank (MRR) as the main evaluation metric, together with Recall @5 and Recall @10 as the secondary metrics.

Mean Reciprocal Rank. Mean Reciprocal Rank (MRR) is a metric used to evaluate the effectiveness of ranking models or algorithms, especially in information retrieval and search engine applications. It measures the quality of the ranked list of items that the system produces in response to a set of user queries, which is formulated as:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i}, \tag{11}$$

where the value $|Q|$ indicates the total count of language queries, while $rank_i$ represents the position of the correct track in the ranking for the i_{th} language query.

Recall @5, Recall @10 and Recall @25. Recall @k is a performance metrics used to evaluate the effectiveness of information retrieval systems, specifically vehicle retrieval. It measures the proportion of relevant retrieved vehicles that were retrieved among the top k retrieved vehicles. The metrics is denoted as:

$$recall\,@k = \frac{N(k)}{|R|} \quad as \quad N(k) = |R \cap D(k)|, \tag{12}$$

where $N(k)$ is the size of the intersection between the set of relevant vehicles R and the set of retrieved vehicles up to rank k.

4.3. Implementation Details

Following Zhao *et al.* [21], we select EfficientNet B2 [17] as our visual encoder backbone, and initialize them with their pre-trained weights. For the natural language encoder backbone, we ultilize RoBERTa [10] and freeze it during the training process. Both the vehicle images and the motion images are resized to 228×228. We set the batch size to 64, and train our models for 400 epochs, which is

optimized with the AdamW optimizer [11] with weight decay (1e-2) and initial learning rate 0.01. We apply a warm-up scheduler for 40 epochs, and a step delay scheduler that decays the learning rate for every 80 epochs. All of our models are trained using a NVIDIA A100-80G GPU with 90GiB RAM.

4.4. Evaluation Results

Method	MRR	Recall@5	Recall@10
Baseline	0.47	0.32	0.69
Baseline+All Attrs	0.44	0.25	0.68
Baseline+Selected Attrs	0.45	0.60	0.87
Baseline+Selected Attrs+NL Aug	0.58	0.78	0.93
Baseline+Selected Attrs+NL Aug+Feat Eng	0.58	0.87	0.94

Table 3. Ablation study on the validation set.

Method	MRR	Recall@5	Recall@10
Baseline	0.23	0.33	0.50
Baseline+Selected Attrs	0.25	0.44	0.57
Baseline+Selected Attrs+NL Aug	0.23	0.36	0.51
Baseline+Selected Attrs+NL Aug+Feat Eng	0.24	0.39	0.58
Ensemble+Post-process	0.35	0.53	0.64

Table 4. Ablation study on the test set.

Ablation Study. We conduct an ablation study involving several experiments with different model choices. The result on the validation and test set are illustrated in Tab. 3 and Tab. 4. "Baseline" denotes the symmetric model without any attribute augmentations. "All Attrs" indicates using all the attributes of the second vehicle, and "Selected Attrs" indicates using only *type2* and *color2*. As shown in Tab. 3, the model performs worse when using all of the second vehicle's attributes because of the significant lack of *size2* and *motion2*, and the Recall@5 and Recall@10 scores increase by a huge amount after removing the two attributes. "NL Aug" denotes applying augmentations to the local and global texts, and "Feat Eng" indicates removing the motion word if the following word is *"lane"* and removing *color2* if *type2* is not found. We can see that the system performs better on the validation set as more techniques are applied, results in the final scores of 0.58 for MRR, 0.87 for Recall@5, and 0.94 for Recall@10. However, it fails to overcome the domain gap as it performs worse in the test set after applying natural language augmentations. After the feature engineering step is applied to remove some of the noises that frequently appear in the test set, the models perform slightly better. We ensemble the models and augment the result with the post-processing steps in Sec. 3.4, and achieve the final results of 0.35 for MRR, 0.53 for Recall@5, and 0.64 for Recall@10.

With the final MRR score of 0.3544, we present our solution as the 7^{th} place among all submissions of the 7^{th} AI City Challenge Track 2.

5. Conclusion

In this paper, we introduce SNDA, a symmetric architecture with attributes augmentation and evaluate it on the natural language tracked-vehicle retrieval task. It contains four main branches to extract the visual and natural language representations, both locally and globally. We present a method to effectively extract the attributes from the text descriptions and augment the local and global text inputs with those attributes. We also develop an attributes enhancement system that takes into account the relationship of two nearby vehicles, which is able to capture different local and global characteristics. Our solution achieves 35.44% MRR accuracy on the test set, yielding rank 7^{th} on the the 7^{th} AI City Challenge Track 2.

As one major limitation of our method lies in its inability to overcome the domain gap, a promising direction for future works is to focus on domain adaptation techniques. More improvements on the fusion mechanism can be made to improve the performance of the augmentation techniques.

References

[1] Shuai Bai, Zhedong Zheng, Xiaohan Wang, Junyang Lin, Zhu Zhang, Chang Zhou, Hongxia Yang, and Yi Yang. Connecting language and vision for natural language-based vehicle retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4034–4043, 2021. 2, 5

[2] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738, 2021. 1, 2

[3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. A CLIP-Hitchhiker's Guide to Long Video Retrieval, 2022. 2

[4] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, 2021. 2

[5] Yunhao Du, Binyu Zhang, Xiangning Ruan, Fei Su, Zhicheng Zhao, and Hong Chen. Omg: Observe multiple granularities for natural language-based vehicle retrieval. pages 3123–3132, 06 2022. 5

[6] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15013–15022, 2021. 1

[7] Alexander Kolesnikov, Alexey Dosovitskiy, Dirk Weissenborn, Georg Heigold, Jakob Uszkoreit, Lucas Beyer, Matthias Minderer, Mostafa Dehghani, Neil Houlsby, Sylvain Gelly, Thomas Unterthiner, and Xiaohua Zhai. An image is worth 16x16 words: Transformers for image recognition at scale. 2021. 2

[8] Hongchao Li, Chenglong Li, Aihua Zheng, Jin Tang, and Bin Luo. MsKAT: Multi-Scale Knowledge-Aware Transformer for Vehicle Re-Identification. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):19557–19568, 2022. 2

[9] Xinchen Liu, Wu Liu, Huadong Ma, and Huiyuan Fu. Large-scale vehicle re-identification in urban surveillance videos. In *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2016. 2

[10] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach, 2019. 2, 6

[11] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 7

[12] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. CLIP4Clip: An empirical study of CLIP for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, 2022. 1

[13] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mohammed Shaiqur Rahman, Archana Venkatachalapathy, Anuj Sharma, Qi Feng, Vitaly Ablavsky, Stan Sclaroff, Pranamesh Chakraborty, Alice Li, Shangru Li, and Rama Chellappa. The 6th ai city challenge, 2022. 1, 5, 6

[14] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2, 5

[15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1

[16] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter, 2020. 2

[17] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020. 2, 6

[18] Zheng Tang, Milind Naphade, Stan Birchfield, Jonathan Tremblay, William Hodge, Ratnesh Kumar, Shuo Wang, and Xiaodong Yang. Pamtri: Pose-aware multi-task learning for vehicle re-identification using highly randomized synthetic data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 211–220, 2019. 1

[19] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18123–18133, 2022. 1

[20] Jiacheng Zhang, Xiangru Lin, Minyue Jiang, Yue Yu, Chenting Gong, Wei Zhang, Xiao Tan, Yingying Li, Errui Ding, and Guanbin Li. A multi-granularity retrieval system for natural language-based vehicle retrieval. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3215–3224, 2022. 1, 4, 5

[21] Chuyang Zhao, Haobo Chen, Wenyuan Zhang, Junru Chen, Sipeng Zhang, Yadong Li, and Boxun Li. Symmetric network with spatial relationship modeling for natural language-based vehicle retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3226–3233, 2022. 2, 3, 4, 5, 6

[22] Shuai Zhao, Linchao Zhu, Xiaohan Wang, and Yi Yang. Centerclip: Token clustering for efficient text-video retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 970–981, 2022. 2

[23] Meng Zheng, Srikrishna Karanam, Ziyan Wu, and Richard J Radke. Re-identification with consistent attentive siamese networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5735–5744, 2019. 2

[24] Zhedong Zheng, Minyue Jiang, Zhigang Wang, Jian Wang, Zechen Bai, Xuanmeng Zhang, Xin Yu, Xiao Tan, Yi Yang, Shilei Wen, and Errui Ding. Going Beyond Real Data: A Robust Visual Representation for Vehicle Re-identification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2550–2558, 2020. 1, 2

[25] 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. 1