

TrafficVLM: A Controllable Visual Language Model for Traffic Video Captioning

Quang Minh Dinh¹ Minh Khoi Ho² Anh Quan Dang² Hung Phong Tran²

¹ Simon Fraser University, BC, Canada

² Hanoi University of Science and Technology, Hanoi, Vietnam

qmd@sfu.ca

{khoi.hm204917, quan.da203909, phong.tnh200465}@sis.hust.edu.vn

Abstract

Traffic video description and analysis have received much attention recently due to the growing demand for efficient and reliable urban surveillance systems. Most existing methods only focus on locating traffic event segments, which severely lack descriptive details related to the behaviour and context of all the subjects of interest in the events. In this paper, we present TrafficVLM, a novel multi-modal dense video captioning model for vehicle ego camera view. TrafficVLM models traffic video events at different levels of analysis, both spatially and temporally, and generates long fine-grained descriptions for the vehicle and pedestrian at different phases of the event. We also propose a conditional component for TrafficVLM to control the generation outputs and a multi-task fine-tuning paradigm to enhance TrafficVLM's learning capability. Experiments show that TrafficVLM performs well on both vehicle and overhead camera views. Our solution achieved outstanding results in Track 2 of the AI City Challenge 2024, ranking us third in the challenge standings. Our code is publicly available at <https://github.com/quangminhdinh/TrafficVLM>.

1. Introduction

The recent advancements in dense video captioning models, which can precisely localize and describe incidents within a continuous video stream, have brought new opportunities and challenges to the field. This capability is particularly crucial in complex urban environments where the dynamic interactions between pedestrians, vehicles, and other elements can lead to accidents. As urban areas continue to grow and traffic becomes denser, the ability to automatically and accurately identify and describe accident scenarios from multiple perspectives becomes essential. This not only aids in immediate response efforts but also informs the development of safer, more intelligent transportation solu-

tions.

Most traditional deep learning systems for this particular domain virtually utilize the supervising training approach to predict frames or localize the incident segments [21, 47, 50]. While models of this kind may capture the nuanced details of the traffic interactions, they lack the ability to explain their predictions, making it difficult to analyze causes, predict outcomes, and develop preventative measures. In response to this challenge, the 8th AI City Challenge [34] introduces the Traffic Safety Description and Analysis task, which involves detailed video captioning of traffic safety scenarios for both vehicles and pedestrians, given the videos from multiple static overhead cameras or moving vehicle ego cameras.

In this work, we introduce TrafficVLM, which leverages the advancements of multi-modality dense video captioning models, specifically adapted to the traffic domain. TrafficVLM extracts different layers of visual features from the vehicle camera frames to locate different phases of the traffic events and then provide detailed descriptions for different targets. Our contributions can be summarized as follows:

- We reformulate the multi-phase Traffic Safety Description and Analysis task as a temporal localization and dense video captioning task with a single sequence as the output and introduce TrafficVLM, a video language model that is adapted specially to this task and domain.
- In accordance with our new fine-tuning objective, we propose a method to model the video features at different levels, enabling our model to effectively capture the fine-grained visual details, both spatially and temporally.
- We make use of the availability of captions for different targets in the dataset to devise a multi-task fine-tuning paradigm, allowing TrafficVLM to effectively learn the alignments between the video and textual features for all phases.
- We achieved the third rank on the blind test set of the

AI City Challenge 2024 Track 2 with high results, which show the competitiveness of our solution.

2. Related Works

2.1. Traffic Accident Detection with Descriptions

Traffic accident detection aims to identify accidental traffic incidents during driving, such as collisions among vehicles and objects and loss of control. Early methods for this task [50] involve using a manual feature extraction process and simply utilizing a Bayesian model to detect traffic incidents, which lacks the ability to generalize and is sensitive to the extracted features. Further advancements in deep neural networks allow researchers to deploy deep learning approaches to reconstruct and predict errors in video frames to detect traffic accidents [4, 8, 21]. In recent years, to counter the problem of hectic backgrounds, researchers have applied a two-stage process. Initially, architectures such as Mask-RCNN [9], FlowNet [12], DeepSort [38] or ORBSLAM [22] are used, for instance by Yao *et al.* [47] to extract visual features such as bounding boxes, optical flow, tracking ids, and ego-motion correspondingly. Then, a detection process is applied to these features for the final classification.

On the other hand, the latest applications in deep learning revolve around the power of multi-modality models. For instance, Liang *et al.* [19] exploit the language features, which are already aligned with visual features by extensive training of CLIP [25], as supervised signals while capturing the dynamic changes of driving scenes in the high temporal domain. A few other methods even approach the problem from a generative perspective, such as TRIVIA [24], which infuses traffic-domain knowledge into a large video-language model to elevate the combined capabilities derived from having been multi-modality pre-trained.

2.2. Video Visual Language Models

Together with the advancements of image-text pre-training models [1, 6, 11, 17, 25, 30–32, 36, 49, 55], recent works are moving toward video-text pre-training, with some of the representatives are [18, 35, 40, 43, 56]. Similar to image-text pre-training, while these models are excellent at capturing global semantic understanding of the videos, they still struggle with temporal localization and are not suitable to use off-the-shelf in a generative manner.

2.3. Dense Video Captioning and Localization

Dense video captioning takes a further step ahead to understand events in the video as it uses timestamp information. Solutions for this task often interest in capturing all the details in a video that describe major events that happen inside it. The general flow for dense video captioning consists of three steps: (1) Video Feature Extraction, (2) Tempo-

ral Event Localization, and (3) Dense Caption Generation. Video feature extraction commonly uses strong video visual language models mentioned above, while traditional approaches for temporal event localization can be categorized into two types: proposal-based [3, 52, 53] and proposal-free methods [7, 51, 54]. Proposal-based techniques generate candidate proposals before ranking them based on relevance. In contrast, proposal-free methods directly predict the start and end boundaries of the target moment. Early approaches often complete the dense caption generation task by adding dedicated transformer models [41, 42, 44], while more recent approaches [10, 33, 45, 46] in this field have witnessed a shift towards joint training of captioning and localization modules. For instance, Vid2Seq [46] enhances a language model by incorporating specific time tokens, enabling the model to generate event boundaries and textual descriptions within the unified output sequence. Furthermore, VTimeLLM [10] leverages the power of a large language model to enable natural language interaction with humans, while presenting excellent generalization in video understanding and event localization.

3. Method

Traffic Safety Description and Analysis is a challenging task which involves the long fine-grained captioning of different continuous phases of traffic safety scenarios for multiple targets, given the camera video, the timestamp of each phase, and the target bounding box information for a number of frames. Specifically, for each phase, the goal is to describe the surrounding context, attention, location, and behaviour of the pedestrian and vehicle in detail. In this section, we present TrafficVLM, a video language model that involves fine-tuning a temporal transformer encoder and the decoder of a large language model. Following Vid2Seq [46], we reformulate this task as a temporal localization and dense video captioning task wherein the model learns to predict both the event boundaries and the descriptions for a target as a single sequence of tokens. In Sec. 3.2, we introduce a controllable component which allows TrafficVLM to generate multiple captions for different targets. We also show how to make use of different target captions to enhance our fine-tuning paradigm in Sec. 3.3.

Problem Formulation. Given a vehicle camera video $V \in \mathbb{R}^{T \times H \times W \times C}$ with T frames, the event boundary sequence $s = \{(start_i, end_i)\}_{i=1}^P$ with P phases, and a list of pedestrian bounding boxes $b = \{(x_i^{st}, x_i^{ed}, y_i^{st}, y_i^{ed})\}_{i=1}^n$ for $n < T$ frames, the goal is to generate two sequences $\tilde{v} = \{\tilde{v}_i\}_{i=1}^{L^v}$ for *vehicle* and $\tilde{p} = \{\tilde{p}_i\}_{i=1}^{L^p}$ for *pedestrian* that contain both the temporal information and the textual descriptions for all P phases. The final P captions of both targets, $v = \{v_i\}_{i=1}^P$ and $p = \{p_i\}_{i=1}^P$, can be decoded and extracted directly from \tilde{v} and \tilde{p} .

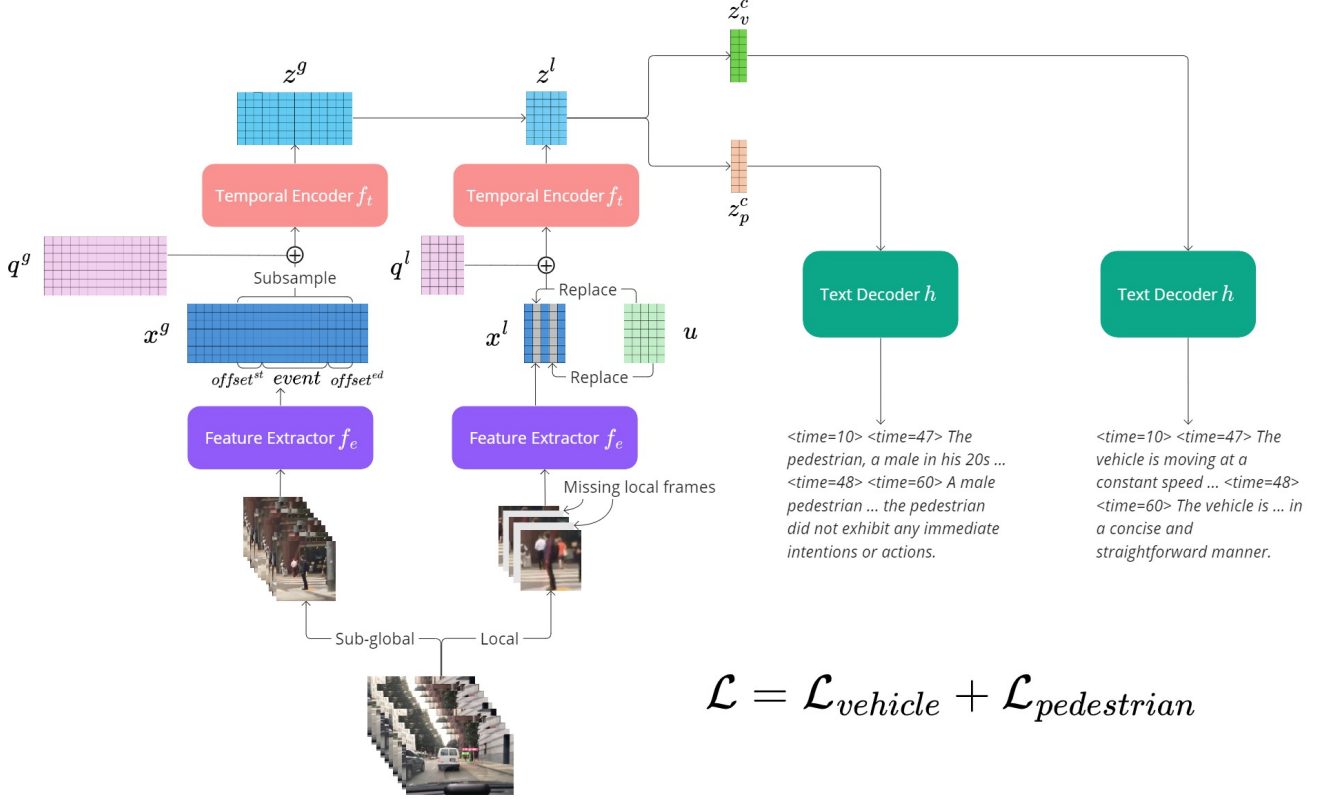


Figure 1. Overview of our method. First the sub-global and local frame sequences are extracted from the vehicle camera video. Some local frames might be missing, depending on the availability of the bounding boxes in the event segment. A visual feature extractor f_e is then applied to both of them to get the two visual embeddings x^g and parts of x^l . The sub-global embedding is trimmed to the event segment and subsampled to create the sub-global feature \tilde{x}^g . Feature vectors in the learnable local tensor u are added to the local embedding as the embeddings for the missing phases. Positional embeddings are then applied to both visual features, followed by the temporal encoder f_t . The final embeddings z^g and z^l are concatenated with the conditional embedding z_v^c for *vehicle* or z_p^c for *pedestrian* to control the generation output. The text decoder h receives the concatenated embedding as the input and autoregressively generates the output sequence. For fine-tuning, the final loss is calculated by combining the losses for generating the *vehicle* and *pedestrian* output sequences.

3.1. Visual Feature Extraction

Main Feature. To remove redundant information, we extract our main video feature at a sub-global level by cropping each frame equally to a target segment that contains all local pedestrian features across T frames. Explicitly, we select a segment $(x_{min}^{st}, x_{max}^{ed}, y_{min}^{st}, y_{max}^{ed})$ from the bounding box information b and extend the shorter dimension to length

$$\tilde{W} = \max(y_{max}^{ed} - y_{min}^{st}, x_{max}^{ed} - x_{min}^{st}) \quad (1)$$

to make the selected segment square. We crop all T frames to this area, which results in $\tilde{V} \in \mathbb{R}^{T \times \tilde{W} \times \tilde{W} \times C}$. Our visual feature extractor, which is denoted as f_e , is a frozen CLIP ViT-L/14 [25] that processes at resolution 224×224 pixels. We resize each cropped frame to the target resolution and encode it independently to get the visual embedding for the

entire video:

$$x^g = \{f_e(\tilde{V}_i) | \tilde{V}_i \in \tilde{V}\} \in \mathbb{R}^{T \times d}, \quad (2)$$

with d being the dimension of the embedding.

We continue to use the event boundary sequence s to trim the video feature to the segment that is related to the traffic event. To diversify the segment duration and the phase timestamps for training, we randomly select two offset durations $offset^{st}$ and $offset^{ed}$ between 0s and 5s at the video extracted frame rate and change the segment start and end frame accordingly to increase the duration. Specifically, the video feature x^g is trimmed to

$$\begin{aligned} &(\max(0, start_1 - offset^{st}), \\ &\min(end_P + offset^{ed}, T)). \end{aligned} \quad (3)$$

We subsample the trimmed feature to a smaller frame rate and then continue to subsample or zero-pad it to F frames.

The final feature $\tilde{x}^g \in \mathbb{R}^{F \times d}$ is used as the main video feature for training.

Local Feature. For each phase $P_i \in P$, we randomly select a single pedestrian bounding box $b_{P_i} \in b$ to construct our local features. Similar to how we process our main feature, we square crop each frame corresponding to bounding box b_{P_i} to the segment that is close to the bounding box and resize it to 224×224 pixels, represented as $l_i \in \mathbb{R}^{224 \times 224 \times C}$. The cropped segment is then separately encoded with our visual extractor to get:

$$x_i^l = f_e(l_i) \in \mathbb{R}^d. \quad (4)$$

As illustrated in Fig. 1, there can be several scenarios where there is no bounding box b_{P_j} in relation to phase $P_j \in P$. To address this issue, we add a learnable tensor $u = \{u_i\}_{i=1}^P \in \mathbb{R}^{P \times d}$, and replace the missing visual feature x_j^l with the learnable embedding u_j at the corresponding phase. All phase visual features are aggregated to form the final local embedding $x^l = \{x_i^l\}_{i=1}^P \in \mathbb{R}^{P \times d}$.

In practice, we extract all raw CLIP features x^g and x^l prior to training to save computation time and resources. Details about the selection of features will be explained in Sec. 4.2.

3.2. Architecture

Temporal Visual Encoder. We model the temporal dynamics for the frames of both the sub-global feature \tilde{x}^g and the local feature x^l using two learnable positional embeddings $q^g \in \mathbb{R}^{F \times d}$ and $q^l \in \mathbb{R}^{P \times d}$, and a vision transformer [5] f_t , which results in the final visual embeddings z^g and z^l :

$$z^g = f_t(\tilde{x}^g + q^g) \in \mathbb{R}^{F \times d} \quad (5)$$

$$z^l = f_t(x^l + q^l) \in \mathbb{R}^{P \times d} \quad (6)$$

Generation Output Control. We make use of how the transformer decoder uses the multi-head attention layer to attend to the input of the encoders and design the conditional module in a way that allows the model to learn the condition itself. Simply, for both targets *vehicle* and *pedestrian*, we add the additional learnable conditional embeddings $z_v^c \in \mathbb{R}^{k \times d}$ and $z_p^c \in \mathbb{R}^{k \times d}$, where k is the extended dimension of the embeddings.

Text Decoder. To generate the output sequences, we employ T5-Base [26] as our transformer decoder h . The output sequences contain both the pseudo-timestamps and the textual descriptions for all phases:

$$\tilde{v} = \{\tilde{v}_i\}_{i=1}^{L^v} = h(z^g, z^l, z_v^c) \quad (7)$$

$$\tilde{p} = \{\tilde{p}_i\}_{i=1}^{L^p} = h(z^g, z^l, z_p^c), \quad (8)$$

where L^v and L^p are the lengths of the *vehicle* and *pedestrian* sequences correspondingly. As the main video features are trimmed to the target scenario segments, we do not include the temporal boundaries information in the inputs to the decoder to force the model to learn the temporal alignment between the visual features of each phase and the matching textual description. Details about the output sequences will be given in the following parts.

Time Tokenization. As T5-Base was selected for our text decoder, we also use the T5 tokenizer, which is based on the SentencePiece tokenizer [16] and is publicly available on the HuggingFace library [39]. Following Vid2Seq, we extend the tokenizer by adding $N = 100$ additional time tokens, which represent the relative timestamps in each video segment.

Output Sequence Construction. This part explains how we construct the two output sequences to be used as groundtruths during training and their format. We first adjust the event boundary sequence s to the new start time $start^n = \max(0, start_1 - offset^{st})$ and end time $end^n = \min(end_P + offset^{ed}, T)$ at Eq. (3) to create the new event boundary sequence s^g :

$$\begin{aligned} s^g &= \{(start_i^g, end_i^g)\}_{i=1}^P \\ &= \{(start_i - start^n, end_i - start^n)\}_{i=1}^P \end{aligned} \quad (9)$$

The new duration D of the video segment is obtained by:

$$D = end^n - start^n \quad (10)$$

Each timestamp in the new event boundary sequence s^g is then rescaled to an integer between 0 and $N - 1$ to construct the quantized event boundary sequence s^q :

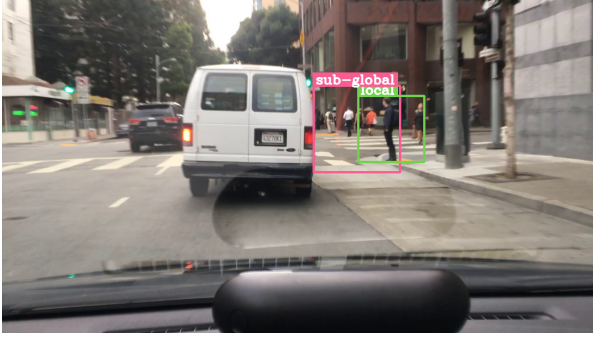
$$s^q = \left\{ \left(\left\lfloor \frac{start_i^g \times N}{D} \right\rfloor, \left\lfloor \frac{end_i^g \times N}{D} \right\rfloor \right) \right\}_{i=1}^P \quad (11)$$

We map each quantized timestamp to one of N time tokens, the result sequence is denoted as t :

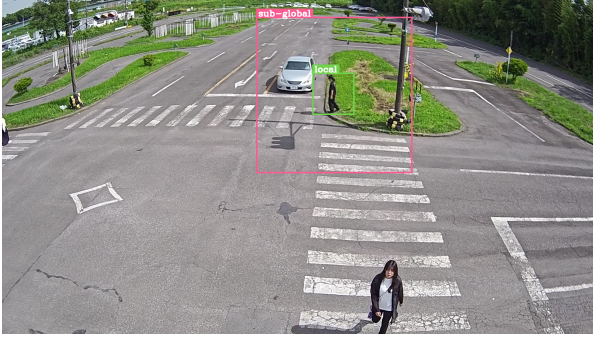
$$t = \{(t_i^{st}, t_i^{ed})\}_{i=1}^P \quad (12)$$

For the *vehicle* textual tokens, we first tokenize all phase descriptions of the caption sequences v to obtain $v^t = \{v_i^t\}_{i=1}^P$, where $v_i^t = \{v_{i,j}^t\}_{j=1}^{L_i^v}$ is the textual token sequence for phase i . Similar to Vid2Seq, we construct a new sequence for each phase i by concatenating the start time token t_i^{st} , the end time token t_i^{ed} , and all the textual tokens $v_{i,j}^t$ in v_i^t . We concatenate all such sequences in increasing order of the start time and add a *BOS* token at the beginning and a *EOS* token at the end of the sequence. The result is the final output sequence \tilde{v} :

$$\begin{aligned} \tilde{v} &= \{\tilde{v}_i\}_{i=1}^{L^v} \\ &= [BOS, t_1^{st}, t_1^{ed}, v_{1,1}^t, \dots, v_{1,L^v}^t, t_2^{st}, \dots, EOS] \end{aligned} \quad (13)$$



(a) Vehicle view.



(b) Overhead view.

Figure 2. Sub-global and local segments for vehicle and overhead camera views.

The output sequence \tilde{p} is obtained in the same manner:

$$\begin{aligned} \tilde{p} &= \{\tilde{p}_i\}_{i=1}^{L^p} \\ &= [BOS, t_1^{st}, t_1^{ed}, p_{1_1}^t, \dots, p_{1_{L^p}}^t, t_2^{st}, \dots, EOS] \end{aligned} \quad (14)$$

Phase Captions Reconstruction. During inference, the P captions for both *vehicle* and *pedestrian* are recovered by using the tokenizer to decode \tilde{v} and \tilde{p} , and regular expression to extract each of the captions from the combined sequences.

3.3. Fine-tuning

To leverage the two different sets of captions provided by the dataset and let the model learn both the spatial and temporal alignments between the video and textual features better, we directly use *vehicle captioning* and *pedestrian captioning* as our two fine-tuning tasks. For both tasks, we apply the standard maximum likelihood objective, with teacher forcing [37] and a cross-entropy loss:

$$\mathcal{L}_\theta^c(z^g, z^l, z^c, y) = -\mathbb{E} \left[\sum_{i=1}^{L-1} \log p_\theta(y_{i+1} | z^g, z^l, z^c, y_{<i}) \right], \quad (15)$$

where z^g is the sub-global video feature, z^l is the local video feature, z^c is the conditional embedding, y is the target output sequence, and L is the length of y .

Split	Subset	No. scenes	No. videos
Train	Main	97	425
	Normal trimmed	70	114
	External	2430	
Valid	Main	48	210
	Normal trimmed	34	61
	External	972	
Test	Main	49	224
	Normal trimmed	35	65
	External	375	

Table 1. Statistics of the WTS dataset.

The final loss is the combination of the losses of both tasks:

$$\mathcal{L}_\theta = \mathcal{L}_\theta^c(z^g, z^l, z_v^c, \tilde{v}) + \mathcal{L}_\theta^c(z^g, z^l, z_p^c, \tilde{p}) \quad (16)$$

4. Experiments

4.1. Experimental Setup

Datasets. For fine-tuning and evaluation, we use the WTS dataset [15], which is introduced along with the Traffic Safety Description and Analysis track in the AI City Challenge 2024. The dataset contains 155 scenarios and 810 videos from both fixed overhead cameras and vehicle cameras, as illustrated in Tab. 1. In addition, the dataset also provides 3402 vehicle camera videos extracted from the BDD100K dataset [48]. Each scenario has around 5 phases, for which there is an annotation file capturing the location, attention, behaviour, and context information of both the vehicle and pedestrian in detail. Target pedestrian bounding boxes are provided for a number of frames in each video, and target vehicle bounding boxes are also provided for the overhead videos.

As the vehicle camera videos take up the majority of the WTS dataset, our fine-tuning data pipeline mainly extracts the visual features from the vehicle videos. We still randomly add in some overhead camera videos, which occupy up to 10% of the data pipeline to create diversity in the fine-tuning data. All of our experiments are evaluated on the main subset of the WTS validation set. We generate vehicle and pedestrian captions for all subsets in the WTS test set and submit them to the AI City Challenge portal to obtain the test results.

Evaluation Metrics. Following the evaluation setups of Track 2 of the 8th AI City Challenge, we adopt BLEU-4 [23], ROUGE-L [20], METEOR [2], and CIDEr [29] as our main evaluation metrics. BLEU and ROUGE are both standard metrics for assessing the quality of machine translation systems based on n-grams overlap that focus on precision and recall respectively. METEOR is an improvement of the BLEU score by using an F_3 measure of unigram similarity,

Global	Sub-global	Local	Phase Encoder	Vehicle Captioning				Pedestrian Captioning				Score
				BLEU	ROUGE	METEOR	CIDEr	BLEU	ROUGE	METEOR	CIDEr	
	✓			0.395	0.584	0.478	0.607	0.302	0.37	0.424	0.323	33.08
✓				0.426	0.591	0.489	0.316	0.302	0.376	0.42	0.512	33.55
✓	✓	✓	✓	0.402	0.589	0.472	0.354	0.316	0.376	0.429	0.882	33.84
✓	✓			0.432	0.594	0.495	0.703	0.313	0.375	0.425	0.257	34.12
	✓	✓		0.411	0.598	0.479	0.647	0.326	0.375	0.437	0.412	34.15
✓		✓	✓	0.433	0.589	0.499	0.555	0.315	0.386	0.437	0.482	34.54
	✓	✓	✓	0.443	0.591	0.5	0.785	0.317	0.367	0.431	0.507	34.74

Table 2. Ablation study results for our TrafficVLM variations on the WTS main validation set, vehicle view. *Phase Encoder* denotes temporal modelling for the local feature. The final score used for selecting the models is calculated in the same manner as the evaluation score of the 8th AI City Challenge Track 2.

Global	Sub-global	Local	Phase Encoder	Vehicle Captioning				Pedestrian Captioning				Score
				BLEU	ROUGE	METEOR	CIDEr	BLEU	ROUGE	METEOR	CIDEr	
✓				0.394	0.564	0.467	0.384	0.297	0.373	0.416	0.171	32.08
	✓	✓		0.401	0.591	0.474	0.218	0.322	0.378	0.432	0.392	33.23
✓	✓	✓	✓	0.413	0.597	0.479	0.252	0.306	0.377	0.433	0.345	33.31
	✓			0.395	0.588	0.474	0.601	0.314	0.374	0.43	0.661	33.76
✓	✓			0.419	0.596	0.49	0.577	0.317	0.384	0.428	0.459	34.23
	✓	✓	✓	0.434	0.593	0.49	0.574	0.32	0.373	0.436	0.423	34.33
✓		✓	✓	0.419	0.589	0.493	0.609	0.316	0.378	0.438	0.61	34.43

Table 3. Ablation study results for our TrafficVLM variations on the WTS main validation set, overhead view.

alongside a penalty p for displacements of words in generated sentences. CIDEr measures the similarity between the reference and the candidate sentences via the cosine similarity of their TF-IDF [27] weights. The final score which is used to rank the models is calculated as the combination of all 4 metrics:

$$Score = \frac{1}{4} \times [100 \times (B + M + R) + 10 \times C] \quad (17)$$

where B , M , R , and C stand for BLEU-4, METEOR, ROUGE-L, and CIDEr respectively.

Implementation Details. We initialize both the temporal visual encoder and the text decoder with checkpoint *vid2seq-htmchaptersvitt* from VidChapters [45], which is publicly available. The local temporal positional embedding q^l is initialized by subsampling the sub-global positional embedding q^g , which is loaded from the VidChapters checkpoint. All videos are extracted at 30 FPS, and the main video features are subsampled to 3 FPS after the trimming. We set $F = 100$ as the final number of frames for the sub-global feature. The text decoder is truncated or padded to $L^v = L^p = 1024$ tokens. We apply the Adam optimizer [14] with a learning rate $3 \cdot 10^{-4}$, a cosine learning rate decay and a warm-up period. All of our models are fine-tuned with batch size 1 for 30 epochs on an NVIDIA RTX-3060 GPU for around 8 hours. We select the best checkpoint for each model based on the validation metrics.

4.2. Ablation Studies

To study the effects of different modules and feature levels on the performance of our models, we mix and match them to create different variations of TrafficVLM, the results of which are shown in Tab. 2 and Tab. 3.

Choice of features. In addition to the sub-global and the local features used in the main model, we conduct some experiments with the features extracted at a global level by square cropping the entire camera frames centering on all the pedestrian bounding boxes. In Tab. 2, we present the ablation results for the Traffic Safety Description and Analysis task on the vehicle branch of the main WTS validation set. As our main concern is the vehicle camera view, we use the results in this table to rank the models. It can be seen that using a combination of two or more features is significantly better than just using one of them. In the experiments where the sub-global feature is directly used in comparison with its global counterpart (row 1 vs row 2 and row 7 vs row 6), the results of both are mostly comparable. The two best models use the global or sub-global feature alongside the local feature with temporal modelling. Of the two of them, the one with the sub-global feature (row 7) outperforms its counterpart in 6 out of 8 metrics. Unfortunately, any combination of both the global and sub-global features does not give a high result compared to other options.

Performance on the Overhead Scenarios. Tab. 3 shows the results of our experiments on the overhead branch

Rank	Team Name	Score
1	AliOpenTrek	33.4308
2	AIO.ISC	32.8877
3	Lighthouse	32.3006
4	VAI	32.2778
5	Santa Claude	29.7838
6	UCF-SST-NLP	29.0084
7	Monitor	28.7485
8	X	27.7771
9	HCMUS_AGAIN	22.7371

Table 4. Public leaderboard of the 8th AI City Challenge Track 2.

of the main WTS validation set. Although overhead videos only take up to 10% of our training set due to their availability compared to the vehicle videos, it is apparent that our models still perform relatively well in this scenario, as most of the numbers are close to those in Tab. 2. Two models with the global or sub-global feature used together with the temporal encoded local feature still perform the best in this scenario.

Local Temporal Modelling. As most scenarios only have around 5 phases, it is one of our designing concerns on whether to do temporal modelling for the local feature (which is denoted as *Phase Encoder* in Tab. 2 and Tab. 3). We observe that in both scenarios, adding temporal modelling to the local feature significantly improves the model performance on most of the metrics (row 5 vs row 7 for Tab. 2 and row 2 vs row 6 for Tab. 3), which show the effectiveness of our design.

4.3. Performance in the Challenge

In Track 2 of the AI City Challenge 2024, we generate the result samples for the WTS internal blind test set (which contains both traffic scenarios and normal scenarios) with a model that uses only the sub-global feature. For the external blind set, we employ an ensemble method to combine that same model with the two best checkpoints of our main model using the generation confidence scores of the text decoder.

Tab. 4 presents the final public leaderboard of the 8th AI City Challenge Track 2. Our team (with the team name Lighthouse) achieved third place with a final score of 32.3006. The result of our solution is only behind the top two teams by a small margin, which shows the competitiveness of TrafficVLM for the Traffic Safety Description and Analysis task.

5. Conclusion

In this paper, we present TrafficVLM, a visual language model tailored to perform dense video captioning for traffic incidents, as well as an effective fine-tuning paradigm

leveraging the multi-target nature of the Traffic Safety Description and Analysis task. TrafficVLM achieved the third position in Track 2 of the AI City Challenge 2024 with an impressive score, demonstrating its effectiveness for the task. Besides dense video captioning and temporal localization, the current implementation of TrafficVLM could be extended to a variety of new tasks, including traffic video question answering, traffic video summarization, and other video understanding tasks. We believe that future works can enhance TrafficVLM further by exploring the use of different large language models, such as Llama2 [28] or Mistral [13] and employ different data augmentation strategies. The controllable design of TrafficVLM can be strengthened by using a text encoder to impose different conditions on the description generation.

References

- [1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Bińkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 23716–23736. Curran Associates, Inc., 2022. 2
- [2] Satandeep Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved 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. 5
- [3] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. Temporally grounding natural sentence in video. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 162–171, Brussels, Belgium, 2018. Association for Computational Linguistics. 2
- [4] Yong Shean Chong and Yong Haur Tay. Abnormal event detection in videos using spatiotemporal autoencoder. In *Advances in Neural Networks - ISNN 2017*, pages 189–196, Cham, 2017. Springer International Publishing. 2
- [5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 4
- [6] Zi-Yi Dou, Yichong Xu, Zhe Gan, Jianfeng Wang, Shuohang Wang, Lijuan Wang, Chenguang Zhu, Pengchuan Zhang, Lu Yuan, Nanyun Peng, Zicheng Liu, and Michael Zeng. An empirical study of training end-to-end vision-and-language

- transformers. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18145–18155, 2022. [2](#)
- [7] Soham Ghosh, Anuva Agarwal, Zarana Parekh, and Alexander Hauptmann. ExCL: Extractive Clip Localization Using Natural Language Descriptions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1984–1990, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. [2](#)
- [8] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K. Roy-Chowdhury, and Larry S. Davis. Learning temporal regularity in video sequences. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 733–742, 2016. [2](#)
- [9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. [2](#)
- [10] Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. Vtimellm: Empower llm to grasp video moments, 2023. [2](#)
- [11] Zhicheng Huang, Zhaoyang Zeng, Bei Liu, Dongmei Fu, and Jianlong Fu. Pixel-bert: Aligning image pixels with text by deep multi-modal transformers, 2020. [2](#)
- [12] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. FlowNet 2.0: Evolution of optical flow estimation with deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. [2](#)
- [13] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Léo Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. [7](#)
- [14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. [6](#)
- [15] Quan Kong, Yuki Kawana, Rajat Saini, Ashutosh Kumar, 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. [5](#)
- [16] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Conference on Empirical Methods in Natural Language Processing*, 2018. [4](#)
- [17] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, 2022. [2](#)
- [18] K. Li, Y. Wang, Y. Li, Y. Wang, Y. He, L. Wang, and Y. Qiao. Unmasked teacher: Towards training-efficient video foundation models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 19891–19903, Los Alamitos, CA, USA, 2023. IEEE Computer Society. [2](#)
- [19] Rongqin Liang, Yuanman Li, Jiantao Zhou, and Xia Li. Text-driven traffic anomaly detection with temporal high-frequency modeling in driving videos, 2024. [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. [5](#)
- [21] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection - a new baseline. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6536–6545, 2018. [1](#), [2](#)
- [22] Raúl Mur-Artal, J. M. M. Montiel, and Juan D. Tardós. Orbslam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015. [2](#)
- [23] 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 on Association for Computational Linguistics*, page 311–318, USA, 2002. Association for Computational Linguistics. [5](#)
- [24] Ehsan Qasemi, Jonathan M. Francis, and Alessandro Oldramari. Traffic-domain video question answering with automatic captioning, 2023. [2](#)
- [25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. [2](#), [3](#)
- [26] Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2019. [4](#)
- [27] Stephen Robertson. Understanding inverse document frequency: On theoretical arguments for idf. *Journal of Documentation - J DOC*, 60:503–520, 2004. [6](#)
- [28] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. [7](#)

- [29] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015. 5
- [30] Jianfeng Wang, Xiaowei Hu, Zhe Gan, Zhengyuan Yang, Xiyang Dai, Zicheng Liu, Yumao Lu, and Lijuan Wang. Ufo: A unified transformer for vision-language representation learning, 2021. 2
- [31] Jianfeng Wang, Xiaowei Hu, Pengchuan Zhang, Xiujun Li, Lijuan Wang, Lei Zhang, Jianfeng Gao, and Zicheng Liu. Minivlm: A smaller and faster vision-language model, 2021.
- [32] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. GIT: A generative image-to-text transformer for vision and language. *Transactions on Machine Learning Research*, 2022. 2
- [33] Junke Wang, Dongdong Chen, Chong Luo, Bo He, Lu Yuan, Zuxuan Wu, and Yu-Gang Jiang. Omnivid: A generative framework for universal video understanding. In *CVPR*, 2024. 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
- [35] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, Sen Xing, Guo Chen, Juntong Pan, Jiashuo Yu, Yali Wang, Limin Wang, and Yu Qiao. Internvideo: General video foundation models via generative and discriminative learning, 2022. 2
- [36] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. SimVLM: Simple visual language model pretraining with weak supervision. In *International Conference on Learning Representations*, 2022. 2
- [37] Ronald J. Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280, 1989. 5
- [38] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649. IEEE, 2017. 2
- [39] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Hugging-face’s transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771, 2019. 4
- [40] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metzger, Luke Zettlemoyer, and Christoph Feichtenhofer. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6787–6800, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. 2
- [41] Kashu Yamazaki, Sang Truong, Khoa Vo, Michael Kidd, Chase Rainwater, Khoa Luu, and Ngan Le. Vlcap: Vision-language with contrastive learning for coherent video paragraph captioning. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 3656–3661, 2022. 2
- [42] Kashu Yamazaki, Khoa Vo, Quang Sang Truong, Bhiksha Raj, and Ngan Le. Vltint: Visual-linguistic transformer-in-transformer for coherent video paragraph captioning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(3):3081–3090, 2023. 2
- [43] Shen Yan, Tao Zhu, Zirui Wang, Yuan Cao, Mi Zhang, Soham Ghosh, Yonghui Wu, and Jiahui Yu. Videococa: Video-text modeling with zero-shot transfer from contrastive captioners, 2023. 2
- [44] Xu Yan, Zhengcong Fei, Shuhui Wang, Qingming Huang, and Qi Tian. Dvcflow: Modeling information flow towards human-like video captioning, 2021. 2
- [45] Antoine Yang, Arsha Nagrani, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vidchapters-7m: Video chapters at scale. In *NeurIPS*, 2023. 2, 6
- [46] 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. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10714–10726, 2023. 2
- [47] Yu Yao, Xizi Wang, Mingze Xu, Zelin Pu, Yuchen Wang, Ella Atkins, and David J. Crandall. Dota: Unsupervised detection of traffic anomaly in driving videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):444–459, 2023. 1, 2
- [48] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2633–2642, 2018. 5
- [49] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *Trans. Mach. Learn. Res.*, 2022, 2022. 2
- [50] Yuan Yuan, Dong Wang, and Qi Wang. Anomaly detection in traffic scenes via spatial-aware motion reconstruction. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1198–1209, 2017. 1, 2
- [51] Yitian Yuan, Tao Mei, and Wenwu Zhu. To find where you talk: Temporal sentence localization in video with attention based location regression. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):9159–9166, 2019. 2
- [52] Yitian Yuan, Lin Ma, Jingwen Wang, Wei Liu, and Wenwu Zhu. Semantic conditioned dynamic modulation for tempo-

- ral sentence grounding in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2725–2741, 2022. 2
- [53] Da Zhang, Xiyang Dai, Xin Wang, Yuan-Fang Wang, and Larry S. Davis. Man: Moment alignment network for natural language moment retrieval via iterative graph adjustment. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1247–1257, 2019. 2
- [54] Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. Span-based localizing network for natural language video localization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6543–6554, Online, 2020. Association for Computational Linguistics. 2
- [55] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5579–5588, 2021. 2
- [56] Long Zhao, Nitesh B. Gundavarapu, Liangzhe Yuan, Hao Zhou, Shen Yan, Jennifer J. Sun, Luke Friedman, Rui Qian, Tobias Weyand, Yue Zhao, Rachel Hornung, Florian Schroff, Ming-Hsuan Yang, David A. Ross, Huisheng Wang, Hartwig Adam, Mikhail Sirotenko, Ting Liu, and Boqing Gong. Videoprism: A foundational visual encoder for video understanding, 2024. 2