EVENTS ARE NOT UNIT TOKENS: LEARNING THRESHOLD-CONSISTENT REPRESENTATIONS FOR EVENT CAMERAS

Anonymous authors
Paper under double-blind review

ABSTRACT

Event cameras encode a brightness change as an asynchronous event once the log-intensity variation reaches a contrast threshold C. This sensing rule gives event streams their high temporal resolution, but it also means that an event is not a threshold-independent unit token: the same visual change can produce many events at a low threshold and only a few events at a high threshold. Most event representations aggregate counts, timestamps, polarities, or recurrent tokens without preserving the physical contrast carried by each event, so a change in sensor threshold can be mixed with a change in visual content. To address this issue, we propose Pixel Contrast Flow (PCF), a threshold-consistent event representation that uses physical contrast mass as the state-update variable. PCF combines contrast-mass encoding, a semigroup contrast flow, and pixel-local temporal integration before spatial mixing. The resulting update is unchanged when the same contrast is split into finer event quanta, while pixel locality removes sensitivity to threshold-induced reordering across different pixels. On the paired Caltech101-CT benchmark, PCF maintains a nominal accuracy of 49.59% while achieving the best accuracy at $0.25C_0$, $0.5C_0$, and $4C_0$, the highest worst-case OOD accuracy of 19.06%, and the lowest paired feature drift with a RelL2 of 0.215. Detailed robustness tests further show that PCF obtains the best Global Mean, Global Worst, polarity-asymmetry accuracy, and competitive pixel-mismatch performance. These results support a simple principle: event representations should follow the physical quantity measured by the sensor rather than the arbitrary number of emitted tokens.

1 INTRODUCTION

Event cameras report brightness changes asynchronously, providing high temporal resolution, low latency, sparse output, and high dynamic range compared with conventional frame cameras (Lichtsteiner et al., 2008; Gallego et al., 2022; Posch et al., 2014; Brandli et al., 2014). A positive or negative event is emitted when the log-intensity change at a pixel reaches an effective contrast threshold $C_p(\mathbf{x})$ (Serrano-Gotarredona & Linares-Barranco, 2013). This thresholding mechanism is what makes event cameras efficient: static regions stay silent, while moving edges generate events only after sufficient visual change has accumulated.

Existing event-learning methods have made substantial progress by converting asynchronous events into representations that are easier for neural networks to process. Time surfaces, HATS-like descriptors, voxel grids, graph structures, matrix-LSTM representations, asynchronous convolutions, and recurrent tokens all organize sparse event streams into useful spatial-temporal features (Lagorce et al., 2017; Sironi et al., 2018; Gehrig et al., 2019; Bi et al., 2019; Cannici et al., 2020; Messikommer et al., 2020). These designs are effective for capturing where and when events occur. However, most of them still treat an emitted event as a unit token. This abstraction is convenient, but it hides an important physical fact: an event is produced only after the sensor has accumulated a thresholded amount of log contrast.

This omission matters when the contrast threshold changes. For a monotone log-intensity change of magnitude Q, the number of emitted events is approximately $N_C \simeq Q/C$. The same scene and motion can therefore produce many events under a low threshold and only a few events under a

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.

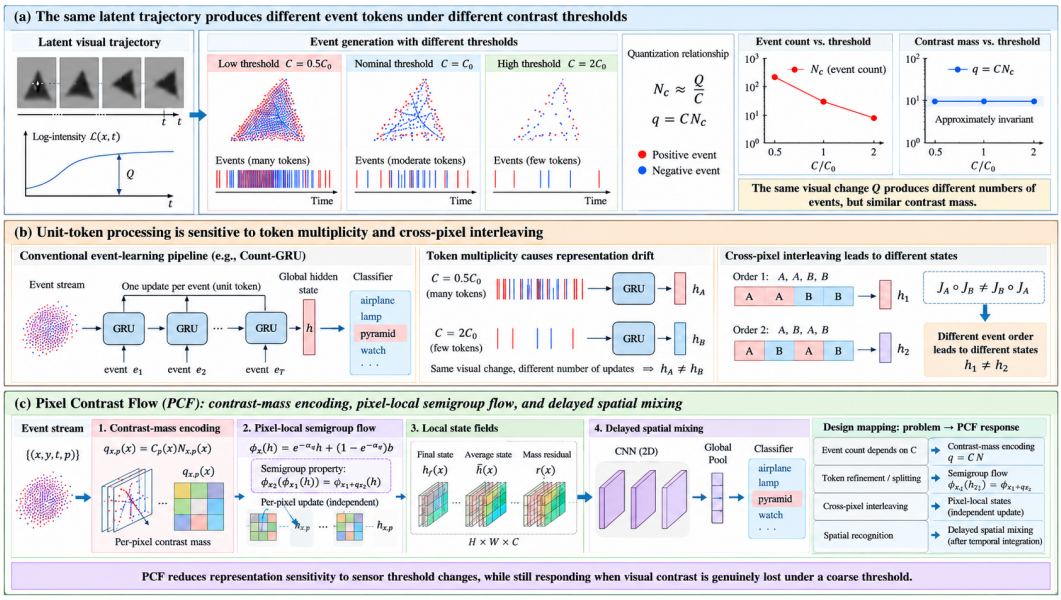


Figure 1: Why threshold consistency matters and how PCF addresses it. (a) The same latent visual trajectory can generate different event streams under different contrast thresholds: lower thresholds produce more event tokens, while the contrast mass $q = CN_C$ remains much more stable. (b) Conventional unit-token processing applies one update per event and uses a global hidden state, making the representation sensitive to both token multiplicity and cross-pixel event interleaving, which can cause representation drift under threshold changes. (c) Pixel Contrast Flow (PCF) first converts event counts into contrast mass, then applies a pixel-local semigroup flow to build local temporal states, and finally performs delayed spatial mixing for classification. This design reduces threshold-induced representation drift while still responding when genuinely coarse thresholds remove useful visual contrast.

high threshold. If a model uses event count as its implicit measure of visual change, then a sensor-only threshold change can be misread as a change in visual content. This problem is not just a simulator artifact. Real event cameras can have asymmetric positive and negative thresholds, and pixel-level threshold mismatch is common because of circuit bias and manufacturing variation (Wang et al., 2020). Realistic simulators such as ESIM and v2e also expose threshold settings and sensor non-idealities (Rebecq et al., 2018; Hu et al., 2021). Data augmentation can improve robustness by exposing the model to corrupted or removed events (Gu et al., 2021), but dropping events does not reproduce the physical re-triggering process caused by a different contrast threshold: crossing times can move, event multiplicity changes, and events from different pixels can be re-interleaved.

To address this limitation, we propose Pixel Contrast Flow (PCF), a threshold-consistent representation framework for event streams. Instead of treating event count as the primitive input, PCF converts event multiplicity into physical contrast mass, $q = CN$. It then evolves local states with a contrast-parametrized flow whose update satisfies a semigroup property: splitting the same contrast mass into finer event quanta gives the same local state. Finally, PCF keeps temporal integration pixel-local and delays spatial mixing until after local contrast-flow states are formed. This design directly targets two threshold-induced nuisances: changes in event multiplicity and changes in the cross-pixel event order.

Experiments on paired Caltech101-CT streams show that this physical and compositional bias improves both threshold robustness and representation stability. On the main paired benchmark, PCF maintains a nominal accuracy of 49.59% while achieving the best accuracy at $0.25C_0$, $0.5C_0$, and $4C_0$, the highest worst-case OOD accuracy of 19.06%, and the lowest paired feature drift with a RelL2 of 0.215. These gains are particularly pronounced under coarse unseen thresholds, where count-based and global-state representations degrade more substantially. Detailed robustness tests further show that PCF achieves the best Global Mean, Global Worst, and polarity-asymmetry accuracy, while remaining competitive under spatially varying pixel mismatch. Direct mechanism diagnostics also

confirm the intended behavior: pixel-local integration reduces threshold-induced cross-pixel ordering sensitivity to numerical precision, whereas a shared global flow exhibits measurable state drift. Moreover, PCF remains stable under moderate threshold-estimation error, with accuracy changing by at most 0.41 percentage points for $\hat{C}/C \in [0.8, 1.2]$. Together, these results indicate that threshold robustness is improved when event representations preserve the physical contrast underlying event generation rather than treating emitted events as threshold-independent unit tokens.

The main contributions of this work are summarized as follows:

- We identify threshold inconsistency as a representation-level failure mode in event learning. We show that event counts mix visual change with the sensor contrast threshold, and formulate physical contrast mass as a more stable primitive for event representation.
- We propose Pixel Contrast Flow, a threshold-consistent event representation framework that combines contrast-mass encoding, semigroup contrast flow, and pixel-local temporal integration before spatial mixing. This design makes local state evolution consistent under contrast refinement and reduces sensitivity to threshold-induced event reordering.
- We build a paired-threshold evaluation protocol that regenerates the same latent visual trajectory under different contrast thresholds. Experiments on paired Caltech101-CT streams and sensor-variation robustness tests show that Pixel Contrast Flow improves unseen-threshold robustness and substantially reduces paired representation drift compared with count-based, normalization-based, and augmentation-based baselines.

2 Related Work

Event representations. A large body of event-based vision research focuses on converting asynchronous events into representations suitable for learning. Time-surface methods retain recent temporal activity (Lagorce et al., 2017; Sironi et al., 2018); voxel and learned-grid representations aggregate events into structured tensors (Gehrig et al., 2019); graph-based methods operate directly on sparse event neighborhoods (Bi et al., 2019); recurrent surfaces and asynchronous convolutions preserve more of the temporal structure (Cannici et al., 2020; Messikommer et al., 2020). These methods differ in how they preserve coordinates, polarity, timestamp, and sparsity, but the event itself is typically used as a unit contribution. Consequently, the physical contrast represented by one event is not an explicit state-evolution variable, and changing the threshold can alter token multiplicity before the learned representation sees the data.

Event augmentation and robustness. Event-based augmentation improves generalization by perturbing the observed stream. EventDrop, for example, removes events according to several policies and improves recognition performance on N-Caltech101 and N-Cars (Gu et al., 2021). Subsequent approaches introduce event-specific geometric transformations, spatiotemporal event mixing, and relevance-guided augmentation (Li et al., 2022; Shen et al., 2023; Sun et al., 2024). These methods improve generalization to event-stream perturbations, but they do not reproduce threshold-dependent event generation. A different C changes the event-generation boundaries themselves, moving crossing times and changing cross-pixel ordering. More generally, robustness learned from finite perturbation samples does not guarantee that two different quantizations of the same physical contrast compose consistently.

Sensor modeling, simulation, and calibration. ESIM generates events from a continuous rendering process and supports separate positive and negative thresholds (Rebecq et al., 2018); v2e converts videos to realistic DVS events and models threshold mismatch, finite bandwidth, and noise (Hu et al., 2021). Calibration methods further show that per-pixel contrast thresholds can be biased and should not always be modeled as a single global constant (Wang et al., 2020). Robust e-NeRF also considers contrast-threshold variation in a reconstruction-specific setting through a more realistic event model and normalized optimization objectives (Low & Lee, 2023). These works concentrate on sensor simulation, calibration, or radiance-field reconstruction rather than a generic representation rule for classification. Together, they show that threshold variation is a real sensing factor and that physically normalized processing can be valuable. These observations directly motivate our design: use the measured contrast quantum inside the representation itself, and make local state evolution consistent with how that quantum can be partitioned.

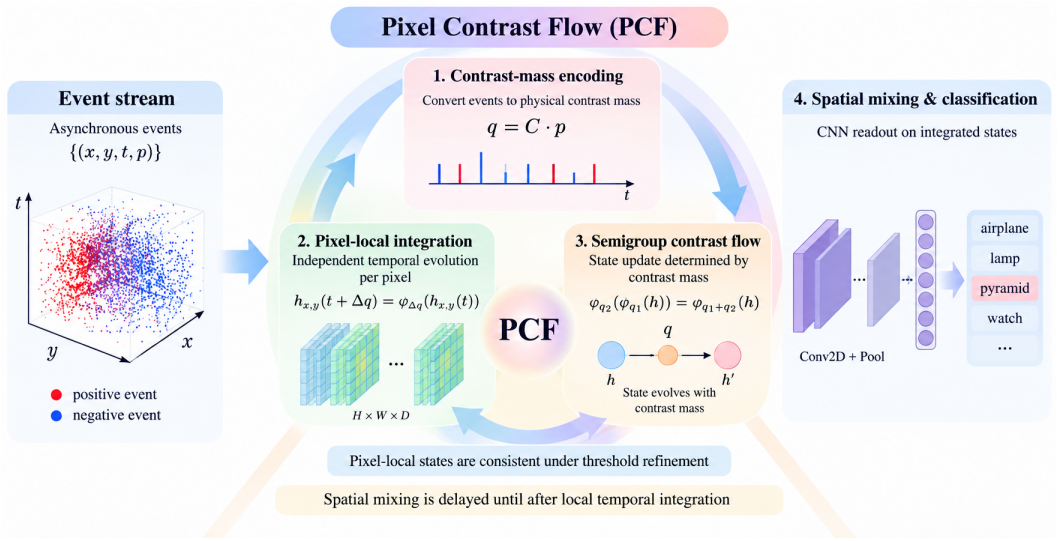


Figure 2: Architecture of Pixel Contrast Flow (PCF). PCF converts event multiplicity into physical contrast mass, performs pixel-local temporal integration with a semigroup contrast flow, and applies spatial mixing only after local temporal states are formed.

3 METHOD

3.1 DESIGN MOTIVATION AND OVERVIEW

An event camera emits an event at pixel $\mathbf{x}$ and polarity $p \in \{-1, +1\}$ when

$$p\left[L(\mathbf{x}, t) - L(\mathbf{x}, t^-)\right] \geq C_p(\mathbf{x}), \tag{1}$$

where $L = \log I$ and $C_p(\mathbf{x})$ is the effective contrast threshold. For a monotone accumulated contrast Q, the event count satisfies $N_C \approx Q/C$. Thus, the same physical change can produce different event multiplicities under different thresholds.

As shown in Figure 2, Pixel Contrast Flow (PCF) replaces event multiplicity with physical contrast mass and evolves pixel-local states along the contrast coordinate. The flow is compositional under contrast refinement, while delaying spatial mixing removes dependence on cross-pixel event interleaving during temporal integration.

3.2 CONTRAST-MASS FLOW

We divide an event window into T shared temporal bins and denote by $N_{t,p}(\mathbf{x})$ the number of polarity-p events at pixel $\mathbf{x}$ in bin t. Rather than treating $N_{t,p}(\mathbf{x})$ as the primitive signal, PCF converts event multiplicity into physical contrast mass,

$$q_{t,p}(\mathbf{x}) = C_p(\mathbf{x}) N_{t,p}(\mathbf{x}), \tag{2}$$

which represents the accumulated log-contrast carried by the events in the bin. For numerical consistency across threshold settings, we use the normalized form

$$\bar{q}_{t,p}(\mathbf{x}) = \frac{C_p(\mathbf{x})}{C_0} N_{t,p}(\mathbf{x}), \tag{3}$$

where C_0 denotes the nominal sensor threshold.

PCF then evolves a local state independently for each pixel, polarity, and state channel. Let $h_{t,p,k}(\mathbf{x})$ denote the k-th local state after processing bin t. Temporal memory is first updated through a learned relaxation,

$$\widetilde{h}_{t,p,k}(\mathbf{x}) = d_{p,k} h_{t-1,p,k}(\mathbf{x}), \qquad d_{p,k} = \exp[-\mathrm{softplus}(\gamma_{p,k})]\,, \tag{4}$$

where different channels can retain information over different temporal scales.

The relaxed state is then advanced along the contrast coordinate according to

$$h_{t,p,k}(\mathbf{x}) = \Phi^{p,k}_{\bar{q}_{t,p}(\mathbf{x})}\left(\widetilde{h}_{t,p,k}(\mathbf{x})\right), \tag{5}$$

with

$$\Phi^{p,k}_{q}(h) = e^{-a_{p,k}q}h + \left(1 - e^{-a_{p,k}q}\right) b_{p,k}, \qquad a_{p,k} = \mathrm{softplus}(\tilde{a}_{p,k}) + \epsilon. \tag{6}$$

Here, $a_{p,k}$ controls the response rate to accumulated contrast, while $b_{p,k}$ defines the corresponding target state. Multiple channels therefore span different temporal and contrast-response scales.

The central property of this flow is compositional consistency with respect to contrast mass.

Proposition 1 (Contrast refinement consistency)**.** *For any* $q_1, q_2 \geq 0$*,*

$$\Phi^{p,k}_{q_2}\left(\Phi^{p,k}_{q_1}(h)\right) = \Phi^{p,k}_{q_1+q_2}(h). \tag{7}$$

Therefore, a fixed amount of accumulated contrast produces the same local state whether it arrives as one coarse contrast increment or is partitioned into multiple finer event quanta. This makes the state evolution invariant to event refinement induced purely by the contrast threshold. The proof follows directly from the exponential form of Eq. (6) and is given in Appendix A.

The contrast-mass construction is also consistent with the underlying sensing process. For a monotone accumulated contrast Q and event count N_C,

$$|CN_C - Q| < C, \tag{8}$$

so contrast mass remains tied to the physical log-intensity change up to one threshold quantum. The derivation is provided in Appendix B.

3.3 Pixel-local temporal integration

Threshold changes affect not only event multiplicity but also event timing and the merged ordering of events from different pixels. A global recurrent state can therefore change when threshold variation alters cross-pixel interleaving. PCF avoids this dependence by evolving each $h_{t,p,k}(\mathbf{x})$ independently during temporal integration, with no convolution, pooling, attention, or other cross-pixel interaction inside the flow loop. Since updates at distinct pixels operate on disjoint state coordinates, they commute before spatial mixing. A formal two-pixel analysis of this ordering property is provided in Appendix C.

After processing T temporal bins, PCF summarizes each local trajectory using three complementary components:

$$H^{\mathrm{final}}_{p,k}(\mathbf{x}) = h_{T,p,k}(\mathbf{x}), \tag{9}$$

$$H^{\mathrm{avg}}_{p,k}(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} h_{t,p,k}(\mathbf{x}), \tag{10}$$

$$R_p(\mathbf{x}) = \log\left(1 + \sum_{t=1}^{T} \bar{q}_{t,p}(\mathbf{x})\right). \tag{11}$$

Here, H^{final} captures the endpoint of the local contrast-driven trajectory, H^{avg} preserves its temporal evolution, and R_p retains the accumulated contrast magnitude. The resulting local representation is

$$Z(\mathbf{x}) = \mathrm{Concat}_{p,k}\left[H^{\mathrm{final}}_{p,k}(\mathbf{x}), H^{\mathrm{avg}}_{p,k}(\mathbf{x}), R_p(\mathbf{x})\right]. \tag{12}$$

Only after the local temporal representation has been formed do we introduce spatial interaction. A compact spatial encoder aggregates object-level structure,

$$z = f_{\mathrm{spatial}}(Z), \tag{13}$$

and the final embedding is obtained by combining average- and max-pooled features,

$$e = \mathrm{MLP}\left([\mathrm{GAP}(z), \mathrm{GMP}(z)]\right). \tag{14}$$

The embedding e is used for classification and paired representation-drift analysis. This ordering separates threshold-sensitive temporal integration from subsequent spatial feature extraction. Additional architectural and implementation details are provided in Appendix D.

Table 1: Direct mechanism diagnostics for the algebraic refinement and cross-pixel ordering properties of PCF

Diagnostic	Observed value
Semigroup numerical error	$\approx 10^{-7}$
Mechanical PCF refinement error	0
Re-simulation vs. mechanical split, event-stream L1	553.0
Global-flow interleaving error	1.647677×10^{-2}
Pixel-local interleaving error	1.110223×10^{-16}

4 EXPERIMENTS

4.1 EXPERIMENTAL PROTOCOL

A threshold-robustness experiment is meaningful only when different-threshold streams correspond to the same latent visual sample. We therefore generate one continuous log-intensity trajectory for each source image and trigger events independently at each tested threshold, rather than mechanically duplicating or deleting events. Unless otherwise stated, we evaluate

$$C/C_0 \in \{0.25, 0.5, 1, 2, 4\}. \tag{15}$$

All threshold variants share the same image identity, class label, motion trajectory, temporal window, data split, and random seed.

Our evaluation has two levels. E1/E2 use a lightweight geometric simulator to isolate the proposed mechanisms. The main benchmark uses Caltech101 source images with fixed motion trajectories and realistic event generation, following the same principle as N-Caltech101: static images are converted into event streams by controlled camera motion (Orchard et al., 2015). This paired construction enables the same latent motion to be evaluated under different contrast thresholds.

Events are accumulated into T shared temporal bins.

Baselines. Count-GRU uses $\log(1+N)$ and ignores C. Mass-GRU replaces raw counts with the physical-mass input $\log(1+NC/C_0)$. Conditioned-GRU appends $\log(C/C_0)$ to each temporal feature. Count-GRU+Aug is trained on $\{0.5C_0, C_0, 2C_0\}$, while all other methods are trained only at the nominal threshold C_0. Global ContrastFlow uses contrast mass with a shared global state, whereas PCF uses pixel-local contrast-flow states before spatial mixing. All matched models use the same split and validation policy.

Metrics. We report accuracy at each threshold, nominal C_0 accuracy, mean OOD accuracy over $C \neq C_0$, worst OOD accuracy, and paired feature drift. For paired embeddings $z_i^{(r)}$ at threshold ratio r,

$$\mathrm{RelL2}(r) = \frac{1}{n}\sum_i \frac{\left\| z_i^{(r)} - z_i^{(1)} \right\|_2}{\left\| z_i^{(1)} \right\|_2 + \epsilon}. \tag{16}$$

Accuracy measures task performance, while RelL2 quantifies representation stability across threshold changes.

The robustness study further evaluates global threshold extrapolation, polarity-specific threshold asymmetry, and spatially varying pixel-level threshold mismatch using the same trained checkpoints as the main benchmark.

4.2 E1: DIRECT MECHANISM DIAGNOSTICS

E1 directly evaluates the two structural properties independently of model training. Table 1 summarizes the completed diagnostics. The semigroup test gives only floating-point-scale error. More importantly, a two-pixel example changes the merged event order after threshold refinement: the global flow changes by 1.65×10^{-2}, whereas the pixel-local update changes only at numerical precision. At the same time, the L1 difference between a physically re-simulated stream and a

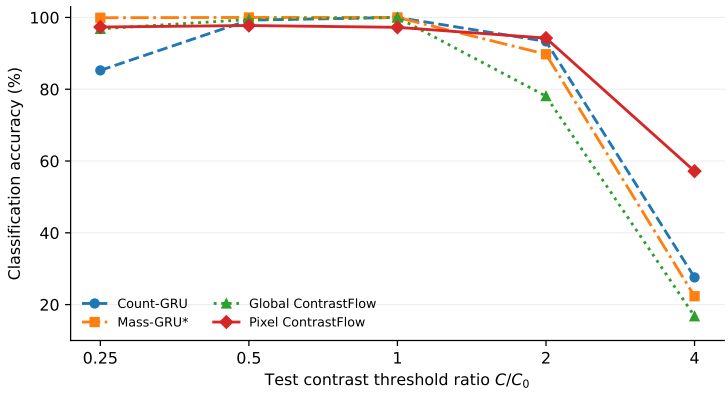


Figure 3: Synthetic E2 threshold sweep. This controlled diagnostic isolates threshold-induced event multiplicity changes and shows that Pixel ContrastFlow degrades more gradually than count-based and global-flow baselines at coarse unseen thresholds.

Table 2: Synthetic E2 threshold-sweep results. Accuracy is reported in %. Mean OOD averages $0.25C_0, 0.5C_0, 2C_0$, and $4C_0$.

Method	$0.25C_0$	$0.5C_0$	C_0	$2C_0$	$4C_0$	Mean OOD
Count-GRU	85.25	99.25	**100.00**	93.33	27.58	76.35
Mass-GRU†	99.92	**100.00**	**100.00**	89.75	22.33	78.00
Global ContrastFlow	96.83	99.42	**100.00**	78.08	16.75	72.77
PCF	**97.33**	97.75	97.25	**94.25**	**57.17**	**86.63**

mechanically split stream is 553, confirming that threshold variation cannot be reduced to simple event duplication.

4.3 E2: TRAIN AT C_0, TEST AT UNSEEN THRESHOLDS

We train all lightweight classifiers at C_0 for 30 epochs using 6000/1200/1200 training/validation/test samples and evaluate them across the five threshold settings. Figure 3 and Table 2 show the results.

The result exposes two useful points. First, simply using a flow globally is not enough: Global ContrastFlow is the weakest method on mean OOD accuracy, consistent with the interleaving diagnostic. Second, Pixel ContrastFlow gives up 2.75 percentage points at C_0 in this run but is substantially more stable as the threshold becomes coarse. The $4C_0$ gap is especially large.

4.4 E3: MAIN REALISTIC BENCHMARK

The main benchmark uses Caltech101 source images to construct paired event streams under controlled camera motion. For each source image, we first generate one fixed latent log-intensity trajectory and then trigger events independently at $C/C_0 \in \{0.25, 0.5, 1, 2, 4\}$. All threshold variants share the same image identity, class label, motion trajectory, temporal window, and data split. This paired design ensures that the evaluation measures threshold robustness rather than differences in sampled visual content.

Models are trained at the nominal threshold C_0, except for Count-GRU+Aug, which observes $\{0.5C_0, C_0, 2C_0\}$ during training. All methods are selected using validation accuracy at C_0 and then evaluated on the full threshold sweep without further tuning. We report accuracy at each threshold, Mean OOD accuracy over the four non-nominal thresholds, Worst OOD accuracy, and paired feature drift. As shown in Table 3, this benchmark tests whether a method can preserve discriminative information when the same visual trajectory is quantized by different contrast thresholds. We evaluate whether the representation preserves nominal performance at C_0, improves robustness to unseen

Table 3: Main Paired-Caltech101-CT benchmark. Accuracy is reported in %. Mean OOD averages the four non-nominal thresholds, and Worst OOD denotes the minimum accuracy among them. Best and second-best values are denoted by boldface and underlining.

Method	$0.25C_0$	$0.5C_0$	C_0	$2C_0$	$4C_0$	Mean OOD	Worst OOD	RelL2↓
Count-GRU	36.67	44.29	48.87	34.09	7.51	30.64	7.51	0.910
Mass-GRU	40.42	45.15	48.76	35.70	11.07	33.06	11.07	<u>0.691</u>
Conditioned-GRU	23.83	39.74	48.14	34.81	6.95	26.33	6.95	0.968
Count-GRU+Aug	<u>43.96</u>	<u>47.01</u>	47.09	**45.48**	<u>18.97</u>	**38.86**	<u>18.97</u>	0.738
Global ContrastFlow	42.69	43.27	**50.97**	33.12	10.74	32.46	10.74	0.982
PCF	**44.10**	**49.51**	<u>49.59</u>	<u>39.98</u>	**19.06**	<u>38.16</u>	**19.06**	**0.215**

thresholds, and reduces paired representation drift. Mass-GRU separates the effect of physical-mass normalization from the additional effect of pixel-local contrast-flow integration.

4.5 E4: Robustness and ablations

We further evaluate robustness under three sensor-variation families. **Global threshold shift** tests extrapolation beyond the training threshold. **Polarity asymmetry** sets different ON/OFF thresholds $C_+ \neq C_-$. **Pixel mismatch** samples spatially varying thresholds around the nominal bias. These variations are chosen because they directly perturb the assumptions behind contrast-mass normalization and pixel-local integration.

The ablation study follows the method design. Starting from Count-GRU, we compare physical mass normalization, threshold conditioning, global contrast flow, and the proposed pixel-local contrast flow. This separates the benefit of knowing the contrast threshold from the benefit of enforcing refinement-consistent local state composition. We also evaluate paired feature drift to verify whether improved accuracy is accompanied by more stable representations across threshold changes.

4.6 E5: Sensitivity to Threshold-Estimation Error

PCF uses the sensor contrast threshold to convert event multiplicity into physical contrast mass. We therefore evaluate its sensitivity to imperfect calibration by keeping the generated event stream fixed at the true threshold C while perturbing only the threshold estimate supplied to the representation, $\hat{C} = \alpha C$. Figure 4 shows that PCF remains stable under moderate estimation errors. For $\hat{C}/C \in [0.8, 1.2]$, accuracy stays within 0.41 percentage points of exact calibration, while calibration-induced feature drift remains below 0.05 RelL2. Even at $\pm 30\%$ error, the maximum accuracy degradation is 0.72 percentage points.

The representation drift increases as the threshold estimate departs from the true value, whereas classification accuracy remains nearly unchanged over the moderate-error range. This suggests that small calibration errors perturb the embedding without substantially changing the resulting decision. At the tested extremes, larger errors produce more pronounced representation drift, with threshold underestimation causing the larger accuracy loss.

4.7 Discussion

PCF reframes threshold variation as a representation problem rather than a data-augmentation problem. By converting event multiplicity into contrast mass and integrating it through pixel-local semigroup dynamics, the representation becomes less dependent on how a fixed visual change is discretized into events. The paired threshold experiments support this interpretation: PCF improves robustness at unseen thresholds and reduces representation drift, while the comparison with Mass-GRU and Global ContrastFlow shows that physical normalization alone is insufficient without the proposed local dynamical structure.

The current formulation assumes access to an estimate of the effective contrast threshold. Our calibration study shows that PCF remains stable under moderate estimation errors, suggesting that exact calibration is not required in practice. A remaining direction is to extend PCF to sensors

Table 4: Detailed robustness results on realistic paired streams. Values report classification accuracy (%). Global Mean and Global Worst are computed over the six non-nominal threshold shifts, excluding C_0. Polarity and Pixel report mean accuracy within each robustness family. Best and second-best values are denoted by boldface and underlining.

Method	$0.125C_0$	$0.25C_0$	$0.5C_0$	C_0	$2C_0$	$4C_0$	$8C_0$	Global Mean	Global Worst	Polarity	Pixel mismatch
Count-GRU	30.94	36.67	44.29	48.87	34.09	7.51	1.45	25.83	1.45	30.63	48.57
Mass-GRU	37.56	40.42	45.15	48.76	35.70	11.07	2.58	28.75	2.58	31.39	48.75
Count-GRU+Aug	<u>41.43</u>	<u>43.96</u>	<u>47.01</u>	47.09	**45.48**	18.97	5.41	<u>33.71</u>	5.41	<u>33.89</u>	47.44
Global ContrastFlow	42.51	42.69	43.27	**50.97**	33.12	10.74	1.70	29.01	1.70	33.80	**49.82**
PCF	**42.97**	**44.10**	**49.51**	<u>49.59</u>	<u>39.98</u>	<u>19.06</u>	**6.79**	**33.74**	**6.79**	**44.00**	<u>49.65</u>

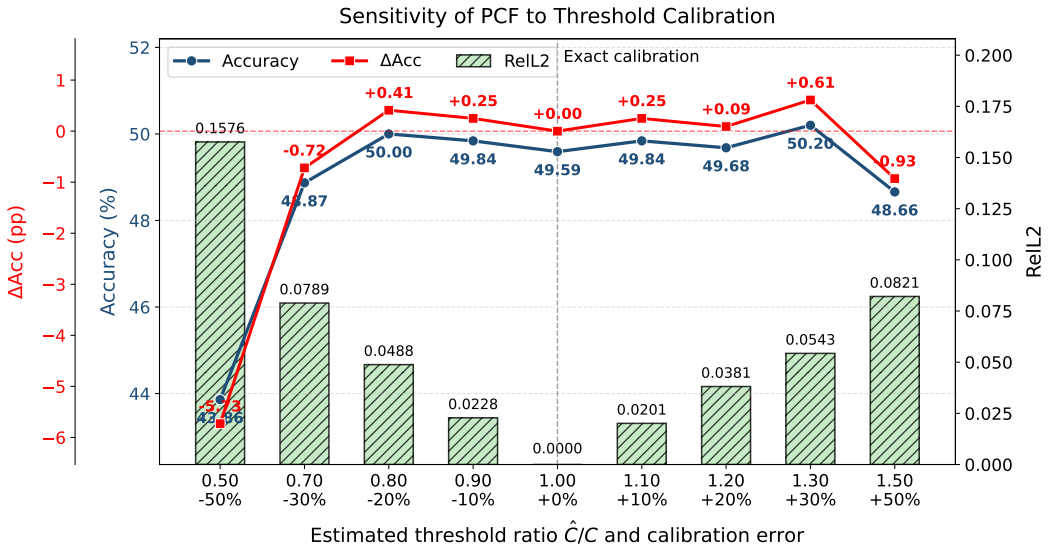


Figure 4: Sensitivity of PCF to threshold-estimation error. Accuracy and ΔAcc are shown as line plots, while RelL2 denotes the representation drift relative to exact calibration ($\hat{C}/C = 1$). PCF maintains nearly unchanged classification accuracy under moderate calibration errors, while representation drift increases as the estimated threshold deviates from the true value.

with stronger spatial, temporal, and polarity-dependent threshold variation and to jointly estimate threshold fields from data. More broadly, the contrast-flow formulation may provide a useful basis for threshold-consistent event representations beyond classification, including detection, tracking, and other event-based perception tasks.

5 Conclusion

This paper proposed Pixel Contrast Flow, a threshold-consistent event representation framework that treats physical contrast mass as the primitive quantity for temporal state evolution. Pixel Contrast Flow combines contrast-mass encoding, a semigroup contrast flow, and pixel-local temporal integration before spatial mixing. The resulting update is consistent when the same contrast is split into finer event quanta, while pixel-local states avoid unnecessary sensitivity to cross-pixel event reordering. Experiments on the paired Caltech101-CT benchmark show that this structural bias improves both robustness and representation stability. On the paired Caltech101-CT benchmark, PCF maintains a nominal accuracy of 49.59% while achieving the best accuracy at $0.25C_0$, $0.5C_0$, and $4C_0$, the highest worst-case OOD accuracy of 19.06%, and the lowest paired feature drift with a RelL2 of 0.215. In the detailed robustness study, Pixel Contrast Flow further achieves the best Global Mean, Global Worst, and polarity-asymmetry accuracy, while remaining competitive under pixel mismatch. These results suggest that preserving the physical measure behind event tokens is an effective principle for threshold-robust event representation learning.

Reproducibility Statement

The released code will include the paired Caltech101-CT trajectory generator, event binning pipeline, model implementations, training scripts, and evaluation scripts. The implemented models include Count-GRU, Mass-GRU, Conditioned-GRU, Count-GRU+Aug, Global ContrastFlow, and PCF. The evaluation code reports per-threshold accuracy, Mean OOD accuracy, Worst OOD accuracy, paired-feature RelL2 drift, polarity-asymmetry robustness, and pixel-mismatch robustness.

For reproducibility, we will release the dataset split identifiers, trajectory seeds, threshold settings, training hyperparameters, checkpoint-selection rule, per-seed result files, and scripts used to generate the tables. All main results are selected using validation accuracy at C_0 only, and the test thresholds are never used for model selection.

AI Use Disclosure

Generative AI tools were used to assist with language polishing of the manuscript. All text generated with AI assistance was independently checked by the authors. The authors take full responsibility for the final contents of the submission.

References

Yin Bi, Aaron Chadha, Alhabib Abbas, Eirina Bourtsoulatze, and Yiannis Andreopoulos. Graph-based object classification for neuromorphic vision sensing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 491–501, 2019.

Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. A 240 $\times$ 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits*, 49(10):2333–2341, 2014. doi: 10.1109/JSSC.2014.2342715.

Marco Cannici, Marco Ciccone, Andrea Romanoni, and Matteo Matteucci. A differentiable recurrent surface for asynchronous event-based data. In *Computer Vision – ECCV 2020*, 2020.

Guillermo Gallego, Tobi Delbruck, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J. Davison, Jorg Conradt, Kostas Daniilidis, and Davide Scaramuzza. Event-based vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):154–180, 2022. doi: 10.1109/TPAMI.2020.3008413.

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.

Fuqiang Gu, Weicong Sng, Xuke Hu, and Fangwen Yu. Eventdrop: Data augmentation for event-based learning. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pp. 700–707, 2021. doi: 10.24963/ijcai.2021/97.

Yuhuang Hu, Shih-Chii Liu, and Tobi Delbruck. v2e: From video frames to realistic dvs events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1312–1321, 2021. doi: 10.1109/CVPRW53098.2021.00144.

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. doi: 10.1109/TPAMI.2016.2574707.

Yuhang Li, Youngeun Kim, Hyoungseob Park, Tamar Geller, and Priyadarshini Panda. Neuromorphic data augmentation for training spiking neural networks. In *Computer Vision – ECCV 2022*, pp. 631–649. Springer, 2022. doi: 10.1007/978-3-031-20071-7_37.

Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128$\times$128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, 2008. doi: 10.1109/JSSC.2007.914337.

Weng Fei Low and Gim Hee Lee. Robust e-nerf: Nerf from sparse & noisy events under non-uniform motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18335–18346, 2023.

Nico Messikommer, Daniel Gehrig, Antonio Loquercio, and Davide Scaramuzza. Event-based asynchronous sparse convolutional networks. In *Computer Vision – ECCV 2020*, 2020. doi: 10.1007/978-3-030-58598-3_25.

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. doi: 10.3389/fnins.2015.00437.

Christoph Posch, Teresa Serrano-Gotarredona, Bernabe Linares-Barranco, and Tobi Delbruck. Retinomorphic event-based vision sensors: Bioinspired cameras with spiking output. *Proceedings of the IEEE*, 102(10):1470–1484, 2014. doi: 10.1109/JPROC.2014.2346153.

Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. Esim: an open event camera simulator. In *Proceedings of the 2nd Conference on Robot Learning*, volume 87 of *Proceedings of Machine Learning Research*, pp. 969–982, 2018.

Teresa Serrano-Gotarredona and Bernabe Linares-Barranco. A 128 $\times$ 128 1.5% contrast sensitivity 0.9% fpn 3 μs latency 4 mw asynchronous frame-free dynamic vision sensor using transimpedance preamplifiers. *IEEE Journal of Solid-State Circuits*, 48(3):827–838, 2013. doi: 10.1109/JSSC.2012.2230553.

Guobin Shen, Dongcheng Zhao, and Yi Zeng. Eventmix: An efficient data augmentation strategy for event-based learning. *Information Sciences*, 644:119170, 2023. doi: 10.1016/j.ins.2023.119170.

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.

Mingyuan Sun, Donghao Zhang, Zongyuan Ge, Jiaxu Wang, Jia Li, Zheng Fang, and Renjing Xu. Eventrpg: Event data augmentation with relevance propagation guidance. In *International Conference on Learning Representations*, 2024.

Ziwei Wang, Yonhon Ng, Pieter van Goor, and Robert Mahony. Event camera calibration of per-pixel biased contrast threshold. *arXiv preprint arXiv:2012.09378*, 2020.

A Proof of Contrast Refinement Consistency

The contrast flow for one pixel, polarity, and state channel is

$$\Phi_q(h) = e^{-aq}h + \left(1 - e^{-aq}\right) b = b + e^{-aq}(h - b), \tag{17}$$

where $a > 0$ and $q \geq 0$ denotes accumulated contrast mass.

Contrast refinement consistency. For any $q_1, q_2 \geq 0$,

$$\begin{aligned}
\Phi_{q_2}(\Phi_{q_1}(h)) &= b + e^{-aq_2}\left[\Phi_{q_1}(h) - b\right] && (18)\\
&= b + e^{-aq_2}e^{-aq_1}(h - b) && (19)\\
&= b + e^{-a(q_1+q_2)}(h - b) && (20)\\
&= \Phi_{q_1+q_2}(h). && (21)
\end{aligned}$$

By repeated application, for any partition $q = \sum_{j=1}^{m} q_j$,

$$\Phi_{q_m} \circ \cdots \circ \Phi_{q_1}(h) = \Phi_q(h). \tag{22}$$

Therefore, within a shared temporal bin, the local flow depends on the total contrast mass rather than on its partition into individual contrast quanta. This is the refinement-consistency property used by PCF.

State contraction. The same flow is contractive with respect to its incoming state. For two states h and h' driven by the same contrast mass q,

$$|\Phi_q(h) - \Phi_q(h')| = \left|e^{-aq}(h - h')\right| \tag{23}$$

$$= e^{-aq} \left|h - h'\right|. \tag{24}$$

Since $a > 0$ and $q \geq 0$, the flow is non-expansive, and is strictly contractive whenever $q > 0$.

Including the learned temporal relaxation

$$\widetilde{h}_t = dh_{t-1}, \qquad 0 < d < 1, \tag{25}$$

two trajectories driven by the same contrast sequence $\{q_t\}_{t=1}^{T}$ satisfy

$$|h_T - h'_T| = d^T \exp\left(-a \sum_{t=1}^{T} q_t\right) |h_0 - h'_0|. \tag{26}$$

Thus, differences in the incoming state are not amplified by the local contrast-flow dynamics.

Stability to contrast-mass perturbation. The flow also varies smoothly when the supplied contrast mass is perturbed. For $q, q' \geq 0$,

$$|\Phi_q(h) - \Phi_{q'}(h)| = \left|e^{-aq} - e^{-aq'}\right| |h - b|. \tag{27}$$

Applying the mean-value theorem to e^{-aq} gives

$$\left|e^{-aq} - e^{-aq'}\right| \leq ae^{-a\min(q,q')} |q - q'|, \tag{28}$$

and therefore

$$|\Phi_q(h) - \Phi_{q'}(h)| \leq ae^{-a\min(q,q')} |q - q'| |h - b|. \tag{29}$$

This bound directly characterizes threshold-estimation error. If the event stream is fixed while the estimated threshold satisfies $\hat{C} = \alpha C$, then the contrast mass supplied to PCF changes from q to $\hat{q} = \alpha q$. Equation (29) gives

$$|\Phi_{\hat{q}}(h) - \Phi_q(h)| \leq aqe^{-aq\min(1,\alpha)} |\alpha - 1| |h - b|. \tag{30}$$

Hence, for a fixed observed stream and incoming state, the induced local-state perturbation is bounded continuously with respect to the relative threshold estimation error. This provides the local stability counterpart to the calibration experiment in Section 4.6.

B Proof of the Contrast-Mass Quantization Bound

Consider a monotone log-intensity interval with accumulated physical contrast Q and contrast threshold C. Let $r_0 \in [0, C)$ denote the residual contrast relative to the current event reference at the beginning of the interval. After an additional contrast change Q, the number of emitted events is

$$N_C = \left\lfloor \frac{r_0 + Q}{C} \right\rfloor. \tag{31}$$

The remaining residual after these events is

$$r_1 = r_0 + Q - CN_C, \qquad r_1 \in [0, C). \tag{32}$$

Rearranging gives

$$CN_C - Q = r_0 - r_1. \tag{33}$$

Because both residuals lie in $[0, C)$,

$$|CN_C - Q| < C. \tag{34}$$

Thus, although the raw event count N_C scales approximately as Q/C, the contrast mass CN_C tracks the underlying physical contrast up to one threshold quantum.

For two thresholds C and C', applying Eq. (34) separately gives

$$|CN_C - C'N_{C'}| \le |CN_C - Q| + |C'N_{C'} - Q| \tag{35}$$
$$< C + C'. \tag{36}$$

PCF uses the normalized contrast mass

$$\bar{q}_C = \frac{C}{C_0} N_C. \tag{37}$$

Dividing Eq. (34) by C_0 yields

$$\left| \bar{q}_C - \frac{Q}{C_0} \right| < \frac{C}{C_0}. \tag{38}$$

Similarly, for two thresholds,

$$|\bar{q}_C - \bar{q}_{C'}| < \frac{C + C'}{C_0}. \tag{39}$$

These bounds formalize the distinction between event count and contrast mass: the former changes inversely with the threshold, whereas the latter remains anchored to the same physical log-intensity change, with an error determined by the finite threshold quantum.

For trajectories containing both polarities, the argument applies independently to each monotone positive or negative contrast segment. The same derivation also applies locally when the effective threshold is pixel- or polarity-dependent by replacing C with $C_p(\mathbf{x})$.

C Cross-Pixel Interleaving

Threshold variation can change not only event multiplicity but also the merged ordering of events generated at different pixels. This distinction matters for representations that update a shared recurrent state.

Consider first two pixels A and B. For a global hidden state with pixel-dependent update maps J_A and J_B,

$$J_A \circ J_B \neq J_B \circ J_A \tag{40}$$

in general. Consequently, two event streams generated from the same underlying visual trajectory can produce different global hidden states when a threshold change modifies their cross-pixel interleaving.

PCF instead associates independent state coordinates with different pixels. Let U_A and U_B denote the corresponding local update operators:

$$U_A(h_A, h_B) = (J_A(h_A), h_B), \tag{41}$$
$$U_B(h_A, h_B) = (h_A, J_B(h_B)). \tag{42}$$

Since they operate on disjoint coordinates,

$$U_A \circ U_B(h_A, h_B) = (J_A(h_A), J_B(h_B)) \tag{43}$$
$$= U_B \circ U_A(h_A, h_B). \tag{44}$$

General cross-pixel permutation invariance. The same property extends to an arbitrary number of pixels. Let

$$\mathcal{H} = \prod_{\mathbf{x}} \mathcal{H}_{\mathbf{x}} \tag{45}$$

denote the joint local-state space, and let $U_{\mathbf{x}}$ modify only the state block associated with pixel $\mathbf{x}$. For any two distinct pixels $\mathbf{x} \neq \mathbf{y}$,

$$U_{\mathbf{x}} U_{\mathbf{y}} = U_{\mathbf{y}} U_{\mathbf{x}}. \tag{46}$$

Any permutation of cross-pixel updates can be decomposed into adjacent swaps. Because every swap between updates from distinct pixels leaves the joint state unchanged, any two merged event orders that preserve the within-pixel update order produce the same local state field before spatial mixing.

Combined refinement and interleaving consistency. The semigroup property in Appendix A and the commutativity above can be combined into a stronger statement. Consider one shared temporal bin, and suppose the total contrast mass $q_p(\mathbf{x})$ accumulated at every pixel and polarity is fixed. The same mass may be partitioned into arbitrary nonnegative increments,

$$q_p(\mathbf{x}) = \sum_{j=1}^{m_{\mathbf{x},p}} q_{j,p}(\mathbf{x}), \tag{47}$$

and these increments may be interleaved arbitrarily across different pixels.

Under the same bin-level temporal relaxation, the resulting PCF state field is unchanged by either operation. First, contrast refinement consistency gives

$$\Phi_{q_m} \circ \cdots \circ \Phi_{q_1} = \Phi_{\sum_j q_j} \tag{48}$$

for every local state coordinate. Second, updates associated with distinct pixels commute by Eq. (46). Therefore, after all local contrast increments in the bin have been processed, the state field depends only on the accumulated local contrast masses $\{q_p(\mathbf{x})\}$ and not on their within-bin partition or their cross-pixel interleaving.

This result identifies the two complementary roles of PCF. The semigroup flow removes sensitivity to how a fixed local contrast is discretized into event quanta, while pixel-local integration removes sensitivity to how updates from different pixels are merged before spatial processing.

D Additional PCF Details

Multi-scale state channels. PCF maintains multiple state channels for each pixel and polarity. The learned decay coefficient

$$d_{p,k} = \exp\left[-\mathrm{softplus}(\gamma_{p,k})\right] \tag{49}$$

controls temporal memory, while the contrast-flow coefficient $a_{p,k}$ controls the response scale along the contrast coordinate. The flow rates are initialized on a logarithmic scale to span slow and fast contrast-response regimes.

Local temporal summaries. After T temporal bins, PCF retains the final state,

$$H_{p,k}^{\mathrm{final}}(\mathbf{x}) = h_{T,p,k}(\mathbf{x}), \tag{50}$$

the time-averaged state,

$$H_{p,k}^{\mathrm{avg}}(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} h_{t,p,k}(\mathbf{x}), \tag{51}$$

and the accumulated contrast-mass residual,

$$R_p(\mathbf{x}) = \log\left(1 + \sum_{t=1}^{T} \bar{q}_{t,p}(\mathbf{x})\right). \tag{52}$$

These fields are concatenated before the spatial readout.

Spatial readout. The local representation is processed by a compact residual CNN. Average- and max-pooled spatial features are concatenated and projected by an MLP to obtain the final embedding used for classification and paired feature-drift evaluation.

Mass-GRU baseline. Mass-GRU uses the same temporal encoder as Count-GRU while replacing raw counts with

$$u_{t,p}(\mathbf{x}) = \log\left(1 + \bar{q}_{t,p}(\mathbf{x})\right). \tag{53}$$

This isolates physical contrast-mass normalization from the contrast-flow and pixel-local components of PCF.