ITC 1/53

Information Technology and Control
Vol. 53 / No. 1 / 2024
pp.98-114
DOI 10.5755/j01.itc.53.1.35095

TSIC-CLIP: Traffic Scene Image Captioning Model Based on Clip

Received 2023/09/13

Accepted after revision 2023/11/24

HOW TO CITE: Zhang, H., Xu, C., Xu, B., Jian, M., Liu, H., Li, X. (2024). TSIC-CLIP: Traffic Scene Image Captioning Model Based on Clip. *Information Technology and Control*, 53(1), 98-114. https://doi.org/10.5755/j01.itc.53.1.35095

TSIC-CLIP: Traffic Scene Image Captioning Model Based on Clip

Hao Zhang , Cheng Xu, Bingxin Xu

Beijing Key Laboratory of Information Service Engineering, Beijing Union University, Beijing, China;
Institute for Brain and Cognitive Sciences, College of Robotics, Beijing Union University, Beijing, China
e-mails: enjoyzh@foxmail.com; xucheng@buu.edu.cn; xubingxin@buu.edu.cn

Muwei Jiane

School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan, China
e-mail: jianmuweihk@163.com

Hongzhe Liu, Xuewei Li

Beijing Key Laboratory of Information Service Engineering, Beijing Union University, Beijing, China;
Institute for Brain and Cognitive Sciences, College of Robotics, Beijing Union University, Beijing, China
e-mails: liuhongzhe@buu.edu.cn; lixuewei@buu.edu.cn

Corresponding author: xucheng@buu.edu.cn (C. Xu); xubingxin@buu.edu.cn (B. Xu)

Image captioning in traffic scenes presents several challenges, including imprecise caption generation, lack of personalization, and an unwieldy number of model parameters. We propose a new image captioning model for traffic scenes to address these issues. The model incorporates an adapter-based fine-tuned feature extraction part to enhance personalization and a caption generation module using global weighted attention pooling to reduce model parameters and improve accuracy. The proposed model consists of four main stages. In the first stage, the Image-Encoder extracts the global features of the input image and divides it into nine sub-regions, encoding each sub-region separately. In the second stage, the Text-Encoder encodes the text dataset to obtain text features. It then calculates the similarity between the image sub-region features and encoded text features, selecting the text features with the highest similarity. Subsequently, the pre-trained Faster RCNN model extracts local image features. The model then splices together the text features, global image features, and local image features to fuse the multimodal information. In the final stage, the extracted features are fed into the Captioning model, which effectively fuses the different features using a novel global weighted attention pooling layer. The Captioning model then generates natural language image captions. The proposed model is evaluated on the MS-COCO dataset, Flickr 30K dataset, and BUUISE-Image dataset, using mainstream evaluation metrics. Experiments demonstrate significant improvements across all evaluation metrics on the public datasets and strong performance on the BUUISE-Image traffic scene dataset.

KEYWORDS: Contrastive learning, Deep learning, Image captioning, Traffic scene, Transformer.

1. Introduction

The image captioning task integrates computer vision and natural language processing to generate descriptive captions for visual inputs. With the development of Artificial Intelligence, image captioning techniques have been increasingly applied in various fields. Such as medicine [5, 28], fashion and e-commerce [19], aided industry [36], and tourism [4]. Moreover, this technology shows immense potential in traffic applications. The traditional computer vision task mainly detects and classifies targets such as pedestrians, obstacles, signage, etc. It then enables traffic monitoring [3], road condition analysis [24], and automated driving [9]. However, these methods lack an understanding of relationships between detected entities. Therefore, we propose converting traffic scene keyframes into natural language captions and using richer semantic information can replace detecting individual entities. This approach shows promise for assisting visually impaired individuals [12, 23], driving safety [1], and describing traffic accidents [18].

Traditional image captioning methods rely on template and rule-based methods, which cannot handle context and generate diverse captions. Most current mainstream image captioning methods are based on deep learning, commonly adopting an encoding-decoding framework. The earliest deep learning-based generalized image captioning model [33] extracts image features by convolutional neural networks and then inputs these features into recurrent neural networks to generate natural language captions, significantly improving over traditional methods. Therefore, the current mainstream image captioning methods mainly focus on deep learning. We base on Stefanini et al. [30] the title of the point of view of the mainstream generalized image captioning model for research and base the different decoders into two categories, respectively, based on Long Short Term Memory methods and based on the Transformer methods. The image captioning methods mainly use the structure of encoder and decoder, where the encoder is responsible for extracting the image features, and the decoder is responsible for receiving the image features and converting them into captions corresponding to the image.

In 2015, Xu et al. [41] proposed an LSTM-based method incorporating a visual attention mechanism for the first time, which can selectively focus on the image preference region. The model extracts the image features by CNN and generates the Caption by LSTM, which better solves the problem that RNN (Recurrent Neural Network) Series Networks can only maintain short-term memory. Building on prior work, Anderson et al. [2]extract local target features using a pre-trained Faster R-CNN model and compute average feature representations to focus on salient image regions. Top-down Attention LSTM and Language LSTM are then utilized to obtain averaged and target-specific features for generating image captions. However, LSTM models are prone to vanishing and exploding gradient issues when processing long sequences due to the limited dimensionality of the LSTM memory units. Following the success of Transformer models [32] for natural language processing in 2017, numerous Transformer-based approaches emerged for image captioning. Unlike LSTMs, Transformers can directly capture long-range dependencies across the full sequence via self-attention, better capturing contextual information. Consequently, most state-of-the-art image captioning methods now utilize Transformer-based architectures. Zhu et al. [45] first proposed a CNN-Transformer framework. However, by only using global image features as input to the decoder, this method fails to capture fine-grained contextual details, instead encoding irrelevant information that yields inaccurate and verbose captions. To address these limitations, Xian et al. [39] propose a Transformer-based method that optimizes region feature representations in the encoding stage using mesh features and geometric information. Wang et al. [37] pioneer the use of Swin Transformers as encoders for image captioning, helping to address prior limitations in this field. They incorporate global visual features into each decoder block to enhance cross-modal interactions and more effectively capture global context. Cornia et al. [6] aim at this problem, encoding low-level and high-level object features as prior knowledge and using prior knowledge to assist in semantic captioning at different levels in the decoding stage. Luo et al. [15] proposed another transformer-based image captioning method using a diffusion model. A cross-modal retrieval module first retrieves sentences highly similar to the image as semantic priors. Multiple diffusion transformers are then stacked in a cascade. Each diffusion transformer conditionally generates output based on the previous model's output to better capture dependencies between words. Therefore, the current transformer-based image captioning method calculates the interaction between each position due to the self-attention mechanism, leading to many parameter models. Most of the above models use a single visual local feature or visual global feature as the input of the decoder, which may cause problems such as inaccurate captions in complex traffic scenes. This problem has been improved with the emergence of contrastive learning methods, large-scale pre-training models break through the constraints between text and image, and the categories of object detection cover a larger amount. The image vector extraction method based on CLIP (Contrastive Language-Image Pre-training) has been widely used [25].

OpenAI proposes CLIP, and its core idea is to pre-train utilizing comparative learning, which maps image and text embeddings to a common feature space by calculating the similarity between image and text. Its application areas are wide, such as image classification, image retrieval, image description, etc. Mokady et al. [17] first proposed a CLIP-based image captioning method by extracting image features from CLIP and using a mapping network to connect the two modalities of image and language. They only fine-tuned the mapping network and finally generated image captions from pre-trained GPT-2. However, this method only extracted visual feature information through CLIP and mapped it to the textual space without considering the intrinsic contextual semantic information of the image. As a result, the generated captions lack details. Furthermore, using a single mapping network to align the two modalities needs further refinement. Subsequently, Nukrai et al. [22] addressed this problem by proposing an image captioning model that matches the two modalities' mapping by injecting noise into the training process. This results in a more accurate alignment of the two modalities. However, their method still needs to consider the contextual semantic information inherent in the image. On the other hand, Dai et al. [8] proposed a method to align CLIP's multimodal encoder and BART's text encoder to the same multimodal space. They used a cross-modal LM loss to harmonize the performance of the BART encoder and decoder. Cho et al. [7] proposed a training strategy to improve the descriptive power by maximizing the multimodal similarity score of CLIP and fine-tuning its text encoder. Current CLIP-based image captioning methods mostly use pre-trained weights directly, and the model is fine-tuned directly on the dataset, or only the final fully connected layer is trained. This could produce overfitting and lead to forgetting the weights when the dataset is too small. Therefore, the above methods cannot maximize CLIP's performance on image captioning tasks for traffic scenes and need more personalization.

To address the abovementioned challenges, we propose the Traffic Scene Image Captioning model based on Contrastive Language-Image Pretraining (TSIC-CLIP). The model consists of two main models. Firstly, the feature extraction model utilizes a fine-tuned CLIP model to extract global image features. It also leverages a pre-trained Faster R-CNN to extract local image features and a CLIP-based text retrieval module to obtain textual features of image sub-region descriptions. These feature vectors are then concatenated as inputs to the captioning model to combine the local features with the global features and textual features of image sub-regions, enhancing the effectiveness and efficiency of captioning. The textual features of image sub-regions enrich semantic information, leading to more accurate captions with fewer redundancies. Additionally, we freeze the CLIP model parameters and design a novel adapter layer for fine-tuning the model on both public and our traffic scene datasets called BUUISE-Image. This ensures the CLIP model's robustness and generalizability while adapting it to traffic scenes, resulting in a more personalized model.

In the captioning model, we replace the attention mechanism layer in the Transformer with a Global Weighted Attention Pooling (WGA-Pooling) layer as the token mixer. First, the features extracted from the feature extraction model are word-embedded. These embedded features are then fed into the WGA-Pooling layer. The WGA-Pooling layer aims to mix feature information while accounting for contextual dependencies between sequences, which allows the model to better capture long-range input relationships and significantly reduce parameters. The pooled features are fed into a series of fully connected layers. Ultimately, the model generates natural language text captions that more closely align with the underlying semantic information of the traffic scene.

Finally, we constructed the BUUISE-Image dataset, which focuses specifically on traffic scene image captioning. We evaluated the proposed method on the publicly available MS-COCO and Flickr30k datasets and our BUUISE-Image dataset. The experimental results demonstrate that the method performs excellently on public and proprietary datasets.

The paper is structured into five sections. Section 1 is the introduction, delineating the research background of image captioning in traffic scenes. It thoroughly explores the strengths and weaknesses of the generic and CLIP-based image captioning models. Furthermore, it offers a comprehensive summarisation and analysis of the encountered challenge. Lastly, the section highlights the innovations and enhancements incorporated in our work. Section 2 is the related work. It involves investigating and analyzing the potential and significance of image captioning techniques within traffic scenes. The section also encapsulates a summary of issues extracted from pertinent literature. In response to these issues, we curate the BUUISE-Image dataset tailored to traffic scenes, introducing the dataset itself. Section 3 is the methodology part. First, we summarize the framework of the model and outline its flow. Second, we divide the model into feature extraction and captioning models. On this basis, we detail the work's innovative aspects. Section 4 is the experiment and discussion part. In detail, we introduce the experimental environment, parameter configurations, commonly used datasets, evaluation metrics, and the self-built BUUISE-Image dataset. We also analyze and discuss the experimental results. Section 5 is the conclusions, and we summarize the contributions of our work, present the remaining deficiencies, and provide an outlook on future research directions.

2. Related Work

2.1. Research Related to Image Captioning in Traffic Scenes

With the rapid development of Artificial Intelligence, image captioning shows broad application prospects in many fields. Especially in the field of traffic, image captioning is playing an important role. In this section, image captioning in traffic scenes is investigated and analyzed regarding application scenes and methods.

Li et al. [13] have demonstrated that image captioning of traffic scenes can provide richer semantic information for Advanced Driver Assistance Systems (ADAS) to make decisions. Appropriate driving suggestions generated from captions can improve driver safety. Voykinska et al. [34] also suggested that a blind person can obtain the necessary information to understand the situation of an invisible target with the help of a trusted friend who describes the target. Thus, image captioning techniques can help blind people see the information in a traffic scene and thus avoid dangerous situations. Unlike previous methods, Xu et al. [40] proposed an end-to-end autonomous driving model. The model takes a sequence of video frames as input to train a model that maps visual information to driving operations. The method can observe previous self-motion state and traffic scene conditions from a monocular camera to generate image captions of the vehicle's future motion behaviours. This informs the user in advance and enhances the user's safety and driving experience. On the other hand, Mori et al. [20] proposed a method to alert drivers to risks by image captioning. Based on previous research, Mori et al. [21] proposed a method for interpreting automated driving decisions based on in-vehicle cameras. The method fuses the visual information captured by the camera and the acceleration and angular velocity information from the vehicle sensors. It uses them as inputs to the model, and the output interprets the vehicle's driving state. The method can effectively reduce the psychological burden on passengers and prevent accidents. Kim et al. [11] proposed an image captioning model for interpreting autonomous driving planning and control. Unlike previous methods, their model also considers the driver's attention. It generates captions for interpreting vehicle behavior by acquiring information about vehicle control parameters and visual information. On the contrary, Srihari et al. [29] proposed a semantic segmentation-based model for image captioning of traffic scenes, which can be used for labeling video captions of traffic scenes and autonomous driving assistance. Unlike others, Wu et al. [38] applied image captioning to traffic scene modelling. They divided the image into several sub-regions and generated corresponding captions. Finally, they performed modelling based on the captions of each region.

Overall, image captioning has shown great promise in traffic scenes, not only for assisted driving but also for improving the safety of the blind and the elderly and helping human users better understand and monitor the operating status of autonomous driving systems. In the future, the technology will be able to analyze images and videos taken by road surveillance cameras in real-time, detect traffic conditions and events, and generate text reports for traffic management authorities to analyze. This could significantly improve the efficiency of monitoring and managing complex traffic environments.

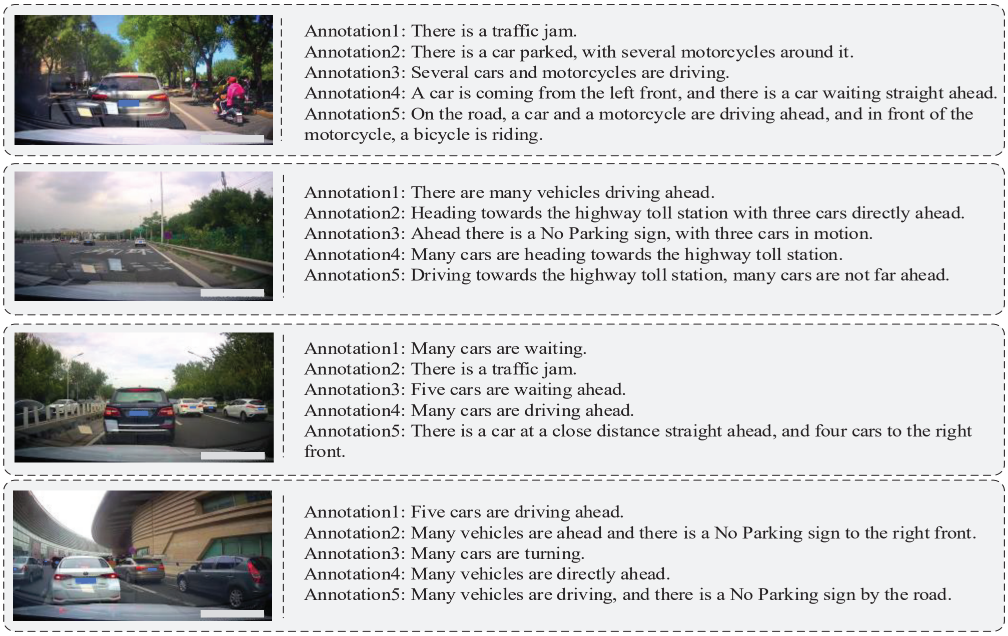
2.2. Image Captioning Dataset in Traffic Scenes

The traffic scene presents unique challenges due to its complexity, specificity, diversity, and uncertainty [16]. The complexity stems from the simultaneous presence of diverse vehicles, pedestrians, and traffic signals, variable road topologies, and highly interdependent traffic flows. The specificity shows in differing traffic conditions across environments like cities, villages, and highways and from weather, time of day, and seasons. Diversity arises from the possibility of multiple vehicle types, pedestrian behaviours, and road conditions, even including unknown objects beyond target detection categories. Uncertainty comes from random factors like weather, accidents, and unexpected events. Therefore, training models on traffic scene datasets remains essential for addressing the unique challenges in this domain. Seifi et al. [27] proposed a method to select ten classes of images in traffic scenes and their corresponding descriptions in the MS-COCO dataset and use them as a separate dataset for model training and evaluation. However, due to the complexity and diversity of the traffic scenes, using only ten classes may cause limitations of the model in practical applications. In contrast, Rochel et al. [26] created a 5,000 images dataset of traffic accidents, divided into 4,000 training and 1,000 test images with five matched captions each. However, the smaller size of datasets risks limitations in capturing the full diversity of traffic accidents. During model training and evaluation, the size of the dataset and the richness of its samples affect generalization ability and model performance.

A more extensive and diverse dataset may improve model robustness and accuracy for such a complex and variable domain. Therefore, we built the BUUISE-Image dataset dedicated to image captioning in traffic scenes, containing over 10,000 images, each with five manually generated captions. As shown in Figure 1, the BUUISE-Image dataset is self-col-

Figure1

BUUISE-Image Traffic Scene Image captioning Dataset



Annotation1: There is a traffic jam.
Annotation2: There is a car parked, with several motorcycles around it.
Annotation3: Several cars and motorcycles are driving.
Annotation4: A car is coming from the left front, and there is a car waiting straight ahead.
Annotation5: On the road, a car and a motorcycle are driving ahead, and in front of the motorcycle, a bicycle is riding.

Annotation1: There are many vehicles driving ahead.
Annotation2: Heading towards the highway toll station with three cars directly ahead.
Annotation3: Ahead there is a No Parking sign, with three cars in motion.
Annotation4: Many cars are heading towards the highway toll station.
Annotation5: Driving towards the highway toll station, many cars are not far ahead.

Annotation1: Many cars are waiting.
Annotation2: There is a traffic jam.
Annotation3: Five cars are waiting ahead.
Annotation4: Many cars are driving ahead.
Annotation5: There is a car at a close distance straight ahead, and four cars to the right front.

Annotation1: Five cars are driving ahead.
Annotation2: Many vehicles are ahead and there is a No Parking sign to the right front.
Annotation3: Many cars are turning.
Annotation4: Many vehicles are directly ahead.
Annotation5: Many vehicles are driving, and there is a No Parking sign by the road.

lected from Beijing, Tianjin, Vietnam, and other cities and is screened and cleaned. The dataset focuses on the accuracy and diversity of captions. Each image has multiple captions covering different aspects of information, such as objects, attributes, relationships, scenes, etc. In addition, the dataset provides rich metadata information, such as the time, location, and labeling of the images, which can be used for a broader range of image understanding tasks. The BUUISE-Image dataset can be used to evaluate the performance of image captioning and can also be used to develop and train new image captioning algorithms.

3. Research Methodology

The feature extraction model uses a pre-trained Faster R-CNN as Object-Dector to extract local image features, focusing on the target object efficiently and reducing irrelevant redundancy. The model extracts global features from the image using a fine-tuned CLIP Image Encoder as the Image-Encoder. The CLIP text encoder as Text-Encoder encodes the BUUISE-Image dataset attribute relations to obtain encoded text features. Then, the image divides into nine sub-regions, each encoded by the Image Encoder. Image features for each sub-region calculate similarity with the encoded text to obtain the most similar text features. Then, the most similar text features, global image features, and local features concatenate together. Finally, the concatenated features input into the WGA-PoolFormer captioning model to generate corresponding image captions. The methods mentioned above are shown in Figure 2, the TSIC-CLIP image captioning model.

3.1. Feature Extraction Model

The feature extraction model consists of the Object-Detector module, the Image-Encoder module and the Text Retrieval module.

3.1.1. Object-Detector

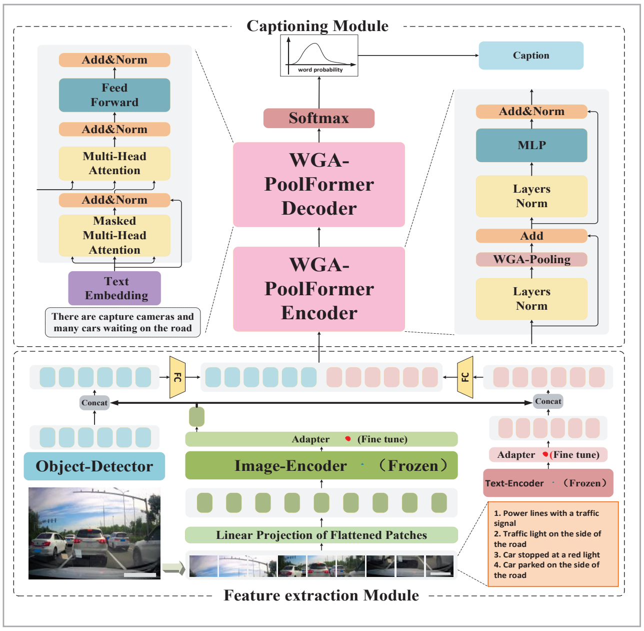
Object-Detector adopts the pre-trained Faster R-CNN model. First, the image $\mathrm{I} \in \mathrm{R}^{H \times W \times C}$ is input to Object-Detector to extract local features such as vehicles and pedestrians in the image $\mathrm{o} = \{o_1, o_2 \,\mathrm{K}\; o_n\}$, where H represents the image height; W represents the image width; C represents the number of channels 3; then. The local features are concatenated with the global image features extracted by Image-Encoder as shown in Equation (1):

$$\hat{O}_m = drop(fc_o(norm_o([o_m, V^*_{global}]))), \qquad (1)$$

where V^*_{global} represents the global image feature vector extracted by Image-Encoder; o_m represents the target feature vector extracted by Object-Detector; $[\cdot,\cdot]$ represents the concatenation operation; norm_o represents layer normalization; fc_o represents the fully connected layer; drop represents the Dropout operation; $\hat{\mathrm{O}}_m$ represents the features for stitching the local and global images; firstly, the features o_m are combined with the features V^*_{global} are concatenated. Their layers are normalized and input to the fully connected layer. And then the overfitting is reduced by Dropout.

Figure2

TSIC-CLIP image captioning model

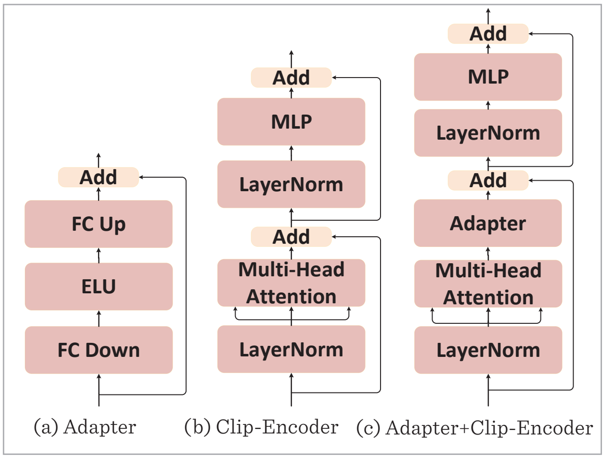


3.1.2. Image-Encoder

The Image-Encoder module utilizes an adapter layer fine-tuned on the CLIP image encoder to extract global features from the image. The parameters of the pre-trained CLIP model are frozen, with only the adapter layer trained on the BUUISE-Image traffic scene dataset. This approach ensures the generalization of the CLIP model while significantly reducing CLIP model training costs. The image encoder is depicted in Figure 3.

Figure 3

Adapter layer



(a) Adapter (b) Clip-Encoder (c) Adapter+Clip-Encoder

The proposed Image-Encoder enables capturing the semantics of the entire scene, focusing the CLIP model on traffic scenes through fine-tuning in the traffic dataset, which makes the model more personalized and helps to generate more contextualized captions of the situation. For an input image I, the Image-Encoder first encodes it to obtain global features, as shown in Equation (2):

$$\mathrm{V}_{\mathrm{global}} = \mathrm{CLIP\_image(I)}, \tag{2}$$

where $\mathrm{Clip\_image}$ denotes the pre-trained CLIP image encoder and V_{global} represents the extracted global image features.

Then, the extracted global image features are fed into the adapter layer, as shown in Equation (3):

$$\mathrm{FA}_{\mathrm{image}}(\mathrm{V}_{\mathrm{global}}) = \mathrm{ELU}\left(\mathrm{V}_{\mathrm{global}}{}^{\mathrm{T}}\mathbf{W}_1^{\mathrm{I}}\right)\mathbf{W}_2^{\mathrm{I}}, \tag{3}$$

where $\mathrm{W}_1^{\mathrm{I}}$ and $\mathrm{W}_2^{\mathrm{I}}$ represent fully connected layers used to adjust the original image features and capture new relevant feature information. ELU represents the activation function. ELU takes an exponential form in the negative region, producing larger gradients to avoid vanishing gradients. Furthermore, ELU's near-zero mean and constant variance accelerate neural network convergence speed and enhance model robustness. $\mathrm{FA}_{\mathrm{image}}$ represents the adapter layer.

The features adjusted by the adapter layer are fed into the residual block, and the Equation as shown in Equation (4):

$$\mathrm{V}^*_{\mathrm{global}} = \alpha \mathrm{FA}_{image}(\mathrm{V}_{\mathrm{global}})^T + (1-\alpha)\mathrm{V}_{\mathrm{global}}, \tag{4}$$

where α represents the residual ratio, which is used to adjust the original features; $\mathrm{V}^*_{\mathrm{global}}$ represents the global image features after adapter layer adjustment.

3.1.1. Text-Encoder

The text encoder utilizes the pre-trained CLIP text encoder, and the weights of the predictive classifier are adjusted using the adapter layer as shown in Equation (5):

$$\mathrm{FA}_{\mathrm{context}}(\mathrm{W}) = \mathrm{ELU}\left(\mathrm{W}^T\mathrm{W}_1^C\right)\mathrm{W}_2^C, \tag{5}$$

where W represents the classifier weights; $\mathrm{FA}_{\mathrm{context}}$ represents the adapter layer used to fine-tune the CLIP text encoder.

The classifier weights are first fed into the adapter layer to map the features to the new space used to obtain the relevant features. Then, they are fed into the residual block for adjustment, as shown in Equation (6):

$$\mathrm{W}^{å} = \beta \mathrm{FA}_{\mathrm{context}}(\mathrm{W})^T + (1-\beta)\mathrm{W}, \tag{6}$$

where β is the residual ratio; $\mathrm{C}^{å}$ represents the weights adjusted by the adapter layer.

Then the features encoded by Image-Encoder $\mathrm{V}^*_{\mathrm{global}}$ and the classifier weights of Text-Encoder W^* are used to calculate the class probability of the image by softmax as shown in Equation (7):

$$\mathrm{p_i} = \frac{\exp((\mathrm{W}_\mathrm{i}^*)^T \mathrm{V}^*_{\mathrm{global}})/\tau}{\sum_{\mathrm{j=1}}^{\mathrm{N}} \exp((\mathrm{W}_\mathrm{j}^*)^T \mathrm{V}^*_{\mathrm{global}})/\tau}, \tag{7}$$

where $\exp(\cdot)$ represents the exponential function; W_i^* represents the weight corresponding to the ith output unit; $\mathrm{V}^*_{\mathrm{global}}$ represents the global image features; and τ represents the temperature parameter used to adjust the softmax, which controls the smoothing degree of the probability distribution. Here, i represents the ith class; N represents the total number of classes; $\mathrm{p_i}$ represents the predicted probability of the ith category.

Finally, the Image-Encoder and Text-Encoder are optimized by cross-entropy loss function as shown in Equation (8):

$$\mathrm{L} = -\frac{1}{\mathrm{N}}\sum_{\mathrm{i}}^{\mathrm{N}}\sum_{\mathrm{c=1}}^{\mathrm{M}} y_{\mathrm{ip}} \log\left(\mathrm{p_i}\right), \tag{8}$$

where N represents the total number of samples; M represents the number of classes; p_i represents the probability that the sample i belongs to the predicted class p; y_{ip} represents the true labeling of the sample i for the class p, and if the sample i belongs to the class p, then $y_{ip} = 1$, else it will be 0; L represents the loss function.

3.1.4. Text Retrieval Module

The text retrieval module is used to calculate and get the text features that have the highest similarity with the nine sub-regions of the image, and its model is shown in Figure 4 Text Retrieval Module.

The input image is first split into nine sub-regions in the text retrieval module. Each image sub-region is then encoded by the Image-Encoder, as shown in Equation (9):

$$v_x = \mathrm{ImageEncoder}(I_{sub}), \quad (9)$$

where the vector v_x is the x th image subregion feature, which is taken as the query key.

Then, Text-Encoder encodes $T_{BUU} = \{T_1, T_2, \ldots, T_n\}$, as shown in Equation (10):

$$u_q = \mathrm{TextEncoder}(T_{BUU}). \quad (10)$$

The cosine similarity is calculated between the query image feature v_x and each text feature u_q as shown in Equation (11):

$$\mathrm{sim}(v_x, u_j) = \frac{v_x \cdot u_q}{|v_x| \cdot |u_q|}. \quad (11)$$

Finally, the k text features $t_{j,k}$ with the highest similarity to V_x are returned, as shown in Equation (12):

$$t_{j,k} = \mathrm{topK}(\mathrm{sim}(v_x \cdot u_q)). \quad (12)$$

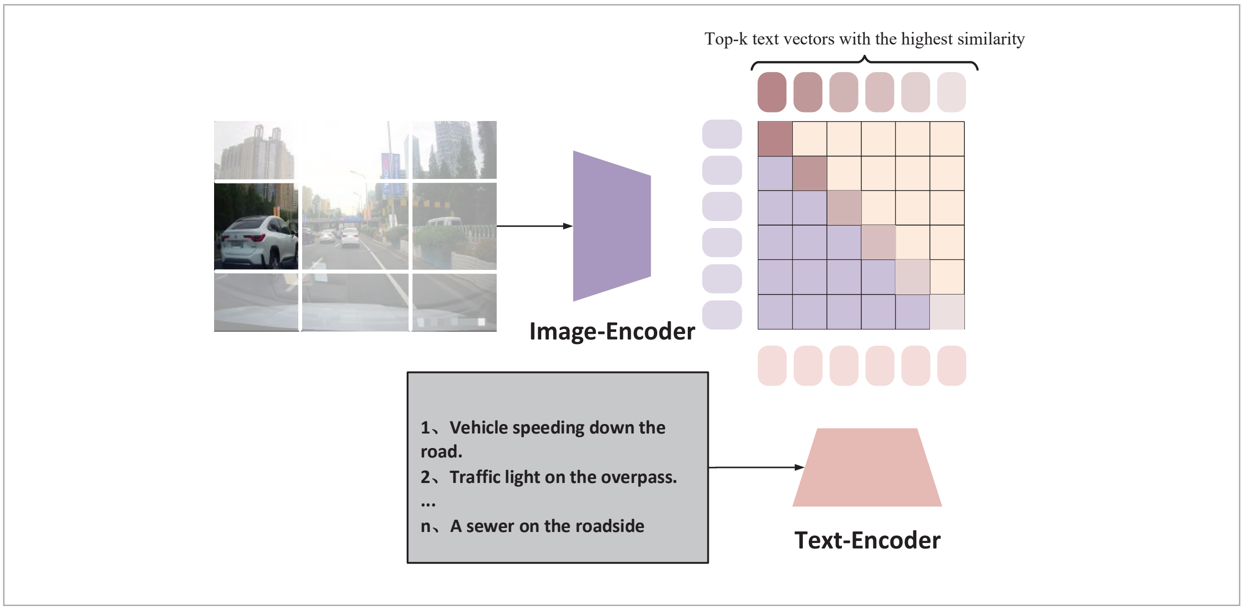
The image sub-region features v_x are then concatenated with the caption vectors $t_{j,k}$ having the highest similarity to their corresponding sub-regions. This combined representation is processed through fully connected layers, layer normalization, and dropout, as shown in Equation (13):

$$\hat{t}_{j,k} = \mathrm{drop}(\mathrm{fc}_t(\mathrm{norm}_t([t_{j,k}, v_x]))), \quad (13)$$

where $t_{j,k}$ is the text feature with the highest similarity; j represents the numbering of the image sub-regions; k represents the top k text descriptions with the highest cosine similarity; and finally the model will be fused by the Object-Detector to the feature vector $O = \{\hat{o}_1, \hat{o}_2, K, \hat{o}_n\}$ are concatenated with the fused

Figure 4

Text retrieval module



1、Vehicle speeding down the road.
2、Traffic light on the overpass.
...
n、A sewer on the roadside

vectors $\hat{T} = \{\hat{t}_{j,k} \mid \forall j, k\}$ from the text retrieval module to obtain the target with global features, respectively. In order to obtain the detection feature vector and the text retrieval encoded feature vector with global features, respectively, the dimension is adjusted by the fully connected layer and the two are concatenated to obtain the feature information V, which is used to be fed into the image captioning module. Compared with traditional image captioning models based on object detection, fine-tuning the pre-trained CLIP model with an adapter layer can minimize model parameters while maintaining the generalization ability of the pre-trained CLIP model. The CLIP model can quickly adapt to new downstream tasks by replacing task-specific adapters, enabling more effective image captioning for traffic scenes.

3.2. WGA-PoolFormer

3.2.1. WGA-PoolFormer Encoder

Traditional Transformer models often have a large number of parameters, which can lead to overfitting when trained on small datasets. To address this, we propose a model called WGA-PoolFormer (Weighted Global Attention PoolFormer) based on MetaFormer [42]. As shown in Figure 5, (a) is the traditional Transformer, and (b) is the proposed WGA-PoolFormer. Compared to the traditional Transformer, we replace the original multi-head self-attention in the encoder with a new WGA-Pooling layer to fuse through token mixing. This replacement reduces model parameters while retaining the ability to model semantic information, enabling the model to capture key features.

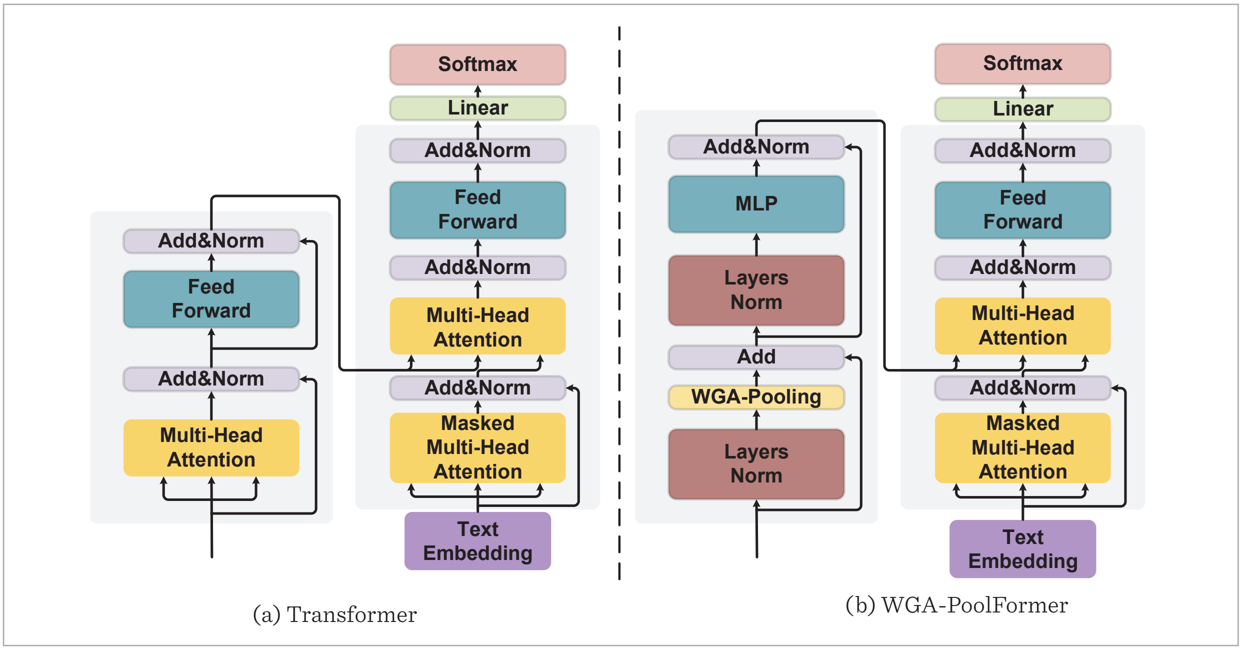
First, the feature vector V, output from the feature extraction model, is input into the WGA-Poolformer for layer normalization. Next, the weighted attention pooling layer (WGA-Pooling) performs token mixing to aggregate the spatial information between tokens at different locations. Subsequently, the residual connection sums the weighted attention pooling features with the original features, followed by layer normalization. Finally, the residual connection is performed after passing through the fully connected layer, as shown in Equation (14):

$$g = \mathrm{WGApool}(\mathrm{norm}(V)) + V, \tag{14}$$

where, $\mathrm{norm}(\cdot)$ is the layer normalization, and $\mathrm{WGApool}(\cdot)$ represents the Weighted Global Attention Pooling module for mixing the spatial informa-

Figure 5

Comparison between Transformer model and WGA-PoolFormer model



(a) Transformer

(b) WGA-PoolFormer

tion of all word tokens within the window, as shown in Equation (15):

$$V'_{:,i,j} = \frac{1}{K \times K} \sum_{p,q=1}^{K} M_{p,q}(V_{i+p-\frac{k+1}{2},j+q-\frac{k+1}{2}} - V_{:,i,j}), \quad (15)$$

where K is the pooling window size; M is a learnable $K \times K$ weight matrix, which assigns different weights to features at different positions; p and q are the row and column indices of the weight matrix M, respectively; $V'_{:,i,j}$ is the output feature matrix; $V_{:,i,j}$ is the input feature matrix; i, j indicate the position of the feature mapping. In Equation (12), the summation goes through each position in the $K \times K$ pooling window, computing the difference between the input feature V at each position and the centre position (i, j), weighted by the learnable weight matrix M. This allows aggregating spatial information by obtaining a weighted fusion of the features around the centre location.

The feature g aggregated with spatial information via WGA-Pooling is fed into the next sub-module. First, layer normalization is applied. g is then input to a fully connected layer for dimension adjustment, followed by ReLU activation to filter features. Another fully connected layer further adjusts dimensions. Finally, a residual connection is added with the original feature vector, as shown in Formula (16), generating an enhanced feature representation:

$$z = \mathrm{Re\,LU}(\mathrm{norm}(g)W_1)W_2 + g, \quad (16)$$

where, W_1 and W_2 represent fully connected layers; norm represents the layer normalization operation; and g represents the features obtained by WGA-Pooling.

This paper introduces a learnable weight matrix M to learn the relationships between neighbouring features, thereby modelling local spatial information. The weight parameters in M can weigh different positional features and learn their importance. In the weighted summation process, global features are considered, and surrounding local features are aggregated so that the model can capture the mutual relationships between local and global features. This delicate spatial information modelling enhances the model's semantic judgment and key information extraction abilities. This method can better analyze the intrinsic correlations between data from different modalities in cross-modal tasks and effectively improve the model's joint representation learning and downstream task performance. In summary, this paper achieves an adaptive fusion of local and global features through learnable weights, strengthening the feature expression ability of the model in cross-modal tasks.

3.2.2. WGA-PoolFormer Decoder

In the WGA-PoolFormer Decoder, the model embeds text captions $\omega \le t = (\omega_1, \omega_2, \cdots, \omega_t)$ to obtain embedded vectors $e \le t = (e_1, e_2, \cdots, e_t)$. These vectors and the encoder output z are input to the model.

First, the query vector Q, key vector K, and value vector V are computed as shown in Equation (17):

$$Q = eW_{Q_0}, K = eW_{K_0}, V = eW_{V_0}, \quad (17)$$

where e is the vector by word embedding; W_{Q_0}, W_{K_0}, and W_{V_0} are the learned weight matrices.

The masked multi-attention mechanism is then employed, taking as input the vectors obtained from the computation, as shown in Equation (18):

$$O = \mathrm{MaskedMultiHead}(Q, K, V, M), \quad (18)$$

where Q, K, V are the Query, Key, and Value computed through the embedding layer and linear mapping, and M is the mask matrix to prevent information leakage;

Next, layer normalization and residual connection are performed on the masked multi-attention output O, which is then input to the decoder's second sublayer, as shown in Equations (19)-(20):

$$O' = \mathrm{WGApool}(\mathrm{norm}(O)) + O \quad (19)$$

$$O'' = \mathrm{Attention}(Q_z, K_z, O'') = \mathrm{soft\,max}(\frac{Q_z K_z^T}{\sqrt{d_k}})O' \quad (20)$$

$$Q_z = zW_{Q_z}, K_z = zW_{Kz}, \quad (21)$$

where O' is the output vector of the masked multi-head attention mechanism. The matrices Q_z and K_z are obtained by linear mapping of the encoder output feature z as shown in Equation (21), where W_{Q_z} and W_{K_z} are learnable weight matrices that map encoder features z to the query and key vector spaces, generating the query and key matrices required for the attention mechanism. O' is obtained from the Masked Multi-Head Attention output.

Finally, the feature vector O''' is obtained by FFN and residual layer processing, and after adjusting the length of the vocabulary list by a linear layer, it is inputted into the softmax function to generate the word probability, whose formula is shown in (22):

$$O_{output} = \mathrm{Soft\,max}(\mathrm{Linear}(O''')), \tag{22}$$

where O_{output} is the word generated at the current timestamp, and O''' is the vector output by the decoder. The model repeats the decoding step until generating the complete textual caption for the image.

4. Experiment and Results Discussion

4.1. Experimental Environment and Parameter Configuration

The experiments were conducted using Ubuntu 18.04, an Intel Xeon E5-2637 v4 CPU, 32GB Samsung RAM, and four NVIDIA Titan V GPUs. The software stack comprised PyTorch 1.10, Python 3.7, CUDA 11.4 and cuDNN 8.2.4. To ensure effective experiments, batchsize was set to 200 and epochs to 60 during training. Training proceeded in two phases: cross-entropy and reinforcement learning. The learning rate was 1e-4 for cross-entropy and 5e-6 for reinforcement learning. Model optimization used the AdamW optimizer.

4.2. General Dataset Introduction and Evaluation Metrics

4.2.1. MS-COCO

The MS-COCO (Microsoft Common Objects in Context) dataset [14] is widely used for image recognition and captioning, containing over 330,000 images annotated with at least five manually generated captions each. These diverse captions, created by different annotators, cover scenes involving people, animals, transportation, furniture, food, and more. Each caption contains about ten words that can describe objects, attributes, actions, etc., in the image. MS-COCO provides instance segmentation, semantic segmentation, and keypoint annotations, enabling diverse image understanding tasks. This rich annotation has been invaluable for advancing image understanding algorithms.

4.2.2. MS-COCO

The Flickr 30K dataset brings about 31,000 real-world images from the Flickr image-sharing platform, providing five high-quality text captions for each image. Created by human annotators, these captions capture not only the objects, scenes, and situations in the images but also rich information such as emotions and contexts. Thus, one of the features of this dataset is the diversity of image-text pairs covering a wide range of scenes, objects and situations.

4.2.3. Evaluation Metrics

We use four commonly used evaluation metrics in image captioning to evaluate the proposed model: BLEU-4, METEOR, ROUGE-L and CIDEr.

BLEU-4 measures n-gram overlap between the generated and reference captions to evaluate accuracy, using up to 4-gram information.

METEOR incorporates semantic information by considering synonyms and stem matching instead of purely exact word matching, better capturing semantic consistency.

ROUGE-L computes the longest common subsequence between captions, reflecting similarity.

CIDEr leverages n-gram co-occurrence statistics between generated and reference captions to assess accuracy and diversity. Higher CIDEr scores indicate greater conformance to human captions.

4.3. Experimental Results Discussion and Comparison

The MS-COCO, Flickr 30k, and BUUISE-Image traffic scene datasets were utilized for training and evaluation to fully validate the model's performance. The model's performance was quantitatively analyzed using common evaluation metrics: BLEU-4 (B@4), METEOR (M), ROUGE-L (R), and CIDEr (C). To verify the model's generalizability, we first evaluated it on the MS-COCO dataset; the results are shown in Table 1. Different algorithms were evaluated on MS-COCO and compared to other image captioning models. The results demonstrate the proposed method obtained effective scores across all metrics, achieving the highest scores compared to the second-ranked S2 model. Specifically, the proposed method scored 40.3% for BLEU-4, 0.2% higher; 30.1% for METEOR, 0.5% higher; 59.6% for ROUGE-L, 0.1% higher; and 137.9 for CIDEr, 5.3% higher.

Table 1

Evaluation results of different algorithms on the MS-COCO dataset

Metrics Methods	B@4	M	R	C
VLKD [8]	36.5	29.1	-	117.1
CTE [7]	38.2	28.7	58.5	124.9
LWDSFUSION [31]	31.3	25.7	54.0	99.9
GAT [35]	39.9	-	59.1	129.8
S2 [43]	40.1	29.6	59.5	132.6
TRANSKG [44]	34.4	27.7	56.3	112.6
ClipCap [17]	33.5	27.4	-	113.0
OURS	40.3	30.1	59.6	137.9

Flickr 30K evaluation results are shown in Table 2, comparing the proposed model against other image captioning methods. Our model achieves state-of-the-art performance on BLEU-4, METEOR, ROUGE-L, and CIDEr, with scores of 26.8%, 23.3%, 48.1%, and 63.4%, respectively. Compared to the second-best TRANSKG model, our model shows improvements of 0.3% on BLEU-4, 1.6% on METEOR, 0.2% on ROUGE-L, and 6.8% on CIDEr.

Table 2

Evaluation results of different algorithms on Flickr 30k dataset

Metrics Methods	B@4	M	R	C
MetaLM [10]	-	-	-	43.3
LWDSFUSION	23.8	20.5	47.0	50.8
TRANSKG	26.5	21.7	47.9	56.6
ClipCap	21.7	22.1	47.3	53.5
OURS	26.8	23.3	48.1	63.4

In order to verify the ability of the CLIP-based image captioning model (TSIC-Clip) proposed in this paper to generate image captions in traffic scenes, we trained and evaluated the method based on pre-trained CLIP on the BUUISE-Image dataset, and the results are shown in Table 3. The evaluation results show that, compared with other image captioning methods based on pre-trained CLIP, the methods in this paper have obvious advantages by adding an adapter layer to CLIP to fine-tune the BUUISE-Image dataset for traffic scenes and by proposing a decoder based on WGA-Poolformer. These methods perform better than the global image feature encoder using only CLIP. Specifically, the model in this paper achieves a score of 39.6% in BLEU-4, 29.7% in METEOR, 59.3% in ROUGE-L, and 136.5% in CIDEr. Compared with the second-ranked CTE model, the model in this paper improves the BLEU-4 by 2.8%, the METEOR by 0.1%, the ROUGE-L by 3.1%, and the CIDEr by 16.1%. The method proposed in this paper is more effective in generating image captions in traffic scenes.

Table 3

Evaluation results of different algorithms on the BUUISE-image dataset

Metrics Methods	Feature Extractor	B@4	M	R	C
ClipCap	CLIP-encoder	32.6	26.4	47.2	117.1
CTE	CLIP-encoder	36.8	29.6	56.2	120.4
VLKD	CLIP-encoder	35.7	29.6	53.2	114.3
OURS	CLIP-encoder	39.6	29.7	59.3	136.5

The number of parameters of the WGA-PoolFormer model proposed in this paper is validated on the BUUISE-Image dataset and compared with three models, CTE, VLKD and Clipcap, which also use the Transformer structure. As shown in Table 4 demonstrates the comparison of different decoders and their parameters under the CLIP-based approach. Under the same visual feature extractor CLIP, the WGA-Poolformer decoder proposed in this paper not only enhances the feature representation capability but also reduces the number of parameters to a certain extent by introducing a learnable weighted full-attention pooling layer for adaptive fusion of local and global features. Specifically, the number of parameters of the model proposed in this paper is 41M, which is 2M lower than that of the Clipcap model with the smallest number of parameters. The number of parameters is 82M lower than that of the CTE model with the second highest scores in the four evaluation

Table 4

Comparison of different decoders and their parameters under the CLIP method

Metrics / Methods	Feature Extractor	Decoder	Params
CTE	CLIP-encoder	Transformer+GPT2	123M
VLKD	CLIP-encoder	BART	86M
ClipCap	CLIP-encoder	Transformer+GPT2	43M
OURS	CLIP-encoder	WGA-Poolformer	41M

indexes of BLEU-4, METEOR, ROUGE-L, and CIDEr in Table 3, which shows that the method of this paper is effective in reducing the number of parameters while ensuring the quality of the caption. Reduces the number of parameters.

Figure 6

Comparison of the number of parameters with BLEU4 and CIDEr values for different CLIP-based methods

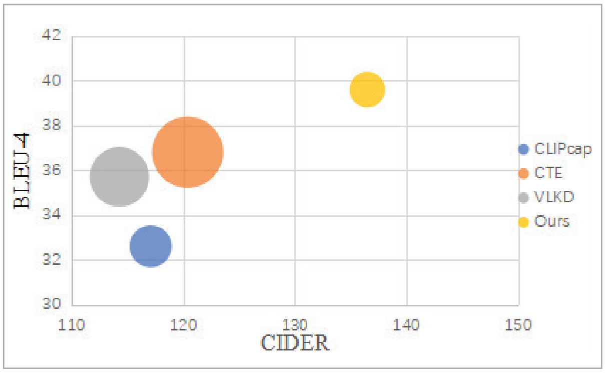
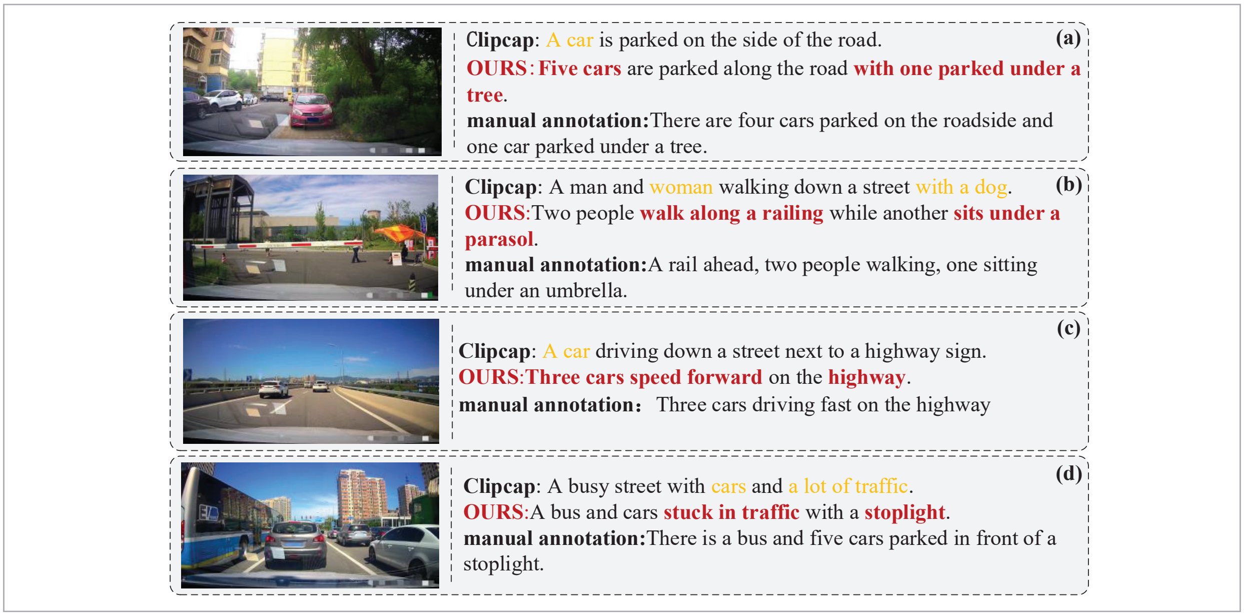


Figure 6 presents a comparison among different CLIP-based methods in terms of parameter count, BLEU-4 scores, and CIDEr values. The graph employs a Cartesian coordinate system, where the horizontal axis represents CIDEr scores, the vertical axis signifies BLEU-4 scores, and the size of each bubble correlates with the corresponding parameter count. Within Figure 4, our model is depicted by a yellow bubble, the CTE model by an orange one, and the CLIPcap model by a blue one. Notably, our model excels in both the horizontal (CIDEr) and vertical (BLEU-4) coordinates. Furthermore, when considering bubble size (indicating parameter count), our proposed model boasts the smallest area ,and thus ,the lowest parameter count.

In order to further evaluate and analyze the captioning performance of the proposed model in this paper on traffic scenes, four images were randomly selected from the BUUISE-Image dataset, and the manually labelled ground truth of each image was provided for evaluation. The results of comparing this paper's model with the same Transformer architecture-based approach are visualized in Figure 7. It can be observed

Figure 7

Example of image captioning in traffic scenes



Clipcap: A car is parked on the side of the road. **(a)**
OURS: **Five cars** are parked along the road **with one parked under a tree**.
manual annotation: There are four cars parked on the roadside and one car parked under a tree.

Clipcap: A man and woman walking down a street with a dog. **(b)**
OURS:Two people **walk along a railing** while another **sits under a parasol**.
manual annotation:A rail ahead, two people walking, one sitting under an umbrella.

(c)
Clipcap: A car driving down a street next to a highway sign.
OURS:**Three cars speed forward** on the **highway**.
manual annotation: Three cars driving fast on the highway

Clipcap: A busy street with cars and a lot of traffic. **(d)**
OURS:A bus and cars **stuck in traffic** with a **stoplight**.
manual annotation: There is a bus and five cars parked in front of a stoplight.

that on the traffic scene dataset, the model proposed in this paper generates richer semantic information in the image captions compared to the same method based on CLIP and Transformer architectures. As shown in example (a) in Figure 5, Clipcap can accurately recognize “A car is parked on the side of the road” but ignores the details of other vehicles on the side of the road and a car parked under a tree, which leads to inaccurate captions as highlighted in a yellow font in the figure. The method in this paper, as highlighted in red font, can accurately describe the number of vehicles in the figure and provide specific details (a car parked under a tree). Furthermore, as depicted in Figure 5(b), Clipcap’s method generates inaccurate descriptions, as highlighted in yellow, since no dog or woman is in the figure. In contrast, the method proposed in this paper accurately states the existence of a railing, a person sitting under an umbrella, and two people walking, as highlighted in red font.

The above analysis demonstrates that compared to the same method based on CLIP and Transformer architectures, the image captioning method proposed in this paper benefits from fine-tuning via adapter layers on BUUISE-Image, which captures the key information in the image more precisely and deepens the understanding of the image. Additionally, token mixing using the weighted global attention pooling module incorporates global and local feature information, making fuller use of semantic information. This enables the generated descriptions to focus on the key parts of the image and describe them more accurately.

4.4. Ablation Experiments

Our approach has two main innovations: First, in the image encoder module, CLIP is adopted as the feature extractor with frozen model parameters, and the CLIP model is fine-tuned by inserting an adapter layer.

This enables learning new features from fewer traffic scene samples. Second, a novel WGA-Pooling layer is proposed in the image captioning module to replace the traditional multi-head self-attention layer, reducing model parameters while maintaining performance. To validate the efficacy of these innovations in the proposed TSIC-Clip model, ablation experiments were conducted on the MS-COCO dataset. The results are shown in Table 5 ablation Experiments.

In the first experiment, the feature extractor was fixed as Vit/b16, and the token mixer was the only variable. Methods using WGA-Pooling and MSA (Multi-Head Self-Attention) as the token mixer were compared. Results show that the WGA-Pooling-based method scores slightly lower than the MSA method on the B@4, M, R, and C metrics, with a value of about 1% difference. However, the WGA-Pooling had 35M fewer parameters than the MSA. Thus, WGA-PoolFormer reduces parameters while maintaining performance, validating WGA-Pooling.

In the second experiment, Vit/b16 was fixed as the feature extractor, WGA-Pooling as the token mixer, and fine-tuning as the only variable. Results show adapter-based fine-tuning had 33.7M parameters; full training had 116M; thus, the adapter reduced parameters by 82.3M. Additionally, adapter-based fine-tuning improved all metrics over full training, with gains of 0.8% in BLEU-4, 0.8% in METEOR, 2.7% in ROUGE-L and 4% in CIDEr.

In the third experiment, the CLIP encoder was fixed as the feature extractor with all parameters frozen, and the token mixer was the only variable.

Table 5
Ablation experiments

Feature Extractor	Token mixer	Fine tuning	B@4	M	R	C	Params
Vit/b16	MSA	Full	37.5	28.7	56.5	127.1	151M
Vit/b16	WGA-POOL	Full	37.1	28.3	55.5	126.5	116M
Vit/b16	WGA-POOL	Adapter	37.9	29.1	58.2	130.5	33.7M
CLIP-encoder	MSA	Freezing	39.7	29.4	59.0	135.3	66M
CLIP-encoder	WGA-POOL	Freezing	39.2	29.2	58.6	134.5	33.5M
CLIP-encoder	WGA-POOL	Adapter	40.3	30.1	59.6	137.9	33.7M

Results show WGA-Pooling had 33.5M parameters versus 66M for MSA. Thus, WGA-Pooling had 32.5M fewer parameters than MSA, with minimal metric fluctuations, maintaining performance.

In the fourth experiment, the CLIP encoder was fixed as a feature extractor, all parameters were frozen, WGA-Pooling was the token mixer, and fine-tuning was the only variable. Results show that adapter-based CLIP fine-tuning improved all metrics over direct CLIP freezing, with gains of 1.1% in BLEU-4, 0.9% in METEOR, 1% in ROUGE-L, and 3.4% in CIDEr.

In summary, ablation experiments verified the efficacy of the two proposed innovations in TSIC-Clip - adapter fine-tuning and the WGA-Pooling layer.

5. Conclusions

In this paper, we propose a CLIP-based image captioning model for traffic scenes to solve current problems of image captioning in traffic scenes, such as imprecise captions, large model sizes, and lack of personalization. In this work, by adding an adapter layer to the CLIP model and fine-tuning public and BUUISE-Image datasets, the CLIP model is adjusted to enable personalized traffic scene captioning while ensuring generalization. Furthermore, considering the large parameter size of Transformer-based image captioning models, we propose a new model, WGA-PoolFormer, replacing the self-attention mechanism in the Transformer with a global weighted attention pooling layer. This allows effective fusion of different features and capturing multi-level, multi-perspective information while reducing model parameters, further improving performance.

However, real-time deployment of image captioning models is still a problem in practical applications. Future work should continue to focus on model reduction and lightweighting approaches, such as knowledge distillation and pruning, to reduce model size and computation. This will help the models to be deployed on resource-constrained mobile or embedded devices. On the other hand, in order to enhance the generalisation ability of the model, future work should also continue to expand the size and scene coverage of the image captioning dataset of traffic scenes, and collect images containing different regions, time of day, weather, etc., so as to adapt the model to a wider range of real-world usage scenarios and improve robustness. The in-depth study of these directions will help to advance the generation of traffic scene image captioning to practical applications.

Appendix A

The download addresses of the four datasets used in this article are as follows:

MS-COCO: https://cocodataset.org/

Flickr 30K: http://web.engr.illinois.edu/577~bplumme2/Flickr30kEntities/

BUUISE-Image: The dataset involve state-owned enterprise confidentiality cannot be disclosed publicly.

Acknowledgement

This work was supported, the National Natural Science Foundation of China (Grant No. 62006020, 62102033, 62171042), the R&D Program of Beijing Municipal Education Commission (Grant No. KZ202211417048), The Project of Construction and Support for high-level Innovative Teams of Beijing Municipal Institutions (Grant No. BPHR20220121), Beijing Natural Science Foundation (Grant No. 4232026), the Academic Research Projects of Beijing Union University (No. ZKZD202302).

References

1. Ahmadian, N., Khosravi, A., Sarhadi, P. Driver Assistant Yaw Stability Control via Integration of AFS and DYC. Vehicle System Dynamics, 2022, 60(5), 1742-1762. https://doi.org/10.1080/00423114.2021.1879390
2. Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., Zhang, L. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, 6077-6086. https://doi.org/10.1109/CVPR.2018.00636
3. Bisio, I., Garibotto, C., Haleem, H., Lavagetto, F., Sciarrone, A. A Systematic Review of Drone-Based Road Traffic Monitoring System. IEEE Access, 2022. https://doi.org/10.1109/ACCESS.2022.3207282

4. Bounab, Y., Oussalah, M., Ferdenache, A. Reconciling Image Captioning and User's Comments for Urban Tourism. In 2020 Tenth International Conference on Image Processing Theory, Tools and Applications, 2020, 1-6. IEEE. https://doi.org/10.1109/IPTA50016.2020.9286602

5. Chai, Y., Liu, H., Xu, J., Samtani, S., Jiang, Y., Liu, H. A Multi-Label Classification with an Adversarial-Based Denoising Autoencoder for Medical Image Annotation. ACM Transactions on Management Information Systems 2023, 14(2), 1-21. https://doi.org/10.1145/3561653

6. Cornia, M., Stefanini, M., Baraldi, L., Cucchiara, R. Meshed-Memory Transformer for Image Captioning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, 10578-10587. https://doi.org/10.1109/CVPR42600.2020.01059

7. Cho, J., Yoon, S., Kale, A., Dernoncourt, F., Bui, T., Bansal, M. Fine-Grained Image Captioning with Clip Reward. arXiv preprint arXiv:2205.13115, 2022. https://doi.org/10.18653/v1/2022.findings-naacl.39

8. Dai, W., Hou, L., Shang, L., Jiang, X., Liu, Q., Fung, P. Enabling Multimodal Generation on CLIP via Vision-Language Knowledge Distillation. arXiv preprint arXiv:2203.06386, 2022. https://doi.org/10.18653/v1/2022.findings-acl.187

9. Gupta, A., Anpalagan, A., Guan, L., Khwaja, A. S. Deep Learning for Object Detection and Scene Perception in Self-Driving Cars: Survey, Challenges, and Open Issues. Array, 2021, 10, 100057. https://doi.org/10.1016/j.array.2021.100057

10. Hao, Y., Song, H., Dong, L., Huang, S., Chi, Z., Wang, W., Wei, F. Language Models Are General-Purpose Interfaces. arXiv preprint arXiv:2206.06336, 2022.

11. Kim, J., Rohrbach, A., Darrell, T., Canny, J., Akata, Z. Textual Explanations for Self-Driving Vehicles. In Proceedings of the European Conference on Computer Vision, 2018, 563-578. https://doi.org/10.1007/978-3-030-01216-8_35

12. Li, M., Zhang, H., Xu, C., Yan, C., Liu, H., Li, X. MFVC: Urban Traffic Scene Video Caption Based on Multimodal Fusion. Electronics, 2022, 11(19), 2999. https://doi.org/10.3390/electronics11192999

13. Li, W., Qu, Z., Song, H., Wang, P., Xue, B. The Traffic Scene Understanding and Prediction Based on Image Captioning. IEEE Access, 2020, 9, 1420-1427. https://doi.org/10.1109/ACCESS.2020.3047091

14. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C. L. Microsoft COCO: Common Objects in Context. In Computer Vision-ECCV 2014: 13th European Conference, 2014, Part V 13, 740-755. https://doi.org/10.1007/978-3-319-10602-1_48

15. Luo, J., Li, Y., Pan, Y., Yao, T., Feng, J., Chao, H., Mei, T. Semantic-Conditional Diffusion Networks for Image Captioning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, 23359-23368. https://doi.org/10.1109/CVPR52729.2023.02237

16. Medina-Salgado, B., Sanchez-DelaCruz, E., Pozos-Parra, P., Sierra, J. E. Urban Traffic Flow Prediction Techniques: A Review. Sustainable Computing: Informatics and Systems, 2022, 35, 100739. https://doi.org/10.1016/j.suscom.2022.100739

17. Mokady, R., Hertz, A., Bermano, A. H. Clipcap: Clip Prefix for Image Captioning. arXiv preprint arXiv:2111.09734, 2021.

18. Mohammed, S.I. An Overview of Traffic Accident Investigation Using Different Techniques. Automotive Experiences, 2023, 6(1), 68-79. doi: 10.31603/ae.7913. https://doi.org/10.31603/ae.7913

19. Moratelli, N., Barraco, M., Morelli, D., Cornia, M., Baraldi, L., Cucchiara, R. Fashion-Oriented Image Captioning with External Knowledge Retrieval and Fully Attentive Gates. Sensors, 2023, 23(3), 1286. https://doi.org/10.3390/s23031286

20. Mori, Y., Fukui, H., Hirakawa, T., Nishiyama, J., Yamashita, T., Fujiyoshi, H. Attention Neural Baby Talk: Captioning of Risk Factors While Driving. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019, 4317-4322. IEEE. https://doi.org/10.1109/ITSC.2019.8917187

21. Mori, Y., Hirakawa, T., Yamashita, T., Fujiyoshi, H. Image Captioning for Near-Future Events from Vehicle Camera Images and Motion Information. In 2021 IEEE Intelligent Vehicles Symposium, 2021, 1378-1384. IEEE. https://doi.org/10.1109/IV48863.2021.9575562

22. Nukrai, D., Mokady, R., Globerson, A. Text-Only Training for Image Captioning Using Noise-Injected Clip. arXiv preprint arXiv:2211.00575, 2022. https://doi.org/10.18653/v1/2022.findings-emnlp.299

23. Ouali, I., Halima, M. B., Wali, A. An Augmented Reality for an Arabic Text Reading and Visualization Assistant for the Visually Impaired. Multimedia Tools and Applications, 2023, 1-29. https://doi.org/10.1007/s11042-023-14880-6

24. Ranyal, E., Sadhu, A., Jain, K. Road Condition Monitoring Using Smart Sensing and Artificial Intelligence: A Review. Sensors, 2022, 22(8), 3044. https://doi.org/10.3390/s22083044

25. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I. Learning Transferable Visual Models from Natural Language Supervision. Proceedings of the International Conference on Machine Learning, July 2021, 8748-8763

26. Rochel, S. N. S., Luc, R. J., Thomas, M., Victor, M. Deep Learning: Traffic Accident Captioning Model in Madagascar Mother Language. In 2022 8th International Conference on Control, Decision and Information Technologies, 2022, 1, 996-1001. IEEE. https://doi.org/10.1109/CoDIT55151.2022.9804080

27. Seifi, P., Chalechale, A. Traffic Captioning: Deep Learning-Based Method to Understand and Describe Traffic Images. In 2022 8th Iranian Conference on Signal Processing and Intelligent Systems, 2022, 1-6. IEEE. https://doi.org/10.1109/ICSPIS56952.2022.10044082

28. Selivanov, A., Rogov, O. Y., Chesakov, D., Shelmanov, A., Fedulova, I., Dylov, D. V. Medical Image Captioning via Generative Pretrained Transformers. Scientific Reports, 2023, 13(1), 4171. https://doi.org/10.1038/s41598-023-31223-5

29. Srihari, K., Sikha, O. K. Partially Supervised Image Captioning Model for Urban Road Views. In Intelligent Data Communication Technologies and Internet of Things: Proceedings of ICICI 2021, 2022. https://doi.org/10.1007/978-981-16-7610-9_5

30. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Polosukhin, I. Attention Is All You Need. Advances in Neural Information Processing Systems, 2017, 30.

31. Vinyals, O., Toshev, A., Bengio, S., Erhan, D. Show and Tell: A Neural Image Caption Generator. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, 3156-3164. https://doi.org/10.1109/CVPR.2015.7298935

32. Voykinska, V., Azenkot, S., Wu, S., Leshed, G. How Blind People Interact with Visual Content on Social Networking Services. In Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, 2016, 1584-1595. https://doi.org/10.1145/2818048.2820013

33. Wang, C., Shen, Y., Ji, L. Geometry Attention Transformer with Position-Aware LSTMs for Image Captioning. Expert Systems with Applications, 2022, 201, 117174. https://doi.org/10.1016/j.eswa.2022.117174

34. Wang, S., Zeng, Q., Ni, W., Cheng, C., Wang, Y. ODP-Transformer: Interpretation of Pest Classification Results Using Image Caption Generation Techniques. Computers and Electronics in Agriculture, 2023, 209, 107863. https://doi.org/10.1016/j.compag.2023.107863

35. Wang, Y., Xu, J., Sun, Y. End-to-End Transformer-Based Model for Image Captioning. In Proceedings of the AAAI Conference on Artificial Intelligence, 2022, 36(3), 2585-2594. https://doi.org/10.1609/aaai.v36i3.20160

36. Wu, C., Li, Y., Li, L., Wang, L., Liu, Y. Caption Generation from Road Images for Traffic Scene Construction. In 2020 IEEE Intelligent Vehicles Symposium, 2020, 1271-1276. IEEE. https://doi.org/10.1109/IV47402.2020.9304746

37. Xian, T., Li, Z., Zhang, C., Ma, H. Dual Global Enhanced Transformer for Image Captioning. Neural Networks, 2022, 148, 129-141. https://doi.org/10.1016/j.neunet.2022.01.011

38. Xu, H., Gao, Y., Yu, F., Darrell, T. End-to-End Learning of Driving Models from Large-Scale Video Datasets. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, 2174-2182. https://doi.org/10.1109/CVPR.2017.376

39. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In International Conference on Machine Learning, 2015, 2048-2057.

40. Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., Yan, S. Metaformer Is Actually What You Need for Vision. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, 10819-10829. https://doi.org/10.1109/CVPR52688.2022.01055

41. Zeng, P., Zhang, H., Song, J., Gao, L. S2 Transformer for Image Captioning. In Proceedings of the International Joint Conferences on Artificial Intelligence, 2022, 5. https://doi.org/10.24963/ijcai.2022/224

42. Zhang, Y., Shi, X., Mi, S., Yang, X. Image Captioning with Transformer and Knowledge Graph. Pattern Recognition Letters, 2021, 143, 43-49. https://doi.org/10.1016/j.patrec.2020.12.020

43. Zhu, X., Li, L., Liu, J., Peng, H., Niu, X. Captioning Transformer with Stacked Attention Modules. Applied Sciences, 2018, 8(5), 739. https://doi.org/10.3390/app8050739