Thank you very much for recognizing our detailed understanding of traffic scenarios and modeling the interactive relationships among different aspects of traffic scenes in our work. Below is our detailed response to your comments and questions. If any of our responses fail to adequately address your concerns, please let us know, and we will promptly follow up.

> **[W1]** The current approach focuses on predefined aspects of traffic scenes, overlook other potentially relevant features or dynamics of the traffic environment.

Thank you for giving us the opportunity to clarify this point. The properties of traffic scenes have different taxonomy for different studies, such as static and dynamic, predefined and potential. The choice of taxonomy is closely related to the design of the model. In our model, we classify properties of traffic scenes as two categories: **Static Properties** and **Time-varying Properties**. This classification serves the design of our model.

In our model, the properties of traffic scenes are used to generate a learnable prompt for the CLIP model to understand an image taken by a vehicle. For the images taken at the same road segment but at different time, the **static properties** are invariable. On the contrary, for the **time-varying properties**, when the time of image taking changes, even the image obtained on the same road segment, their time-varying properties are also different. This feature of the **time-varying properties** allows us to generate a different prompt for each different image.

From the perspective of other taxonomy, both static and time-varying properties can be classified as predefined and potential. For example, static potential properties can be static representations learned from the road network using graph representation learning [1-3], while time-varying potential properties can be dynamic time series representations learned from dynamics of the traffic environment and traffic states [4-5]. How to learn the potential static and time-varying properties is out of scope of our paper. Nonetheless, we believe these potential properties can be easily incorporated in the ST-Context representation of our model. A straightforward method is to concatenate them with the predefined **static** and **time-varying properties** in Eq. (7) and Eq. (8). Our model is compatible for most of potential traffic scenes feature representation learning approaches.

> **[W2]** The integration of spatio-temporal data and the multi-aspect prompt learning approach could introduce additional computational complexity and overhead compared to simpler models.

Thanks for this meaningful concern. The introduction to the computational complexity of the model is provided in **Appendix C**, following Algorithm 1. Here, we briefly summarize it.

Since the image and text encoders of ST-CLIP are kept frozen, the primary sources of the model's time complexity lie in two aspects: the **spatio-temporal context representation learning** and **ST-aware multi-aspect prompts learning**. The time complexity of the former one can be approximated as $\mathcal{O}(D^2)$, where $D$ denotes the dimension of the ST-context representation vector. Here, we set the window size of tracklets a small constant value to reduce the complexity. The time complexity of the latter one can be roughly given as $\mathcal{O}((M \times P)^2 D)$, where M represents the length of learnable prompt and P is number of aspects. In conclusion, the overall time complexity of ST-CLIP is $\mathcal{O}(D^2 + (M \times P)^2 D)$.

To validate the efficiency of our model, we also supplemented relevant experiments on both training and inference speed. We conducted experiments on the Beijing dataset with 16-shot setting to test the training and inference time of ST-CLIP and other baseline models using two visual backbone networks: ResNet-50 and ViT-B/32. For specific experimental details, please refer to **Appendix B**. We compared using the official code of baseline models and the results are shown in the following table:

| Model | Training Time (ResNet-50) | Inference Speed (ResNet-50) | Training Time (ViT-B/32) | Inference Speed (ViT-B/32) |
|---|---|---|---|---|
| **CLIP$_{\text{ZS}}$** | \ | 387.8 item/s | \ | 387.8 item/s |
| **CoOp** | 10min16s | 19.2 item/s | 10min40s | 19.0 item/s |
| **CoCoOp** | 12min38s | 14.4 item/s | 13min11s | 14.2 item/s |
| **CLIP-Adapter** | 6min21s | 19.0 item/s | 7min14s | 19.4 item/s |
| **Tip-Adapter** | 6min24s | 375.4 item/s | 6min40s | 375.1 item/s |
| **Tip-Adapter-F** | 21min46s | 373.2 item/s | 22min4s | 372.5 item/s |
| **ST-CLIP** | 11min5s | 35.3 item/s | 11min33s | 34.4 item/s |

- The training time of ST-CLIP is moderate. Compared to the optimal baseline model Tip-Adapter-F, it significantly reduces training time while achieving better experimental results.
- Regarding inference speed, ST-CLIP constructs unique text features for each input image, requiring the use of the text encoder each time. In contrast, for Tip-Adapter-F, the text features for all images are the same and fixed, so they can be pre-computed and reused, resulting in faster inference speed. Compared to other baseline models, ST-CLIP demonstrates faster inference speed due to its ability to simultaneously address all aspects without the need for one-by-one processing.

> **[Q1]** The model's performance may rely on the availability and accuracy of spatio-temporal data, would model consistently outperform other methods?

Thank you for your insightful thoughts. The reliability of spatio-temporal data is a key factor in enhancing the performance of our model. Below we provide more detailed experimental explanations regarding the availability and accuracy of spatio-temporal data.

We have presented experimental results in the **Section 4.4 Ablation Study**, showing the performance of our model without spatio-temporal context, denoted as **NST**, and experiments conducted solely with segment-level context but without trajectory features, denoted as **NT**.

To further demonstrate the impact of the availability and accuracy of spatio-temporal data on experimental results, we introduce two additional experimental settings:

- **NDST**: It removes the dynamic spatio-temporal information, including trajectory features and dynamic properties of road segments (TC and MS), retaining only the static features of the segments.
- **RST**: It replaces the original spatio-temporal features $\mathbf{r}_{e_i}$ in Eq. (15) with a random vector.

Experimental results are compared in the following table:

| Model | Beijing Scene | Beijing Surface | Beijing Width | Beijing Accessibility |
|---|---|---|---|---|
| **RST** | 0.544 | 0.643 | 0.223 | 0.345 |
| **NST** | 0.728 | 0.832 | 0.541 | 0.730 |
| **NDST** | 0.726 | 0.835 | 0.543 | 0.746 |
| **NT** | 0.731 | 0.834 | 0.550 | 0.751 |
| **Tip-Adapter-F** | 0.670 | 0.851 | 0.576 | 0.728 |
| **ST-CLIP** | **0.757** | **0.859** | **0.596** | **0.799** |

The experimental result indicates that the availability of spatio-temporal data has a significant impact on task performance. Compared to **ST-CLIP**, the absence of trajectory information (**NT**), the absence of dynamic features of road segments (**NDST**), and the absence of segment attributes (**NST**) all lead to a decrease in performance. In some aspects, these variants perform less effectively than **Tip-Adapter-F**. Furthermore, it is worth noting that random spatio-temporal features (**RST**) significantly impair the task performance, which further underscores the importance of availability and accuracy of spatio-temporal data and the effectiveness of incorporating spatio-temporal data into prompt learning to assist vision-language models.

> **[Q2]** How does the ST-CLIP model handle scenarios where spatio-temporal data is sparse or noisy?

Thanks for such detailed questions. In response to the sparsity and noise of spatio-temporal data, we provide more comprehensive experimental results.

We design a new variant to verify the impact of sparsity of spatio-temporal data on model performance.

- **SparseST**: It randomly removes points from the original trajectories according to ratio $\beta$, then the trimmed trajectories are used as input for the model.

The experimental results are compared in the following table:

| Model | Beijing Scene | Beijing Surface | Beijing Width | Beijing Accessibility |
|---|---|---|---|---|
| **SparseST** $_{\beta=1.0}$ | 0.731 | 0.834 | 0.550 | 0.751 |
| **SparseST** $_{\beta=0.8}$ | 0.694 | 0.803 | 0.501 | 0.719 |
| **SparseST** $_{\beta=0.6}$ | 0.712 | 0.820 | 0.522 | 0.732 |
| **SparseST** $_{\beta=0.4}$ | 0.743 | 0.834 | 0.563 | 0.769 |
| **SparseST** $_{\beta=0.2}$ | 0.755 | 0.861 | 0.588 | 0.795 |
| **ST-CLIP** | **0.757** | **0.859** | **0.596** | **0.799** |

The experimental result indicates that as the sparsity of spatio-temporal data $\beta$ increases, the task performance gradually declines. When $\beta = 1$, we remove the trajectory information, namely **SparseST** $_{\beta=1.0}$=**NT**, which actually reduces the uncertainty.

**References**


[1] Wu N, Zhao X W, Wang J, et al. Learning effective road network representation with hierarchical graph neural networks[C]//Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining. 2020: 6-14.

[2] Chen Y, Li X, Cong G, et al. Robust road network representation learning: When traffic patterns meet traveling semantics[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2021: 211-220.

[3] Zhang L, Long C. Road Network Representation Learning: A Dual Graph-based Approach[J]. ACM Transactions on Knowledge Discovery from Data, 2023, 17(9): 1-25.

[4] Fu T Y, Lee W C. Trembr: Exploring road networks for trajectory representation learning[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2020, 11(1): 1-25.

[5] Jiang J, Pan D, Ren H, et al. Self-supervised trajectory representation learning with temporal regularities and travel semantics[C]//2023 IEEE 39th international conference on data engineering (ICDE). IEEE, 2023: 843-855.