VGER: VOXEL-GUIDED GLOBAL EVENT RANKING FOR EVENT CLOUD ATTRIBUTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Event cameras produce sparse and asynchronous event streams that provide rich spatio-temporal information for efficient perception. Recent advances in event-based models have demonstrated strong performance by directly modeling asynchronous events without dense frame reconstruction. However, identifying the event-level evidence behind their predictions is crucial for improving model transparency and reliability. Directly adapting point-level saliency methods from point clouds provides fine-grained attribution but overlooks event-specific spatio-temporal structures. To address this limitation, we propose Voxel-Guided Global Event Ranking (VGER), a training-free attribution framework for point-based event cloud networks. VGER combines event-level gradient evidence with task-aware voxel perturbation evidence, transferring regional contribution into event-level attribution scores while preserving fine-grained resolution. Furthermore, VGER introduces a unified event ranking strategy, where high-ranked events are expected to be prediction-critical and low-ranked events are expected to have limited influence on predictions. We evaluate VGER on three event-based benchmarks with PointNet, PointNet++, and EventMamba. Across nine dataset-backbone settings, VGER consistently improves both high-tail and low-tail deletion performance over point-level saliency baselines.

1 INTRODUCTION

Event cameras capture sparse and asynchronous event streams, providing high temporal resolution, low latency, and high dynamic range perception (Gallego et al., 2022). Unlike conventional cameras that rely on dense frame representations, event-based methods directly exploit asynchronous changes to preserve fine-grained temporal dynamics while reducing redundant computation (Gehrig et al., 2019; Sironi et al., 2018). Recently, point-based event cloud networks have emerged as an effective paradigm for event-based recognition by representing raw events as spatio-temporal point sets and processing them with point-based architectures (Qi et al., 2017a;b; Ren et al., 2025). These methods have demonstrated promising performance across various event-based benchmarks (Amir et al., 2017; Orchard et al., 2015; Li et al., 2017), benefiting from their ability to preserve the intrinsic sparsity and temporal characteristics of event streams. However, as event cloud networks become increasingly powerful, their predictions are still largely determined by complex interactions among massive asynchronous events, making it difficult to understand which events provide critical evidence for model decisions (Ribeiro et al., 2016; Lundberg & Lee, 2017).

Figure 1 illustrates the event-level saliency maps produced by VGER on representative streams from the three evaluation datasets.

Attribution methods for point-based models provide a natural starting point for interpreting event cloud networks (Zheng et al., 2019). By assigning importance scores to individual points, point-level saliency approaches can identify prediction-related inputs with fine-grained spatial resolution (Simonyan et al., 2013). However, directly transferring these methods from conventional point clouds to event streams overlooks the unique characteristics of events. Unlike static 3D points, events are asynchronously generated, and their semantic information often emerges from local spatio-temporal structures formed by correlated event groups (Gallego et al., 2022). For example, motion patterns and object boundaries are represented by groups of temporally correlated events rather than isolated observations. Therefore, event attribution requires not only identifying individual important

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

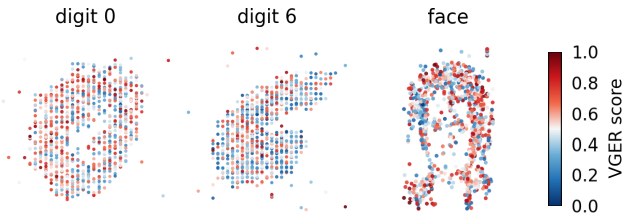

Figure 1: Event-level saliency maps produced by VGER on representative streams from the three evaluation datasets. Colors encode normalized saliency (red: high, blue: low). High-ranked events follow compact spatio-temporal structures that contribute to the corresponding predictions.

events but also capturing regional structural evidence that contributes collectively to model predictions (Fong & Vedaldi, 2017; Petsiuk et al., 2018). This motivates a new attribution paradigm that bridges event-level precision and spatio-temporal structural reasoning.

A straightforward solution is to introduce regional perturbation analysis, where groups of events are removed or modified to measure their influence on model predictions (Fong & Vedaldi, 2017; Petsiuk et al., 2018). Voxel-based perturbation methods provide task-aware regional evidence by evaluating prediction changes caused by perturbing local spatio-temporal regions (Deng et al., 2022). However, assigning a single importance score to all events within the same voxel inevitably sacrifices event-level discrimination. Conversely, point-level saliency preserves individual event resolution but lacks awareness of collective regional contributions. These observations reveal a fundamental granularity dilemma in event attribution: fine-grained event importance and regional structural evidence are complementary but difficult to obtain simultaneously.

To address this challenge, we propose Voxel-Guided Global Event Ranking (VGER), which combines voxel-level task evidence with event-level attribution signals to generate a more faithful global ranking of asynchronous events. VGER is a training-free attribution framework designed for point-based event cloud networks. It first derives task-aware regional evidence through voxel perturbation and transfers this evidence back to individual events. Meanwhile, event-level gradient attribution provides fine-grained sensitivity information (Simonyan et al., 2013; Zheng et al., 2019). By integrating these two complementary signals, VGER produces a unified event ranking that preserves both regional structural awareness and individual event resolution. Beyond conventional attribution evaluation that only examines whether important events affect predictions, we introduce a two-tail ranking perspective: highly ranked events should be prediction-critical, while low-ranked events should have limited influence (Petsiuk et al., 2018; Hooker et al., 2019). We evaluate VGER on three event-based benchmarks with multiple point-based architectures, including PointNet, PointNet++, and EventMamba (Qi et al., 2017a;b; Ren et al., 2025; 2024a). Extensive experiments demonstrate that VGER consistently improves attribution faithfulness over point-level saliency baselines across diverse dataset-backbone settings. The main contributions of this work are summarized as follows:

- We introduce VGER, the first attribution framework for point-based event cloud networks, enabling fine-grained interpretation of asynchronous event streams.
- We propose a voxel-guided evidence fusion strategy that combines task-aware regional perturbation evidence with event-level gradient attribution to preserve both structural context and event-level resolution.
- We develop a global event ranking evaluation strategy with complementary high-tail and low-tail assessments, providing a more comprehensive evaluation of event attribution faithfulness.

2 Related Work

Event representations and recognition. Event representations range from dense time surfaces, learned grids, and recurrent surfaces to sparse recursive, graph-based, and voxel-based forms (Sironi et al., 2018; Gehrig et al., 2019; Cannici et al., 2020; Sekikawa et al., 2019; Schaefer et al., 2022; Deng et al., 2022; Ren et al., 2026a). EvT exploits sparsity by retaining activated event-derived patches, while GET groups events by timestamp and polarity (Sabater et al., 2022; Peng et al., 2023). E2PNet, PEPNet, EventMamba, and SECNet develop task-specific spatial and temporal processing

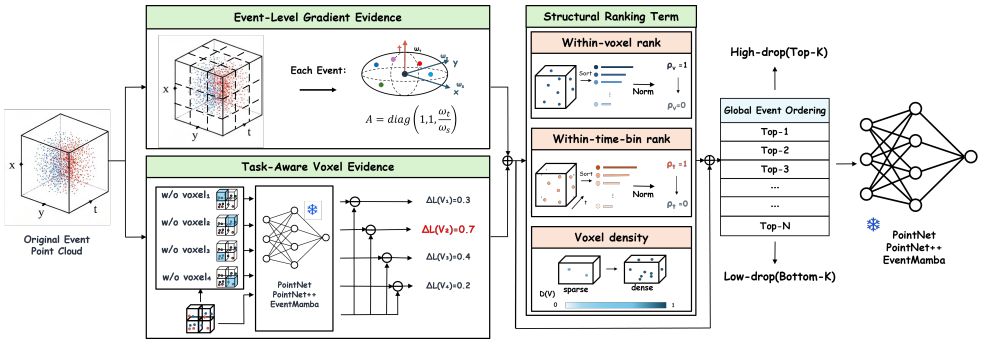

Figure 2: Overview of VGER. Event-level gradient and voxel perturbation evidence are fused into a global event ranking evaluated at both deletion tails.

for event data (Lin et al., 2023; Ren et al., 2024b; 2025; 2026b). General point-based architectures such as PointNet, PointNet++, DGCNN, and Point Transformer provide operations that can be applied to spatio-temporal event points (Qi et al., 2017a;b; Wang et al., 2019; Zhao et al., 2021). These developments expand how event structure is processed across event-based tasks; VGER complements them by identifying the event-level evidence behind a trained model's prediction.

Fine-grained and regional attribution. Gradient Saliency, Integrated Gradients, LRP, Grad-CAM, and SmoothGrad derive attribution from input derivatives or internal activations (Simonyan et al., 2013; Sundararajan et al., 2017; Bach et al., 2015; Selvaraju et al., 2017; Chattopadhay et al., 2018; Smilkov et al., 2017). LIME fits a local interpretable surrogate, while Kernel SHAP uses a weighted linear explanation model (Ribeiro et al., 2016; Lundberg & Lee, 2017). Perturbation methods measure prediction changes under controlled input interventions (Zeiler & Fergus, 2014; Fong & Vedaldi, 2017; Petsiuk et al., 2018). For point clouds, PCS uses a differentiable center-shifting surrogate to score individual points (Zheng et al., 2019); later work explores features, local surrogates, semantic groups, interpretable architectures, and factorized attribution maps (Levi & Gilboa, 2024; Ahmadi et al., 2024; Mulawade et al., 2025; Feng et al., 2024; Liu et al., 2024). EventPoint detects event keypoints for matching and registration (Huang et al., 2023), and gradient attribution has also been studied for spiking-network inputs (Bitar et al., 2023). The attribution methods in this group span fine-grained input evidence, group-level perturbation, feature-based explanations, and inherently interpretable architectures. In contrast, VGER transfers regional task evidence to individual events before globally ranking them, directly connecting the two granularities identified in the Introduction.

Evaluation of explanations. Deletion and insertion curves evaluate an attribution ordering by tracking model behavior as inputs are removed or restored (Petsiuk et al., 2018). Their interpretation depends on a controlled perturbation protocol (Hooker et al., 2019; Tomsett et al., 2020; Gomez et al., 2022), while sanity checks, input-invariance and robustness tests, and infidelity/sensitivity metrics assess complementary properties of explanations (Adebayo et al., 2018; Kindermans et al., 2019; Alvarez-Melis & Jaakkola, 2018; Yeh et al., 2019). We hold deletion cardinality and backbone-specific operators fixed across attribution methods and evaluate both ends of the same global ordering. This two-tail design tests whether high-ranked events are prediction-critical and low-ranked events have limited influence under one consistent protocol.

3 Problem Formulation

We consider the attribution problem for point-based event cloud networks. An event stream is represented as a set of asynchronous events:

$$E = \{e_i\}_{i=1}^{N}, \qquad e_i = (x_i, y_i, t_i, p_i), \tag{1}$$

where (x_i, y_i, t_i) denotes the spatio-temporal coordinate and p_i denotes the event polarity. A frozen event cloud model f_θ takes E as input and produces prediction logits, with $\mathcal{L}(f_\theta(E), y)$ denoting the task loss. Given an event stream E and its label y, the objective is to assign an importance score $S(e_i)$ to each event while keeping θ unchanged. Unlike conventional point attribution, event

attribution should capture not only individual event importance but also the contribution of local spatio-temporal structures.

For a deletion ratio r and $K = \lfloor rN \rfloor$, the top- and bottom-ranked event subsets are defined as:

$$\begin{aligned} D_{\text{high}}(r) &= \{e_i \mid S(e_i) \in \text{TopK}(S, K)\}, \\ D_{\text{low}}(r) &= \{e_i \mid S(e_i) \in \text{BottomK}(S, K)\}, \end{aligned} \tag{2}$$

where TopK and BottomK denote the sets of events with the largest and smallest attribution scores, respectively. Given a unified event ranking, Eq. (2) provides two complementary evaluation perspectives. Removing D_{high} is expected to cause larger prediction degradation, while removing D_{low} should have limited impact on recognition. The deletion operator $\mathcal{D}$ removes the selected events with a fixed cardinality and follows the input representation requirements of each backbone. For fair comparison, the same deletion procedure is applied to all attribution methods under each backbone.

4 VOXEL-GUIDED GLOBAL EVENT RANKING

VGER addresses the attribution challenge through four stages. Figure 2 summarizes the framework, which combines event-level gradient evidence, voxel perturbation evidence, and structural ranking to produce a global event ordering. It first derives fine-grained sensitivity from event-level gradients and then measures the task contribution of local spatio-temporal voxels. The voxel evidence is transferred back to individual events and combined with a structural term that organizes their relative ranks. Finally, the fused event scores define one global ordering for both high-tail and low-tail evaluation.

4.1 EVENT-LEVEL GRADIENT EVIDENCE

Event-level gradients provide the fine-grained component of VGER. Because spatial coordinates and timestamps have different units and ranges, we normalize each $z_i = (x_i, y_i, t_i)^\top$ using a robust stream center $c = (c_x, c_y, c_t)^\top$ and a positive coordinate-wise scale $s = (s_x, s_y, s_t)^\top$:

$$\bar{z}_i = \left(\frac{x_i - c_x}{s_x}, \frac{y_i - c_y}{s_y}, \frac{t_i - c_t}{s_t} \right)^\top. \tag{3}$$

This normalization prevents the temporal coordinate from dominating the score solely through its numerical range.

We adapt PCS to the spatio-temporal event domain by measuring task-loss sensitivity along a differentiable center-contraction path. To distinguish temporal displacement from spatial displacement, we introduce the anisotropic metric

$$A_\rho = \text{diag}(1, 1, \rho), \qquad \rho = \frac{\omega_t}{\omega_s}, \tag{4}$$

where $\rho > 0$ controls the weight of temporal displacement relative to each spatial coordinate. VGER uses the anisotropic default $\rho = 0.5$, which assigns the temporal coordinate half the weight of each spatial coordinate in A_ρ. The equal-weight setting $\rho = 1$ is the isotropic special case, rather than the default. The corresponding anisotropic radius is

$$r_i^\rho = \sqrt{\bar{z}_i^\top A_\rho \bar{z}_i}. \tag{5}$$

We define the anisotropic center-contraction path as

$$\bar{z}_i(\tau) = \exp(-\tau A_\rho)\bar{z}_i, \qquad \tau \geq 0. \tag{6}$$

Its instantaneous displacement direction is

$$\left. \frac{\partial \bar{z}_i(\tau)}{\partial \tau} \right|_{\tau=0} = -A_\rho \bar{z}_i. \tag{7}$$

Therefore, the anisotropic event-level PCS score is defined as

$$S_{\text{pcs}}^\rho(e_i) = -\left\langle \nabla_{\bar{z}_i} \mathcal{L}, \frac{A_\rho \bar{z}_i}{r_i^\rho + \epsilon} \right\rangle (r_i^\rho)^{1+\alpha}, \tag{8}$$

where $\mathcal{L}$ is the task loss, ϵ is a small constant, and α controls the radial scaling. Eq. (8) reduces to the original isotropic PCS formulation when $\rho = 1$.

The score in Eq. (8) measures the first-order increase in task loss when an event is suppressed along the anisotropic spatio-temporal contraction direction. Adjusting ρ balances temporal and spatial displacement while preserving a separate score for every event.

4.2 Task-Aware Voxel Evidence

Event-level gradients distinguish individual events but do not explicitly measure the finite task effect of a complete local spatio-temporal region. To obtain this complementary regional evidence, we partition the normalized event domain into $G_s \times G_s \times G_t$ voxels. Equivalently, voxelization can be performed in the anisotropically transformed coordinates

$$u_i = A_\rho^{1/2} \bar{z}_i, \tag{9}$$

so that the spatial and temporal extent of each voxel follows the metric in Eq. (4).

Let V_j denote an occupied voxel. We construct $\widetilde{E}^{(j)}$ by moving the events in V_j toward the robust stream center while preserving the number, polarity, and temporal order of events. The task-aware evidence of V_j is

$$\Delta\mathcal{L}(V_j) = \mathcal{L}(f_\theta(\widetilde{E}^{(j)}), y) - \mathcal{L}(f_\theta(E), y), \tag{10}$$

where f_θ is the frozen task network and y is the ground-truth label. This voxel-level loss change is broadcast to each constituent event:

$$S_{\text{vox}}(e_i) = \Delta\mathcal{L}(V(e_i)). \tag{11}$$

Eq. (11) performs the transfer described in the Introduction. Rather than treating a voxel as the final attribution unit, VGER assigns its measured task contribution to each constituent event. The subsequent fusion therefore retains event-level resolution while incorporating evidence about the local structure surrounding each event.

4.3 Structural Ranking Term

The event-level gradient and voxel perturbation terms address the two sides of the granularity dilemma. The former distinguishes individual events within a local region, whereas the latter captures the finite task effect of the region as a whole. We first combine these complementary signals into a base event score:

$$S_{\text{base}}(e_i) = \beta_p \,\text{Norm}\big(S_{\text{pcs}}^{\rho}(e_i)\big) + \beta_v \,\text{Norm}(S_{\text{vox}}(e_i))\,, \tag{12}$$

where $\text{Norm}(\cdot)$ denotes stream-wise score normalization, and β_p and β_v control the contributions of event-level gradient evidence and voxel-level task evidence, respectively.

We then organize the base scores according to their local and temporal contexts. Let

$$\begin{aligned} \mathcal{E}_V(e_i) &= \{e_k \in E : V(e_k) = V(e_i)\}\,, \\ \mathcal{E}_T(e_i) &= \{e_k \in E : T(e_k) = T(e_i)\}\,, \end{aligned} \tag{13}$$

where $V(e_i)$ denotes the voxel containing e_i, and $T(e_i)$ denotes its temporal-bin index. Thus, $\mathcal{E}_V(e_i)$ contains the events in the same spatio-temporal voxel as e_i, while $\mathcal{E}_T(e_i)$ contains the events assigned to the same temporal bin.

For an event subset $\mathcal{Q}$ containing e_i, let $r_{\mathcal{Q}}(e_i)$ be the rank of $S_{\text{base}}(e_i)$ among the events in $\mathcal{Q}$, with larger base scores receiving larger ranks. We define the normalized within-set rank as

$$\rho_{\mathcal{Q}}(e_i) = \begin{cases} \dfrac{r_{\mathcal{Q}}(e_i) - 1}{|\mathcal{Q}| - 1}, & |\mathcal{Q}| > 1, \\ \dfrac{1}{2}, & |\mathcal{Q}| = 1. \end{cases} \tag{14}$$

The singleton case is assigned the neutral value $1/2$. The voxel-wise and temporal ranks are consequently given by

$$\rho_V(e_i) = \rho_{\mathcal{E}_V(e_i)}(e_i), \qquad \rho_T(e_i) = \rho_{\mathcal{E}_T(e_i)}(e_i). \tag{15}$$

Both quantities lie in $[0, 1]$ and preserve the relative position of an event within its corresponding local context.

We additionally characterize the occupancy of each voxel. Let $\mathcal{V}_{\text{occ}}$ denote the set of occupied voxels and let $n(V_j)$ be the number of events assigned to voxel V_j. We define

$$\begin{aligned} n(V_j) &= \sum_{k=1}^{N} \mathbb{I}[V(e_k) = V_j], \\ D(V_j) &= \frac{n(V_j)}{\max\limits_{V_\ell \in \mathcal{V}_{\text{occ}}} n(V_\ell)}. \end{aligned} \tag{16}$$

Accordingly, $D(V_j) \in (0, 1]$ for every occupied voxel. A larger value indicates a region containing more events, whereas $1 - D(V_j)$ represents its relative sparsity within the current event stream.

Using these contextual quantities, we define the structural ranking score as

$$\begin{aligned} S_{\text{struct}}(e_i) = w_b S_{\text{base}}(e_i) + w_c \rho_V(e_i) + w_t \rho_T(e_i) \\ + w_r\big(1 - D(V(e_i))\big) - w_d D(V(e_i)). \end{aligned} \tag{17}$$

Here, w_b retains the underlying attribution evidence, w_c and w_t control the voxel-wise and temporal rank contributions, and w_r and w_d calibrate the relative sparsity and occupancy of the event's voxel. The two occupancy components are written separately to expose their structural interpretations, while their combined ranking effect is governed by $w_r + w_d$.

The structural term therefore refines the global event ordering without replacing the gradient and voxel attribution signals. The within-voxel rank preserves fine-grained comparisons among events receiving the same regional evidence, the temporal rank incorporates their relative temporal context, and the occupancy terms account for how events are distributed across spatio-temporal regions.

4.4 Voxel-Guided Global Event Ranking

The final VGER score fuses regional, event-level, and structural evidence into one scalar for every event:

$$\begin{aligned} S_{\text{VGER}}(e_i) = \lambda_v \operatorname{Norm}(S_{\text{vox}}(e_i)) + \lambda_p \operatorname{Norm}(S^{\rho}_{\text{pcs}}(e_i)) \\ + \lambda_r \operatorname{Norm}(S_{\text{struct}}(e_i)). \end{aligned} \tag{18}$$

Here, $\operatorname{Norm}(\cdot)$ denotes stream-wise score normalization, and λ_v, λ_p, and λ_r control the contributions of voxel-level, gradient-level, and structural evidence, respectively. VGER then performs a stable global ranking over the complete stream:

$$\pi = \operatorname{argsort}_{e_i \in E} S_{\text{VGER}}(e_i). \tag{19}$$

5 Experiments

5.1 Datasets, Backbones, and Baselines

We evaluate DVS Gesture (Amir et al., 2017), N-MNIST, and N-Caltech101 (Orchard et al., 2015), the standard event-classification benchmarks alongside CIFAR10-DVS (Li et al., 2017). The backbones are PointNet (Qi et al., 2017a), PointNet++ (Qi et al., 2017b), and EventMamba (Ren et al., 2025). The verified baselines shared by the archived 3×3 matrix are Random and PCS-XYT. A separate PointNet study compares Basic Voxel with PCS-XYT on all three datasets.

Random uses the archived event permutations and provides a task-agnostic deletion reference. PCS-XYT applies Eq. (8) when $\rho = 1$ without voxel evidence. Basic Voxel assigns the same task-loss change to all events in one voxel; its ties are resolved at event level while respecting the exact deletion budget. We restrict the quantitative comparison to these baselines because they are the methods represented by verified, common-format curves in the supplied archive.

The PointNet, PointNet++ and EventMamba results use the full archived test counts: 6,623 for DVS Gesture, 10,000 for N-MNIST, and 17,869 for N-Caltech101.

Table 1: VGER improves both macro AUC tails over PCS-XYT. Values are unweighted averages over nine archived settings.

Method	High AUC ↓	Low AUC ↑
Random	0.4977	0.4977
PCS-XYT	0.4368	0.4934
VGER	**0.3836**	**0.5404**

Table 2: VGER wins both AUC tails in all nine settings. H is high-tail deletion and L is low-tail deletion; bold indicates the better AUC within each pair.

Backbone	Dataset	n	VGER H ↓	PCS H ↓	VGER L ↑	PCS L ↑
PointNet	DVS Gesture	6,623	**0.5532**	0.6351	**0.8121**	0.7781
PointNet	N-MNIST	10,000	**0.5959**	0.6422	**0.7339**	0.6971
PointNet	N-Caltech101	17,869	**0.0576**	0.0756	**0.2665**	0.2180
PointNet++	DVS Gesture	6,623	**0.4665**	0.5311	**0.6035**	0.5315
PointNet++	N-MNIST	10,000	**0.2655**	0.2740	**0.3811**	0.3489
PointNet++	N-Caltech101	17,869	**0.0799**	0.0817	**0.0981**	0.0918
EventMamba	DVS Gesture	6,623	**0.4429**	0.5765	**0.7369**	0.6290
EventMamba	N-MNIST	10,000	**0.7729**	0.8616	**0.8935**	0.8491
EventMamba	N-Caltech101	17,869	**0.2180**	0.2533	**0.3381**	0.2972

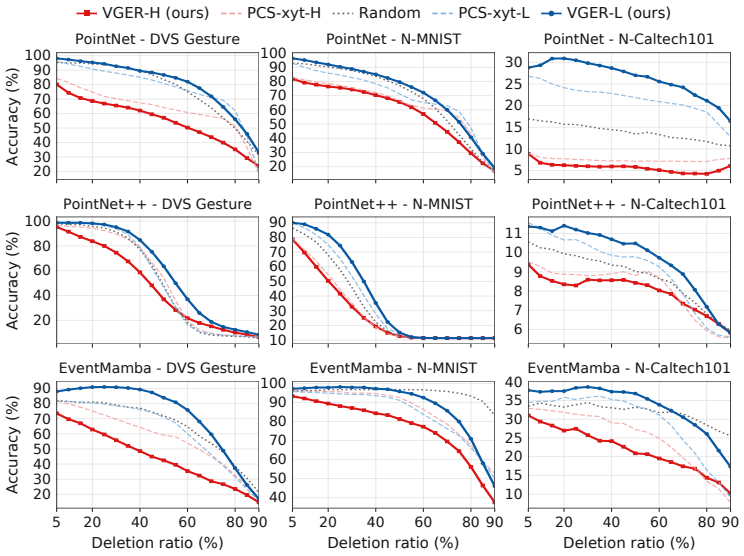

Figure 3: Two-tail deletion curves for the 3×3 matrix of backbones and datasets. Each panel deletes 5%–90% of events in score order and reports accuracy. VGER's high tail (VGER-H) drops below PCS-XYT-H and Random in most settings and ratios, and its low tail (VGER-L) stays above PCS-XYT-L in most settings and ratios, so the two VGER tails separate decision-critical from decision-irrelevant events across the whole protocol.

5.2 Two-Tail Protocol

All settings are compared over

$$\mathcal{R} = \{0, 0.05, \ldots, 1.00\}.$$

At each ratio, exactly $\lfloor rN \rfloor$ events are deleted. Accuracy curves are summarized by trapezoidal AUC; lower is better for the high tail and higher is better for the low tail:

$$\mathrm{AUC}(A) = \sum_{k=1}^{20} \frac{A(r_k) + A(r_{k+1})}{2}(r_{k+1} - r_k).$$

We report exact ratio-wise wins for every setting. The two summaries answer different questions: AUC weights the magnitude of the curve difference at each ratio, whereas a win count gives every sampled ratio one vote, so a method cannot dominate either summary through a single strong region of the protocol.

5.3 Main Two-Tail Fidelity

Table 1 reports the macro comparison across the archived matrix.The macro comparison shows that VGER reduces high-tail AUC by 0.0532 and increases low-tail AUC by 0.0470 relative to

PCS-XYT. Its macro two-tail gap, low AUC minus high AUC, is 0.1568 compared with 0.0566 for PCS-XYT. These averages weight each dataset–backbone setting equally. The rows behind them are uniformly directional and are all produced by frozen archived configurations, so the macro gains are not carried by any single backbone or dataset.Table 2 gives the corresponding setting-level AUCs.

Figure 3 presents the main experiment of this paper: the complete two-tail deletion curves behind these counts. In most settings and deletion ratios, removing VGER's top-ranked events damages accuracy earlier and deeper than removing PCS-XYT's or Random's, whereas removing its bottom-ranked events preserves accuracy longer than PCS-XYT's. The appendix further averages the two-tail curves over all nine settings and condenses each setting into its two-tail separation, which VGER widens over PCS-XYT in all nine settings, showing that one global ordering simultaneously isolates decision-critical and decision-irrelevant events.

Additional implementation details and supporting evidence are provided in the appendix, including the setting-specific fusion weights used in part of the experiments, the full numerical data underlying Figure 3, and the auxiliary results from hyperparameter search and related diagnostic studies. These materials are included to make the reported two-tail fidelity results traceable beyond the summarized AUC values and to facilitate independent verification of the experimental protocol.

5.4 Results on PointNet

PointNet uses single max-pooling, making predictions depend on a compact critical event subset. High-tail deletion therefore tests whether VGER's top-ranked events are truly decisive. VGER lowers high-tail AUC from 0.6351 to 0.5532 on DVS Gesture and from 0.0756 to 0.0576 on N-Caltech101. Low-tail AUC also improves over PCS-XYT on all datasets, with gains of 0.0340, 0.0368, and 0.0485. This consistent two-tail behavior shows reliable importance estimates for globally pooled representations.

5.5 Results on PointNet++

PointNet++ learns hierarchical features from nested spatial neighborhoods, so attribution depends on both local geometry and global structure. Although PCS-XYT already has low high-tail AUCs, VGER improves both tails on every dataset. On DVS Gesture, high-tail AUC drops from 0.5311 to 0.4665, while low-tail AUC rises from 0.5315 to 0.6035. Figure 3 further shows that top-ranked deletion disrupts recognition faster, whereas bottom-ranked deletion preserves accuracy longer. The smaller gains on N-MNIST and N-Caltech101 remain consistent.

5.6 Results on EventMamba

EventMamba models long-range temporal dependencies through selective state updates, making event-level attribution less direct than in point-wise networks. Still, VGER improves both tails in all three archived runs. The largest gain appears on DVS Gesture, where high-tail AUC decreases from 0.5765 to 0.4429 and low-tail AUC increases from 0.6290 to 0.7369. N-MNIST and N-Caltech101 show the same pattern: removing highly ranked events causes sharper degradation, while removing low-ranked events better preserves accuracy.

5.7 Ablation Study

We ablate the key evidence channels on the deterministic PointNet++/N-MNIST development subset using the same 5%–90% deletion protocol. Figure 4 compares four representative rankings: PCS-XYT, PCS-XYT with voxel evidence, the full voxel-structural fusion, and Random. The top panel reports high-tail deletion, where lower accuracy indicates that the removed events are more decision-critical. The bottom panel reports low-tail deletion, where higher accuracy indicates that the removed events are less important.

The high-tail results show that voxel evidence is the main source of decision-critical localization. Compared with PCS-XYT, adding the voxel channel causes accuracy to drop earlier across most deletion ratios. The full fusion, which further incorporates structural ranking, remains among the strongest high-tail curves and stays well below Random, indicating that it preserves the ability to

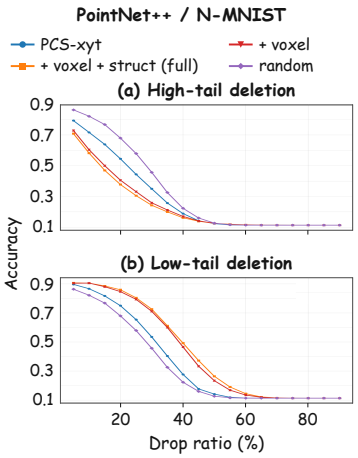

Figure 4: Component ablation on the PointNet++/N-MNIST development subset. Top: high-tail deletion AUC (lower is better). Bottom: low-tail deletion AUC (higher is better).

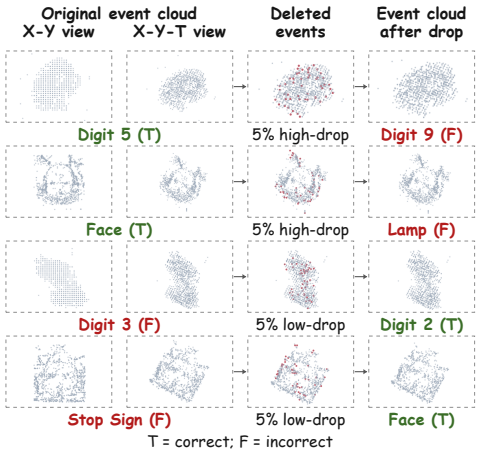

Figure 5: Representative prediction changes after deleting ranked events under VGER. Red points denote deleted events; green/red text marks correct/incorrect predictions.

identify influential events. The low-tail results reveal the complementary role of the structural component. While voxel evidence alone improves high-tail deletion, the full voxel-structural fusion gives the best low-tail preservation over most ratios, keeping accuracy higher than PCS-XYT, voxel-only, and Random before the curves converge at large deletion ratios. Together, the two panels show that voxel perturbation sharpens the head of the ranking, while structural fusion improves the tail. This supports the design of VGER: combining regional task evidence with global structural ordering yields a ranking that is effective for both explanation and safe event removal.

5.8 Representative Prediction Changes

Figure 5 shows representative prediction changes caused by deleting events selected by VGER. Each row contains the original event cloud, the deleted events highlighted in red, and the retained event cloud after deletion. Green labels indicate correct predictions, while red labels indicate incorrect predictions.

The first two rows show failure-inducing high-tail deletions. Removing a small set of highly ranked events changes originally correct predictions into incorrect ones, from digit 5 to digit 9 and from face to lamp. These cases suggest that the removed events contain decision-critical evidence for the original class.

The last two rows show the complementary behavior. For samples that are initially misclassified, deleting ranked events suppresses misleading evidence and recovers the correct class, changing digit 3 to digit 2 and stop sign to face. Overall, these examples connect the aggregate two-tail deletion curves with concrete event-level behavior: the ranking identifies influential events whose removal can alter or even correct the model decision.

6 Conclusion

We presented VGER, a training-free method that fuses event-level gradient evidence with task-aware voxel perturbations to produce a global event ranking for two-tail deletion. Across nine archived dataset–backbone settings, VGER improves both tails over PCS-XYT in every case. Its head identifies decision-critical events, while its tail identifies removable events that preserve accuracy, so one ranking supports both explanation and compression. The two-tail deletion protocol is architecture-agnostic and reproducible from the archived evidence trail. Beyond individual model interpretation, event cloud attribution provides a general tool for analyzing how asynchronous event streams contribute to perception decisions. Such event-level evidence may further support future research in architecture design, adversarial robustness, event selection, and efficient event processing.

AI USE STATEMENT

Generative AI tools assisted with drafting the appendix clarifications, inspecting implementation details, and checking arithmetic consistency in the reported tables. No new model experiments were conducted for these appendix additions. The authors are responsible for verifying the resulting text and numerical corrections before submission.

REFERENCES

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018.

Seyed Mohammad Ahmadi, Koorosh Aslansefat, Rubén Valcarce-Diñeiro, and Joshua Barnfather. Explainability of point cloud neural networks using smile: Statistical model-agnostic interpretability with local explanations. *arXiv preprint arXiv:2410.15374*, 2024.

David Alvarez-Melis and Tommi S Jaakkola. On the robustness of interpretability methods. *arXiv preprint arXiv:1806.08049*, 2018.

Arnon Amir, Brian Taba, David Berg, Timothy Melano, Jeffrey McKinstry, Carmelo Di Nolfo, Tapan Nayak, Alexander Andreopoulos, Guillaume Garreau, Marcela Mendoza, et al. A low power, fully event-based gesture recognition system. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7243–7252, 2017.

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

Ammar Bitar, Rafael Rosales, and Michael Paulitsch. Gradient-based feature-attribution explainability methods for spiking neural networks. *Frontiers in Neuroscience*, 17:1153999, 2023.

Marco Cannici, Marco Ciccone, Andrea Romanoni, and Matteo Matteucci. A differentiable recurrent surface for asynchronous event-based data. In *European Conference on Computer Vision*, pp. 136–152. Springer, 2020.

Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pp. 839–847. IEEE, 2018.

Yongjian Deng, Hao Chen, Hai Liu, and Youfu Li. A voxel graph cnn for object classification with event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1172–1181, 2022.

Tuo Feng, Ruijie Quan, Xiaohan Wang, Wenguan Wang, and Yi Yang. Interpretable3d: An ad-hoc interpretable classifier for 3d point clouds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 1761–1769, 2024.

Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE international conference on computer vision*, pp. 3429–3437, 2017.

Guillermo Gallego, Tobias Delbrück, Garrick Michael Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J Davison, Jörg Conradt, Kostas Daniilidis, et al. Event-based vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(1): 154–180, 2022.

Daniel Gehrig, Antonio Loquercio, Konstantinos G Derpanis, and Davide Scaramuzza. End-to-end learning of representations for asynchronous event-based data. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5633–5643, 2019.

Tristan Gomez, Thomas Fréour, and Harold Mouchère. Metrics for saliency map evaluation of deep learning explanation methods. In *International Conference on Pattern Recognition and Artificial Intelligence*, pp. 84–95. Springer, 2022.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. *Advances in neural information processing systems*, 32, 2019.

Ze Huang, Li Sun, Cheng Zhao, Song Li, and Songzhi Su. Eventpoint: Self-supervised interest point detection and description for event-based camera. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 5396–5405, 2023.

Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un) reliability of saliency methods. In *Explainable AI: Interpreting, explaining and visualizing deep learning*, pp. 267–280. Springer, 2019.

Meir Yossef Levi and Guy Gilboa. Fast and simple explainability for point cloud networks. *arXiv preprint arXiv:2403.07706*, 2024.

Feifei Li, Mi Zhang, Zhaoxiang Wang, and Min Yang. Infocons: Identifying interpretable critical concepts in point clouds via information theory. *arXiv preprint arXiv:2505.19820*, 2025.

Hongmin Li, Hanchao Liu, Xiangyang Ji, Guoqi Li, and Luping Shi. Cifar10-dvs: an event-stream dataset for object classification. *Frontiers in neuroscience*, 11:244131, 2017.

Xiuhong Lin, Changjie Qiu, Siqi Shen, Yu Zang, Weiquan Liu, Xuesheng Bian, Matthias Müller, Cheng Wang, et al. E2pnet: event to point cloud registration with spatio-temporal representation learning. *Advances in Neural Information Processing Systems*, 36:18076–18089, 2023.

Shuai Liu, Boyang Li, Zhiyu Fang, Mingyue Cui, and Kai Huang. Ffam: feature factorization activation map for explanation of 3d detectors. *Advances in Neural Information Processing Systems*, 37:80363–80382, 2024.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

Raju Ningappa Mulawade, Christoph Garth, and Alexander Wiebel. Xai for point cloud data using perturbations based on meaningful segmentation. *IEEE Access*, 2025.

Garrick Orchard, Ajinkya Jayawant, Gregory K Cohen, and Nitish Thakor. Converting static image datasets to spiking neuromorphic datasets using saccades. *Frontiers in neuroscience*, 9:437, 2015.

Yansong Peng, Yueyi Zhang, Zhiwei Xiong, Xiaoyan Sun, and Feng Wu. Get: Group event transformer for event-based vision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6038–6048, 2023.

Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017a.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017b.

Hongwei Ren, Yue Zhou, Xiaopeng Lin, Yulong Huang, Haotian Fu, Jie Song, and Bojun Cheng. Spikepoint: An efficient point-based spiking neural network for event cameras action recognition. In *International Conference on Learning Representations*, volume 2024, pp. 27827–27846, 2024a.

Hongwei Ren, Jiadong Zhu, Yue Zhou, Haotian Fu, Yulong Huang, and Bojun Cheng. A simple and effective point-based network for event camera 6-dofs pose relocalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18112–18121, 2024b.

Hongwei Ren, Yue Zhou, Jiadong Zhu, Xiaopeng Lin, Haotian Fu, Yulong Huang, Yuetong Fang, Fei Ma, Hao Yu, and Bojun Cheng. Rethinking efficient and effective point-based networks for event camera classification and regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Hongwei Ren, Youxin Jiang, Tuopusen Huang, and Xiangqian Wu. A systematic survey on event camera representation learning. *arXiv preprint arXiv:2606.23078*, 2026a.

Hongwei Ren, Fei Ma, Yuetong Fang, Hongxiang Huang, Yue Zhou, Yulong Huang, Haotian FU, Ziyi Yang, Youxin Jiang, Xiangqian Wu, et al. Scalable event cloud network for event-based classification. In *Forty-third International Conference on Machine Learning*, 2026b.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Alberto Sabater, Luis Montesano, and Ana C Murillo. Event transformer. a sparse-aware solution for efficient event data processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2677–2686, 2022.

Simon Schaefer, Daniel Gehrig, and Davide Scaramuzza. Aegnn: Asynchronous event-based graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12371–12381, 2022.

Yusuke Sekikawa, Kosuke Hara, and Hideo Saito. Eventnet: Asynchronous recursive event processing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3887–3896, 2019.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Wen Shen, Qihan Ren, Dongrui Liu, and Quanshi Zhang. Interpreting representation quality of dnns for 3d point cloud processing. *Advances in Neural Information Processing Systems*, 34: 8857–8870, 2021.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Amos Sironi, Manuele Brambilla, Nicolas Bourdis, Xavier Lagorce, and Ryad Benosman. Hats: Histograms of averaged time surfaces for robust event-based object classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1731–1740, 2018.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pp. 3319–3328. PMLR, 2017.

Richard Tomsett, Dan Harborne, Supriyo Chakraborty, Prudhvi Gurram, and Alun Preece. Sanity checks for saliency metrics. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 6021–6029, 2020.

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)*, 38(5): 1–12, 2019.

Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Suggala, David I Inouye, and Pradeep K Ravikumar. On the (in) fidelity and sensitivity of explanations. *Advances in neural information processing systems*, 32, 2019.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 16259–16268, 2021.

Tianhang Zheng, Changyou Chen, Junsong Yuan, Bo Li, and Kui Ren. Pointcloud saliency maps. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1598–1606, 2019.

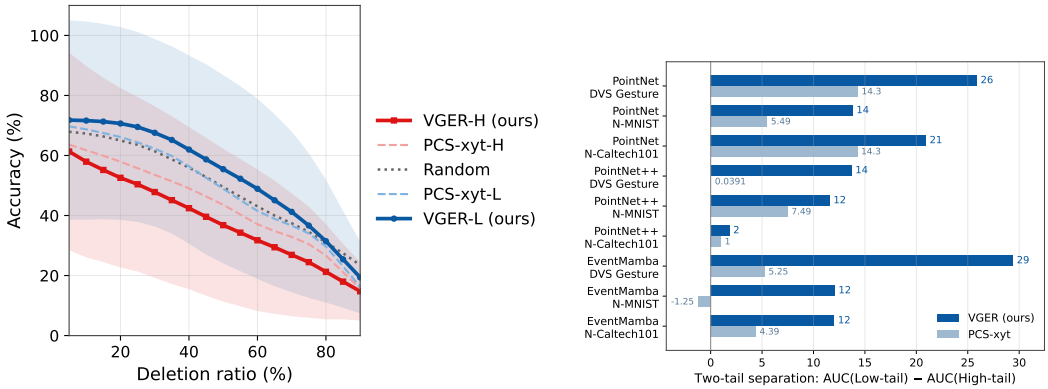

Figure 6: Archived aggregate visualization. Left: macro-average two-tail curves (shading: ± 1 standard deviation across settings, not a confidence interval). Right: two-tail separation (low-tail AUC minus high-tail AUC) per setting. This figure is retained as the archived version; its values are not fully consistent with the detailed tables in Appendix A.

A Detailed 3×3 Results

Tables 3–11 report the detailed setting-level curves. AUC is normalized over the common 5%–90% deletion interval. High-tail AUC is better when lower; low-tail AUC is better when higher under the reported task-accuracy criterion. Win counts compare VGER with PCS-xyt at 18 deletion ratios. These detailed tables contain discrepancies with the main summary in Table 2; their result provenance remains to be reconciled.

The PointNet++/N-MNIST row illustrates why the main paper reports both curve integrals and pointwise counts: AUC weights the magnitude of curve differences, whereas a win count gives every sampled ratio one vote.

Tables 3–11 report the full 5%–90% deletion curves for the nine archived settings. Each table has five curves, eighteen deletion ratios, and a final normalized AUC column as well. The two VGER rows are shaded in light cyan. Bold indicates the best value: lower accuracy/AUC is better for high-tail deletion, and higher accuracy/AUC is better for low-tail deletion.

B Macro Two-Tail View

Figure 6 presents the archived macro two-tail visualization and setting-level separations. Its numerical provenance is not fully consistent with the detailed tables in Appendix A. In particular, an improvement in every setting is not supported by the detailed-table version of the results.

C Representative Prediction Changes

Figure 7 uses two cases from the archive's precomputed clarity subset. The first shows that deleting the top 5% of ranked events changes a correct N-Caltech101 prediction. The second shows a selected N-MNIST case in which deleting the bottom 5% corrects the prediction. These cases make the deletion operator inspectable at the event level. The second case demonstrates a change in prediction, rather than establishing that the removed events are decision-irrelevant; Appendix E clarifies this distinction.

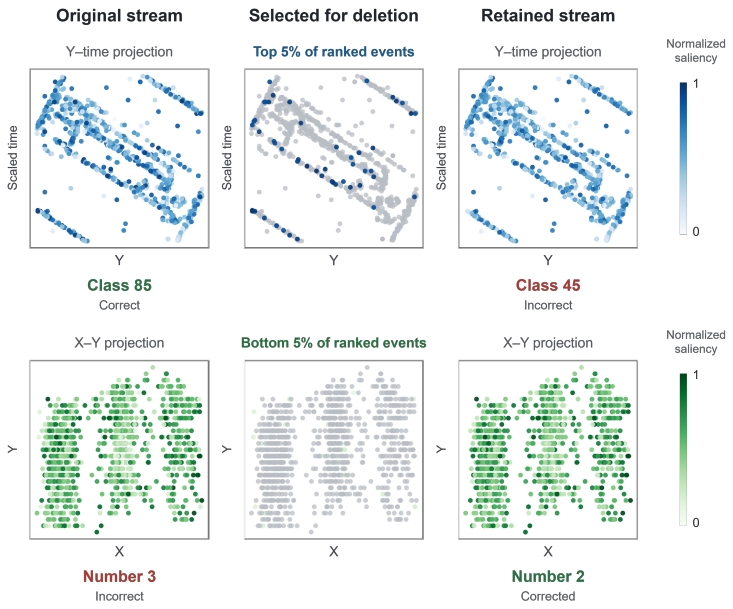

Figure 7: Representative PointNet cases. Top: N-Caltech101 sample 15,615 changes from the correct class 85 to class 45 after high-tail 5% deletion. Bottom: N-MNIST sample 2,169 changes from class 3 to the ground-truth class 2 after low-tail 5% deletion. Colors encode saliency within each row; gray points denote the deleted mask.

Table 3: PointNet on DVS Gesture: accuracy after deleting each ratio of events. The final column reports normalized AUC over 5%–90%. H denotes high-tail deletion and L denotes low-tail deletion.

Method	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	AUC
VGER-H	**0.8010**	**0.7438**	**0.7077**	**0.6867**	**0.6693**	**0.6554**	**0.6411**	**0.6210**	**0.5952**	**0.5713**	**0.5368**	**0.5034**	**0.4717**	**0.4373**	**0.3995**	**0.3532**	**0.2928**	0.2371	**0.5532**
PCS-XYT-H	0.8433	0.8135	0.7785	0.7489	0.7186	0.7026	0.6882	0.6745	0.6624	0.6417	0.6248	0.6080	0.5946	0.5795	0.5658	0.5143	0.3668	**0.1842**	0.6351
VGER-L	**0.9808**	**0.9731**	**0.9623**	**0.9529**	**0.9438**	0.9268	0.9150	0.8946	**0.8839**	**0.8670**	**0.8470**	**0.8202**	**0.7761**	**0.7193**	0.6458	0.5612	**0.4601**	**0.3316**	**0.8121**
PCS-XYT-L	0.9612	0.9422	0.9263	0.9076	0.8925	0.8801	0.8655	0.8519	0.8323	0.8087	0.7847	0.7583	0.7321	0.7134	**0.6924**	**0.6206**	0.4419	0.1946	0.7781
Random	0.9530	0.9496	0.9493	0.9432	0.9381	**0.9283**	**0.9153**	**0.8976**	0.8688	0.8350	0.8027	0.7512	0.6965	0.6402	0.5698	0.4983	0.4084	0.3136	0.7780

Table 4: PointNet on N-MNIST: accuracy after deleting each ratio of events. The final column reports normalized AUC over 5%–90%. H denotes high-tail deletion and L denotes low-tail deletion.

Method	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	AUC
VGER-H	**0.8187**	**0.7918**	**0.7775**	**0.7647**	**0.7559**	**0.7437**	**0.7264**	**0.7037**	**0.6821**	**0.6555**	**0.6187**	**0.5709**	**0.5089**	**0.4456**	**0.3724**	**0.2945**	**0.2235**	0.1698	**0.5959**
PCS-XYT-H	0.8296	0.8113	0.7966	0.7846	0.7733	0.7547	0.7393	0.7235	0.6991	0.6632	0.6312	0.6131	0.6052	0.5949	0.5463	0.4308	0.2611	**0.1495**	0.6422
VGER-L	**0.9624**	**0.9510**	**0.9347**	**0.9199**	**0.9032**	**0.8886**	**0.8691**	**0.8504**	**0.8245**	**0.7960**	**0.7606**	**0.7218**	**0.6673**	0.6001	0.5152	0.4070	**0.2913**	**0.1898**	**0.7339**
PCS-XYT-L	0.9285	0.8999	0.8759	0.8593	0.8449	0.8281	0.8106	0.7828	0.7550	0.7129	0.6715	0.6521	0.6401	**0.6294**	**0.5890**	**0.4740**	0.2841	0.1539	0.6971
Random	0.9287	0.9245	0.9132	0.9024	0.8903	0.8765	0.8586	0.8367	0.8038	0.7716	0.7275	0.6798	0.6108	0.5235	0.4252	0.3255	0.2294	0.1633	0.6968

Table 5: PointNet on N-Caltech101: accuracy after deleting each ratio of events. The final column reports normalized AUC over 5%–90%. H denotes high-tail deletion and L denotes low-tail deletion.

Method	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	AUC
VGER-H	**0.0888**	**0.0689**	**0.0636**	**0.0633**	**0.0626**	**0.0611**	**0.0598**	**0.0607**	**0.0608**	**0.0597**	**0.0559**	**0.0524**	**0.0489**	**0.0457**	**0.0445**	**0.0448**	**0.0513**	**0.0611**	**0.0576**
PCS-XYT-H	0.0889	0.0815	0.0789	0.0782	0.0773	0.0761	0.0750	0.0741	0.0738	0.0733	0.0733	0.0733	0.0736	0.0731	0.0727	0.0722	0.0762	0.0779	0.0756
VGER-L	**0.2888**	**0.2931**	**0.3086**	**0.3101**	**0.3066**	**0.3002**	**0.2944**	**0.2868**	**0.2805**	**0.2707**	**0.2673**	**0.2587**	**0.2495**	**0.2422**	**0.2266**	**0.2137**	**0.1953**	**0.1648**	**0.2665**
PCS-XYT-L	0.2686	0.2619	0.2498	0.2418	0.2370	0.2340	0.2329	0.2289	0.2230	0.2187	0.2151	0.2101	0.2068	0.2035	0.1977	0.1870	0.1600	0.1286	0.2180
Random	0.1680	0.1661	0.1617	0.1575	0.1554	0.1505	0.1481	0.1449	0.1432	0.1391	0.1372	0.1343	0.1275	0.1255	0.1217	0.1182	0.1102	0.1073	0.1399

Table 6: PointNet++ on DVS Gesture: accuracy after deleting each ratio of events. The final column reports normalized AUC over 5%–90%. H denotes high-tail deletion and L denotes low-tail deletion.

Method	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	AUC
VGER-H	**0.9683**	**0.9579**	**0.9485**	**0.9319**	**0.9052**	**0.8794**	**0.8386**	0.7820	0.6944	0.5704	0.4261	0.2941	0.1915	0.1333	0.1033	0.0861	0.0758	0.0667	0.5492
PCS-XYT-H	0.9689	0.9595	0.9509	0.9410	0.9233	0.8973	0.8528	0.7864	0.6647	0.5365	0.3545	0.1918	0.1214	0.0935	0.0800	0.0735	0.0696	0.0667	0.5303
VGER-L	**0.9872**	**0.9888**	**0.9884**	**0.9837**	0.9663	0.9318	0.8600	0.7272	0.5485	0.3669	0.2340	0.1523	**0.1083**	0.0865	0.0775	0.0717	0.0670	0.0571	0.5107
PCS-XYT-L	0.9863	0.9855	0.9840	0.9774	**0.9678**	**0.9395**	**0.8913**	**0.7877**	0.6268	0.4635	0.2999	0.1591	0.1080	**0.0904**	**0.0815**	**0.0747**	**0.0702**	**0.0634**	**0.5313**
Random	0.9793	0.9734	0.9707	0.9574	0.9453	0.9127	0.8638	**0.7723**	**0.6461**	**0.4785**	**0.3033**	**0.1809**	**0.1012**	**0.0799**	**0.0743**	**0.0701**	**0.0667**	**0.0563**	**0.5244**

Table 7: PointNet++ on N-MNIST: accuracy after deleting each ratio of events. The final column reports normalized AUC over 5%–90%. H denotes high-tail deletion and L denotes low-tail deletion.

Method	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	AUC
VGER-H	**0.7867**	**0.6971**	**0.5995**	**0.5038**	**0.4161**	**0.3276**	**0.2518**	0.1930	0.1485	0.1261	0.1170	0.1144	0.1140	**0.1135**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	**0.2655**
PCS-XYT-H	0.7936	0.7169	0.6389	0.5453	0.4443	0.3503	0.2577	**0.1890**	**0.1423**	**0.1235**	**0.1150**	**0.1136**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	0.2740
VGER-L	**0.9001**	**0.8897**	**0.8581**	**0.8191**	**0.7440**	**0.6324**	**0.4995**	**0.3526**	**0.2231**	**0.1514**	**0.1207**	**0.1145**	**0.1136**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	**0.3811**
PCS-XYT-L	0.8967	0.8667	0.8174	0.7502	0.6547	0.5354	0.4038	0.2771	0.1773	0.1417	0.1198	0.1143	**0.1136**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	0.3489
Random	0.8635	0.8220	0.7680	0.6801	0.5797	0.4576	0.3258	0.2235	0.1602	0.1270	0.1163	0.1137	**0.1135**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	**0.1135**	0.3194

Table 8: PointNet++ on N-Caltech101: accuracy after deleting each ratio of events. The final column reports normalized AUC over 5%–90%. H denotes high-tail deletion and L denotes low-tail deletion.

Method	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	AUC
VGER-H	**0.0938**	**0.0877**	**0.0853**	**0.0835**	**0.0829**	**0.0859**	**0.0856**	**0.0856**	**0.0858**	**0.0843**	**0.0830**	**0.0803**	**0.0785**	0.0734	0.0703	0.0669	0.0629	0.0585	**0.0799**
PCS-XYT-H	0.0953	0.0928	0.0893	0.0888	0.0885	0.0880	0.0883	0.0890	0.0900	0.0881	0.0905	0.0870	0.0805	**0.0730**	**0.0648**	**0.0593**	**0.0561**	**0.0555**	0.0817
VGER-L	0.1137	**0.1130**	**0.1113**	**0.1141**	**0.1120**	**0.1102**	**0.1092**	**0.1069**	**0.1047**	**0.1048**	**0.1013**	**0.0973**	**0.0934**	**0.0888**	**0.0806**	**0.0717**	0.0627	0.0581	**0.0981**
PCS-XYT-L	**0.1156**	0.1128	0.1091	0.1066	0.1068	0.1040	0.1007	0.0986	0.0977	0.0979	0.0959	0.0927	0.0866	0.0785	0.0684	0.0606	0.0570	0.0561	0.0918
Random	0.1056	0.1026	0.1017	0.0994	0.0984	0.0966	0.0956	0.0935	0.0930	0.0902	0.0893	0.0864	0.0846	0.0787	0.0740	0.0681	**0.0632**	**0.0595**	0.0881

Table 9: EventMamba on DVS Gesture: accuracy after deleting each ratio of events. The final column reports normalized AUC over 5%–90%. H denotes high-tail deletion and L denotes low-tail deletion.

Method	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	AUC
VGER-H	**0.7356**	**0.6970**	**0.6701**	**0.6287**	**0.5961**	**0.5579**	**0.5208**	**0.4860**	**0.4498**	**0.4247**	**0.3954**	**0.3536**	**0.3240**	**0.2866**	**0.2668**	**0.2355**	**0.1954**	**0.1468**	**0.4429**
PCS-XYT-H	0.8167	0.7993	0.7765	0.7513	0.7247	0.6974	0.6695	0.6435	0.6147	0.5914	0.5804	0.5398	0.4954	0.4484	0.4000	0.3278	0.2520	0.1613	0.5765
VGER-L	**0.8804**	**0.8928**	**0.9020**	**0.9093**	**0.9103**	**0.9088**	**0.9037**	**0.8933**	**0.8742**	**0.8390**	**0.8091**	**0.7581**	**0.6814**	**0.5967**	**0.4888**	0.3726	0.2618	0.1693	**0.7369**
PCS-XYT-L	0.8188	0.8138	0.8116	0.8049	0.7972	0.7853	0.7783	0.7614	0.7418	0.7103	0.6716	0.6003	0.5313	0.4623	0.3891	0.3148	0.2322	0.1558	0.6290
Random	0.8202	0.8150	0.8066	0.8104	0.8061	0.7912	0.7761	0.7706	0.7459	0.7198	0.6909	0.6465	0.5851	0.5365	0.4818	**0.3844**	**0.3073**	**0.2112**	0.6582

Table 10: EventMamba on N-MNIST: accuracy after deleting each ratio of events. The final column reports normalized AUC over 5%–90%. H denotes high-tail deletion and L denotes low-tail deletion.

Method	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	AUC
VGER-H	**0.9320**	**0.9205**	**0.9062**	**0.8937**	**0.8806**	**0.8703**	**0.8585**	**0.8426**	**0.8332**	**0.8128**	**0.7912**	**0.7724**	**0.7396**	**0.6950**	**0.6441**	**0.5618**	**0.4638**	**0.3754**	**0.7729**
PCS-XYT-H	0.9619	0.9598	0.9591	0.9564	0.9557	0.9483	0.9489	0.9459	0.9362	0.9263	0.8960	0.8634	0.8238	0.7858	0.7316	0.6766	0.5890	0.5270	0.8616
VGER-L	**0.9716**	**0.9742**	**0.9778**	**0.9778**	**0.9807**	**0.9785**	**0.9781**	**0.9718**	**0.9685**	0.9570	0.9449	0.9250	0.8953	0.8563	0.7990	0.7085	0.5808	0.4604	0.8935
PCS-XYT-L	0.9562	0.9536	0.9507	0.9484	0.9446	0.9414	0.9415	0.9348	0.9275	0.9109	0.8763	0.8369	0.8001	0.7616	0.7249	0.6625	0.5895	0.5021	0.8491
Random	0.9571	0.9619	0.9620	0.9669	0.9673	0.9671	0.9668	0.9671	0.9683	**0.9666**	**0.9662**	**0.9659**	**0.9617**	**0.9573**	**0.9477**	**0.9331**	**0.9050**	**0.8316**	**0.9544**

Table 11: EventMamba on N-Caltech101: accuracy after deleting each ratio of events. The final column reports normalized AUC over 5%–90%. H denotes high-tail deletion and L denotes low-tail deletion.

Method	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	AUC
VGER-H	**0.2021**	**0.2104**	**0.2031**	**0.1816**	**0.1641**	**0.1665**	**0.1528**	**0.1367**	**0.1270**	**0.1255**	**0.1201**	**0.1050**	**0.1064**	**0.0918**	**0.0884**	**0.0820**	**0.0610**	**0.0498**	**0.1323**
PCS-XYT-H	0.2358	0.2319	0.2271	0.2231	0.2222	0.2153	0.2104	0.1978	0.2012	0.1938	0.1880	0.1782	0.1611	0.1426	0.1216	0.0947	0.0874	0.0591	0.1791
VGER-L	**0.2998**	**0.2886**	**0.2954**	**0.2998**	**0.2837**	**0.2793**	**0.2769**	**0.2725**	**0.2739**	**0.2617**	**0.2490**	**0.2451**	0.2158	0.2061	0.1792	0.1475	0.1167	0.0742	**0.2399**
PCS-XYT-L	0.2539	0.2524	0.2485	0.2568	0.2520	0.2617	0.2632	0.2534	0.2495	0.2383	0.2388	0.2261	0.2075	0.1758	0.1567	0.1255	0.1060	0.0781	0.2164
Random	0.2461	0.2534	0.2441	0.2334	0.2471	0.2368	0.2402	0.2358	0.2393	0.2319	0.2344	0.2285	**0.2222**	**0.2114**	**0.2026**	**0.1890**	**0.1719**	**0.1484**	0.2247

Table 12: Main-evaluation settings.

Shared setting	Archived value
Events per stream; deletion ratios	1,024; 5%–90% by 5%.
Default voxel grid	$G_x \times G_y \times G_t = 8 \times 8 \times 4$.
Anisotropic default $\rho = \omega_t/\omega_s$	0.5; $\rho = 1$ is the equal-weight special case.

Table 13: Archived values of the structural weights used by the structural score.

Structural weight	Role in structural score
w_b $(w_{\text{base}}) = 1$	Retains the endpoint base evidence.
w_c $(w_{\text{cell}}) = 1$	Weights the event rank within its spatio-temporal voxel.
w_t $(w_{\text{time}}) = 1$	Weights the event rank within its temporal bin.
w_r $(w_{\text{rare}}) = 0.5$	Rewards relatively sparse occupied voxels through $1 - \bar{D}$.
w_d $(w_{\text{dense}}) = 0.1$	Penalizes normalized voxel density $\bar{D}$.

D Evaluation Configuration and Sensitivity

Tables 12 and 13 separate the reported shared evaluation protocol from the structural-score configuration. VGER uses the anisotropic default $\rho = 0.5$. Under Eq. (4), this gives unequal temporal and spatial weights; $\rho = 1$ is only the equal-weight, isotropic special case. Table 13 records the structural weights listed with the manuscript. These tables do not by themselves identify every setting-specific fusion configuration; Appendix G discusses this reporting limitation.

Figure 8 reports the archived temporal-to-spatial weight sensitivity on N-Caltech101 test samples with PointNet. Its horizontal axis retains the plotting notation λ_t/λ_s. The dashed line at one denotes an equal-weight comparison reference, not VGER's anisotropic default $\rho = 0.5$. An explicit mapping from the plotting parameters to $\rho = \omega_t/\omega_s$ is needed to identify this curve with the metric in Eq. (4). The displayed AUCs change modestly across the tested values. This observation is confined to the shown setting and range; it does not establish sensitivity to all fusion weights or robustness across all dataset–backbone combinations.

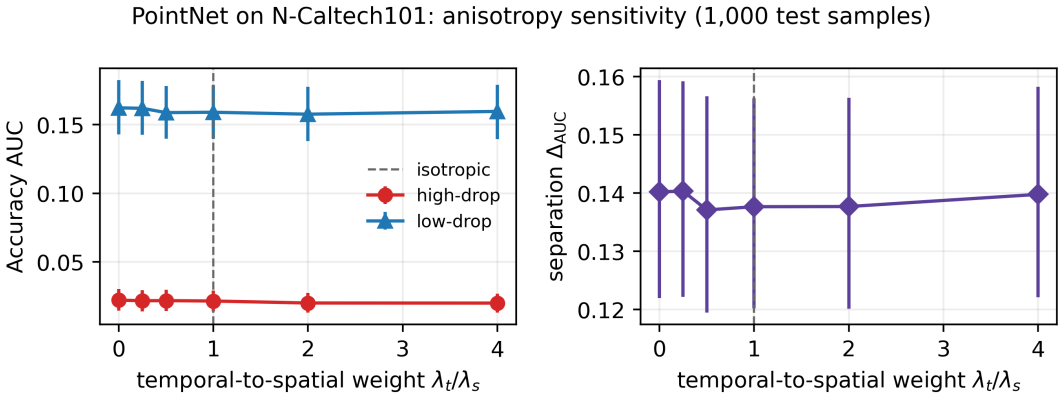

Figure 8: Archived temporal-to-spatial weight sensitivity on PointNet/N-Caltech101 test samples. Left: high-tail and low-tail accuracy AUC. Right: low-tail AUC minus high-tail AUC. The horizontal axis uses λ_t/λ_s. The dashed line at one marks an equal-weight comparison reference; VGER's default is anisotropic, with $\rho = 0.5$. The plotting notation and the metric parameter require an explicit mapping.

E Attribution Target and Interpretation

E.1 Ground-truth-conditioned attribution

The gradient and voxel terms in Eqs. (8) and (10) use the task loss with the ground-truth label y. Consequently, the reported attribution is conditional on a supervised target. It characterizes sensitivity and finite loss changes relative to that target. For an initially misclassified stream, evidence that reduces the ground-truth loss need not be evidence supporting the model's original predicted class. Ground-truth targeting is therefore suitable for analyzing task performance and error-related interventions, but the present results do not establish an explanation of every original model prediction.

A prediction-conditioned evaluation would instead determine the target from the unperturbed prediction and hold it fixed throughout gradient computation, voxel perturbation, and deletion evaluation. Recomputing the target after every intervention would change the question along the curve. Although the gradient routine supports a predicted-class fallback when no label is supplied, that software capability does not constitute an evaluated predicted-class result: the evaluated scoring path explicitly supplies the dataset labels. The ground-truth-conditioned results should be interpreted within this scope.

E.2 SIGNED EFFECTS AND SCORE NORMALIZATION

The underlying evidence is signed. A positive voxel loss difference in Eq. (10) means that the specified regional intervention increases target loss; a negative difference means that it decreases target loss. The PCS term similarly estimates a signed loss change along its contraction direction, with the radial weighting in Eq. (8). These quantities measure intervention-dependent effects rather than unsigned influence magnitudes.

The scoring implementation applies stream-wise min–max normalization. This transformation preserves the ordering of unequal raw scores; it neither takes absolute values nor makes a negative effect equivalent to a negligible effect. The minimum normalized value identifies the smallest raw score, which may be strongly negative. In addition, the final score includes structural ranks and fusion weights, so its numerical zero is not a reference point for zero task effect. Its ordering must be interpreted using the fixed fusion configuration and score direction.

E.3 WHAT LOW-TAIL ACCURACY DOES AND DOES NOT MEASURE

High accuracy after low-tail deletion measures task performance after removing the selected events. It does not, by itself, establish that those events have limited influence on the model output. The prediction correction in Figure 7 directly illustrates this distinction: removing low-ranked events changes an incorrect prediction into the ground-truth class. Such a change is consistent with the removal of counterevidence relative to the supervised target, rather than with complete decision irrelevance.

Likewise, aggregate accuracy can conceal both correct-to-incorrect and incorrect-to-correct transitions. Preserving the original prediction and preserving accuracy relative to ground truth are distinct objectives. Accordingly, the low-tail curves provide evidence about target-conditioned deletion behavior; they do not establish a safe-removal guarantee or a prediction-preserving compression method. Those stronger interpretations would require measurements of original-prediction agreement and output changes, as well as validation of joint event removal. Small individual effects would not alone control interactions among multiple deleted events.

F PERTURBATION SEMANTICS AND COMPUTATIONAL COST

F.1 CENTER REPLACEMENT AND EVENT DELETION

The voxel-scoring routine replaces the selected events' three coordinates with the coordinate-wise median of the original stream. This preserves the number of events and their polarity values, but changes their spatial coordinates and timestamps. The routine keeps array positions and does not explicitly sort the modified timestamps. Retaining array order should therefore not be interpreted as preserving chronological order after replacement.

The resulting voxel score is the loss change under center replacement. The evaluation routine instead gathers a subset of retained events. These interventions are not equivalent: center replacement can create a concentrated event configuration, whereas deletion changes the input cardinality. Neither operator is guaranteed to remain within the training distribution. Using the same deletion operator across attribution methods controls one aspect of the comparison, but it does not remove this distributional limitation or prove that center replacement estimates the effect of true regional deletion. The regional evidence is therefore interpreted relative to its stated perturbation operator.

F.2 FORWARD EVALUATIONS AND BATCHING

Let M denote the number of occupied voxels, and let B denote the number of voxel interventions evaluated in one batch. The voxel branch evaluates the original stream once and constructs one perturbed input for each occupied voxel. It therefore requires $M+1$ input evaluations, implemented as one original-input forward call and $\lceil M/B \rceil$ batched perturbation calls. Batching reduces the number of calls but does not eliminate the computation of the M perturbed inputs.

The gradient branch additionally requires a forward evaluation and an input-gradient computation. If a single-stream forward evaluation costs $C_f(N)$ and the input-gradient computation costs $C_b(N)$ for

N events, the model-computation cost of the two branches is approximately $(M+2)C_f(N)+C_b(N)$ when their original-input forwards are not shared. This accounting excludes voxel assignment, structural-score construction, ranking, and the separate deletion-evaluation curves. Compared with gradient-only scoring, the main additional model cost comes from the M perturbed streams. The implementation materializes batches of these streams, so the batch size also affects memory use. This is an operation-count analysis; it does not supply measured latency, peak memory, or a hardware-specific speedup.

G Fusion and Implementation Conventions

G.1 Repeated evidence in the structural branch

Equation (17) retains the base score, which already contains gradient and voxel evidence, and Eq. (18) adds those channels again. The structural branch is consequently a transformation of the existing evidence and local context, rather than an independent third measurement of model response. Reusing the base score can retain its global ordering while local ranks alter relative positions. This construction alone does not show that the repeated fusion is necessary or superior to a simpler fusion; that is an empirical question.

The occupancy terms admit a direct simplification:

$$w_r(1-D) - w_d D = w_r - (w_r + w_d)D. \tag{20}$$

For fixed stream-level weights, the additive constant w_r does not change event order, and min–max normalization also removes this common offset. Thus, the two density coefficients have a combined ranking effect governed by $w_r + w_d$; they should not be interpreted as two independently identifiable ranking mechanisms.

G.2 Anisotropy and ordering conventions

VGER uses the anisotropic metric $A_\rho = \mathrm{diag}(1, 1, \rho)$ in Eq. (4), with default $\rho = 0.5$ as reported in Table 12. This assigns the temporal coordinate half the weight of each spatial coordinate in the metric. The setting $\rho = 1$ denotes the isotropic special case used to describe the equal-weight limit; it does not describe the default VGER configuration.

The paper defines high-tail deletion as removing larger attribution scores. The exploratory global-ranking code also contains a safety-oriented score convention, negative fusion weights, and separate safety and attack selection objectives. These conventions must not be interchanged when interpreting a saved curve. A reproducible configuration must identify the score direction, the fusion coefficients, and which end of the ordering is deleted.

There are also differences between the idealized protocol and the inspected generic pruning routine. The latter rounds the retained count, retains at least one event, and uses `topk` without an explicit stable tie-breaking rule. It does not directly implement an empty-stream endpoint. These implementation details preclude treating an unrecorded 100% deletion value as a measured result or assuming that the stated floor-based cardinality and stable ordering follow automatically from the routine.

G.3 Configuration provenance

Tables 12 and 13 report shared settings and structural coefficients, but do not specify every setting-specific fusion coefficient and base-score configuration. The exploratory implementation also includes nonlinear rank-calibrated branches and interpolation between them. A mapping from each final result to its exact executed configuration is therefore necessary before the reported curves can be attributed to a single fully specified implementation of Eqs. (12)–(18).

The presence of parameter-search code is not evidence that selection used a held-out validation set. The current reporting does not establish the separation between parameter-selection samples and the samples used in the final curves. No validation/test separation or complete set of unrecorded parameter values is inferred here. Likewise, the sensitivity study for one parameter and one dataset–backbone setting does not establish robustness to all fusion coefficients.

H Relation to Prior Work and Scope of Evidence

VGER shares with regional point-cloud attribution the general principle of analyzing model behavior through local regions, while differing in its objective and attribution granularity. (Shen et al., 2021) investigate point-cloud representations through regional Shapley attribution, whereas InfoCons (Li et al., 2025) uses an information-theoretic framework to identify faithful and conceptually coherent point-cloud subsets. VGER addresses how regional task evidence can be combined with fine-grained event sensitivity to produce an event-level ranking over an entire asynchronous event stream. Its contribution lies in transferring regional evidence to individual-event attribution: voxel perturbations measure task responses to interventions on local spatiotemporal regions, event-level gradients distinguish events within the same voxel, and structural ranking incorporates local and temporal context. The methodological contribution of VGER therefore centers on reconciling regional structure with event-level resolution.

This design addresses a granularity challenge in event attribution. Event-level gradients alone do not directly measure the task response to a finite perturbation of an entire local region, while voxel scores alone assign identical regional evidence to all events within a voxel. VGER combines these complementary signals to incorporate regional measurements while retaining event-level outputs. Its global ranking also enables deletion evaluation at both ends of the same score ordering, revealing how the two tails affect target-conditioned task performance. Here, low-tail results characterize performance after deletion without assuming that low-ranked events have no influence.

The experimental baselines, Random and PCS-XYT, provide a task-agnostic ranking reference and an event-attribution reference derived directly from point-level gradients, respectively. Because PCS-XYT is directly related to VGER's event-level gradient component, this comparison specifically examines whether incorporating regional evidence and structural information improves event ranking. The detailed results show that VGER achieves both lower high-tail deletion AUC and higher low-tail deletion AUC than PCS-XYT across most evaluated dataset–backbone combinations, providing empirical support for the design across multiple settings. These findings support the motivation for combining regional evidence with event-level sensitivity, with performance conclusions scoped to the evaluated baselines, attribution targets, and intervention protocols.