Multi-perspective Traffic Video Description Model with Fine-grained Refinement Approach

Tuan-An To* , Minh-Nam Tran* , Trong-Bao Ho* , Thien-Loc Ha* , Quang-Tan Nguyen* , Hoang-Chau Luong , Thanh-Duy Cao , and Minh-Triet Tran†

University of Science, VNU-HCM, Ho Chi Minh City, Vietnam
Vietnam National University, Ho Chi Minh City, Vietnam
{ttan20,tmnam20,htbao20,htloc20,nqtan20}@apcs.fitus.edu.vn,
{lhchau20,ctduy20}@apcs.fitus.edu.vn, tmtriet@fit.hcmus.edu.vn

Abstract

Analyzing traffic patterns is crucial for enhancing safety and optimizing flow within urban cities. While urban cities possess extensive camera networks for monitoring, the raw video data often lacks the contextual detail necessary for understanding complex traffic incidents and the behaviors of road users. In this paper, we propose a novel methodology for generating comprehensive descriptions of traffic scenarios, combining a vision-language model with rule-based refinements to capture pertinently pedestrian, vehicle, and environment factors. First, a captioning model will generate a general description using processed video as input. Subsequently, this description is refined sequentially through three primary modules: pedestrian-aware, vehicle-aware, and context-aware, enhancing the final description. We evaluate our method on the Woven Traffic Safety datasets in Track 2 of the AI City Challenge 2024, obtaining competitive results with an S2 score of 22.6721. Code will be available at https://github.com/ToTuanAn/AICityChallenge2024_Track2

1. Introduction

Traffic safety remains a paramount concern in nowadays society. The careful monitoring and analysis of traffic patterns play a crucial role in mitigating accidents and enhancing the flow of traffic within urban cities. Recent advancements in urban development have led to the widespread installation of cameras throughout streets and vehicles, enabling the capture of unforeseen events that transpire during traffic participation. Nonetheless, this wealth of video data often lacks the descriptive elements necessary to dissect traffic incidents and evaluate the actions of drivers and pedestrians. To address this challenge, this paper proposes a novel method that combines a vision-language model with rule-based refinement to generate detailed descriptions of the surrounding environment, vehicles, and pedestrians, including their perceptions and actions.

In this work, we utilize a vision-language model (VLM) to generate video descriptions and then enhance these descriptions through the integration of rule-based refinements. We experiment with three distinct VLM: the single-view model utilizes single-view video input, the motion-blur model utilizes an average of multiple continuous images as input and the multi-view model utilizes input from multiple videos at varying views. Our rule-based refinement includes three primary modules: pedestrian-aware, vehicle-aware, and context-aware. These modules refine the video description from VLM about pedestrian appearance, behavior, and awareness, as well as vehicle location, speed, actions, and surrounding environmental factors.

We propose a novel yet efficient method for Traffic Safety Description and Analysis, the 2^{nd} track in AI City Challenge 2024 [34]. We train and test all VLMs in this paper on Woven Traffic Safety Dataset [12]. For testing in this track, there are 376 external cases and 84 internal cases; each case contains at least one video and the bounding box of a pedestrian at the start time of each phase in the corresponding video. The output is the pedestrian and vehicle caption corresponding to the input video. Our motion-blur model with rule-based refinement achieves good results in Track 2 of AI City Challenge 2024 [34] with $S2 = 22.6721$.

In Section 2, we briefly review existing methods related to our problem and solution. Then we present our proposed solution in Section 3. Experiments and evaluation are in Section 4. Conclusion and future work are in Section 5.

*The first five authors share the equal contribution.
†Corresponding author. Email: tmtriet@fit.hcmus.edu.vn

2. Related Work

Vision Foundation Models Recent works have proposed pretrained video encoders that tackle diverse vision-understanding tasks. CLIP [25], and ALIGN [8] are Transformer-based image-text models adept at learning visual concepts through natural language supervision, employing contrastive learning on web-scale noisy samples that have led to robust image-text representations for powerful zero-shot transfer. Building upon CLIP's framework, BLIP [14] enhances performance by leveraging synthesized image captioning data for the pretraining stage, achieving state-of-the-art on several vision and image-language benchmarks. Although these image foundation models show promising performance for video recognition, they fall short on many other video tasks due to a lack of motion and temporal information [39]. Some initial works on developing video foundation models BEVT [33], MaskedFeat [36], VideoMAE [27], VideoMAE2 [32] directly extend masked auto-encoding frameworks to spatio-temporal space. Recent works have shifted to utilize. Notably, All-in-one [31] trains a single backbone with multiple pretraining objectives, LAVENDER [18] unifies the tasks as masked language modeling. MERLOT Reserve [40] learns the joint video representations on the collected 20M video-text-audio pairs with contrastive span matching, achieving leading results across various video tasks.

Large Vision Language Models The evolution of large language models has spurred the development of large vision language models by incorporating expansive language models into VLM architectures. BLIP-2 [16] innovatively employs large language models as text decoders alongside a cross-attention module, effectively merging image and text features. LLaVA [21], combining CLIP as an image encoder and Vicuna [4] as a text decoder, achieves performance levels approaching GPT-4 on multimodal benchmarks. Since then, several models have been introduced with the capability to handle both images and videos with text, including Video-Llava [19], VideoChat [17], VideoLlama [41], Flamingo [2], LanguageBind [43], and Vision Gemini [26], which have been proposed to handle diverse forms of multimedia data. However, they are restricted to single-video/single-view scenarios. This work addresses this limitation using multiview attention in Section 3.2.3.

Rule base refinement To solve problems related to traffic analysis in particular or lifelog difficulties in general, it is common to use rule bases to analyze the actions of objects participating in traffic because it is difficult for vision models to predict information related to the relative position (on the left (or right) side, in front of, or behind) and predict the motion of objects such as *"turn left"*, *"turn right"* or *"go straight"* because of the variety of angles as well as the type of camera on the road. Nguyen *et al.* [22] use the algebraic area of the polygon generated by n points in the motion trajectory to categorize a tracked vehicle's motion into *"turn left"*, *"turn right"* or *"go straight"*. Le *et al.* use the sign of counterclockwise $CCW(A^1, M^2, B^3)$ to determine if the vehicle *"turn left"* or *"turn right"*. In our research, many rules were used to refine the results to increase the accuracy of each description.

3. Proposed Method

3.1. Method Overview

Overall, our proposed method contains two main components: Captioning Model, and Rule Engine Refinement.

The primary captioning module is tasked with generating pseudo captions corresponding to input video data. We present our design for this main component in Section 3.2. Additionally, the refinement phase 3.3 encompasses vehicle-aware, pedestrian-aware, and context-aware modules, which collectively serve to rectify instances of pseudo caption failure associated with corner cases.

Figures 1 show the overview of our method in both the training and inference phases. During the training phase, the provided data undergoes preprocessing to form sets of tuples denoted as <Caption, Visual Embedding>, which are utilized to train the captioning model iteratively. In terms of caption preprocessing, our approach yields two distinct sets of features: chunked captions and full captions. Chunked captions are derived from full captions through a process of text-mining, wherein the latter is divided into five smaller sentences based on semantic categories, namely pedestrian/vehicle description, pedestrian/vehicle position, pedestrian/vehicle action, environment status, and road status.

In the inference phase, we extract visual attributes of the primary pedestrian, primary vehicle, and contextual surroundings from visual data. We input the video track data into our pre-trained captioning model to obtain a preliminary pseudo caption for subsequent refinement. The refinement module leverages the extracted attributes to iteratively reconstruct the final caption result, ensuring coherence and accuracy in the generated output.

3.2. Captioning Model

3.2.1 Single-view model

We wish to design a single-view model capable of discerning relationships among events using visual cues, facilitating the accurate localization and description of these events within untrimmed videos. Inspired by Vid2Seq proposed by Yang *et al.* [37], we conceptualize the single-view model as a sequence-to-sequence problem to address this challenge.

[1] start point of motion trajectory
[2] the point at one fifth of motion trajectory
[3] end point of motion trajectory

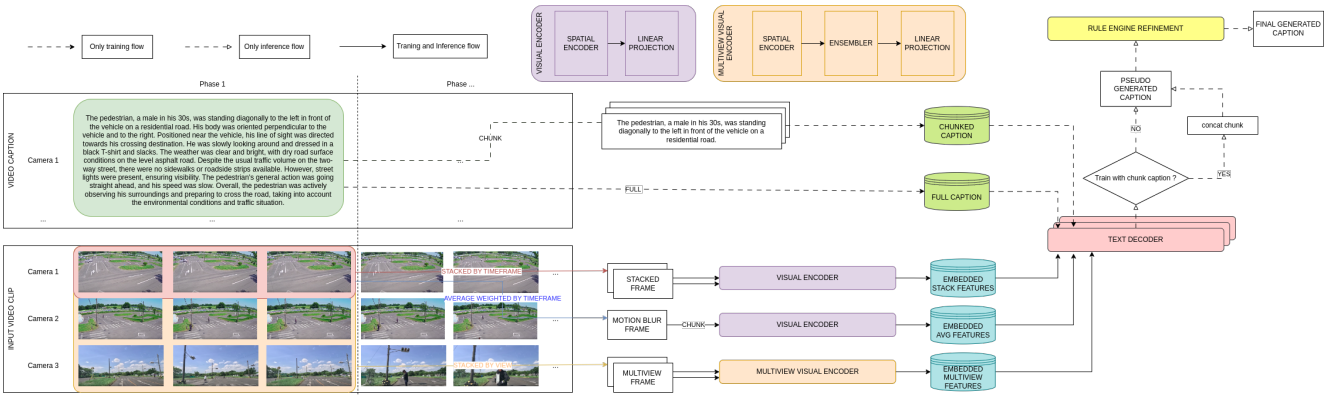

Figure 1. Overview of our proposed methodology. It entails the design of a multiperspective model incorporating three distinct submodels: namely, single-view, motion-blur, and multi-view. Each submodel is characterized by a pair of visual encoders tasked with embedding visual representations, alongside a text decoder dedicated to the generation of textual captions. Subsequent to this, a rule-based engine is employed to iteratively refine the pseudo-captions generated by the submodels, thereby yielding a finalized caption.

In this framework, both the input and output sequences encompass semantic details of the events through natural language descriptions, along with temporal localization information represented by temporal timestamps.

Text and time tokenization. We initialize our text tokenizer with a vocabulary size of V and extend it by incorporating T additional time tokens. This augmentation yields a tokenizer with a total of $V + T$ tokens. The training videos consistently maintain a frame rate of 30 frames per second (fps). Hence, the additional tokens correspond to $30 \times D$ equally-spaced timestamps, where D measured in seconds represents the duration of the videos. Specifically, we employ the `SentencePiece` tokenizer [13] and $T = 30 \times D$ to facilitate this process.

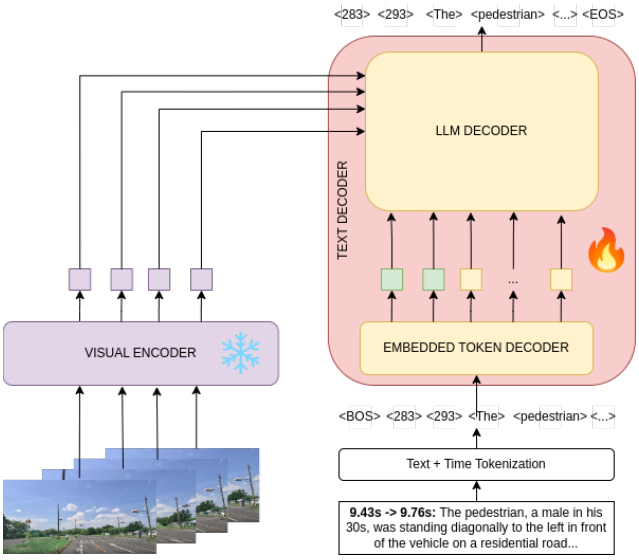

Figure 2. Single-view model architecture.

Visual encoder. The visual encoder learned from a sequence of f frames $x \in \mathbb{R}^{f \times h \times w \times c}$ where c, h, and w denote the channels, height, and width of every frame. Initially, a visual backbone encodes each frame individually and yields frame embeddings. In the challenge, we use the visual backbone ViT-L/14@336px [25] at the resolution of 336 × 336 pixels. Using a contrastive loss function, this backbone is pre-trained to map images to textual descriptions. To ensure computational efficiency, we maintain the backbone in a frozen state.

Text decoder. The output sequence z is generated by the text decoder, utilizing the visual embeddings from the encoder. During each autoregressive step k, the text decoder cross-attends to the encoder outputs and self-attends to previously generated tokens to produce a contextualized representation. Subsequently, a large language model forecasts a probability distribution across the comprehensive vocabulary of text and time tokens, foreseeing the subsequent token in the event sequence. We next explain the construction of our output event sequence z. In the challenge, each phase k is characterized by a textual caption, a start time, and an end time. We first construct for each phase k a sequence by concatenating its start time token t_{start} , its end time token t_{end} and its text tokens $[z_0, z_1, ..., z_k, ..., z_n]$. Finally, the event sequence is derived by adding a BOS and an EOS token at the beginning and end of the sequence, respectively, to mark the start and conclusion of the sequence i.e., $z = [BOS, t_{start}, t_{end}, z_0, z_1, \ldots, EOS]$.

3.2.2 Motion-blur model

After chunking the description into several parts mentioned in Section 3.1, we use images in the same phase as the description to train the Image Captioning model. The `start_time` frame only tells the model how the situation

Figure 3. Example of Motion Blur Image of Vehicle View

started, and the `end_time` frame only tells how the scenario ends. Therefore, combining all the frames in the video is necessary to give the model the most overview with just one image. To do that, we accumulate a sequence of images with the weight w_i (see Equation 2) to create a motion blur image in Equation 1. The closer the image is to `start_time`, the smaller the weight is. Using weights here helps the image keep the temporal order of each frame. Figure 3 is an example of using motion blur images across a sequence of frames of vehicle view.

$$\texttt{motion\_blue\_image} = \sum_{i=1}^{\texttt{n\_frames}} w_i * \texttt{frame}_i \quad (1)$$

where

$$w_i = \frac{i}{\sum_{j=1}^{n} j} \quad (2)$$

In this section, we are mainly finetuning pretrained image captioning models such as BEiT-3 [35] or BLIP-2 [15].

3.2.3 Multi-view model

Rather than relying solely on a single video view, the system adopts a more comprehensive approach by incorporating multiple video views via a vision encoder. These views are then processed through a Multiview Ensembler, which aggregates the video features along the time axis before feeding them into the large language model (see Figure 4).

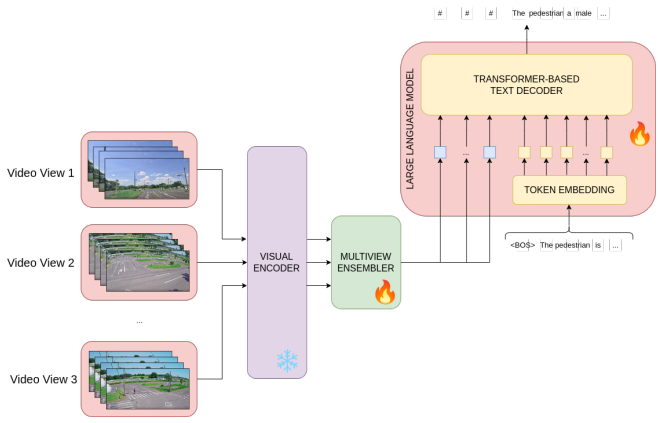

Figure 4. Multiview Model with Multiview Ensembler to combine features from various views before feeding into the LLM.

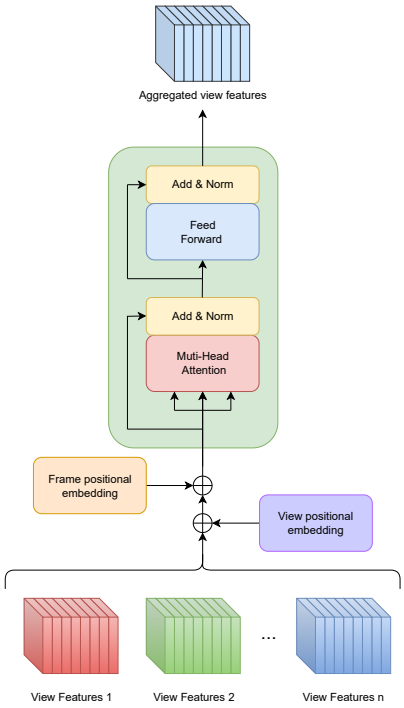

Figure 5. Mutliview Ensembler module with frame positional embedding and view positional embedding. Inputs are sequences of features from multiple views, while output is a single aggregated sequence of frame features.

The architecture resembles Vaswani *et al.*'s Transformer block [29] but replaces the multihead attention with the multiview attention (see Figure 5). Utilizing the multiview attention module, numerous sequences of features are consolidated into a unified sequence of features. First, the assemblage of view features undergoes augmentation with view positional embedding and frame positional embedding. Incorporating view positional embedding is crucial due to the necessity of computing attention scores across all frames across all videos. This enables the multiview attention mechanism to evaluate the interplay of each frame in each video concerning all other frames across all views, facilitating the model's capacity to discern cross-view patterns along the timestamp axis.

Subsequently, the output of the Multiview Ensembler is concatenated with the text embedding before being processed through the transformer blocks. This combined input configuration aids the model in predicting descriptions based on inputs from multiple videos, enhancing its capability to generate contextually relevant descriptions. The text decoder is a generative pre-trained language model, such as Llama [28], Vicuna [4], Phi [7], and Mistral [9]. The text decoder receives the aggregated view features. It creates detailed captions to describe the scenario through multiple video view features, similar to the text decoder of the single-view model in Section 3.2.1.

3.3. Rule Engine Refinement

3.3.1 Pedestrian-aware module.

Pedestrian description color. For each video and each phase in videos, we extract the bounding box image of the target pedestrian in the caption as pbbox $= [\text{bbox}_{v_1}^{p_0}, \text{bbox}_{v_1}^{p_1}, \text{bbox}_{v_1}^{p_2}, \text{bbox}_{v_1}^{p_3}, \text{bbox}_{v_1}^{p_4} ... \text{bbox}_{v_n}^{p_4}]$, where $\text{bbox}_{v_i}^{p_j}$ is the bounding box image in video i-th at phase j-th, $j \in [0, 4]$. The bounding boxes are divided horizontally into two halves for cloth color detection. We use the first half to detect the top clothing color and the second half to detect the clothing color at the bottom.

We sequentially fit all pixels of the first half $\text{bbox}_{v_i}^{p_j}$ in pbbox with K-mean clustering at $k = 15$ (limit to maximum 15 color clustering), calculate the histogram of all clusters and their centroid pixels to format human-readable color. We select the color with the maximum frequency in the histogram as the detected color. After applying K-mean clustering for each bounding box image in pbbox, we will have a list of possible detected colors. We then apply majority voting to determine the top clothing color with the highest frequency within this list. We repeat the same process to determine the bottom clothing color.

Pedestrian awareness. Intuitively, to detect whether the pedestrian is aware of the approaching vehicle, it is necessary to know where the pedestrian is looking. Inspired by this, we estimate the pedestrian's gaze vector, and the pedestrian notices the vehicle if the gaze vector intersects the vehicle bounding box; otherwise. For gaze estimation, we take advantage of the off-the-shelf GazeNet [23] that allows us to calculate the gaze direction without needing eyes' visual features, which is an advantage compared with the existing methods. Besides the mentioned advantage, this method also has a sophisticated preprocessing step. Specifically, it requires the head position, the whole body image, and the body velocity vector. Therefore, we employ the BoT-SORT [1] to track all pedestrians and vehicles. CLIP [25] is used to extract the tracks that contain the target pedestrians and cars by retrieving the highest similarity score between the candidate tracks and the annotated bounding boxes since the provided bounding boxes are missing for some phases in each video and contain errors. After obtaining visible bounding boxes of the pedestrian across frames, we employ the AlphaPose [6] to detect 26 full-body keypoints. The head, neck, nose, left and right of both ears and eyes coordinates corresponding to the $0, 1, 2, 3, 4, 17, 18$-th keypoints are collected and expanded to create a head bounding box. The other keypoints are used to get the body bounding boxes; assume that the body bounding box at the i-th frame is $[t_i, l_i, h_i, w_i]$, the center of the body at this frame would be $[x_i, y_i]$ where $x_i = l_i + w_i/2$ and $y_i = t_i + h_i/2$. The body velocity vector of a frame would be the vector between 2 body centers of the current frame and its consecutive frame, which is calculated by $\vec{v} = [y_{i+1} - y_i, x_{i+1} - x_i]$. During inference, ten frames are taken from each video's camera view from each phase. If a frame exists where the pedestrian's gaze vector intersects with the vehicle's bounding box, then the pedestrian is aware of the approaching vehicle at that phase.

Figure 6. Example of the pedestrian is not aware (left) and aware (right)

3.3.2 Vehicle-aware module

In this section, the rules we apply to refine the vehicle description are described in detail.

Vehicle Position Rule. Mainly, to consider the relative position between two objects on the road, we have four main positions: "on the left side," "on the right side," "in front of," and "behind." The most important thing that affects the relative position of two objects is the viewing direction of the landmark object. We determine the viewing direction of the object here, which is the vehicle, by identifying 2 points as the centers of 2 bounding boxes of 2 consecutive phases. With the current-phase vehicle point $A(x_A, y_A)$ and next-phase vehicle point $B(x_B, y_B)$, we define $P(x_P, y_P)$ is the pedestrian point in current phase. We then rely on the sign and value of counterclockwise $CCW(A, M, B)$. We consider that whether a human is on the right or left side of the car when the value of CCW falls outside the overlap value between the two cases, a threshold of 0 cannot be selected because there can be no guar-

Figure 7. An example of utilizing the algebraic area to identify vehicle action using overhead view.

antee that the car will go extremely straight. After analysis, we choose $\alpha_L = 7208.9652$ for the left side and $\alpha_R = -6419.8733$ for the right side.

$$side = \begin{cases} "left", & ccw \geq \alpha_L \\ "right", & ccw \leq \alpha_R \end{cases} \quad (3)$$

Vehicle motion. In the vehicle motion analysis task, the target predicts the vehicle's speed for a video sequence.

The speed corresponding to each video segment is extracted from the vehicle captions using regular expression (regex) techniques to achieve this. These speeds are categorized into seven distinct labels: $\{0, 5, 10, 15, 20, 25, 30\}$.

The preprocessing phase entails using sliding window methodology and motion blur imaging to construct the input for the speed prediction model. Initially, a sequence of frames is isolated for each segment, after which sliding windows are employed to generate numerous 5-frame segments. Subsequently, a motion blur image is generated for each segment by averaging the frames.

Following this preprocessing, a pretrained Vision Transformer model [5] is finetuned with the data extracted from the training subset. The model's efficacy is then evaluated on the validation subset of the competition dataset.

Vehicle action. For vehicle action detection, we replicate research proposed by Nguyen *et al.* [22], which is used only on overhead view. We exploit a series of bounding box centers created by tracking the target vehicle. The movement behaviors of the car (turning left, turning right, or going straight) are described by the sign and magnitude of the algebraic area of the polygon formed from these sequences of bounding box centers. Figure 7 shows how the algebraic area is applied.

3.3.3 Context-aware module

The context-aware module is precisely engineered to reconstruct pseudo captions by incorporating external contextual factors beyond pedestrian and vehicle objects. These factors encompass environmental variables such as brightness levels, weather conditions, and the state of the road surface.

Context brightness. Our approach involves randomly cropping shots from training videos and analyzing the optimal lowest threshold for average color intensity across width, height, and RGB channels. This threshold distinguishes "dark" from "bright" scenes. Following an analysis of the training set, in instances where the average color intensity of an image registers below 60, all occurrences of the critical phrase "bright" are systematically substituted with "dark."

Context weather. We employ a straightforward method based on the number of detected umbrellas to classify the context as "sunny" or "rainy." Using YOLOv8 [10], we detect umbrellas within entire videos, focusing solely on the umbrella class. If the count of detected umbrellas exceeds two (accounting for occasional use by pedestrians on sunny days), we classify the context as "rainy."

Context road surface. Initially, we employ random shot cropping followed by applying the Segment-Anything - SAM [11] technique to delineate the main road within the videos. Subsequently, a pretrained PSPNet [42] model is utilized to segment water puddles from the main road. If the ratio of the water puddle area to the image area exceeds 0.3, we modify the descriptions of the road surface from "dry" to "wet."

4. Experiment and Evaluation

This section presents our proposed method's experimental findings and evaluations, employing quantitative and qualitative methodologies. In Section 4.1, we delineate the configurations within various modules of our captioning and refinement methodologies. Additionally, we conduct a preliminary ablation study, examining the impact of diverse configuration strategies on the performance outcomes. We offer illustrative examples for qualitative assessment in Section 4.2, providing insightful analyses of specific cases to supplement the quantitative results.

4.1. Experiment Results

In Track 2 of the AI City Challenge 2024 [34], our methodology meticulously adheres to the stipulated guidelines, limiting our utilization exclusively to the training data supplied by the competition organizers. Both the training and test sets encompass proprietary internal data - WTS [12] dataset and externally sourced data - BDD_PC_5K [38] dataset. Our model undergoes training on the entirety of the training dataset, and predictions are made across the entirety of the test set to ascertain the model's capacity for generalization across a spectrum of scenarios.

The evaluation process for the generated captions will be conducted based on four distinct metrics: BLEU_4 [24],

METEOR [3], ROUGE_L [20], and CIDEr [30]. These metrics gauge the textual overlap and semantic correspondence between the generated and ground truth captions. Subsequently, the challenge will employ a weighted sum to amalgamate these metrics, thereby deriving the $S2$ score. This score will be further refined by computing the average $S2$ score across both the internal WTS [12] and external BDD_PC_5K [38] datasets for ranking purposes.

Table 1 presents the initial ablation analysis concerning various configuration methodologies aimed at assessing the individual contributions of each element within our proposed framework. In Version V1, we adopt a multi-view architecture trained from scratch in conjunction with a rule engine, yielding an $S2$ score of 18.9794 across the entire test dataset. Initiating training of the multi-view model from scratch, devoid of any pretraining, may incur minimal generalization costs during the evaluation on the test set. However, upon integrating motion-blur (BEiT) [35] with the rule engine in Version V2, the $S2$ score demonstrates a notable enhancement, reaching 21.5941, particularly exhibiting efficacy within the interval test subset, with a score of 13.0185. Remarkably, Version V3, incorporating motion blur (BLIP2) [16], demonstrates superior performance compared to all other model versions with the highest score of 22.6721.

Furthermore, we anticipate further enhancement in results by incorporating a pretrained single-view model. Consequently, we examine Versions V4 and V5, where we deploy a single-view architecture equipped with a pretrained T5 decoder. In Version V3, lacking refinement, the $S2$ score reaches 21.8448. In contrast, Version V5, incorporating refinement techniques, achieves a higher S2 score of 22.6630. Despite the improved performance observed in both versions, whether with or without the rules engine, in comparison to Versions 1 and 2, they still fall short of the exceptional performance exhibited by Version V3.

Table 1. Ablation experiment on different configuration strategies

No.	Configuration	S2[i]	S2[e]	S2
V1	Multi-view (scratch) + RE	11.3999	26.5590	18.9794
V2	Motion-blur (BEiT3) + RE	**13.0185**	30.1698	21.5941
V3	Motion-blur (BLIP2) + RE	12.8885	**32.4557**	**22.6721**
V4	Single-view (T5 3B) + No RE	12.4882	31.2015	21.8448
V5	Single-view (T5 3B) + RE	12.9084	32.4176	22.6630

4.2. Case Study

In this section, we offer illustrative examples generated by various versions of our model, facilitating an in-depth ablation study.

In Figure 8, we present a comparative analysis of our model's performance with and without action refinement during the post-processing phase. Our findings reveal instances where the captioning model exhibits limitations in accurately describing crucial textual elements. By incorporating refinement techniques, discernible enhancements are observed, particularly in the fidelity of captioned content about pedestrian awareness, vehicle dynamics, and environmental contextualization.

In Figure 9, we present a comparative analysis of various architectural models focusing on their respective performance metrics. The multi-view model demonstrates proficiency in capturing overarching environmental features such as weather conditions and road surfaces; however, it exhibits significant limitations in delineating the precise spatial relationship between pedestrians and vehicles. Conversely, the single-view and motion-blur-image models accurately describe pedestrian and vehicle attributes. Evaluation across precision, comprehensiveness, semantic fidelity, and overall quality consistently places the motion-blur model as the highest-performing model.

5. Conclusion

In conclusion, this research paper introduces a solution for Track 2 in the AI City Challenge 2024 [34] for the Traffic Safety Description and Analysis task. The proposed approach provides various multimodal techniques to increase the video visual features effectively, therefore unlocking the potential of linking the motion information with visual features by the motion blur model, as well as the comprehensive relationship across multiple frames and camera views with the multiview ensembler. A rule-based refinement engine was seamlessly integrated to rectify the final description based on multi-contextual attribute information. Our proposed approach achieves a competitive result of 22.6721 S2 score, demonstrating its promising performance in describing and analyzing the real-world traffic safety scenarios.

Acknowledgement

This research was funded by Vingroup and supported by Vingroup Innovation Foundation (VINIF) under project code VINIF.2019.DA19.

References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking, 2022. 5

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022. 2

[3] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved

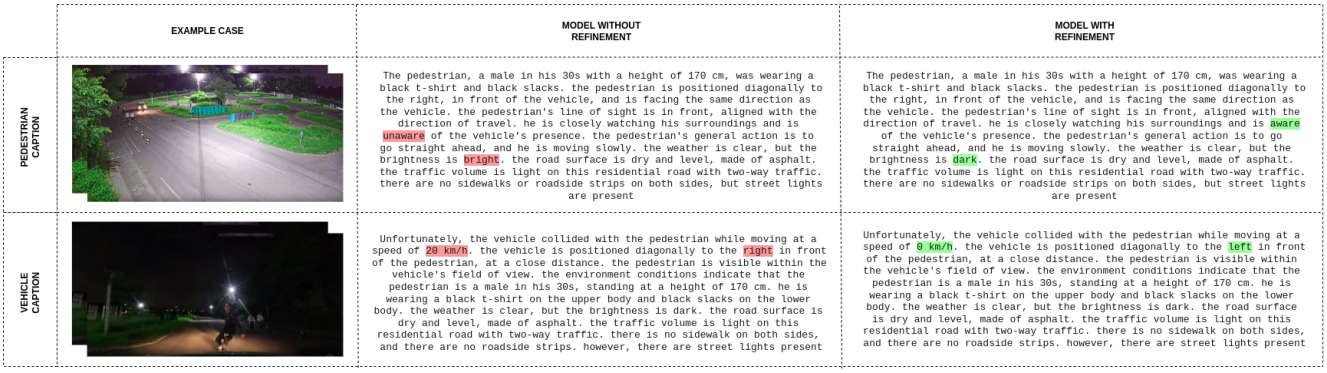

Figure 8. Case study on rule engine refinement

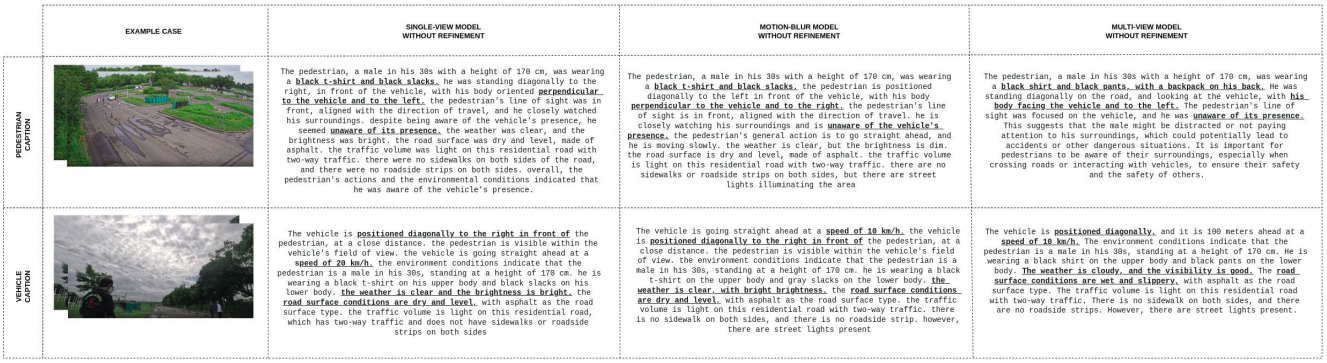

Figure 9. Case study on three models architecture

correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, 2005. Association for Computational Linguistics. 7

[4] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 2, 4

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 6

[6] Hao-Shu Fang, Jiefeng Li, Hongyang Tang, Chao Xu, Haoyi Zhu, Yuliang Xiu, Yong-Lu Li, and Cewu Lu. Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 5

[7] Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023. 4

[8] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 2

[9] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023. 4

[10] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO, 2023. 6

[11] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023. 6

[12] Quan Kong, Yuki Kawana, Rajat Saini, Ashutosh Ku-

mar, Jingjing Pan, Ta Gu, Yohei Ozao, Balazs Opra, David C. Anastasiu, Yoichi Sato, and Norimasa Kobori. Wts: A pedestrian-centric traffic video dataset for fine-grained spatial-temporal understanding. 2024. 1, 6, 7

[13] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *CoRR*, abs/1808.06226, 2018. 3

[14] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022. 2

[15] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*. JMLR.org, 2023. 4

[16] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 2, 7

[17] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 2

[18] Linjie Li, Zhe Gan, Kevin Lin, Chung-Ching Lin, Zicheng Liu, Ce Liu, and Lijuan Wang. Lavender: Unifying video-language understanding as masked language modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23119–23129, 2023. 2

[19] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023. 2

[20] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, 2004. Association for Computational Linguistics. 7

[21] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023. 2

[22] Tien-Phat Nguyen, Ba-Thinh Tran-Le, Xuan-Dang Thai, Tam V. Nguyen, Minh N. Do, and Minh-Triet Tran. Traffic video event retrieval via text query using vehicle appearance and motion attributes. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 4160–4167, 2021. 2, 6

[23] Soma Nonaka, Shohei Nobuhara, and Ko Nishino. Dynamic 3d gaze from afar: Deep gaze estimation from temporal eye-head-body coordination. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2192–2201, 2022. 5

[24] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, 2002. Association for Computational Linguistics. 6

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 3, 5

[26] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 2

[27] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35: 10078–10093, 2022. 2

[28] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 4

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[30] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. *CoRR*, abs/1411.5726, 2014. 7

[31] Jinpeng Wang, Yixiao Ge, Rui Yan, Yuying Ge, Kevin Qinghong Lin, Satoshi Tsutsui, Xudong Lin, Guanyu Cai, Jianping Wu, Ying Shan, et al. All in one: Exploring unified video-language pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6598–6608, 2023. 2

[32] Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan

Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14549–14560, 2023. 2

[33] Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Yu-Gang Jiang, Luowei Zhou, and Lu Yuan. Bevt: Bert pretraining of video transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14733–14743, 2022. 2

[34] Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mohammed Shaiqur Rahman, Meenakshi S. Arya, Anuj Sharma, Pranamesh Chakraborty, Sanjita Prajapati, Quan Kong, Norimasa Kobori, Munkhjargal Gochoo, Munkh-Erdene Otgonbold, Ganzorig Batnasan, Fady Alnajjar, Ping-Yang Chen, Jun-Wei Hsieh, Xunlei Wu, Sameer Satish Pusegaonkar, Yizhou Wang, Sujit Biswas, and Rama Chellappa. The 8th AI City Challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2024. 1, 6, 7

[35] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, and Furu Wei. Image as a foreign language: Beit pretraining for all vision and vision-language tasks, 2022. 4, 7

[36] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14668–14678, 2022. 2

[37] Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *CVPR*, 2023. 2

[38] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving video database with scalable annotation tooling. *CoRR*, abs/1805.04687, 2018. 6, 7

[39] Liangzhe Yuan, Nitesh Bharadwaj Gundavarapu, Long Zhao, Hao Zhou, Yin Cui, Lu Jiang, Xuan Yang, Menglin Jia, Tobias Weyand, Luke Friedman, et al. Videoglue: Video general understanding evaluation of foundation models. *arXiv preprint arXiv:2307.03166*, 2023. 2

[40] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. Merlot reserve: Neural script knowledge through vision and language and sound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16375–16387, 2022. 2

[41] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 2

[42] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6230–6239, 2017. 6

[43] Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, Wang HongFa, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, Cai Wan Zhang, Zhifeng Li, Wei Liu, and Li Yuan. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment, 2023. 2