Transport and Communications Science Journal

A CLIP-BASED DUAL-STREAM METHOD FOR TEXT BASED VEHICLE SEARCH

Quang Huy Can[1], Phuong Dung Nguyen[1,2], Thuy Binh Nguyen[3*], Hong Quan Nguyen[4], Thien Linh Vo[3], Thi Lan Le[1]

[1]SigM Lab, School of Electrical and Electronic Engineering (SEEE), Hanoi University of Science and Technology, Hanoi, Vietnam

[2]Thuyloi University, Hanoi, Vietnam

[3]Universiy of Transport and Communications, Hanoi, Vietnam

[4]Viet-Hung Industrial University, Hanoi, Vietnam

ARTICLE INFO

TYPE: Research Article
Received: 08/09/2024
Revised: 22/10/2024
Accepted: 10/01/2025
Published online: 15/01/2025
https://doi.org/10.47869/tcsj.76.1.2

**Corresponding author*
Email: thuybinh_ktdt@utc.edu.vn

Abstract. A text-based vehicle search refers to a system where users can find vehicles or route information by entering text-based queries. The primary objective of text-based vehicle search is to identify the most relevant vehicle in a given dataset using a natural language description as a query. This approach leverages natural language processing (NLP) to understand and interpret description queries and provide relevant results. Despite significant progress, this task still faces several challenges due to the complexity and diversity of natural language, as well as inherent difficulties in the vision domain. Moreover, few studies have focused on tracked-vehicle retrieval, where vehicle tracklets are considered instead of single images. In this paper, we propose a novel framework for natural language-based tracked-vehicle retrieval based on CLIP model, one of the most effective models for image-text matching task. This framework leverages both appearance and motion information to enhance the matching accuracy of vehicle tracklet retrieval. Some experiments are conducted on the CityFlow-NL dataset, provided by the 6-th AI City Challenge, an annual competition. The results are comparable to state-of-the-art methods, achieving an MRR score of 46.63%, Rank@5 of 67.02%, and Rank@10 of 81.82%.

Keywords: Vehicle retrieval, CLIP-based model, natural language, vision and text matching.

1. INTRODUCTION

Vehicle search has indeed become crucial for the advancement of intelligent traffic systems in smart cities. The ability to efficiently locate and manage vehicles offers numerous benefits, including enhanced security, improved traffic flow, and better overall management of the transportation network. Consequently, in the last decade, vehicle retrieval problem has attracted much attention of research community on over the world. In the literature, studies addressing this problem are classified into two main approaches: vision-based and text-based vehicle retrieval. Vision-based vehicle retrieval, also treated as the re-identification problem, aims to identify images of the same vehicle as it moves across different surveillance cameras. The main challenge here is to match vehicles accurately despite the changes in camera angles, lighting conditions, and potential occlusions. In contrast, text-based vehicle retrieval involves finding the most suitable vehicle in the database that matches a given description sentence. This approach is significantly more challenging than vision-based retrieval due to the diversity and complexity of description sentences. However, researchers often favour the latter approach due to its convenience and naturalness, as it does not need to provide an image of the vehicle of interest. In text-based vehicle search, description sentences may vary greatly in terms of detail and language use, making it difficult to match them accurately with vehicle images. Additionally, cross-modalities matching also bring much more difficult for text-image retrieval. To overcome the challenges of this problem, several models have been proposed focusing on either feature extraction or image-text alignment [1, 2, 3]. However, the obtained performance is still limited.

Recently, the CLIP (Contrastive Language-Image Pre-training) model [4], trained on a large dataset of image-text pairs, has been introduced. In this paper, to leverage CLIP's feature extraction capabilities, we propose a CLIP-based framework for text-based vehicle search. The contribution of this paper is two-fold. Firstly, to enhance the feature extraction process, we propose strategies to generate both appearance and motion data in the visual and language domains from the original vehicle tracklet and natural language description. This approach not only transforms a sequence-level framework into an image-level one but also leverages both static and dynamic information. Secondly, we explore the effectiveness of processing appearance and motion data within either a single model or two independent models, evaluating the impact on overall framework performance. Some experiments are conducted on the dataset provided by 6-th AI City Challenge Track 2. The values of MRR score, Rank@5, and Rank@10 are 46.63%, 67.02%, and 81.82%, respectively. These obtained results show the superiority of the proposed framework compared to some existing methods for natural language-based vehicle retrieval problem.

2. RELATED WORKS

2.1. Natural language-based video retrieval

Natural language-based video retrieval is an emerging field that leverages advancements in natural language processing and computer vision. The main target of this task is to seek the most relevant video matching the given language description from a large amount of candidate videos [1]. This approach is increasingly relevant in an era where video content is proliferating across platforms, and the need for efficient search mechanisms is paramount. Although have been achieving some important milestones, studies on language-based video retrieval problem have to cope with many challenges from both vision and language domains. For example, the strong variation in illumination, color, and occlusion in images or the diversity of natural language description. Besides, visual-text alignment is also a significant difficulty when handling this task.

To overcome the afore-mentioned challenges, most existing work have focused on two main approaches including dual- and single-stream methods. Due to the difference of the modality and the limitation of the network, most of the early work have focused on the dual-stream framework in which visual and textual data are processed in two separated branches. In these frameworks, some effective deep-learned networks are used for visual representation. Meanwhile, several networks with the recurrent structure are exploited for textual representation. By this way, the computation complexity is significantly reduced.

Recent years have witnessed the successful migration of Transformer [5] from natural language processing to computer vision, CLIP [4], ViT [6], ViLBERT [7], etc. Some studies on natural language-based video retrieval task begin to use Transformer as encoders for both video and natural language. ViLBERT [7] firstly explore to utilize a single-stream Transformer network for dealing with text-based video retrieval. Due to the powerful performance of CLIP model [4] for image-text contrastive learning, a large number of methods are extended from CLIP baseline model. CLIP2Video [8] is an improved structure of CLIP model to develop form image-text retrieval to video-text retrieval.

2.2. Natural language-based vehicle retrieval

The primary goal of natural language-based vehicle retrieval is to identify the most appropriate vehicle image corresponding to a given natural language description. This task is also treated as the branch of natural language-based video retrieval. Consequently, the natural language-based vehicle retrieval task faces challenges similar to those encountered in general natural language processing tasks. Also, most of the existing works belonging to this research direction process vision and text input in either dual or single stream. In the work of Sribano et al. [9] propose a dual-stream framework to handle natural language-base vehicle retrieval, in which BERT model (Bidirectional Encoder Representation from Transformer) [10] is used for text embedding and a CNN network [11] in combination with a Transformer model is utilized for image embedding. To deal with this task, Lee et al. [12] introduce a new segmentation-based network model, called SBNet, consisting of three main modules including natural language, image processing, and multi-modal modules. This multi-modal module is exploited to learn the similarity between visual and textual embedding via an attention mechanism.

In the other way, several works have focused on using Transformer in a single-stream framework for natural language-based vehicle retrieval. Pirazh et al. [13] and Nguyen et al. [14] suggest using only CLIP model [4] for both visual and textual representation. Besides, Sebastian et al. [15] propose encoder-decoder based framework in which the encoder is utilized to extract useful features in both visual and textual domains while the decoder jointly optimizes these embeddings via an input-token-reconstruction task. To connect language and vision domains, Bai et al. [1] introduce a novel framework to jointly train the cropped image and language description in an end-to-end manner.

In summary, we have mentioned several impressive methods for dealing with natural language-based vehicle retrieval. However, there are a few works that refer to motion cues - important information to search a relevant vehicle to a description sentence. In this work, we propose a novel framework that leverages both appearance and motion information to significantly enhance the matching rate in natural language-based vehicle retrieval.

3. PROPOSED METHOD FOR TRACKED-VEHICLE RETRIEVAL

3.1. The overall proposed framework

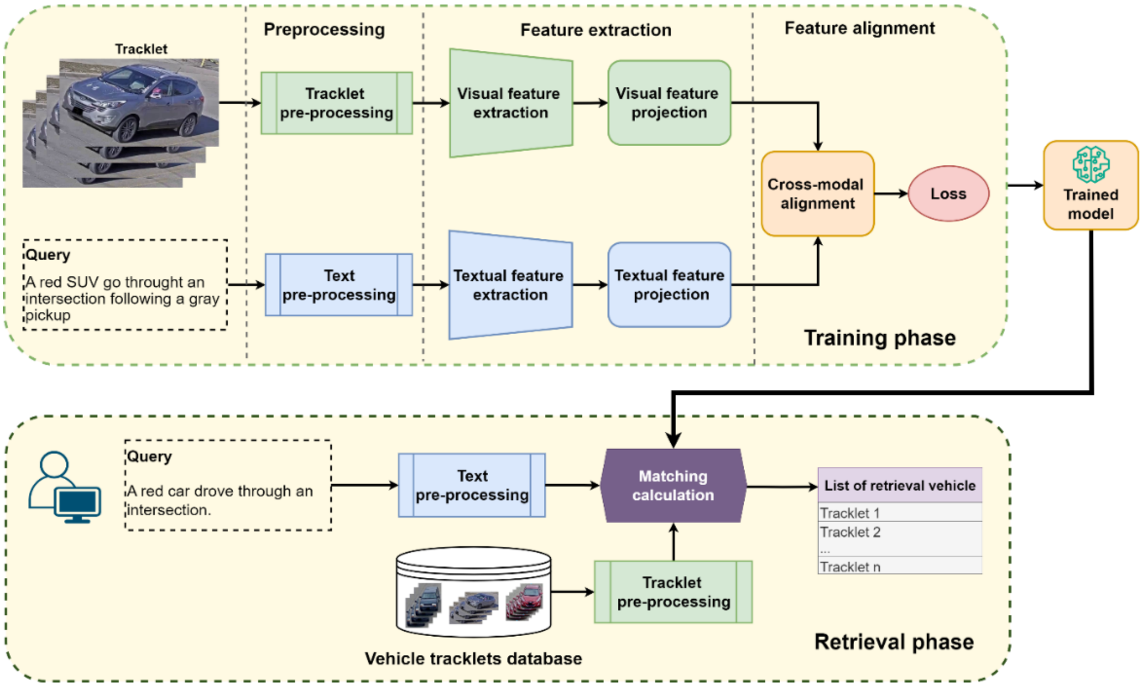


Figure 1. The proposed framework for natural language-based vehicle retrieval problem.

Figure 1 illustrates the overall proposed framework for text-based tracked-vehicle retrieval. Unlike traditional natural language-based vehicle retrieval methods, this framework is specifically designed to search for the most relevant vehicle tracklet based on a description sentence query. The framework operates in two main phases: the training phase and the retrieval phase. The training phase comprises three key stages: pre-processing, feature extraction, and feature alignment. The primary objective of the pre-processing step is to generate appearance and motion information for both the visual and textual streams, serving as a data augmentation technique to enhance training performance. The feature extraction step focuses on capturing valuable features and constructing visual and textual embeddings corresponding to the images and description sentences, respectively. Following this, the extracted features are projected into a common space to learn the visual-textual similarity in the final step. To train the deep-learning model, three loss functions are exploited including Instance loss, Circle loss, and InfoNCE loss. In the retrieval phase, the trained model is utilized to calculate the similarity score between a given natural language description and vehicle tracklets within the database. The output of this framework is a ranked list of vehicle tracklets with the highest similarity scores. Each of these steps will be explained in more detail below.

3.2. Data pre-processing

3.2.1. Visual data pre-processing

Text-based vehicle search is performed on tracked vehicles; this means that the bounding boxes of the vehicle over the video sequence have been determined through detection and tracking process. In our study, for a vehicle, two images that are appearance image and motion image are generated. An appearance image is determined by selecting randomly one bounding box in the tracklet of the vehicle.

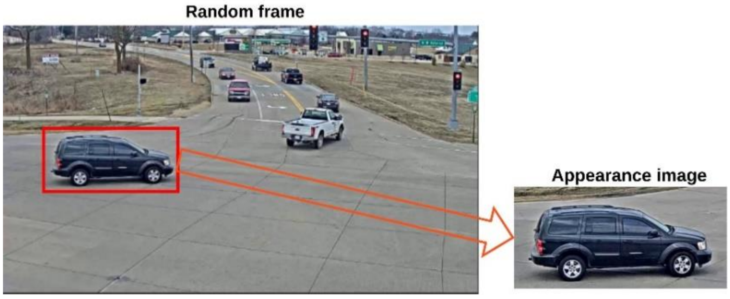


Figure 2. The appearance image of the interested vehicle.

One of the remarkable points of the proposed framework is that the motion image, presenting the movement of the interested vehicle, is exploited. This image is built from the background image and bounding boxes corresponding to locations of the vehicle over consecutive frames. It is worth noting that images of the examined dataset are captured by static cameras. Therefore, the background has not changed much in a short period. Firstly, the background image is created by applying the average mechanism over all consecutive frames within the considered tracklet. Secondly, bounding boxes corresponding to locations of the interested vehicle over time are added in the background image to generate the motion map. This motion map provides not only appearance cues but also motion direction of the interested vehicle. Figure 2 and Figure 3 provide an example of appearance image and motion image in our study.

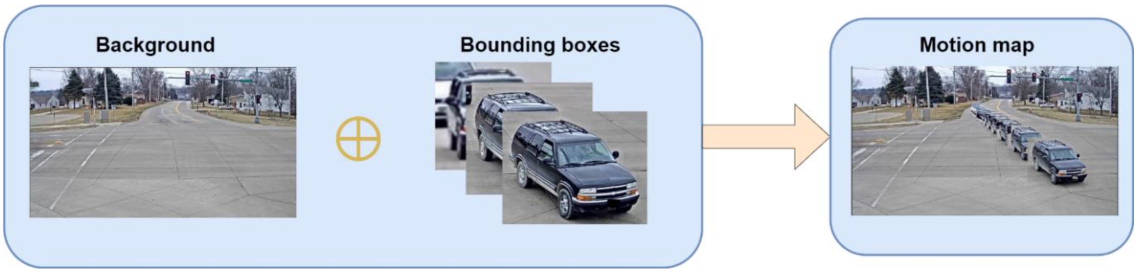


Figure 3. Motion image generation process.

3.2.2. Textual data pre-processing

Like the visual branch, the textual branch focuses on two crucial description sentences containing appearance and motion information, respectively. In our study, we use both motion description and appearance description. When applying the proposed method for the 6-th AI City Challenges Track 2 dataset, as there are total three sentences used for describe the motion of a given vehicle, motion description sentence is generated by randomly taking one of three above sentences. To generate a description for appearance, from the motion description, syntactic parsing to extract nouns or noun-phrases is performed by NLTK (Natural Language Toolkit) [16]. After that, {"*This is a*"} is added to nouns or noun-phrases to form a complete sentence for appearance representation. For example, a given description sentence: {"*A red car drove through an intersection.*"}, after applying NLTK tool on the input sentence, the two noun-phrases are generated: {["*A red car*", "*an intersection*"]}. The noun-phrase corresponding to appearance information is extracted and combined with {"*This is a*"} to build a complete sentence as follows: {"*This is a red car.*"}.

3.3. Data augmentation techniques

3.3.1. Image transformation

a. Data augmentation for appearance images:

As aforementioned, training data plays an important role in enhancing the performance of the training phase. Training data has a significant effect on accuracy as well as generality of the trained model. Especially, in the dataset provided by 6-th AI City Challenges Track 2, only 2,155 samples are available leading to the lack of the training data. Moreover, the examined dataset also suffers from the unbalance of vehicle types, colors, and movement. To overcome these issues, data augmentation techniques are performed by using several image transformations, such as flipping, cropping, translation. In this work, AugMix method [17] a combination between some traditional operators and Mixup algorithm [18] is exploited to generate augmented images from randomly mixing two images in the original dataset.

b. Data augmentation for motion images:

When dealing with image sequences, beside appearance information, motion also plays an important role in providing useful cues for recognition task. For motion images, data augmentation is implemented by changing intensity and flipping the motion direction (from right to left, and vice versa). These augmentation techniques can simulate different real-world scenarios, making the model more robust to variations in the input data, particularly in the case where the amount of available training data is limited.

3.3.2. Back translation

Back translation [19] also known as reverse translation, is one of the oldest data augmentation techniques. Back translation is a process in which a given sentence is translated into another language, after that, the returned sentence is translated back to the original language. The basic principle of the back translation technique is to generate a new sentence which has the same meaning as the original one by exploiting the semantic variances used for training the translator. This is a powerful technique for data augmentation in natural language processing [1]. In this paper, two translation systems are employed that are Google Translate [20] and Systran API [21]. Noted that French is used as the target language. Google Translate will convert the original sentence in English into a new sentence in French. And then, Systran is used for translating the output sentence of Google Translate back to English language. By utilizing two different translators, the augmented sentences are much different from the original ones and enrich the dictionary. For example, given a description sentence: “A red car drove through an intersection”. When using back translation, we obtain a new sentence “A red car crossed an intersection.”

3.4. Feature extraction

After pre-processing step, images and description sentences are fed forward to the next stage to extract worthwhile features. While appearance and motion images are the input of the visual branch, appearance and motion description sentences are forwarded to the textual branch. In the proposed framework, CLIP model is used for both visual and textual embeddings. This model is widely used in many image-text matching tasks because of effective representation the relation between visual and textual information, such as image captioning, visual question answering, etc. It is important to note that this framework can use single or dual CLIP models to handle both appearance and motion information.

3.4.1. Visual feature extraction

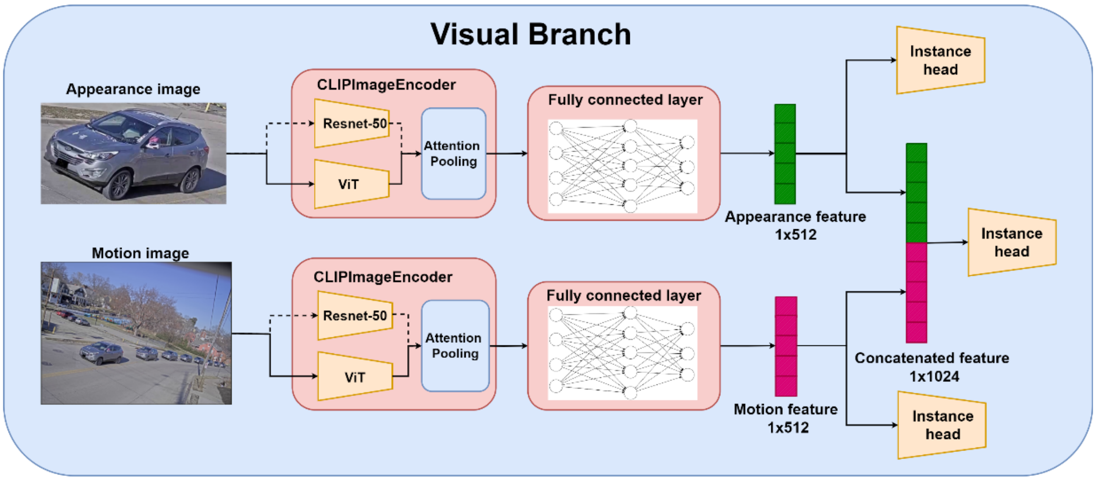


Figure 4. Visual feature extraction branch.

As aforementioned, a given vehicle tracklet is forwarded to pre-processing step to generate two augmented images, appearance and motion images. These two images are resized to 224×224 pixels, split into some patches and fed forwarded to CLIP image encoder. Because the motion map is created by overlaying vehicle bounding boxes onto a background image, it often appears quite different from a natural image. To ensure the effectiveness of the deep-learning model, appearance and motion images should be processed in two separate streams. Consequently, this work proposes a DualCLIP framework, which incorporates two independent CLIP models. Each stream processes appearance and motion information individually, ensuring that both types of data are handled optimally.

In this framework, ViT structure [6] is used as the backbone for feature extraction in both streams. The output is a 768-dimensional feature vector, representing the image patch, ensuring a compact and informative representation for both appearance and motion data. Additionally, a two-layer fully connected layer with ReLU function is added to transform 768-dimension features into 512-dimension ones to reduce the computation complexity. After that, two feature vectors corresponding to appearance and motion images are concatenated to form the overall feature vector. As a result, there are a total of three feature vectors including appearance, motion, and concatenated features at the output of visual branch. Figure 4 illustrates the way to extract these above feature vectors in the visual branch.

3.4.2. Textual feature extraction

Similar to the visual branch, the text branch also extracts features from two different types of description sentences: (1) sentences describing the vehicle's appearance and (2) sentences containing information about the vehicle's movement. Figure 5 describes the block diagram for the textual branch of the proposed model. A CLIPTextEncoder block is used for both appearance and motion sentences. In CLIPTextEncoder, BPE (Byte pair encoding) [22] encode the description sentences and Roberta [23] containing 12 transformer layers with multi-head attention extracts their crucial features. A description sentence is separated into some tokens, and then, two special tokens [EOS] and [SOS] are added. Feature vectors corresponding to token [EOS] are used for representing overall description sentence. The output of the encoding block is passed through a Fully Connected layer, which is added to reduce the dimensionality of the feature vector to 512 and enhance the learning ability.

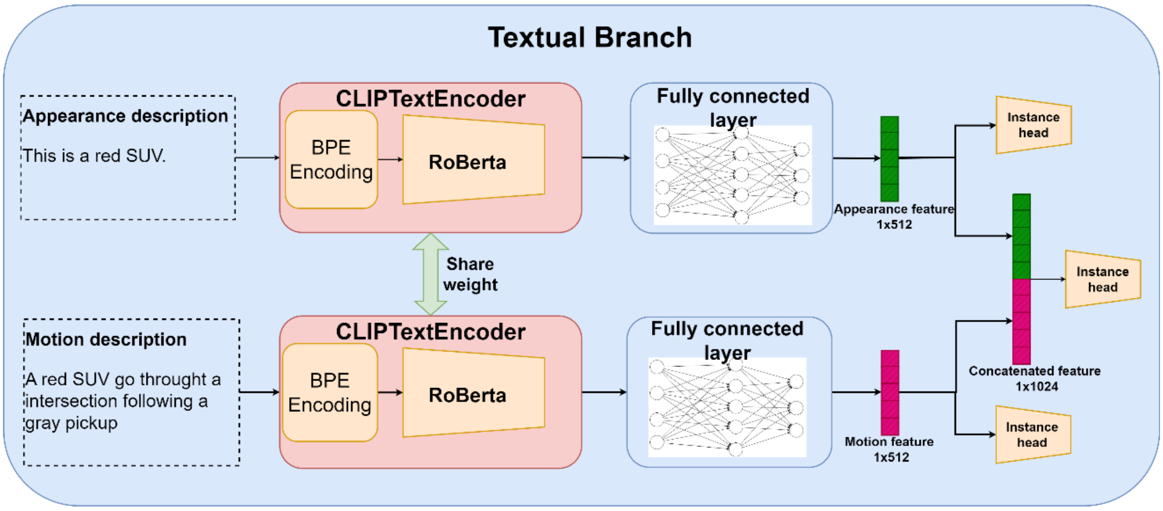


Figure 5. Textual feature extraction branch.

3.5. Image-text alignment

One of the most important challenges in multi-modal data processing is the feature alignment task. After the feature extraction step, the visual and textual feature vectors are projected onto a common subspace to learn cross-modal similarity. In this work, three loss functions including Circle loss, InfoNCE loss, and Instance loss are exploited to optimize the network weights. The formulations of these loss functions will be presented as follows.

- InfoNCE loss [22] (Information Noise-Contrastive Estimation) is also designed to train a model to distinguish between similar and dissimilar data points, which is crucial for learning useful and discriminative representations in contrastive learning. The crucial objective of InfoNCE loss is to encourage the model to assign a higher similarity score to the positive pair compared to the negative pairs. This is achieved by minimizing the negative log-likelihood of the correct pair being the most similar among all candidate pairs. In this work, InfoNCE loss is calculated by the sum of InfoNCE loss from image to language (InfoNCE$_{i2t}$) and InfoNCE loss from language to image InfoNCE$_{t2i}$) as the following Equations:

$$\mathrm{InfoNCE}_{i2t} = \frac{1}{N}\sum_{i=1}^{N} -\log \frac{\exp\left(\cos\left(f_i^{\mathrm{img}}, f_i^{\mathrm{text}}\right)/\tau\right)}{\sum_{j}^{N}\exp\left(\cos\left(f_i^{\mathrm{img}}, f_j^{\mathrm{text}}\right)/\tau\right)}, \tag{1}$$

$$\mathrm{InfoNCE}_{t2i} = \frac{1}{N}\sum_{i=1}^{N} -\log \frac{\exp\left(\cos\left(f_i^{\mathrm{text}}, f_i^{\mathrm{img}}\right)/\tau\right)}{\sum_{j}^{N}\exp\left(\cos\left(f_i^{\mathrm{text}}, f_j^{\mathrm{img}}\right)/\tau\right)}, . \tag{2}$$

$$\mathrm{InfoNCE} = \mathrm{InfoNCE}_{i2t} + \mathrm{InfoNCE}_{t2i} \tag{3}$$

where f_i^{img} and f_j^{text} denote the extracted features from the image and language domains; N is the number of data samples; τ is *temperature learnable parameter*, which is used for tunning the slope of the output distribution and is chosen equal 0.2 in this work.

- Circle loss [23] a variation of the Contrastive loss, one of the most useful loss functions for similarity learning task. In the image-text alignment task, positive and negative pairs refer to the matched and mismatched image-text pairs, respectively. As we known, one of drawbacks of some traditional loss function is to penalize the similarity for positive and negative pairs equally regardless of their differences. To overcome this issue, Circle loss is introduced to optimize the similarity of positive and negative pairs simultaneously, aiming for a more discriminative feature space. Moreover, Circle loss provides a flexible weighting mechanism that assigns different levels of importance to positive and negative pairs based on their similarity scores.

- Instance loss [24] is treated as cross-entropy function for discriminative learning task. This loss function was first introduced for learning representations between images and languages. After that, it has been widely applied in most of the problems of searching between different information domains. Instance loss is calculated on both image and language domains as in Equations (4) and (5). Noted that there are two data domains and three types of features, including appearance features, motion features, and concatenated features. Consequently, there are a total of six loss functions for this instance. The final instance loss function will be the sum of all the loss functions for the two domains.

$$L_{\text{img}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{C} y_{i,k}\log\frac{e^{f_{i,k}^{img}}}{\sum_{j=1}^{C} e^{f_{i,j}^{img}}}, \tag{4}$$

$$L_{\text{text}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{C} y_{i,k}\log\frac{e^{f_{i,k}^{text}}}{\sum_{j=1}^{C} e^{f_{i,j}^{text}}}, \tag{5}$$

$$L_{\text{Instanceloss}} = L_{\text{img}} + L_{\text{text}} \tag{6}$$

where *N* is the number of feature vectors, *C* is the number of vehicles in the examined dataset. The final loss function is a combination of all three above ones with different weights as indicated in the following Equation. The weight assigned for each loss function is chosen through a number of practical experiments.

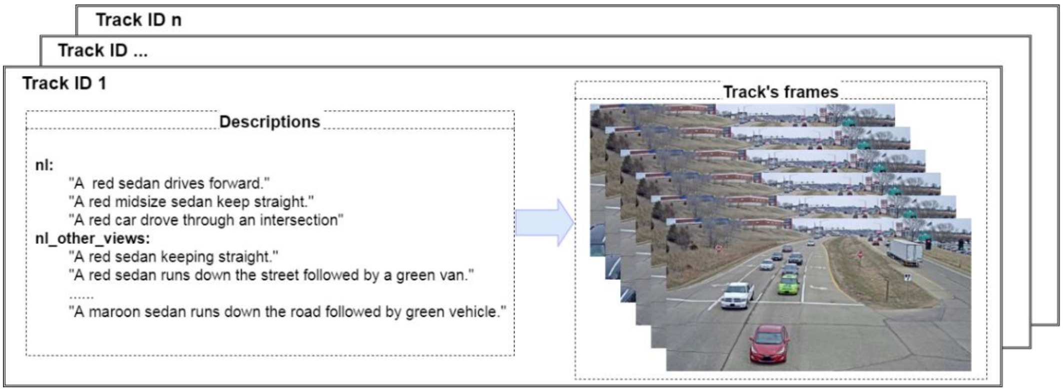
$$L_{\Sigma} = \left[0.2\times L_{\text{Circleloss}} + L_{\text{InfoNCEloss}}\right] + 0.5\times L_{\text{Instanceloss}} \tag{6}$$

4. EXPERIMENT AND RESULTS

4.1. Dataset and evaluation metrics

4.1.1. Dataset

In this study, we employ the dataset provided by the 6-th AI City Challenges Track 2: Tracked-Vehicle Retrieval by Natural Language Descriptions. AI City Challenge is a competition held annually with the main goal of applying AI (Artificial Intelligence) to enhance operational efficiency in city environments. Particularly, in the challenge track "Tracked-vehicle retrieval by natural language descriptions", each participating team is required to find out the most suitable vehicle tracklets for a given natural language description. A vehicle tracklet is defined as a sequence of detections or observations of a vehicle as it moves through a scene, typically captured by a video or camera system. In this task, the natural language descriptions provide both static and dynamic properties of the target vehicles. Static properties provide information about vehicle type, size and color. Meanwhile dynamic properties involve the vehicle's motion, the relations between the target vehicle and others or the environment.



nl:
"A red sedan drives forward."
"A red midsize sedan keep straight."
"A red car drove through an intersection"
nl_other_views:
"A red sedan keeping straight."
"A red sedan runs down the street followed by a green van."
......
"A maroon sedan runs down the road followed by green vehicle."

Figure 6. Descriptive sentences for a tracklet in 6-th AI City Challenges Track 2.

This dataset is captured by various static cameras installed at roads and intersections in America. It contains a total of 2,339 vehicle tracklets, each annotated with three natural language descriptions, tagged as "nl." Notably, the training set also includes several additional descriptions for each tracklet, tagged as "nl_other_views". These descriptions typically involve information about appearance, motion, and the relationship between the interested vehicle and other vehicles as well as the surrounding environment. Each frame of a vehicle tracklet contains a bounding box that indicates the location of the vehicle of interest. For the training phase, 2,155 tracklets are used, while the remaining 184 tracklets are utilized for the test phase. Figure 6 illustrates a sample pair of image sequences and description sentences from the examined dataset. As mentioned in the work of Feng et al. [25], CityFlow-NL is the first benchmark for multi-view multi-target tracking using natural language. With a large number of vehicle tracklets and annotated natural language descriptions, evaluations on this dataset can generalize effectively to other environments and datasets.

4.1.2. Evaluation metric

To evaluate the performance of natural language-based vehicle retrieval framework, Mean Reciprocal Rank is used as the main evaluation metric. The formulation of this metric is depicted in the following Equation:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i}, \tag{7}$$

where $rank_i$ involves rank position of the matched tracklet corresponding to the i-th text description, and Q is the set of description sentences. Additionally, Rank@5 and Rank@10, which represent the matching rates at rank-5 and rank-10 respectively, are also used in the evaluation phase. It is worth noting that the evaluation metrics mentioned above are the most widely used for evaluating the performance of proposed frameworks in retrieval tasks. Additionally, these metrics have been adopted by AI-City Challenges. Therefore, in this work, we use these evaluation metrics to compare our results with those of others.

4.2. Experimental scenarios

To prove the effectiveness of the proposed framework when dealing with natural language-based vehicle retrieval, four experimental scenarios are considered, namely SingleCLIP-

Baseline, SingleCLIP-Fused, DualCLIP-Baseline and DualCLIP-Fused. Some notable points of these four scenarios will be described below.

- SingleCLIP-Baseline: In this case, only a single CLIP model is used for both appearance and motion embeddings. As aforementioned, after feature extraction step, three kinds of features are generated including appearance, motion, and fused features. In this scenario, all of these kinds of features are used for the training phase, however, only one of them is utilized for the test phase.

- SingleCLIP-Fused: This scenario uses the same framework as SingleCLIP-Baseline; however, all three types of features are utilized during the test phase. It is worth noting that the fused features are generated using two different methods: mean fusion and concatenation.

- DualCLIP-Baseline: For this scenario, the proposed framework employs two separate CLIP models to handle appearance and motion data independently. Notably, in this scenario, the CLIPTextEncoder for both appearance and motion sentences operate with shared weights, which enhances the performance during the training phase.

- DualCLIP-Fused: Similar to DualCLIP-Baseline scenario except all three kinds of features are used in the test phase.

In this study, the proposed models are trained in 81 epochs with a batch size of 32; the learning rate is initialized to 0.0001 and reduced by 10 times every 10 epochs. The Adam optimization algorithm is used with a weight decay parameter of 0.01.

4.3. Ablation study

Table 1. Some obtained results in the ablation study.

Scenario	Appearance feature	Motion feature	Concat feature	Dual extractors	Fusion method	MRR (%)	Rank@5 (%)	Rank@10 (%)
SingleCLIP-Baseline	✓					23.78	32.57	47.51
		✓				26.68	37.58	48.51
			✓			35.69	48.50	64.77
SingleCLIP-Fused	✓	✓	✓		mean	**35.50**	**48.50**	**65.52**
	✓	✓	✓		concat	**37.89**	**50.02**	**65.57**
DualCLIP-Baseline	✓			✓		30.31	40.01	62.87
		✓		✓		34.18	48.47	64.32
			✓	✓		44.50	66.66	78.15
DualCLIP-Fused	✓	✓	✓	✓	mean	**43.47**	**61.28**	**77.26**
	✓	✓	✓	✓	concat	**46.63**	**67.02**	**81.82**

To evaluate the role of each kind of features in natural language-based vehicle retrieval task, we conducted an ablation study in which only one of three kinds of features (appearance, motion, and fused features) is used in the test phase. Both above scenarios are investigated in this experiment. Some obtained results in terms of MRR, Rank@5, and Rank@10 are shown in Table 1. As seen in this Table, some conclusions can be drawn as follows.

Firstly, when comparing the three types of features, the appearance feature exhibits the lowest performance, while the fused feature achieves the highest performance. In the SingleCLIP-Baseline scenario, the MRR metric improves by 2.90% with motion features and by 11.91% with fused features, compared to using solely appearance features. In the DualCLIP-Baseline scenario, these improvements are 3.87% and 14.19%, respectively. These results can be explained by the fact that fused features combine both appearance and motion information, providing more comprehensive data that enhances the matching rate in the natural language-based vehicle retrieval task.

Secondly, when exploiting all of three kinds of features for both training and test phase, our framework achieves some impressive MRR score of 35.50% and 37.89% in SingleCLIP-Fused scenario; 43.47% and 46.63% in DualCLIP-Fused scenario corresponding to the cases of fused features generated by average and concatenated mechanism. From these results, we can realize that by leveraging all three kinds of features, MRR score achieves some significant improvements of 11.72% and 14.11% in SingleCLIP-Fused scenario, 13.16% and 16.32% in DualCLIP-Fused scenario when applying average and concatenated mechanisms to create fused features, respectively. However, using average features in both SingleCLIP-Fused and DualCLIP-Fused does not provide a better result compared to the use of only concatenated features in SingleCLIP-Baseline and DualCLIP-Baseline scenarios.

Finally, in the case of dealing with appearance and motion data independently through the two CLIP models, the performance of the proposed framework is remarkably increased in all terms of MRR, Rank@5, and Rank@10. When using DualCLIP-Fused, these values are 46.63%, 67.02%, and 81.82% with some boosts of 8.74%, 17.00%, and 16.25% compared to those in case of SingleCLIP-Fused. Due to the differences in terms of appearance and motion data in both visual and textual streams, these kinds of data should be processed independently in two CLIP models.

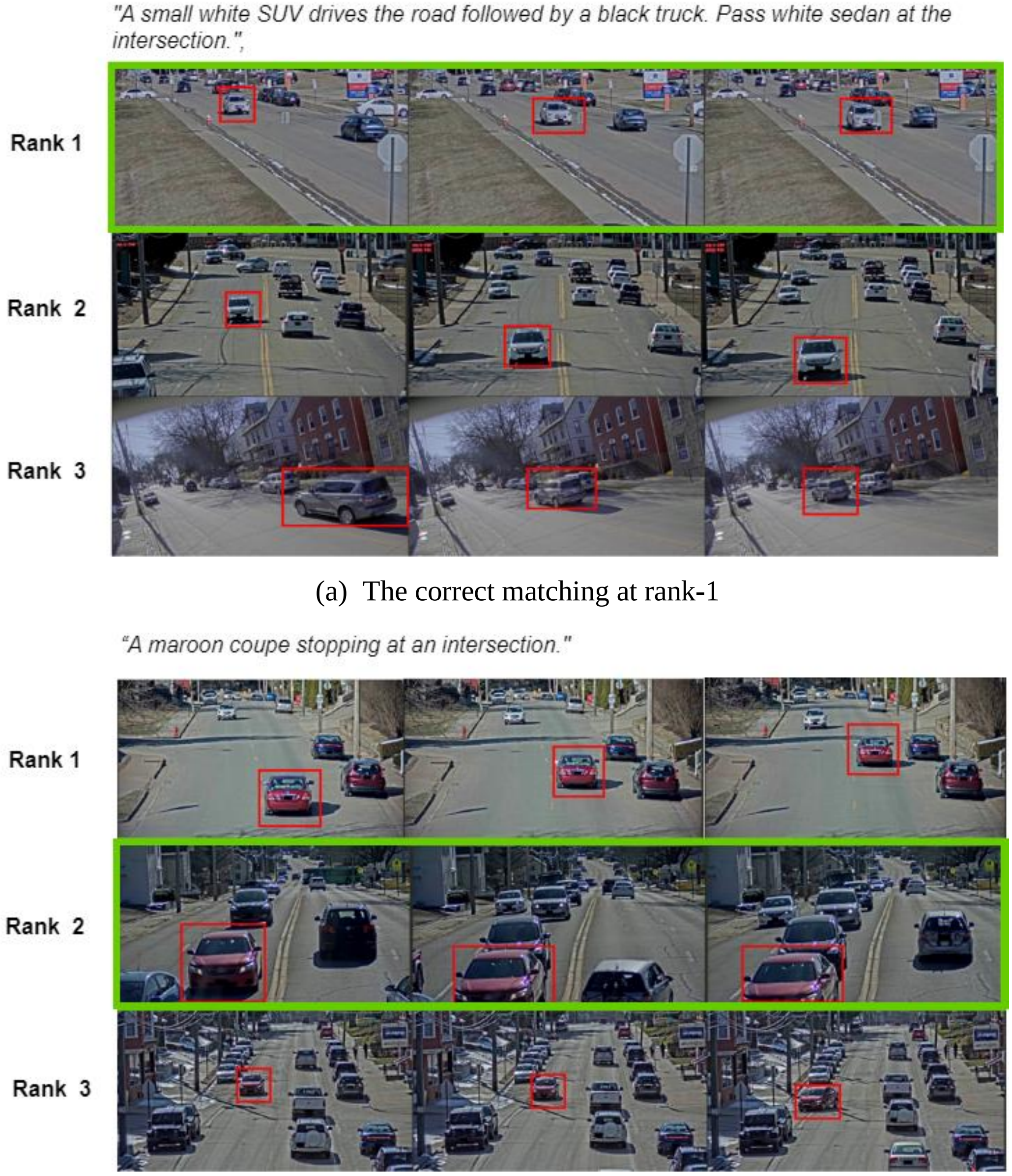
4.4. The comparison with SOTA methods

Table 2. The comparison between the obtained results of the proposed framework and those of the SOTA methods.

Models	**MRR (%)**	**Rank@5 (%)**	**Rank@10 (%)**
Alibaba-UTS-ZJU [1]	24.04	38.37	46.75
ViTAA+Post-processing [2]	23.72	37.37	43.34
OMG [3]	38.08	47.63	62.09
SSM Ensemble [26]	54.65	71.76	82.56
Ours	**46.63**	**67.02**	**81.82**

Table 2 presents a comparison between the results of the proposed framework and those of several state-of-the-art (SOTA) methods, with our results highlighted in bold. According to the table, our framework outperforms almost considered methods, except for the SSM Ensemble [26]. Notably, our method achieves significantly better results compared to the Alibaba-UTS-ZJU [1] and ViTAA+Post-processing [2] methods. Specifically, our framework shows improvements of 8.55%, 19.39%, and 19.73% over the OMG method. Although the SSM Ensemble framework achieves the best overall results, it involves multiple models, leading to increased computational complexity and longer processing times. Moreover, the matching rate at Rank@10 when applying our framework is approximately to that of SSM Ensemble method. This suggests that although the first 10 returned results are comparable between the two methods, SSM Ensemble requires significantly more complex computations to achieve these results. From the above analysis, we can see clearly the effectiveness of the proposed framework in natural language-based tracked-vehicle retrieval.

Figure 7 shows some examples for vehicle tracklet retrieval based on a given natural language description sentence when applying the proposed framework. The correct matching is remarked by green bounding boxes (best view in color). In the first case, the correct matching is found at the first rank and in the other case, it is at the second rank. The proposed framework has a good performance in vehicle tracklet retrieval task based on appearance information, however, it does not exploit motion information effectively.


"A small white SUV drives the road followed by a black truck. Pass white sedan at the intersection.",
Rank 1
Rank 2
Rank 3
"A maroon coupe stopping at an intersection."
Rank 1
Rank 2
Rank 3

(a) The correct matching at rank-1

(b) The correct matching at rank-2

Figure 7. Some examples of vehicle tracklet retrieval.

5. CONCLUSIONS AND FUTURE WORKS

In this paper, we introduce a CLIP-based dual-stream framework for tracked-vehicle retrieval using natural language descriptions as queries. The framework is composed of three key stages: pre-processing, feature extraction, and image-text alignment. The pre-processing stage generates appearance and motion data in both the visual and language domains, with additional augmentation techniques applied to enrich the training data. After feature extraction, three types of features - appearance, motion, and concatenated - are generated. These features are then projected onto a common space to learn image-text similarity. Notably, the framework handles appearance and motion data using either a single or dual CLIP models, referred to as the SingleCLIP or DualCLIP scenarios. The best results are achieved with the DualCLIP model using all three types of features during evaluation, in what we call the DualCLIP-Fused scenario. The proposed framework yields some results comparable to state-of-the-art methods, with an MRR score of 46.67%, a Rank@5 of 67.02%, and a Rank@10 of 81.82%. Although the proposed framework achieves convincing results, it still has some limitations due to the use of two independent CLIP models for processing appearance and motion data. In future work, we plan to explore a novel framework that integrates both types of data within a single model. Additionally, in this paper, we focus solely on English-based vehicle retrieval and do not address the adaptation of the proposed framework to other languages, such as Vietnamese. For this, two approaches could be considered: (1) training models on Vietnamese descriptions and (2) using a pre-processing step to translate Vietnamese descriptions into English through machine translation models. These approaches will also be discussed in future work.

ACKNOWLEDGMENT

This research is funded by University of Transport and Communications (UTC) under grant number T2023-DT-010TD.

REFERENCES

[1]. S. Bai, Z. Zheng, X. Wang, J. Lin, Z. Zhang, C. Zhou, H. Yang, Y. Yang, Connecting Language and Vision for Natural Language-Based Vehicle Retrieval, in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2021. https://doi.org/10.1109/CVPRW53098.2021.00455
[2]. Quang-Huy Can, Hong-Quan Nguyen, Thi-Ngoc-Diep Do, Hoai Phan, Thuy-Binh Nguyen, Thi Thanh Thuy Pham, Thanh-Hai Tran, Thi-Lan Le, Exploring the Effect of Vehicle Appearance and Motion for Natural Language-Based Vehicle Retrieval, in Asian Conference on Intelligent Information and Database Systems, 2022. https://doi.org/10.1007/978-981-19-8234-7_5
[3]. Y. Du, B. Zhang, X. Ruan, F. Su, Z. Zhao, H. Chen, OMG: Observe Multiple Granularities for Natural Language-Based Vehicle Retrieval, in Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2022. https://doi.org/10.1109/CVPRW56347.2022.00352
[4]. A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, Learning transferable visual models from natural language supervision, in: International conference on machine learning, pp. 8748-8763, 2021.
[5]. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in 31st International Conference on Neural Information Processing Systems, 2017. https://doi.org/10.48550/arXiv.1706.03762
[6]. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, in 2021 International Conference on Learning Representations, 2021.
[7]. Jiasen Lu, Dhruv Batra, Devi Parikh, Stefan Lee, ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks, in 33rd International Conference on Neural Information Processing Systems, 2019. https://doi.org/10.48550/arXiv.1908.02265
[8]. H. Fang, P. Xiong, L. Xu, Y. Chen, CLIP2Video: Mastering Video-Text Retrieval via Image CLIP, in arXiv preprint arXiv:2106.11097, 2021. https://doi.org/10.48550/arXiv.2106.11097
[9]. C. Scribano, D. Sapienza, G. Franchini, M. Verucchi, M. Bertogna, All You Can Embed: Natural Language based Vehicle Retrieval with Spatio-Temporal Transformers, in Conference on Computer Vision and Pattern Recognition, 2021. https://doi.org/10.1109/CVPRW53098.2021.00481
[10]. Kyunghyun Cho, B van Merrienboer, Caglar Gulcehre, F Bougares, H Schwenk, Yoshua Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, in

Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), 2014. https://doi.org/10.3115/v1/D14-1179

[11]. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep Residual Learning for Image Recognition, in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, 2016. https://doi.org/10.1109/CVPR.2016.90

[12]. Sangrok Lee, Taekang Woo, Sang Hun Lee, SBNet: Segmentation-based Network for Natural Language-based Vehicle Search, in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 2021. https://doi.org/10.1109/CVPRW53098.2021.00457

[13]. P. Khorramshahi, S. S. Rambhatla, R. Chellappa, Towards Accurate Visual and Natural Language-Based Vehicle Retrieval Systems, in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 2021. https://doi.org/10.1109/CVPRW53098.2021.00472

[14]. Tam Minh Nguyen, Quang Huu Pham, Linh Bao Doan, Hoang Viet Trinh, Viet-Anh Nguyen, Viet-Hoang Phan, Contrastive Learning for Natural Language-Based Vehicle Retrieval, in 2021 Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 2021. https://doi.org/10.1109/CVPRW53098.2021.00480

[15]. Clint Sebastian; Raffaele Imbriaco, Panagiotis Meletis, Gijs Dubbelman, Egor Bondarev, Peter H.N. de With, TIED: A Cycle Consistent Encoder-Decoder Model for Text-to-Image Retrieval, in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 2021. https://doi.org/10.1109/CVPRW53098.2021.00467

[16]. Bird Steven, Ewan Klein, Edward Loper, Natural language processing with Python: analyzing text with the natural language toolkit, O'Reilly Media, Inc, 2009.

[17]. Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, Balaji Lakshminarayanan, AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty, in arXiv:1912.02781v2, 2019. https://doi.org/10.48550/arXiv.1912.02781

[18]. Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, David Lopez-Paz, Mixup: Beyond Empirical Risk Minimization, in arXiv preprint arXiv:1710.09412, 2018. https://doi.org/10.48550/arXiv.1710.09412

[19]. Sennrich, R., B. Haddow, A. Birch, Improving Neural Machine Translation Models with Monolingual Data, in arXiv preprint arXiv:1511.06709, 2016. https://doi.org/10.18653/v1/P16-1009

[20]. Cloud translation documentation, [Online]. Available: https://cloud.google.com/translate/docs. [Accessed 22 August 2024].

[21]. Translate any application with SysTran API, [Online]. Available: https://www.systransoft.com/translation-products/translate-api/. [Accessed 22 August 2024].

[22]. Rico Sennrich, Barry Haddow, Alexandra Birch, Neural Machine Translation of Rare Words with Subword Units, in 54th Annual Meeting of the Association for Computational Linguistics, 2016. https://doi.org/10.18653/v1/P16-1162

[23]. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, in arXiv preprint arXiv:1907.11692, 2019.

[24]. Z. Zheng, L. Zheng, M. Garrett, Y. Yang, M. Xu, Y.-D. Shen, Dual-path convolutional image-text embeddings with instance loss, ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 16(2) (2020) 1–23. https://doi.org/10.1145/3383184

[25]. Feng, Qi Ablavsky, Vitaly Sclaroff, Stan, CityFlow-NL: Tracking and retrieval of vehicles at city scale by natural language descriptions, arXiv preprint arXiv:2101.04741, 2021. https://doi.org/10.48550/arXiv.2101.04741

[26]. Chuyang Zhao, Haobo Chen, Wenyuan Zhang, Junru Chen, Symmetric Network with Spatial Relationship Modeling for Natural Language-based Vehicle Retrieval, in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2022. https://doi.org/10.1109/CVPRW56347.2022.00364