eTraM: Event-based Traffic Monitoring Dataset Supplementary Material

Aayush Atul Verma*, Bharatesh Chakravarthi*, Arpitsinh Vaghela*, Hua Wei, Yezhou Yang
Arizona State University
{averma90, bshettah, avaghel3, hua.wei, yz.yang}@asu.edu

1. *eTraM* Statistics

This section provides additional statistics about our dataset for a more comprehensive understanding of *eTraM*. *eTraM* consists of 10 hours of data collected from the Prophesee EVK4 HD camera [1]. Beyond the annotated static perception data, *eTraM* includes sequences of ego-motion event-based data, offering increased dataset diversity and experimentation opportunities.

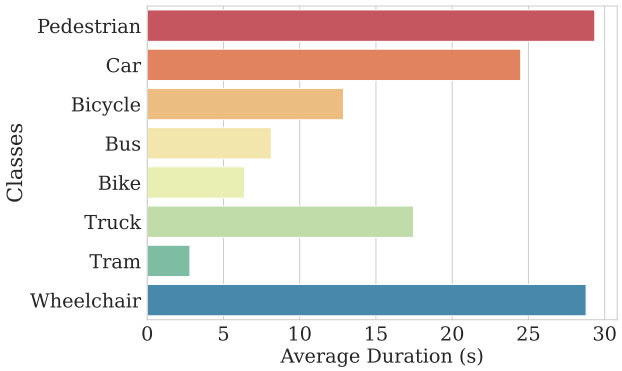

Figure 1. Average Duration Spent by Objects from Each Class: The bar plot illustrates the average duration, in seconds, spent by objects of different classes, providing insights into the temporal characteristics of each class in the dataset.

Figure 1 presents the average duration spent by instances from each class at the traffic sites. This temporal analysis sheds light on the distinctive time dynamics of different classes within the dataset. Participants from the pedestrian and wheelchair classes spend the maximum time at the traffic sites, correlating with their respective movement speeds. In contrast, classes within the vehicle category tend to spend relatively less time in comparison.

Further, we analyze the distribution of different categories (VH, PED, and MM) by the area they cover - small, medium, and large, as shown in Figure 2.

*Equal contribution

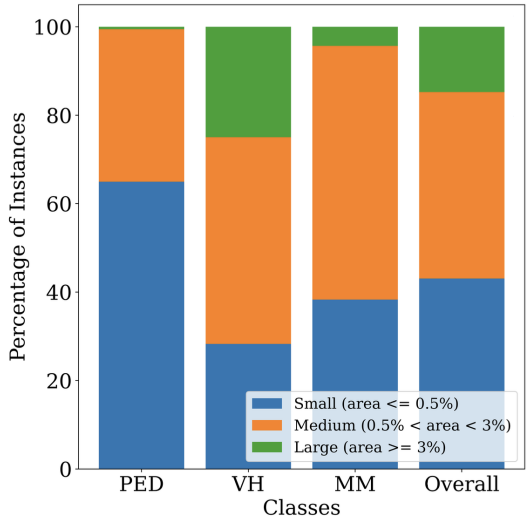

Figure 2. Analysis of the distribution of objects categorized by size (small, medium, and large)

Object Size	RVT				RED			
	PED	VH	MM	All	PED	VH	MM	All
Small	0.308	0.705	**0.276**	0.516	0.324	0.556	**0.274**	0.385
Medium	**0.859**	**0.722**	0.100	**0.722**	**0.661**	**0.763**	0.159	0.561
Large	-	0.637	-	0.637	-	0.701	-	**0.701**

Table 1. Evaluation of object size impact on the performance of RVT and RED.

Based on the size classification, we also establish benchmarks in Table 1. Upon analysis, it becomes evident that both models exhibit similar trends in performance. Specifically, the performance on instances categorized as medium-sized within the pedestrian and vehicle categories is consistently superior to that on small and large-sized instances of their category. Although the performance on vehicles tends to be similar performance across all three size classifications, the performance in the pedestrian category observes a significant drop when evaluated with small-sized instances.

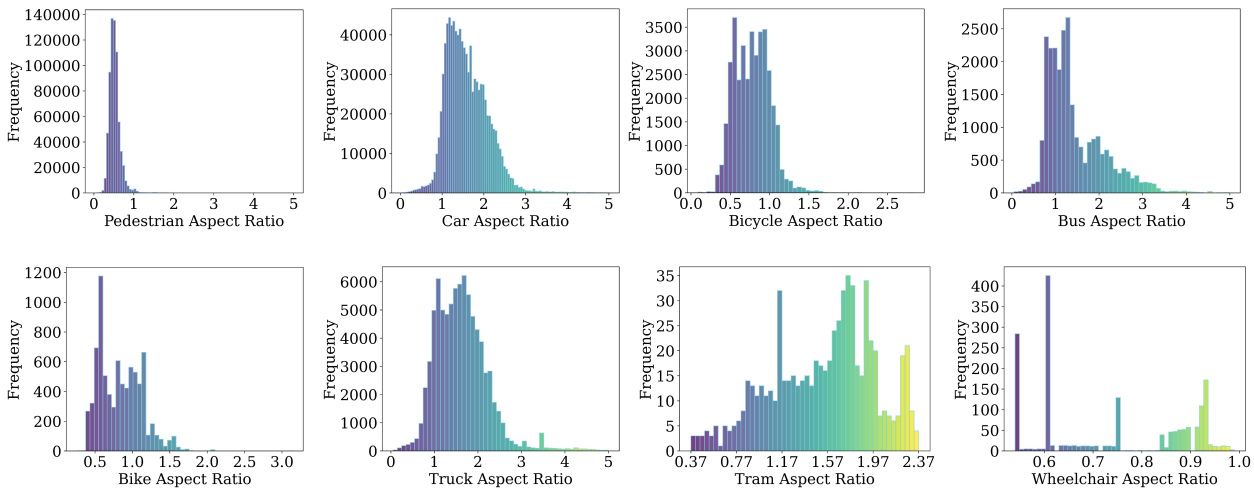

Figure 3. Aspect Ratio Distribution in *eTraM*: A histogram depicting the frequency distribution of aspect ratios across different classes in *eTraM*, providing a comprehensive overview of the dataset's characteristics.

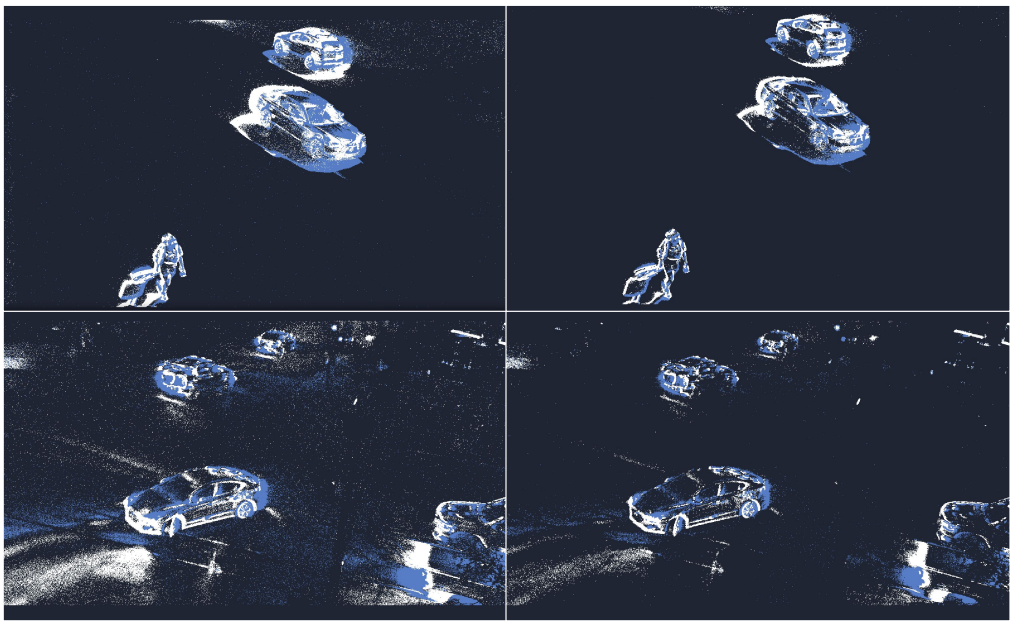

Figure 4. Impact of Spatiotemporal Filtering on Event Camera Data: Comparison of a noisy pre-filtered image (left) and the enhanced clarity achieved post-filtering (right) on daytime (top row) and nighttime data (bottom row).

In contrast, performance on small-sized instances is better than medium-sized for micro-mobility. However, the results of micro-mobility in its best-performing size classification are still worse than the worst-performance of pedestrian and vehicle categories. These results signify a performance degradation when dealing with small-sized objects, particularly micro-mobility. This limitation may stem from the lack of contour and color information in raw event data.

Additionally, Figure 3 shows the frequency of aspect ratios for each class in *eTraM*.

2. Denoising Using Spatiotemporal Filter

To address the noise present in the event stream, particularly heightened during nighttime data with increased levels of reflections and pointed light sources from streets and vehicles, a denoising step is implemented for *eTraM*.

Figure 4 qualitatively illustrates the effectiveness of the spatiotemporal filter [3] by presenting a side-by-side comparison of images before and after applying the filter, showcasing the impact of noise reduction on event data frames.

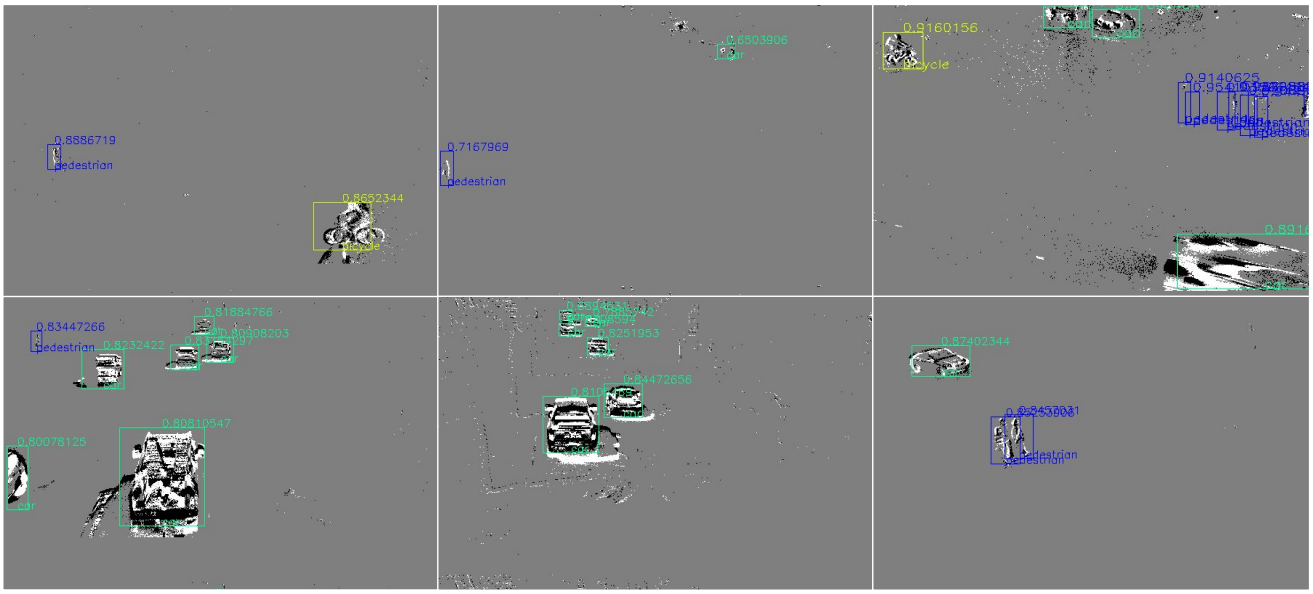

Figure 5. Traffic Participant Object Detection by RVT: Snapshots illustrating the detection results of RVT at various traffic sites, showcasing its performance in diverse real-world scenarios.

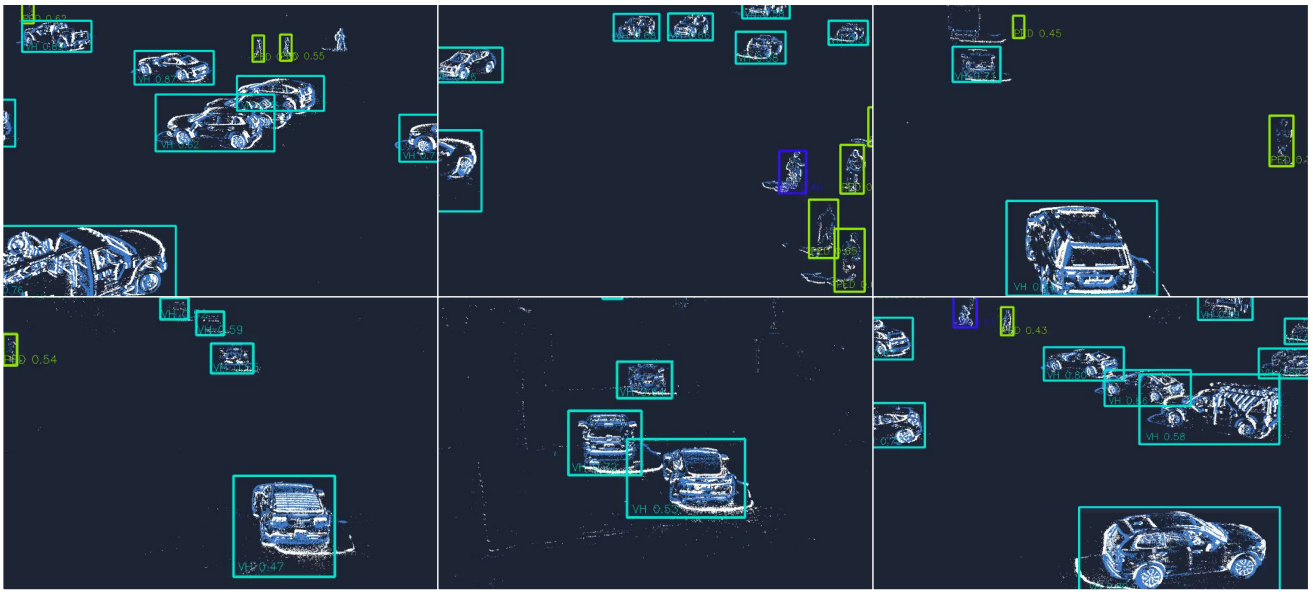

Figure 6. Traffic Participant Object Detection by RED. Snapshots illustrate the detection results of RED at various traffic sites, showcasing its performance in diverse real-world scenarios.

3. Implementation Details

To assess how well event-based models perform on *eTraM*, we trained the state-of-the-art architectures - RVT [4], RED [9], and YOLOv8 [5] on 7 hr of data. We evaluated them on 1.5 hr of validation and 1.5 hr of test data to establish the baselines. Learning rates of 2×10^{-6}, 2×10^{-4}, and 1×10^{-2} are chosen, respectively.

3.1. Input Preprocessing

In this section, we define the input representations used, namely the Histogram of Events [7, 8] and Time Surfaces [6]. The following representations were used to establish baselines and conduct the generalization experiments.

Histogram of Events involves assigning each event to a specific cell based on its position (x, y) and a time bin

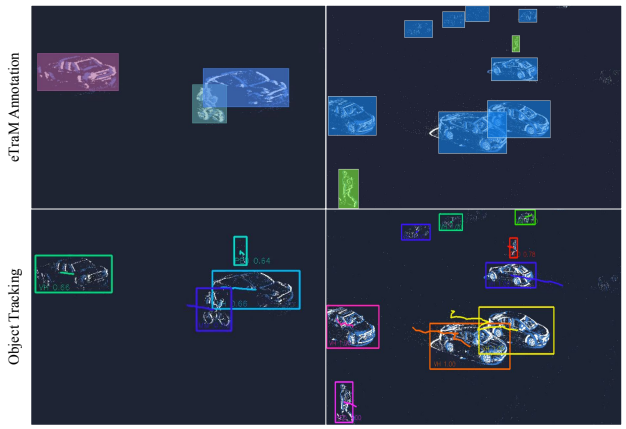

Figure 7. Illustration of Intersection-over-Union based Multi-Object Tracking on the detection results of RVT

determined by its timestamp (t). Subsequently, the total count of events is tallied within each cell and time bin, with separate counts for each polarity recorded in distinct output channels. This process results in a total of two output channels.

Let H represent a four-dimensional tensor with dimensions n, c, h, w, where n represents the index of the timestamp, c represents the channel for the two polarities, h represents the height, and w represents the width of the input event stream. Every new event $\langle x, y, p, t \rangle$ corresponds to a specific histogram decided by the time bin that the timestamp corresponds to. Next, the histogram is updated by adding 1 at the spatial coordinates of the new event. The mathematical representation of the update is as shown in Equation 1. This four-dimensional input representation was used by the tensor-based approaches - RVT and RED.

$$H(\tfrac{t}{\Delta}, p, y, x) = H(\tfrac{t}{\Delta}, p, y, x) + 1 \tag{1}$$

Time Surface, an alternative event processing method, involves recording the timestamp of the most recently received event for each pixel. This technique considers polarities independently, resulting in the output of two channels.

We incorporate an exponential decay to the timestamps to diminish the influence of older events. Assuming $t_0 = 0$ for simplicity, this decay process is implemented. The input representation is represented as a three-dimensional tensor $\langle p, w, h \rangle$, where p represents the polarity, h represents the height, and w represents the width of the input event stream.

For each event $\langle x, y, p, t \rangle$ when $t \leq t_i$, its contribution to the time surface at time t_i can be mathematically represented as shown in Equation 2. The two polarities were considered as the input channels for YOLOv8, and the architecture was updated accordingly.

$$TS_{t_i}(p, y, x) = exp(-\tfrac{t_i - t}{\tau}) \tag{2}$$

4. Detection and Tracking Examples

This section features illustrations of detections by the tensor-based methods - RVT (Figure 5) and RED (Figure 6) across the various traffic scenarios within *eTraM*.

The detection results are been used to perform tracking using an IoU-based thresholding technique [2]. This results in a Multi-Object Tracking Accuracy (MOTA)/Multi-Object Tracking Precision value (MOTP) of 0.18/0.28 on *eTraM*'s test set. It is worth reiterating that the precise evaluation of tracking performance is made possible solely through the inclusion of object IDs within *eTraM*. An example of ground truth objects and their corresponding tracking is illustrated in Figure 7.

References

[1] Event Camera Evaluation Kit 4 HD IMX636 Prophesee-Sony. 1

[2] Erik Bochinski, Volker Eiselein, and Thomas Sikora. High-speed tracking-by-detection without using image information. In *International Workshop on Traffic and Street Surveillance for Safety and Security at IEEE AVSS 2017*, Lecce, Italy, Aug. 2017. 4

[3] Tobias Brosch, Stephan Tschechne, and Heiko Neumann. On event-based optical flow detection. *Frontiers in neuroscience*, 9:137, 2015. 2

[4] Mathias Gehrig and Davide Scaramuzza. Recurrent vision transformers for object detection with event cameras, 2023. 3

[5] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. YOLO by Ultralytics, Jan. 2023. 3

[6] Xavier Lagorce, Garrick Orchard, Francesco Galluppi, Bertram E. Shi, and Ryad B. Benosman. Hots: A hierarchy of event-based time-surfaces for pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(7):1346–1359, 2017. 3

[7] Ana I. Maqueda, Antonio Loquercio, Guillermo Gallego, Narciso Garcia, and Davide Scaramuzza. Event-based vision meets deep learning on steering prediction for self-driving cars. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, June 2018. 3

[8] Diederik Paul Moeys, Federico Corradi, Emmett Kerr, Philip Vance, Gautham Das, Daniel Neil, Dermot Kerr, and Tobi Delbruck. Steering a predator robot using a mixed frame/event-driven convolutional neural network, 2016. 3

[9] Etienne Perot, Pierre de Tournemire, Davide Nitti, Jonathan Masci, and Amos Sironi. Learning to detect objects with a 1 megapixel event camera. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 16639–16652. Curran Associates, Inc., 2020. 3