SUPPLEMENTARY MATERIALS FOR EVENT DENOISING BASED ON ITERATIVE TREE-STRUCTURED INFORMATION AGGREGATION

1. SCHEMATIC DIAGRAM OF THE ITERATIVE INFERENCE PROCESS

The iterative inference process can be divided into two steps which is illustrated in Figure 6. The detailed description is as follows:

(1) *Establish connections*. For the current event $e_{new} = \{x_{new}, y_{new}, t_{new}, p_{new}\}$, identify the pixel set satisfying $x \in [x_{new} - W, x_{new} + W]$ and $y \in [y_{new} - H, y_{new} + H]$ (excluding (x_{new}, y_{new})) and find events with timestamps greater than $t_{new} - T$ as child nodes from the time register. If the number of satisfying pixels exceeds the tree degree K_d, use nearest neighbor pruning (NNP) to select the nearest points. The results of the above process correspond completely to those in Section 3.1.

(2) *Feature aggregation*. After determining the child events of the latest event, the x and y coordinates along with the timestamp of the child events form the 0^{th}-order feature set. Applying the first-order convolution from Section 3.2 to the child nodes' 0^{th}-order features yields the latest event's 1^{st}-order features. Next, reset and write operations are performed: select the register group with the smallest value in the timestamp register at the pixel of the latest event, update it with the latest timestamp t_{new}, set the feature registers from 1st to $D-1$ to 0, and write the new event's first-order features into the reset feature registers. If the event has no child nodes, it is still necessary to reset and write to the time register.

The higher-order convolution of the algorithm is the same as the 1^{st}-order convolution, and the classification module is consistent with the one introduced in Section 3.2.

2. ABLATION STUDY

In the detailed experimental setup, the role of the attention branch in high-order convolution modules is analyzed by removing the branch and retraining the model for evaluation. For the depth of the Relation Tree, configurations of $D = 1$, $D = 2$, and $D = 4$ are explored; for the degree of the Relation Tree, values of $K_d = 8$, $K_d = 16$, and $K_d = 48$ are tested. Additionally, the hyperparameter K_p for PCP pruning and the hyperparameters H/W and T for the Spatiotemporal Window (STW) are included in the testing. To ensure variable independence, other modules remain unchanged across all experiments. The accuracy is evaluated across 16 scenes from the DVSNOISE20 dataset, with the average performance reported.

Table 3. Ablation test of our algorithm

Attention Tests	Removed		
AUC	0.813		
Depth Tests	$D = 1$	$D = 2$	$D = 4$
AUC	0.688	0.792	0.870
Degree Tests	$K_d = 8$	$K_d = 16$	$K_d = 48$
AUC	0.783	0.845	0.869
PCP Tests	$K_p = 1$	$K_p = 3$	$K_p = 4$
AUC	0.844	0.854	0.812
H/W Tests	$H/W = 2$	$H/W = 6$	$H/W = 8$
AUC	0.681	0.859	0.821
T Tests	$T = 0.01$	$T = 0.03$	$T = 0.04$
AUC	0.847	0.858	0.815
Original	$AUC = 0.867$		

From the ablation experiments, we observe that the attention mechanism effectively aggregates useful information, thereby enhancing denoising accuracy. Increasing the depth and degree of the Relation Tree significantly improves denoising performance; however, this improvement becomes marginal when the tree depth reaches $D = 4$ and the degree reaches $K_d = 48$. This suggests that $D = 3$ and $K_d = 32$ are sufficient for capturing spatiotemporal correlations in event streams. Further increasing D to 4 or K_d to 48 results in higher computational costs during the inference phase, reducing inference speed without providing substantial performance gains.

Regarding the hyperparameter K_p for PCP pruning, when K_p increases from 1 to 2 (the original parameter setting), the algorithm's performance improves. However, when K_p continues to increase, the algorithm's performance declines. This is because the algorithm excessively focuses on information from the same pixel while neglecting information from other pixels. Repeated extraction of similar information from the same pixel reduces the algorithm's performance.

For the selection of the spatiotemporal window, the algorithm achieves optimal performance when the hyperparameters are set as $H/W = 4$ and $T = 0.02$. A too small spatiotemporal window leads to insufficient receptive field, which is unfavorable for information extraction, while a too large receptive field extracts information from events that are too distant, both of which hinder the algorithm's information

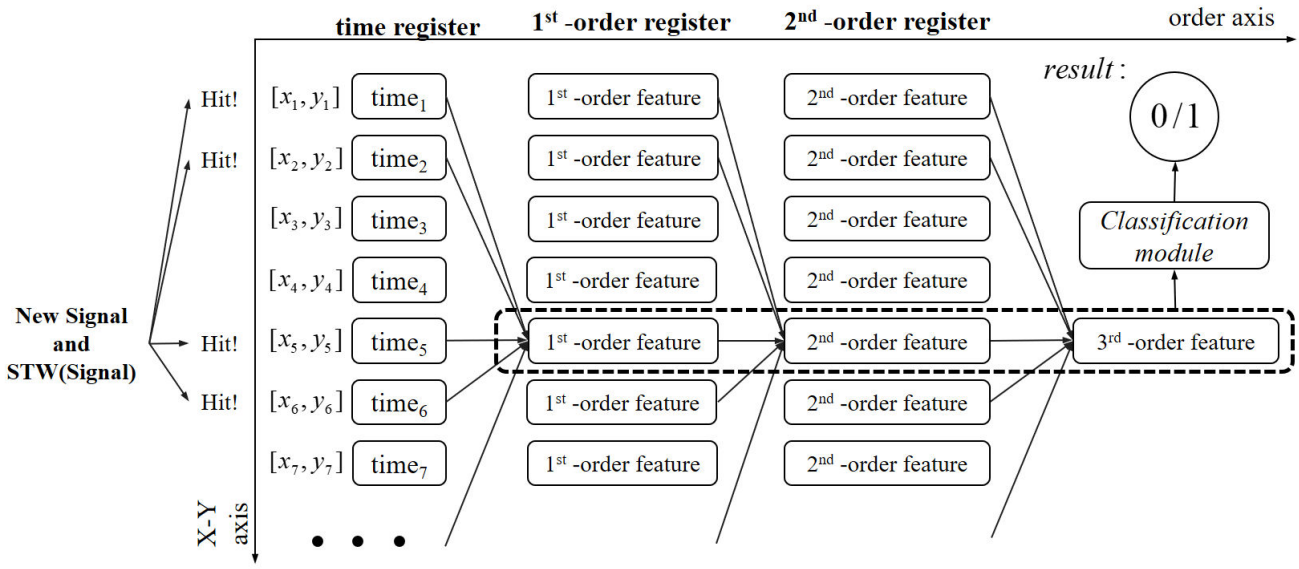

Fig. 6. Illustration of iteration inference process. In the iterative inference process, the sub-node selection is performed first, followed by sequential convolution. For clarity, we set $K_p = 1$, which indicates that only one register group is depicted for each pixel in the figure.

processing.

3. VISUALIZATION

To visually demonstrate the denoising effects, we visualize selected scenes from the DVSNOISE20 dataset. In this process, the event data over a specific time period is converted into frames for display. Positive polarity events are represented in red, while negative polarity events are shown in blue. The results are presented in Figure 7. From the figure, it can be seen that our method effectively removes noise while preserving useful event information compared to other methods.

4. SUPPLEMENTARY ACURRACY COMPARISON ON DVSCLEAN

This experiment is carried out based on the simulated data of the DVSCLEAN[1] dataset. By randomly injecting events into the simulated dataset to simulate actual noise, the model is used to remove these noises, thereby comparing the denoising capabilities of the model under different noise intensities. Specifically, in this experiment, noises with 30%, 50% and 100% of the number of original scene events were introduced into the original DVSCLEAN simulated dataset, and the proposed algorithm was compared with other classic algorithms. The experimental results are shown in Table 4, and the comparison index is the Signal-to-Noise Ratio (SNR). The larger the SNR value, the better the denoising effect. In the scenario of 30% noise injection, this algorithm performed well and obtained the second-best result; while in the scenarios of 50% and 100% noise injection, it achieved the best performance. In addition, for different degrees of noise, the performance of the algorithm did not show a significant decline, indicating that it has strong adaptability to different noise levels.

Table 4. Accuracy comparison on DVSCLEAN

Methods	BAF	MLPF	EdnCNN	TGNN	AEDNet	Ours
30% noise	27.9	**29.3**	27.8	27.7	28.6	29.2
50% noise	23.0	22.7	24.6	25.0	25.8	**26.4**
100% noise	18.9	17.2	18.6	23.2	24.4	**25.2**

5. REFERENCES

[1] Huachen Fang, Jinjian Wu, Leida Li, Junhui Hou, Weisheng Dong, and Guangming Shi, "Aednet: Asynchronous event denoising with spatial-temporal correlation among irregular data," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 1427–1435.

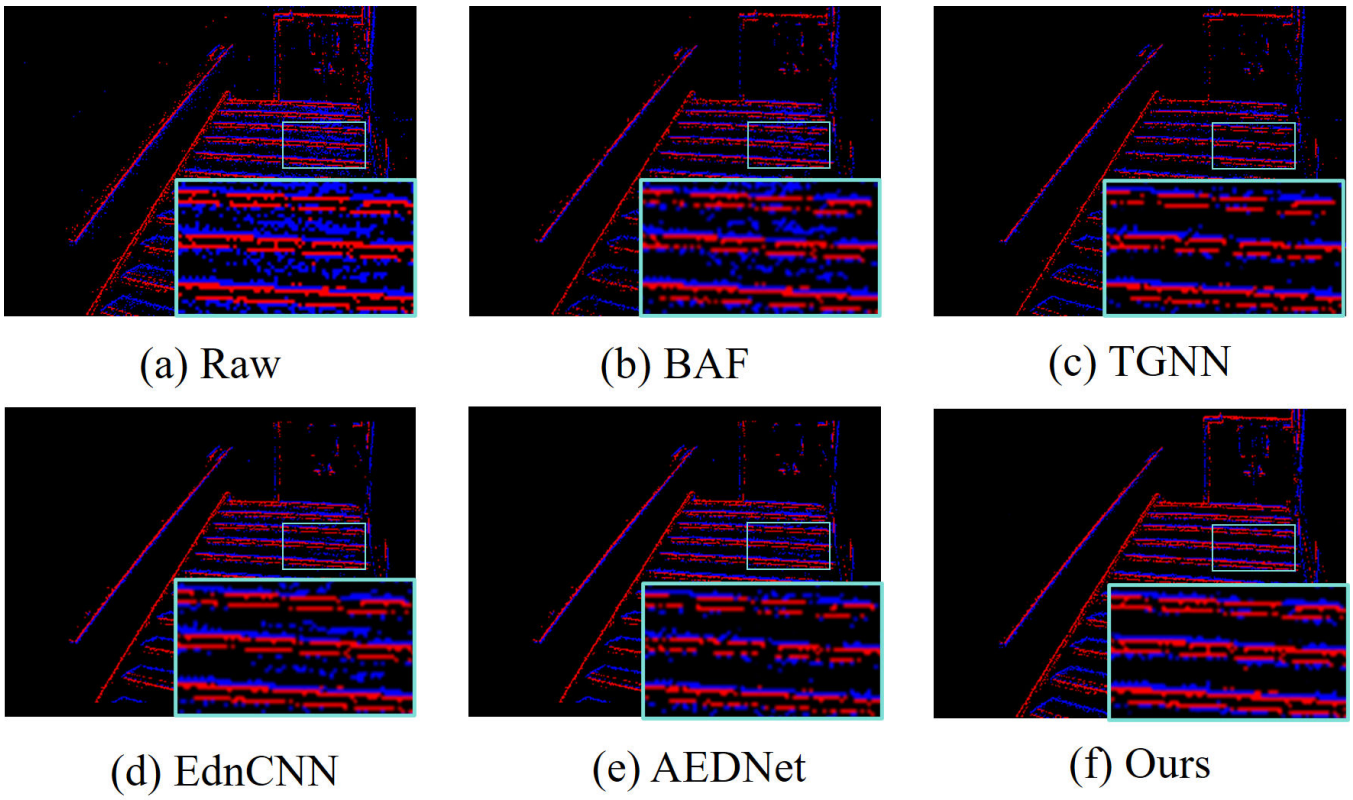

Fig. 7. Visualization of event denoising frames. This figure uses the stairs scene from the DVSNOISE20 dataset to illustrate the denoising effects.