THE AVERAGING PROBLEM IN ON-CHIP LOCAL LEARNING FOR DENSE PREDICTION: MOST OF THE GAP TO BACKPROPAGATION IS MODULATION GRANULARITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Layer-local three-factor rules train spiking networks within the constraints of on-chip learning. On classification, their gap to backpropagation is attributed to the credit that cannot cross layers. On dense prediction, where none of these rules has been evaluated, we find a larger term in that gap. Every published rule forms its modulatory signal through a class-level readout that averages the per-pixel error into one vector per image. The backbone trained this way ends within one point of a frozen random backbone. Block-averaging the per-pixel error is an orthogonal projection, and a coarse rule discards the residual, which concentrates at the target's boundaries. When only the averaging operator is varied, accuracy rises monotonically with granularity on two datasets and under two local rules. Granularity is the larger term at every depth within the chip's budget. At three layers, it is 78% of the distance from uniform modulation to backpropagation and the larger term on every seed. Letting the gradient cross layers recovers little, and so does a partition aligned to the ground-truth classes. The loss therefore comes from averaging within a region rather than from missing label information. Blocks of 16×16 pixels recover 67–77% of the dense gain at 0.4% of the modulation state. Per-pixel modulation is a per-layer dense readout under the on-chip constraints, and it comes within 2.7 points of same-architecture backpropagation.

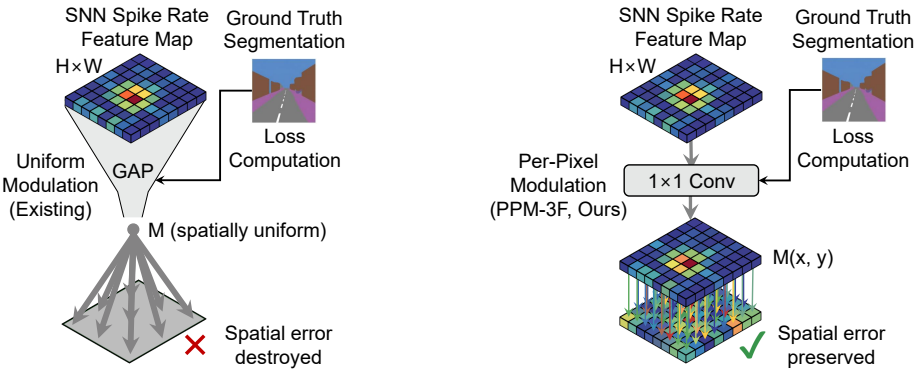
1 INTRODUCTION

Neuromorphic processors (Davies et al., 2018; Orchard et al., 2021; Furber et al., 2014; Akopyan et al., 2015; Pei et al., 2019) promise low-power inference for event cameras (Gallego et al., 2022), but they cannot run backpropagation on-chip. Their on-chip memory is small (128 cores with 192 KB each, about 24 MB, on Intel Loihi 2 (Orchard et al., 2021)), and they have no cross-layer backward pass. Both preclude backpropagation through time (BPTT), which needs $O(L\times T\times N)$ activation memory for L layers run for T timesteps with N neurons per layer. The usual practice is to train on a graphics processing unit (GPU) and deploy the network frozen, so the device cannot learn once deployed. A rule that learns on the chip must therefore be layer-local, keep $O(N)$ memory, and read only its own synaptic weights (the three on-chip constraints of Section 3.1).

Three-factor local learning rules train spiking neural networks (SNNs) under these constraints, yet they have only been validated on image-level outputs (classification and single-object localization (Barchid et al., 2021)). These rules (Frémaux & Gerstner, 2016; Kaiser et al., 2020; Apolinario et al., 2025) update each layer independently from local information with only $O(N)$ memory, and they achieve competitive accuracy for *classification* (Apolinario et al., 2025; Quintana et al., 2024). The gap between local rules and backpropagation is usually attributed to the credit that cannot cross layers (Lillicrap et al., 2020). The rules published since, TESS (Apolinario et al., 2025), ETLP (Quintana et al., 2024), and S-TLLR (Apolinario & Roy, 2025), add temporal credit assignment or a substitute for it. TESS and ETLP keep a class-level readout, and S-TLLR backpropagates the output error across layers. Classification is the favorable case for local learning, because one label per image makes a collapsed error signal lossless.

For *per-pixel* prediction, a collapsed error signal is lossy, because neighboring positions need different updates. Every published layer-local three-factor rule (Kaiser et al., 2020; Apolinario et al., 2025; Quintana et al., 2024) forms its modulatory signal (Frémaux & Gerstner, 2016) from a spatially

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.



(a) Uniform modulation: the averaging problem.

(b) Per-pixel modulation: no averaging.

Figure 1: The two extremes of the granularity range. (a) A class-level readout collapses the spike-rate field by global average pooling (GAP) into one shared vector M broadcast to every position. (b) A per-pixel readout gives each position its own $M(x, y)$.

collapsed error. The signal is one vector for the whole image or, with a random readout, a field whose rank equals the number of classes (Section A). For a dense target, this collapse averages the errors of every position into one value per class, the class-share error of Section 3.2. The weights learn which classes are over- or under-predicted, never where, so positions that need different updates receive the same one (Figure 1). We call this the *averaging problem*. Between one modulatory signal per image and one per pixel lies a range of granularities. We span this range with a sweep of variants, one per granularity, and ask how much the averaging loses and what sets it.

Our claim is that *the granularity of the modulatory signal sets most of what layer-local learning loses on dense prediction under the on-chip constraints*, rather than the credit that cannot cross layers. We develop it in three steps.

1. **Diagnosis** (Section 3.3). We model modulation granularity as a partition of the image. Block-averaging is then an orthogonal projection, and the update a coarse rule loses is the projection's residual mapped through the layer's kernels. The residual shrinks monotonically under refinement, Proposition 1 gives a sufficient condition for descent, and four hypotheses follow.
2. **Measurement** (Section 5). A controlled granularity sweep on two datasets and under two local rules tests the four hypotheses, and the measurements confirm each of them.
3. **Per-pixel modulation against backpropagation** (Section 3.4, Section 5.3, Section 6). In PPM-3F (per-pixel modulated three-factor learning), each layer forms its modulatory field from its own per-pixel readout. It is the dense local head of local learning in artificial neural networks (ANNs), placed inside the three on-chip constraints. It reaches 40.1% mean intersection over union (mIoU), 2.7 percentage points (pp) below same-architecture BPTT at three layers and a shared budget.

2 Related Work

This paper draws on two areas, on-chip learning rules for SNNs and dense prediction with local objectives. Five lines of work are relevant, and none varies the granularity of the learning signal.

SNN semantic segmentation. Spiking segmentation networks build on standard dense architectures (Shelhamer et al., 2017; Ronneberger et al., 2015; Chen et al., 2018), and all of them train off-chip. They train with surrogate-gradient BPTT (Neftci et al., 2019; Wu et al., 2018) at $O(L{\times}T{\times}N)$ memory and reach 45–55% mIoU on DDD17 from events alone (Spike-BRGNet Long et al., 2025a, SLTNet Long et al., 2025b, SpikingEDN Zhang et al., 2025, EvSegSNN Hareb & Martinet, 2024, and Spiking-SSegNet Ye et al., 2025). They take event input at $64{\times}64$ to $200{\times}346$ or larger, with 0.4–8.6M parameters. This paper trains the same task under the on-chip constraints, at the resolution and parameter budget of the chip.

Layer-local three-factor rules. DECOLLE (Kaiser et al., 2020), TESS (Apolinario et al., 2025), and ETLP (Quintana et al., 2024) use spatially collapsed modulation, while S-TLLR (Apolinario & Roy, 2025) obtains per-neuron signals by spatial backpropagation across layers (or a random projection of it). These rules descend from neuromodulated plasticity (Frémaux & Gerstner, 2016)

and e-prop (Bellec et al., 2020), which makes temporal credit local but still reads a task-level error. Online-through-time training (Xiao et al., 2022; Meng et al., 2023; Bohnstingl et al., 2023) removes the factor of T from memory but keeps the cross-layer backward pass and weight transport. This paper varies the modulatory signal's spatial granularity, the one property they all hold fixed.

Other local paradigms. Other local paradigms deliver per-neuron errors by mechanisms outside the on-chip constraint set. Predictive coding (Whittington & Bogacz, 2017), equilibrium propagation (Scellier & Bengio, 2017), target propagation (Lee et al., 2015b), and dendritic or burst-multiplexing microcircuits (Sacramento et al., 2018; Payeur et al., 2021) belong here. Each needs a cross-layer feedback pathway, symmetric or learned feedback weights, or several relaxation phases. None of them satisfies the three constraints of Section 3.1.

Local learning in ANNs. In ANNs, deep supervision (Lee et al., 2015a) adds companion losses to end-to-end training. Greedy layer-wise training (Belilovsky et al., 2019), local error signals (Nøkland & Eidnes, 2019), Forward-Forward (Hinton, 2022), and InfoPro (Wang et al., 2025) are alternatives to end-to-end backpropagation (Lillicrap et al., 2020). They use unconstrained local backpropagation inside each block or a per-layer goodness objective (Forward-Forward), with pooled heads on classification and, in InfoPro, dense heads on segmentation. None satisfies the three on-chip constraints.

Image-level supervision on dense tasks. Weakly supervised segmentation asks the converse question, a dense output from an image-level target. A classifier trained through global average pooling localizes the discriminative parts of each class (Zhou et al., 2016), and priors and learned pixel affinities recover extent and boundaries (Kolesnikov & Lampert, 2016; Ahn & Kwak, 2018). Here the target is dense and the rule collapses the signal, with the same effect: a modulatory signal constant over a class's region locates the class but cannot resolve its shape (Section 5.4).

3 The Averaging Problem

Before the loss from coarse modulation can be measured, the averaging behind it must be defined. This section formalizes granularity as a partition of the image and block-averaging as an orthogonal projection. Section 3.1 states the task and constraints, Section 3.2 shows the loss on a real scene, Section 3.3 gives the projection view and four hypotheses, and Section 3.4 defines PPM-3F.

3.1 Setting

Three constraints define on-chip learning in this paper, and every rule we compare must satisfy them. The task is event-based semantic segmentation. Given an event stream, a spiking network must predict the per-pixel label map $\mathbf{y}^* \in \{1, \dots, K\}^{H \times W}$, where K is the number of classes and $H \times W$ the resolution, and it must be trainable and adaptable *on-chip*. Neuromorphic processors (e.g., Loihi 2 (Davies et al., 2018; Orchard et al., 2021)) impose three constraints.

1. **Layer-local updates.** No bus carries cross-layer gradients.
2. $O(N)$ **memory.** The on-core memory holds one layer's activations, not the computation graph.
3. **No weight transport.** A core sees only its own weights.

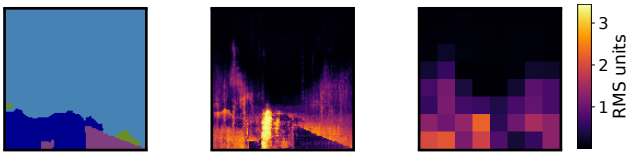
Three-factor learning (Frémaux & Gerstner, 2016; Kaiser et al., 2020) satisfies all three. Layer ℓ outputs a spike-rate field $\mathbf{r}^{(\ell)} \in \mathbb{R}^{C \times H \times W}$ with C channels, and each of its synapses w_{ij} updates as

$$\Delta w_{ij}^{(\ell)} = \eta \cdot \underbrace{s_j^{\text{pre}}}_{\text{Factor 1}} \cdot \underbrace{s_i^{\text{post}}}_{\text{Factor 2}} \cdot \underbrace{M^{(\ell)}}_{\text{Factor 3}}, \tag{1}$$

with learning rate η, pre-/post-synaptic spike activities $s_j^{\text{pre}}, s_i^{\text{post}}$, and a modulatory signal $M^{(\ell)}$ derived from a layer-local loss $\mathcal{L}^{(\ell)}$.

3.2 What Uniform Modulation Discards

In every published layer-local three-factor rule, DECOLLE among them (Kaiser et al., 2020), $M^{(\ell)}$ is computed only *after* the spike-rate field $\mathbf{r}^{(\ell)}$ has been collapsed through a class-level readout. This



(a) Ground truth. (b) Per-pixel field. (c) 16×16 average.

Figure 2: The averaging problem on a real DDD17 scene (color: field magnitude in units of its root mean square). The per-pixel modulation field (b) is concentrated at the class boundaries of (a), and a 16×16 block average (c) keeps 63% of its energy.

collapse is the *averaging problem*. With global average pooling, it is an average over positions:

$$M^{(\ell)} = \frac{\partial \mathcal{L}^{(\ell)}}{\partial \bar{\mathbf{r}}^{(\ell)}} \in \mathbb{R}^C, \qquad \bar{\mathbf{r}}^{(\ell)} = \tfrac{1}{HW} \textstyle\sum_{x,y} \mathbf{r}^{(\ell)}(x,y). \tag{2}$$

Any class-level readout $f_{\text{cls}} : \mathbb{R}^{C\times H\times W} \to \mathbb{R}^K$ loses the same information, the fixed random projections of DECOLLE (Kaiser et al., 2020) included, because the $H\times W$ positions are reduced to K scores before the loss (a span of rank K, Section A). Let $\bar{\mathbf{y}}$ be the class shares of $\mathbf{y}^*$ and $\bar{\mathbf{p}}$ the mean predicted distribution, both computed with the pixel weights of the loss. The only error information the readout transmits is then $\bar{\mathbf{p}} - \bar{\mathbf{y}}$, the *class-share error*. The update of the kernel weights $\mathbf{w}$,

$$\nabla_{\mathbf{w}} \mathcal{L}_{\text{uniform}} = \tfrac{1}{HW}\, M^{(\ell)} \cdot \textstyle\sum_{x,y} \frac{\partial \mathbf{r}^{(\ell)}(x,y)}{\partial \mathbf{w}}, \tag{3}$$

shares one $M^{(\ell)}$ across all positions.

Figure 2 shows what the collapse discards on a real DDD17 scene: the per-position error signal, the modulation field defined in Section 3.3, is concentrated at the class boundaries (Figure 2b). A 16×16 block average keeps 63% of its energy (Figure 2c), and the scalar mean of Equation (2) keeps 10%.

3.3 Granularity as an Orthogonal Projection

Block-averaging the modulation field is an orthogonal projection, and this section's results follow from that fact. Fix a layer and drop the superscript ℓ. Let P be the set of the HW positions and $\mathcal{F} = (\mathbb{R}^C)^P$ the space of *fields*, one C-vector per position, with the entrywise inner product and norm. The *modulation field* $\mathbf{M} \in \mathcal{F}$ collects the per-position error signals $\mathbf{M}_p = \partial \mathcal{L}_p / \partial \mathbf{r}_p$ of a dense loss $\mathcal{L} = \sum_p \mathcal{L}_p$ with one term $\mathcal{L}_p$ per position. Uniform modulation is the special case in which every entry $\mathbf{M}_p$ is forced equal to one shared vector, the third factor $M^{(\ell)}$ of Equation (1).

A *granularity* is a partition Π of P into blocks. It ranges from the scalar partition $\Pi_{\text{scalar}} = \{P\}$, a single block, to the per-pixel partition Π_{pix} of singletons. We write $\Pi \preceq \Pi'$ when Π' refines Π, and A_Π for the averaging operator that replaces each block's values by the block mean.

A coarse rule applies the dense update to the averaged field. Let $\mathbf{J}_p = \partial \mathbf{r}_p / \partial \mathbf{w}$ be the Jacobian at position p, and let $\mathcal{J}$ be the linear map $\mathcal{J}\mathbf{V} = \sum_p \mathbf{J}_p^\top \mathbf{V}_p$ that turns a field into a weight update. The per-pixel update is $g_{\text{pix}} = \mathcal{J}\mathbf{M}$, and the three-factor update at granularity Π is

$$g_\Pi \;=\; \mathcal{J}\, \mathsf{A}_\Pi \mathbf{M}, \tag{4}$$

which is exactly the rule implemented by averaging the backward signal per block (Section 5.1). The scalar partition Π_{scalar} recovers Equation (3), up to the factor HW under squared loss and up to a second-order term for pool-then-classify cross-entropy (Section C).

Because A_Π is the orthogonal projector onto the subspace $\mathcal{F}_\Pi \subseteq \mathcal{F}$ of block-constant fields, the field splits into the part a coarse rule receives and the part it discards,

$$\|\mathbf{M}\|^2 = \|\mathsf{A}_\Pi \mathbf{M}\|^2 + D_\Pi, \qquad D_\Pi(\mathbf{w}) := \|(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}\|^2 = \sum_{B\in\Pi} \sum_{p\in B} \|\mathbf{M}_p - \bar{\mathbf{M}}_B\|^2, \tag{5}$$

with I the identity and $\bar{\mathbf{M}}_B$ the mean of $\mathbf{M}$ over the block B. We call D_Π the *discarded energy*. In the scene of Figure 2, it is 37% of $\|\mathbf{M}\|^2$ for 16×16 blocks and 90% for the scalar partition. Nested partitions compose, $\mathsf{A}_\Pi \mathsf{A}_{\Pi'} = \mathsf{A}_\Pi$ for $\Pi \preceq \Pi'$, which gives the first result.

Corollary 1 (Monotone recovery). *If $\Pi \preceq \Pi'$ then $D_{\Pi'}(\mathbf{w}) \le D_\Pi(\mathbf{w})$ for every $\mathbf{w}$, with $D_{\Pi_{\text{pix}}} \equiv 0$ and $D_{\Pi_{\text{scalar}}}$ equal to the total spatial variance of $\mathbf{M}$.*

What training loses is the update, not the field itself. By linearity of $\mathcal{J}$, the coarse update falls short of the dense one by the residual mapped through the kernels,

$$g_{\text{pix}} - g_{\Pi} = \mathcal{J}(\mathsf{I} - \mathsf{A}_{\Pi})\mathbf{M}, \qquad \|g_{\text{pix}} - g_{\Pi}\| \le \|\mathcal{J}\|_{\text{op}}\sqrt{D_{\Pi}}, \tag{6}$$

with $\|\mathcal{J}\|_{\text{op}}$ the operator norm of $\mathcal{J}$. We call the left side the *discarded gradient*, the quantity that Figure 4a in Section B measures on trained networks. The discarded energy bounds it and shrinks monotonically along a chain of refinements (Corollary 1). The discarded gradient itself depends on how the field and the Jacobians $\mathbf{J}_p$ covary within each block (Proposition 4 in Section A), so it need not shrink. Section 5.4 shows a partition for which it does not, and Equation (6) still gives a sufficient condition for descent.

Proposition 1 (Descent under coarse modulation)**.** *For any per-position loss and any* $\mathbf{w}$*,* $\nabla\mathcal{L} = g_{\Pi} + \mathcal{J}(\mathsf{I}-\mathsf{A}_{\Pi})\mathbf{M}$*. (i) At any stationary point of the coarse dynamics (*$g_{\Pi} = 0$*),* $\|\nabla\mathcal{L}\| \le \|\mathcal{J}\|_{\text{op}}\sqrt{D_{\Pi}}$*. (ii) The coarse update* g_{Π} *is a descent direction for the true* $\mathcal{L}$ *whenever the discarded gradient is smaller than the dense one,* $\|\mathcal{J}(\mathsf{I} - \mathsf{A}_{\Pi})\mathbf{M}\| < \|g_{\text{pix}}\|$*. Figure 4a and Table 3 report this ratio.*

The discarded energy concentrates at the discontinuities of the target. The next proposition quantifies this for a field that is smooth except across boundaries.

Proposition 2 (Where the discarded energy concentrates)**.** *Suppose* $\mathbf{M}$ *is* G*-Lipschitz in space except across a discontinuity set of total length* $\mathcal{B}$ *pixels, with jump magnitudes at most* Δ*, whose interfaces are each at least* b *long, lie more than* $\sqrt{2}\,b$ *apart, and cross each block in a single arc. For the partition* Π_b *into* $b\times b$ *blocks,*

$$D_{\Pi_b} \;\le\; c_1\, HW\, G^2 b^2 \;+\; c_2\, \mathcal{B}\, \Delta^2 b \tag{7}$$

with absolute constants c_1, c_2 *for* $b \le \mathcal{B}$*. For piecewise-constant fields (*$G{=}0$*) whose interfaces cross a constant share of the boundary blocks away from their edges, the boundary term is tight up to constants (Section B).*

Smooth targets incur the b^2 term with the small prefactor G^2. Discontinuous targets incur the b term with the prefactor $\Delta^2\mathcal{B}$, which we call the target's *discontinuity mass*. At class boundaries, the cross-entropy modulation field jumps by $\Theta(1)$, so segmentation targets belong to the discontinuous class. The bound is for this idealized class of fields, Section B checks the scaling law on fields with junctions, and Section A has all proofs.

Hypotheses. The propositions bound what a coarse rule loses, but they do not predict the order of the variants, so the four hypotheses below are tested in Section 5. The propositions concern the gradient g_{Π} of the layer's rate model, the differentiable model of its firing rates, before any optimizer scales it. The surrogate-gradient update of Section 4 is its counterpart. The order of the variants and the split of the distance to BPTT are measured, not derived.

- **H1.** Accuracy rises monotonically with granularity along the block chain, where the bound on the discarded gradient can only shrink (Equation (6) and Corollary 1).
- **H2.** Uniform modulation remains far below per-pixel modulation and loses most where the within-block error concentrates, at discontinuities (Propositions 1 and 2).
- **H3.** The benefit of per-pixel modulation grows with the target's discontinuity mass $\Delta^2\mathcal{B}$, larger on DSEC, whose label boundaries are denser, than on DDD17.
- **H4.** The ordering survives a change of the within-layer update, because the analysis uses only the modulation field and the linearity of $\mathcal{J}$.

3.4 PPM-3F: Per-Pixel Modulation

PPM-3F is the per-pixel end of the granularity range and the rule this paper proposes. Each layer trains its own per-pixel readout $\hat{\mathbf{y}}^{(\ell)}(x,y) = f^{(\ell)}_{\text{read}}(\mathbf{r}^{(\ell)}(x,y))$, a 1×1 convolution from C channels to K scores, and each position receives its own modulatory signal from it:

$$M^{(\ell)}(x,y) = \frac{\partial \mathcal{L}_{\text{CE}}\big(\hat{\mathbf{y}}^{(\ell)}(x,y),\, y^*(x,y)\big)}{\partial \mathbf{r}^{(\ell)}(x,y)}, \tag{8}$$

with $\mathcal{L}_{\text{CE}}$ the per-pixel cross-entropy loss, so that this signal is the field entry $\mathbf{M}_p$ at $p = (x,y)$. Each position therefore contributes to the shared kernel in proportion to its own error, so rare classes and boundaries are not averaged away. It keeps the three constraints of Section 3.1, because $M^{(\ell)}(x,y)$ depends only on the layer's own $f^{(\ell)}_{\text{read}}$ and the labels, and the per-pixel error never crosses layers. A fixed random feedback in place of the readout's transpose retains the benefit (Figure 3, Section E.3).

4 ARCHITECTURE AND TRAINING

The sweep's variants must differ only in the modulation operator, so one architecture and one recipe serve them all. This section states both, as well as the forward-only adaptation that runs on-chip.

LIF neuron. We use Leaky Integrate-and-Fire (LIF) neurons, the standard model of neuromorphic processors (Davies et al., 2018; Orchard et al., 2021):

$$u_i^t = \alpha\, u_i^{t-1}(1 - s_i^{t-1}) + \textstyle\sum_j w_{ij} s_j^t, \qquad s_i^t = \mathbf{1}[\, u_i^t \geq \vartheta\,], \tag{9}$$

with membrane potential u_i^t at timestep t, leak $\alpha{=}0.9$, threshold $\vartheta{=}0.5$, and the indicator $\mathbf{1}[\cdot]$. We train with a surrogate gradient (Zenke & Ganguli, 2018; Wu et al., 2018), the arctangent of SpikingJelly (Fang et al., 2023), applied *within* each layer only. A stop-gradient is applied to the spikes passed between layers and to the membrane carried between timesteps. The update therefore uses the current timestep only, since exact credit through time needs the T-step history that $O(N)$ memory excludes. The layer-local surrogate-gradient update then takes the three-factor form Equation (1), with the surrogate derivative $\sigma'(u_i)$ as the postsynaptic factor and $M^{(\ell)}$ from the readout loss (Kaiser et al., 2020), up to the layer's normalization. The normalization statistics stay on-chip (Section 6). On the GPU, AdamW takes this gradient per layer, with per-synapse moments kept inside the layer.

Backbone and encoding. Since no layer can rely on deeper ones for context, each of the $L{=}3$ layers is multi-scale: atrous spatial pyramid pooling (ASPP Chen et al., 2017), then batch normalization (BN Ioffe & Szegedy, 2015) and LIF. There is no pooling or upsampling, so every layer keeps full resolution. At $C{=}64$ channels and $128{\times}128$, the network has 77K parameters. Inputs are the 5-bin voxel grids of Table 6, one per timestep.

One layer's 2^{20} neurons match a Loihi 2 chip's full complement, $8{,}192$ per core across 128 cores, and its convolutional connections share each kernel across positions, so the weights are stored once rather than replicated per position (Orchard et al., 2021). Its three fields (rate, membrane, and modulation) are 12 MB at 32 bits, a conservative figure against the chip's $\sim$24 MB of core memory (Section 6). One of the three is the modulation field, the state a class-level rule replaces with a single C-vector. Dense modulation is therefore a third of a layer's state, the overhead that Section 6 quantifies in area and power. The depths we measure, three to eight layers, stay within one Loihi 2 board of eight chips, one per layer (Orchard et al., 2021).

Forward-only on-chip adaptation. The network can also adapt on-chip without labels. Re-estimating the BN statistics on unlabeled test data (Schneider et al., 2020) is a form of test-time adaptation (TTA) that needs forward passes only and $2C$ running scalars per BN layer. Under Gaussian input noise at a signal-to-noise ratio (SNR) of 2, it returns PPM-3F from 21.1 to 41.0% mIoU, within the seed spread of its clean 40.1% (three seeds, Section E.6). After adaptation, per-pixel modulation stays 9.3 to 13.8 pp above uniform modulation at every noise level.

5 EXPERIMENTS

This section tests the four hypotheses of Section 3.3. A summary of its results comes first.

- Accuracy rises monotonically with granularity on every seed of both datasets, and $16{\times}16$ blocks recover 67–77% of the dense gain (Section 5.2).
- Granularity accounts for 78% of the distance from uniform modulation to BPTT at three layers and 66% at five and eight, while letting the gradient cross layers recovers 1.0 pp (Section 5.3).
- The loss concentrates at boundaries, a label-aligned partition recovers little of it, and the benefit grows with the target's discontinuity mass (Sections 5.4 and 5.5).
- The ordering holds under a second local rule and on a regression task (Section 5.6).

5.1 SETUP

Every comparison below shares one protocol. This subsection fixes its data and methods.

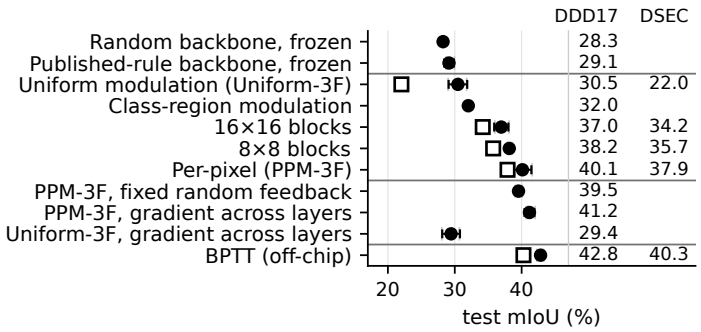


Figure 3: The granularity sweep. Test mIoU on DDD17 (filled circles) and DSEC-Semantic (open squares, sweep and reference only), mean±std over three seeds and each seed a gray tick (Section E.2).

Table 1: DDD17 per-class results across the granularity range (mean±std and per-class means over three seeds). The classes are flat, construction and sky (c+s), object, nature, human, and vehicle.

Method	Rule	On-chip	mIoU	flat	c+s	obj	nat	hum	veh
DECOLLE readout (as published)	Uniform	✓	13.6±0.4	0.0	70.7	0.0	10.8	0.1	0.3
Pool-then-classify	Uniform	✓	20.6±4.6	41.8	58.4	0.0	19.9	0.4	3.2
Uniform-3F	Uniform	✓	30.5±1.4	66.2	81.8	0.2	18.1	0.0	16.6
Label-aligned	Region	✓	32.0±0.3	67.5	73.6	2.7	20.9	5.0	22.4
16×16 blocks	Block	✓	37.0±1.1	70.0	78.2	2.6	26.1	16.8	28.0
8×8 blocks	Block	✓	38.2±0.4	69.4	78.3	4.2	27.9	15.4	33.7
PPM-3F	Per-pixel	✓	**40.1**±1.3	70.6	78.8	4.3	30.4	17.5	39.2
BPTT-SNN	BPTT	✗	42.8±0.5	71.0	83.4	4.9	35.8	18.6	43.3

Datasets and protocol. We use DDD17 (Binas et al., 2017) with labels from EV-SegNet (Alonso & Murillo, 2019) on the standard 6-class split (Zhang et al., 2025; Long et al., 2025b; Sun et al., 2022) and the 11-class DSEC-Semantic (Sun et al., 2022; Gehrig et al., 2021) (every other label frame of the split of Sun et al. (2022)). All methods share the backbone and recipe (per-pixel cross-entropy with inverse square-root frequency class weights, AdamW at learning rate 10^{-3}, cosine-annealed, batch 32, 60 epochs, flip + input-noise augmentation, flip-averaged evaluation). Main results report mean±std over seeds $\{42, 0, 1\}$, and ablations and checkpoint diagnostics use seed 42.

Methods. PPM-3F uses per-pixel modulation with per-layer AdamW. Uniform-3F is identical but with the backward signal spatially averaged (Equation (4), Π_{scalar}). It is DECOLLE's rule with the dense readout kept, and a strengthened form of the class-level mechanism shared by DECOLLE, TESS, and ETLP (Section C). Two weaker controls, pool-then-classify (20.6%) and DECOLLE's readout as published (13.6%), fall between the 12.2% obtained by predicting the largest class everywhere and Uniform-3F (Table 1), so every comparison uses the stronger Uniform-3F. BPTT-SNN is the reference that the on-chip constraints exclude, with full gradient flow from a single final readout.

5.2 Main Results: Accuracy Rises Monotonically with Granularity (H1)

Accuracy rises monotonically with granularity on every seed of both datasets (Figure 3), the ordering that Corollary 1 motivates. The four variants are the scalar (b=128), 16×16, 8×8, and per-pixel (b=1) settings of A_Π on the *backward* field (Equation (4)). The gain is not spread evenly across classes (Table 1). Vehicles, humans, and nature gain 12 to 23 pp, while construction and sky, 73% of the pixels, lose 3 pp, because uniform modulation predicts the majority class almost everywhere.

Across the granularity range, the energy discarded and the accuracy lost decrease together. Over the test set, the discarded fraction is 0.90 under uniform modulation, 0.49 at 16×16 and 0.33 at 8×8 blocks, and 0 per pixel, all at the PPM-3F checkpoint (Section D). Blocks of 16×16 broadcast 64 modulation vectors per layer instead of $16{,}384$ and recover 67% of the gain on DDD17 and 77% on

Table 2: How the distance from uniform modulation to BPTT splits (DDD17, mIoU %, mean±std over three seeds). Gains are the per-seed differences, and from Uniform-3F at 30.5±1.4 the total is 12.4±1.1 pp. The granularity step is the larger one on every seed (per seed 64–95% of the total).

Step	mIoU reached	Gain (pp)	Share
Per-pixel modulation (PPM-3F)	40.1±1.3	9.7±2.7	78%
Gradient across layers	41.2±0.7	1.0±0.7	8%
One objective with credit through time (BPTT-SNN)	42.8±0.5	1.7±1.2	13%

Table 3: Aligning the partition to the label recovers little (mIoU over three seeds). Energy and gradient ratio at the PPM-3F checkpoint, the last column at each variant's own (seed 42, Section B).

	mIoU (%)	energy discarded	gradient ratio (PPM-3F)	gradient ratio (own)
Uniform (one block)	30.5±1.4	0.90	0.83	0.53
Label-aligned	32.0±0.3	0.50	1.19	0.40
16×16 blocks	37.0±1.1	0.49	0.18	0.13

DSEC (Figure 3). Every variant trains its readout at every position and averages only the field that enters the backbone, so blocks save only modulation state.

5.3 Granularity Sets Most of the Gap to BPTT

Under uniform modulation, the backbone learns little. A per-pixel readout reaches 28.3% on a frozen random backbone, 29.1% on a frozen backbone trained under DECOLLE's published readout, and 30.5% in Uniform-3F (Figure 3). All three are within 2.2 pp of one another and at least 9.7 pp below PPM-3F.

Table 2 splits the distance from uniform modulation to BPTT: granularity accounts for 78% of it at three layers. PPM-3F reaches $40.1\pm1.3\%$ mIoU against $42.8\pm0.5\%$ for BPTT-SNN at the shared budget (Table 1). Under the plain update of Equation (1), the two reach 36.0 and 39.6% (seed 42, Section E.1). The split has three steps: per-pixel modulation, then the gradient crossing layers while every layer keeps its loss, then BPTT's single objective with credit through time.

Granularity remains the larger term at greater depth and in the other order of the steps. The share is 66% at both five and eight layers, against 73% at three layers (seed 42, Table 7), and the gap to BPTT is wider at five and eight layers than at three. Letting the gradient cross layers recovers 1.0 pp, positive on every seed, and 2.4 and 2.5 pp at five and eight layers (Table 7). Under uniform modulation, the same change recovers nothing, -1.0 ± 1.1 pp, so granularity accounts for 88% of the distance from uniform modulation with cross-layer gradients to BPTT.

5.4 Where Coarse Modulation Loses Accuracy (H2)

The accuracy loss lies where the modulation field is discontinuous, and it persists under changes in depth, width, optimizer, and training budget (Section E.1). Frame by frame, the mIoU loss of uniform modulation rises with boundary density, from 4.4 pp in the lowest quintile to 10.5 pp in the highest (three seeds, Section B). It assigns whole vehicles to the surrounding class (Section E.5).

Averaging the modulation over each ground-truth class, one block per class, recovers little of the granularity gain: 32.0% against 30.5 for uniform and 37.0 for 16×16 blocks (Table 3). It locates humans and vehicles but leaves them far below their per-pixel values (Table 1). Every class has its own modulatory signal in this variant, so what is missing is not label information but shape within each region. At the PPM-3F checkpoint, the label-aligned partition discards the same energy as 16×16 blocks but a larger share of the kernel gradient, a ratio $\|g_{\mathrm{pix}} - g_{\Pi}\|/\|g_{\mathrm{pix}}\|$ of 1.19 against 0.18 (Table 3). A ratio above one means that the sufficient condition of Proposition 1(ii) fails there, which does not rule out descent. At each variant's own checkpoint, the same ratio orders the coarse variants by accuracy (Table 3, Section B).

5.5 The Benefit Grows with the Target's Discontinuity Mass (H3)

On DSEC-Semantic, whose label boundaries are denser (boundary-pixel fraction 0.038 against 0.020, Section B), the ordering holds (Figure 3). The PPM–Uniform gap grows from $+9.7$ pp

Table 4: Post-synthesis results for the PPM-3F accelerator (TT 0.9 V 25 °C).

Process (nm)	Area (mm^2)	Memory overhead (MB)	Average power (W)
28	48.08	2.52	6.13

Table 5: NVIDIA RTX PRO 6000 versus the implemented ASIC design. GPU entries are means over ten runs, five per dataset; ratios are to the design in the same mode.

		Throughput (frame/sec)	Normalized throughput	Energy consumption (J/frame)	Normalized energy consumption
NVIDIA RTX PRO 6000	training	157.3	×0.59	1.326	×57.4
	inference	476.5	×1.15	0.462	×31.22
Implemented design	training	265.5	×1.00	0.0231	×1.00
	inference	414.1	×1.00	0.0148	×1.00

(DDD17) to $+15.9$ pp while the PPM–BPTT gap stays small (2.4 pp against 2.7 on DDD17), as H3 predicts. Merged to the six DDD17 categories, with a boundary-pixel fraction of 0.029, DSEC gives $+13.7\pm0.3$ pp (three seeds), so the number of classes is not the cause.

5.6 The Ordering Holds Across Rules and Tasks (H4)

H4 states that the ordering survives a change of the within-layer update, and we test it with a change of rule and a change of task. For the rule, we replace each layer's update with the official S-TLLR convolutional layer (Apolinario & Roy, 2025), which has a single dilation rate and weight standardization, and keep a readout per layer. Only the granularity of each layer's modulation varies. The ordering reappears on DDD17: $13.6\pm0.4\%$ with a pooled readout per layer, $29.2\pm1.3\%$ with uniform modulation, and $36.5\pm0.2\%$ with per-pixel modulation (three seeds). For the task, DSEC stereo disparity under the rule of Section 4 is a regression target without a BPTT reference, and the ordering is the same there: a mean absolute error of 1.66 px under uniform modulation against 1.48 per pixel, with the constant predictor at 1.70 (three seeds, Table 9).

6 Discussion

Hardware feasibility: ASIC implementation. Dense modulation is cheap in silicon. We implemented PPM-3F as an application-specific integrated circuit (ASIC) whose learning path covers the 1×1 readout, per-pixel modulation, the three-factor gradient of Equation (1), and the AdamW update. It was described in C++, compiled to RTL with Siemens Catapult high-level synthesis, and synthesized with Synopsys Design Compiler to a 28 nm standard-cell netlist at 200 MHz. The design uses 8-bit forward and 32-bit backward paths; training at this precision changes DDD17 mIoU by -0.7 ± 2.6 pp versus the main recipe across three paired seeds (Section F). Table 4 reports post-synthesis area, on-chip memory, and power at 128×128 input. Tiling keeps one tile's state on-chip: 2.52 MB versus a Loihi 2 chip's 24 MB (Section 3.1); the 12 MB in Section 4 represents the full 32-bit layer. At batch size 1, it sustains 414.1 and 265.5 frames/s for inference and training, respectively. The NVIDIA RTX PRO 6000 achieves 1.15× and 0.59× these rates. The ASIC consumes 0.0148 and 0.0231 J/frame for inference and training, achieving 31.22× and 57.4× higher energy efficiency, respectively, on 28 nm against the GPU's 4 nm (Table 5). ASIC figures are post-synthesis estimates.

Scope and future work. The operating point is five timesteps, three to eight layers, 128×128, and 77K parameters. The next steps are place-and-route, fabrication, and measurement on silicon.

7 Conclusion

When backpropagation is unavailable, the granularity of the modulatory signal is the largest source of loss for dense prediction with spatial discontinuities. It accounts for most of the distance from uniform modulation to backpropagation at every depth within the chip's budget, and the loss comes from averaging within a region rather than from missing label information. Per-pixel modulation removes it and comes within 2.7 points of same-architecture BPTT, and 16×16 blocks keep most of the benefit at 0.4% of the modulation state. For the designer of local rules, the granularity of the modulatory signal is therefore a design variable, not a fixed property of the rule.

AI USE STATEMENT

We wrote the entire initial draft of this manuscript independently. Generative AI tools were used exclusively for copyediting, proofreading, and polishing the English text to improve readability and grammatical correctness. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The assumptions of every proposition are stated with it in Section 3.3, and the proofs are in Section A. The datasets, splits, preprocessing, and the shared training recipe are in Section 5.1 and Section D, with the full configuration in Table 6 and the training loop in Algorithm 1. The variants of the sweep differ only in the modulation operator of Equation (4), and per-seed values of every multi-seed result are in Section E.2. The GPU measurement protocol behind Table 5 is in Section D, and the figures of Table 4 are post-synthesis results of the design described in Section 6. Anonymous source code accompanies the submission as supplementary material. It contains the model, the three-factor training loop with per-pixel modulation, the data download and preprocessing scripts, a Docker environment, and a script that trains PPM-3F on DDD17 and DSEC-Semantic.

REFERENCES

Jiwoon Ahn and Suha Kwak. Learning Pixel-Level Semantic Affinity with Image-Level Supervision for Weakly Supervised Semantic Segmentation . In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4981–4990, Los Alamitos, CA, USA, June 2018. IEEE Computer Society. doi: 10.1109/CVPR.2018.00523. URL https://doi.ieeecomputersociety.org/10.1109/CVPR.2018.00523.

Filipp Akopyan, Jun Sawada, Andrew Cassidy, Rodrigo Alvarez-Icaza, John Arthur, Paul Merolla, Nabil Imam, Yutaka Nakamura, Pallab Datta, Gi-Joon Nam, Brian Taba, Michael Beakes, Bernard Brezzo, Jente B. Kuang, Rajit Manohar, William P. Risk, Bryan Jackson, and Dharmendra S. Modha. Truenorth: Design and tool flow of a 65 mw 1 million neuron programmable neurosynaptic chip. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 34(10): 1537–1557, 2015. doi: 10.1109/TCAD.2015.2474396.

Iñigo Alonso and Ana C. Murillo. Ev-segnet: Semantic segmentation for event-based cameras. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1624–1633, 2019. doi: 10.1109/CVPRW.2019.00205.

Marco P. E. Apolinario, Kaushik Roy, and Charlotte Frenkel. Tess: A scalable temporally and spatially local learning rule for spiking neural networks. In *2025 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–9, 2025. doi: 10.1109/IJCNN64981.2025.11227652.

Marco Paul E. Apolinario and Kaushik Roy. S-TLLR: STDP-inspired temporal local learning rule for spiking neural networks. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=CNaiJRcX84.

Sami Barchid, José Mennesson, and Chabane Djeraba. Deep spiking convolutional neural network for single object localization based on deep continuous local learning. *2021 International Conference on Content-Based Multimedia Indexing (CBMI)*, pp. 1–5, 2021. URL https://api.semanticscholar.org/CorpusID:234470074.

Eugene Belilovsky, Michael Eickenberg, and Edouard Oyallon. Greedy layerwise learning can scale to imagenet. In *International conference on machine learning*, pp. 583–593. PMLR, 2019.

Guillaume Bellec, Franz Scherr, Anand Subramoney, Elias Hajek, Darjan Salaj, Robert Legenstein, and Wolfgang Maass. A solution to the learning dilemma for recurrent networks of spiking neurons. *Nature Communications*, 11, 07 2020. doi: 10.1038/s41467-020-17236-y.

Aleksandr Beznosikov, Samuel Horváth, Peter Richtárik, and Mher Safaryan. On biased compression for distributed learning. *J. Mach. Learn. Res.*, 24(1), January 2023. ISSN 1532-4435.

Jonathan Binas, Daniel Neil, Shih-Chii Liu, and Tobi Delbruck. Ddd17: End-to-end davis driving dataset. In *ICML Workshop*, 2017.

Thomas Bohnstingl, Stanisław Woźniak, Angeliki Pantazi, and Evangelos Eleftheriou. Online spatio-temporal learning in deep neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 34(11):8894–8908, 2023. doi: 10.1109/TNNLS.2022.3153985.

Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation, 2017. URL https://arxiv.org/abs/1706.05587.

Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (eds.), *Computer Vision – ECCV 2018*, pp. 833–851, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01234-2.

Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, Yuyun Liao, Chit-Kwan Lin, Andrew Lines, Ruokun Liu, Deepak Mathaikutty, Steven McCoy, Arnab Paul, Jonathan Tse, Guruguhanathan Venkataramanan, Yi-Hsin Weng, Andreas Wild, Yoonseok Yang, and Hong Wang. Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro*, 38(1):82–99, 2018. doi: 10.1109/MM.2018.112130359.

Wei Fang, Yanqi Chen, Jianhao Ding, Zhaofei Yu, Timothée Masquelier, Ding Chen, Liwei Huang, Huihui Zhou, Guoqi Li, and Yonghong Tian. Spikingjelly: An open-source machine learning infrastructure platform for spike-based intelligence. *Science Advances*, 9(40):eadi1480, 2023. doi: 10.1126/sciadv.adi1480. URL https://www.science.org/doi/abs/10.1126/sciadv.adi1480.

Nicolas Frémaux and Wulfram Gerstner. Neuromodulated spike-timing-dependent plasticity, and theory of three-factor learning rules. *Frontiers in Neural Circuits*, Volume 9 - 2015, 2016. ISSN 1662-5110. doi: 10.3389/fncir.2015.00085. URL https://www.frontiersin.org/journals/neural-circuits/articles/10.3389/fncir.2015.00085.

Steve B. Furber, Francesco Galluppi, Steve Temple, and Luis A. Plana. The spinnaker project. *Proceedings of the IEEE*, 102(5):652–665, 2014. doi: 10.1109/JPROC.2014.2304638.

Guillermo Gallego, Tobi Delbrück, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J. Davison, Jörg 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.

Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. Dsec: A stereo event camera dataset for driving scenarios. *IEEE Robotics and Automation Letters*, 2021. doi: 10.1109/LRA.2021.3068942.

Dalia Hareb and Jean Martinet. Evsegsnn: Neuromorphic semantic segmentation for event data. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2024. doi: 10.1109/IJCNN60899.2024.10650811.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HJz6tiCqYm.

Geoffrey Hinton. The forward-forward algorithm: Some preliminary investigations, 2022. URL https://arxiv.org/abs/2212.13345.

Sergey Ioffe and Christian Szegedy. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning - Volume 37*, ICML'15, pp. 448–456. JMLR.org, 2015.

Jacques Kaiser, Hesham Mostafa, and Emre Neftci. Synaptic plasticity dynamics for deep continuous local learning (decolle). *Frontiers in Neuroscience*, Volume 14 - 2020, 2020. ISSN 1662-453X. doi: 10.3389/fnins.2020.00424. URL https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2020.00424.

Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Stich, and Martin Jaggi. Error feedback fixes signsgd and other gradient compression schemes. In *International conference on machine learning*, pp. 3252–3261. PMLR, 2019.

Alexander Kolesnikov and Christoph H Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *European conference on computer vision*, pp. 695–711. Springer, 2016.

Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. Deeply-supervised nets. In *Artificial intelligence and statistics*, pp. 562–570. Pmlr, 2015a.

Dong-Hyun Lee, Saizheng Zhang, Asja Fischer, and Yoshua Bengio. Difference target propagation. In *Proceedings of the 2015th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part I*, ECMLPKDD'15, pp. 498–515, Gewerbestrasse 11 CH-6330, Cham (ZG), CHE, 2015b. Springer. ISBN 9783319235271. doi: 10.1007/978-3-319-23528-8_31. URL https://doi.org/10.1007/978-3-319-23528-8_31.

Timothy P Lillicrap, Adam Santoro, Luke Marris, Colin J Akerman, and Geoffrey Hinton. Backpropagation and the brain. *Nature Reviews Neuroscience*, 21(6):335–346, 2020.

Xianlei Long, Xiaxin Zhu, Fangming Guo, Chao Chen, Xiangwei Zhu, Fuqiang Gu, Songyu Yuan, and Chunlong Zhang. Spike-brgnet: Efficient and accurate event-based semantic segmentation with boundary region-guided spiking neural networks. *IEEE Trans. Cir. and Sys. for Video Technol.*, 35(3):2712–2724, March 2025a. ISSN 1051-8215. doi: 10.1109/TCSVT.2024.3495769. URL https://doi.org/10.1109/TCSVT.2024.3495769.

Xianlei Long, Xiaxin Zhu, Fangming Guo, Wanyi Zhang, Qingyi Gu, Chao Chen, and Fuqiang Gu. Sltnet: Efficient event-based semantic segmentation with spike-driven lightweight transformer-based networks. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4331–4338, 2025b. doi: 10.1109/IROS60139.2025.11247662.

Stéphane Mallat. *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, Inc., USA, 3rd edition, 2008. ISBN 0123743702.

Qingyan Meng, Mingqing Xiao, Shen Yan, Yisen Wang, Zhouchen Lin, and Zhi-Quan Luo. Towards memory-and time-efficient backpropagation for training spiking neural networks. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6143–6153. IEEE, 2023.

Emre O. Neftci, Hesham Mostafa, and Friedemann Zenke. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6):51–63, 2019. doi: 10.1109/MSP.2019.2931595.

Arild Nøkland and Lars Hiller Eidnes. Training neural networks with local error signals. In *International conference on machine learning*, pp. 4839–4850. PMLR, 2019.

Garrick Orchard, E. Paxon Frady, Daniel Ben Dayan Rubin, Sophia Sanborn, Sumit Bam Shrestha, Friedrich T. Sommer, and Mike Davies. Efficient neuromorphic signal processing with loihi 2. In *2021 IEEE Workshop on Signal Processing Systems (SiPS)*, pp. 254–259, 2021. doi: 10.1109/SiPS52927.2021.00053.

Alexandre Payeur, Jordan Guerguiev, Friedemann Zenke, Blake A Richards, and Richard Naud. Burst-dependent synaptic plasticity can coordinate learning in hierarchical circuits. *Nature neuroscience*, 24(7):1010–1019, 2021.

Jing Pei, Lei Deng, Sen Song, Mingguo Zhao, Youhui Zhang, Shuang Wu, Guanrui Wang, Zhe Zou, Zhenzhi Wu, Wei He, Feng Chen, Ning Deng, Si Wu, Yu Wang, Yujie Wu, Zheyu Yang, Cheng Ma, Guoqi Li, Wentao Han, Huanglong Li, Huaqiang Wu, Rong Zhao, Yuan Xie, and Luping Shi. Towards artificial general intelligence with hybrid tianjic chip architecture. *Nature*, 572:106 – 111, 2019. URL https://api.semanticscholar.org/CorpusID:199056116.

Fernando M Quintana, Fernando Perez-Peña, Pedro L Galindo, Emre O Neftci, Elisabetta Chicca, and Lyes Khacef. Etlp: event-based three-factor local plasticity for online learning with neuromorphic hardware. *Neuromorphic Computing and Engineering*, 4(3):034006, aug 2024. doi: 10.1088/2634-4386/ad6733. URL https://doi.org/10.1088/2634-4386/ad6733.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi (eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4.

João Sacramento, Rui Ponte Costa, Yoshua Bengio, and Walter Senn. Dendritic cortical microcircuits approximate the backpropagation algorithm. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 8735–8746, Red Hook, NY, USA, 2018. Curran Associates Inc.

Benjamin Scellier and Yoshua Bengio. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in computational neuroscience*, 11:24, 2017.

Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):640–651, April 2017. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2572683. URL https://doi.org/10.1109/TPAMI.2016.2572683.

Zhaoning Sun, Nico Messikommer, Daniel Gehrig, and Davide Scaramuzza. Ess: Learning event-based semantic segmentation from still images. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIV*, pp. 341–357, Berlin, Heidelberg, 2022. Springer-Verlag. ISBN 978-3-031-19829-8. doi: 10.1007/978-3-031-19830-4_20. URL https://doi.org/10.1007/978-3-031-19830-4_20.

Yulin Wang, Zanlin Ni, Yifan Pu, Cai Zhou, Jixuan Ying, Shiji Song, and Gao Huang. Infopro: Locally supervised deep learning by maximizing information propagation. *International Journal of Computer Vision*, 133(5):2752–2782, 2025. doi: 10.1007/s11263-024-02296-0.

James C. R. Whittington and Rafal Bogacz. An approximation of the error backpropagation algorithm in a predictive coding network with local hebbian synaptic plasticity. *Neural Computation*, 29(5): 1229–1262, 2017. doi: 10.1162/NECO_a_00949.

Yujie Wu, Lei Deng, Guoqi Li, Jun Zhu, and Luping Shi. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in Neuroscience*, Volume 12 - 2018, 2018. ISSN 1662-453X. doi: 10.3389/fnins.2018.00331. URL https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2018.00331.

Mingqing Xiao, Qingyan Meng, Zongpeng Zhang, Di He, and Zhouchen Lin. Online training through time for spiking neural networks. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Wujian Ye, Shaozhen Chen, Haoxian Liu, Yijun Liu, Yuehai Chen, Youfeng Cui, and Wenjie Lin. The architecture design and training optimization of spiking neural network with low-latency and high-performance for classification and segmentation. *Neural Netw.*, 191(C), November 2025. ISSN 0893-6080. doi: 10.1016/j.neunet.2025.107790. URL https://doi.org/10.1016/j.neunet.2025.107790.

Friedemann Zenke and Surya Ganguli. Superspike: Supervised learning in multilayer spiking neural networks. *Neural Comput.*, 30(6):1514–1541, June 2018. ISSN 0899-7667. doi: 10.1162/neco_a_01086. URL https://doi.org/10.1162/neco_a_01086.

Rui Zhang, Luziwei Leng, Kaiwei Che, Hu Zhang, Jie Cheng, Qinghai Guo, Jianxing Liao, and Ran Cheng. Accurate and efficient event-based semantic segmentation using adaptive spiking encoder–decoder network. *IEEE Transactions on Neural Networks and Learning Systems*, 36(5): 9326–9340, 2025. doi: 10.1109/TNNLS.2024.3437415.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2921–2929, 2016.

A PROOFS

We collect the proofs deferred from Section 3.3. Throughout, A_Π is the block-averaging operator of Equation (4), $\mathcal{F}_\Pi$ the subspace of block-constant fields, and I the identity on $\mathcal{F}$.

Proof of Equation (5). A_Π is idempotent and self-adjoint ($\langle \mathsf{A}_\Pi \mathbf{U}, \mathbf{V}\rangle = \sum_{B\in\Pi} |B|\, \bar{\mathbf{U}}_B^\top \bar{\mathbf{V}}_B$ is symmetric in $\mathbf{U}, \mathbf{V}$), with range $\mathcal{F}_\Pi$. Hence it is the orthogonal projector onto $\mathcal{F}_\Pi$, which gives the energy split, and $\mathsf{A}_\Pi \mathsf{A}_{\Pi'} = \mathsf{A}_\Pi$ for $\Pi \preceq \Pi'$ because $\mathcal{F}_\Pi \subseteq \mathcal{F}_{\Pi'}$. □

Proof of Corollary 1. By Equation (5), $D_\Pi = \|\mathbf{M}\|^2 - \|\mathsf{A}_\Pi \mathbf{M}\|^2$. For $\Pi \preceq \Pi'$, $\mathcal{F}_\Pi \subseteq \mathcal{F}_{\Pi'}$ gives $\|\mathsf{A}_\Pi \mathbf{M}\| \le \|\mathsf{A}_{\Pi'} \mathbf{M}\|$, so $D_{\Pi'} \le D_\Pi$. At Π_{pix}, the projection is the identity, so $D_{\Pi_{\mathrm{pix}}}{=}0$. At Π_{scalar}, it subtracts the global mean, leaving the total spatial variance. □

Proposition 3 (Coarse modulation optimizes a pooled surrogate)**.** *Let $\mathcal{L}_p(\mathbf{r}_p) = \frac{1}{2}\|\mathbf{r}_p - \mathbf{r}_p^*\|^2$ (squared loss against a per-position target rate $\mathbf{r}_p^*$, cross-entropy in Section C) and $\mathcal{L}(\mathbf{w}) = \sum_p \mathcal{L}_p$. Then $g_\Pi = \nabla_{\mathbf{w}} \mathcal{L}_\Pi$ for the* pooled surrogate $\mathcal{L}_\Pi(\mathbf{w}) := \frac{1}{2}\|\mathsf{A}_\Pi(\mathbf{r}(\mathbf{w}) - \mathbf{r}^*)\|^2$*, and for all $\mathbf{w}$ and $\Pi \preceq \Pi'$,*

$$\mathcal{L}_\Pi \;\le\; \mathcal{L}_{\Pi'} \;\le\; \mathcal{L}, \qquad \mathcal{L} - \mathcal{L}_\Pi = \tfrac{1}{2} D_\Pi. \tag{10}$$

Proof of Proposition 3. With $\mathbf{M} = \mathbf{r} - \mathbf{r}^*$, $\nabla_{\mathbf{w}} \frac{1}{2}\|\mathsf{A}_\Pi(\mathbf{r} - \mathbf{r}^*)\|^2 = (\partial \mathbf{r}/\partial \mathbf{w})^\top \mathsf{A}_\Pi^\top \mathsf{A}_\Pi(\mathbf{r} - \mathbf{r}^*) = \mathcal{J}\mathsf{A}_\Pi \mathbf{M} = g_\Pi$, using $\mathsf{A}_\Pi^\top \mathsf{A}_\Pi = \mathsf{A}_\Pi$ (Equation (5)). The sandwich $\mathcal{L}_\Pi \le \mathcal{L}_{\Pi'} \le \mathcal{L}$ is monotonicity of nested projections, and $\mathcal{L} - \mathcal{L}_\Pi = \frac{1}{2}(\|\mathbf{M}\|^2 - \|\mathsf{A}_\Pi \mathbf{M}\|^2) = \frac{1}{2} D_\Pi$ is Equation (5). □

Proof of Proposition 1. Linearity of $\mathcal{J}$ gives $\nabla\mathcal{L} = \mathcal{J}\mathbf{M} = \mathcal{J}\mathsf{A}_\Pi \mathbf{M} + \mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M} = g_\Pi + \mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}$. (i) At $g_\Pi{=}0$, $\|\nabla\mathcal{L}\| = \|\mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}\| \le \|\mathcal{J}\|_{\mathrm{op}} \|(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}\| = \|\mathcal{J}\|_{\mathrm{op}} \sqrt{D_\Pi}$. (ii) With $\mathbf{d} = \mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}$ and $g_\Pi = g_{\mathrm{pix}} - \mathbf{d}$, $\langle \nabla\mathcal{L}, g_\Pi\rangle = \|g_{\mathrm{pix}}\|^2 - \langle g_{\mathrm{pix}}, \mathbf{d}\rangle \ge \|g_{\mathrm{pix}}\|(\|g_{\mathrm{pix}}\| - \|\mathbf{d}\|)$ by Cauchy–Schwarz, which is positive under the stated condition. □

Corollary 2 (Loss gap at a coarse stationary point)**.** *If $\mathcal{L}$ is a differentiable L_s-smooth loss (the rate model of the network, of which the surrogate-gradient update is the counterpart), then at any stationary point of the coarse dynamics $\mathcal{L}(\mathbf{w}) - \min \mathcal{L} \;\ge\; \|\mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}\|^2/(2L_s)$. The norm on the right is the discarded gradient in parameter space, the quantity Section B reports.*

Proof. At $g_\Pi = 0$ the identity of Proposition 1 gives $\nabla\mathcal{L} = \mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}$, and an L_s-smooth function satisfies $f(\mathbf{w}) - \min f \ge \|\nabla f(\mathbf{w})\|^2/(2L_s)$. □

Remark. In optimization terms, A_Π compresses the backward field, not the gradient. The field contraction $\|(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}\|^2 = D_\Pi \le \|\mathbf{M}\|^2$ is unconditional. The gradient-level contraction $\|\mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}\| < \|g_{\mathrm{pix}}\|$ that a δ-compressor descent lemma needs (compressor definition in Karimireddy et al., 2019, descent lemma in Beznosikov et al., 2023) is the hypothesis of Proposition 1(ii), which Table 3 shows can fail. Under that hypothesis, Proposition 1 is that bound specialized to A_Π. The projection view adds the exact geometric identity of the discarded part (D_Π, Equation (5)) and its link to the target geometry (Proposition 2). Proposition 2 carries the approximation rate for piecewise-regular images with edges (Section 9.3 of Mallat, 2008) over to the modulation field.

The same algebra covers any linear collapse. For a rule whose modulation field $\mathbf{M}_S$ lies in a subspace $S \subseteq \mathcal{F}$ (e.g., the rank-K span of a fixed random class-level readout), Equation (5) and Corollary 1 apply to the orthogonal projector P_S with refinement read as subspace inclusion. Proposition 1 holds with $\sqrt{D_\Pi}$ replaced by $\|\mathbf{M} - \mathbf{M}_S\|$, which is at least $\|(\mathsf{I} - \mathsf{P}_S)\mathbf{M}\|$ because $\mathsf{P}_S \mathbf{M}$ is the closest point of S to $\mathbf{M}$. Granularity generalizes to $\dim S$. Error feedback (Karimireddy et al., 2019) would reintroduce a $C{\times}H{\times}W$ residual state (dense modulation delivered through memory rather than through the readout) and is therefore an instance of the diagnosis, not an alternative to it.

Proposition 4 (Shared kernel)**.** *Write the update as a sum over positions, $\mathcal{J}\mathbf{M} = \sum_p \mathbf{J}_p^\top \mathbf{M}_p$, where $\mathbf{J}_p^\top$ maps the modulation at p to the kernel update (the eligibility at p). For a partition Π with block means $\bar{\mathbf{J}}_B = |B|^{-1} \sum_{p\in B} \mathbf{J}_p$ and $\bar{\mathbf{M}}_B$,*

$$\mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M} \;=\; \sum_{B\in\Pi} \sum_{p\in B} (\mathbf{J}_p - \bar{\mathbf{J}}_B)^\top (\mathbf{M}_p - \bar{\mathbf{M}}_B), \qquad \|\mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M}\| \le \sqrt{D_{\mathcal{J}}(\Pi)\, D_\Pi}, \tag{11}$$

with $D_{\mathcal{J}}(\Pi) = \sum_p \|\mathbf{J}_p - \bar{\mathbf{J}}_{B(p)}\|_{\mathrm{op}}^2$ the within-block energy of the eligibility. The discarded gradient is the within-block covariance of eligibility and modulation, and vanishes when either is block-constant.

Proof. $\mathcal{J}(\mathsf{I} - \mathsf{A}_\Pi)\mathbf{M} = \sum_B \sum_{p\in B} \mathbf{J}_p^\top (\mathbf{M}_p - \bar{\mathbf{M}}_B)$, and $\sum_{p\in B} \bar{\mathbf{J}}_B^\top (\mathbf{M}_p - \bar{\mathbf{M}}_B) = 0$ because the block residual of $\mathbf{M}$ sums to zero. Subtracting it gives the identity. The triangle inequality and Cauchy–Schwarz over positions give the bound. □

Unlike the operator-norm bound of Proposition 1, this bound vanishes when the eligibility is block-constant, however large D_Π is. Section B reports the left-hand side measured on the trained network.

Proof of Proposition 2. Under the separation condition, distinct interfaces of the discontinuity set have length at least b, are separated by more than $\sqrt{2}\,b$, and cross each block in a single arc. No block then meets two interfaces or is split into more than two regions. In boundary-free blocks, G-Lipschitzness bounds the deviation from the block mean by $\sqrt{2}\,Gb$, contributing at most $2G^2b^2$ per pixel and hence at most $2\,HWG^2b^2$ in total. The n interfaces of total length $\mathcal{B}$ split into at most $\mathcal{B}/b + n$ arcs of length $\le b$. Each arc has diameter $\le b$ and thus meets at most four b-blocks, so the discontinuity set meets at most $4(\mathcal{B}/b + n) \le 8\mathcal{B}/b$ blocks, since each interface is at least b long. The block diagonal $\sqrt{2}\,b$ is below the separation of distinct interfaces, so a block meets at most one interface and contains at most two regions. Each boundary-block pixel deviates from its block mean by at most $\Delta + \sqrt{2}\,Gb$, and the square of this bound distributes across the two terms of Equation (7). The boundary-block count is at most $\min\{8\mathcal{B}/b,\ HW/b^2\}$, which absorbs the cross terms and gives the boundary term for $b \le \mathcal{B}$ with an absolute constant. Tightness: a block crossed transversally with area fractions $\lambda, 1-\lambda$ contributes $\lambda(1-\lambda)b^2\Delta^2$, so a piecewise-constant field attains the boundary term up to constants. Section B measures the resulting linear growth in b. □

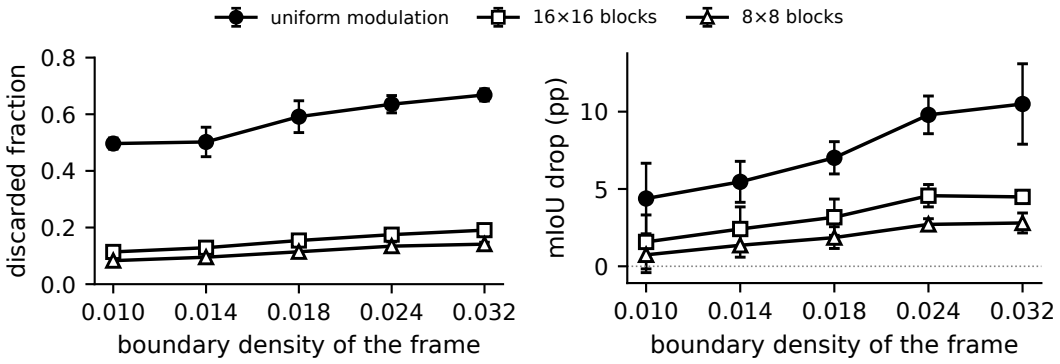
B Tightness of the Discontinuity-Mass Bound

This appendix gives the boundary-level measurements behind H2 and H3. They are the accuracy in a band at class boundaries, the discarded gradient on trained networks, the scaling check of Proposition 2, and the boundary densities of the two datasets.

Accuracy near class boundaries is the first measurement. In a 3-pixel band around the boundaries, Uniform-3F retains $59.5\pm0.4\%$ of its interior mIoU against $71.7\pm1.6\%$ for PPM-3F and $70.0\pm2.0\%$ for BPTT-SNN (three seeds). The split holds at band widths of 1 to 5 pixels and under a ±1-pixel displacement of the label boundaries.

The second measurement is the discarded gradient of Proposition 4 on trained networks. At the PPM-3F checkpoint (test frames), the fraction of the kernel gradient a coarse variant discards, $\|g_{\mathrm{pix}} - g_\Pi\|/\|g_{\mathrm{pix}}\|$ averaged over batches of 16 frames, is 0.83, 0.18, and 0.09 for scalar, 16×16, and 8×8 blocks. Figure 4a reports the same ratio per frame at each coarse variant's own checkpoint, which is why its scalar values are lower. Measured at each coarse variant's own checkpoint with the same protocol, the fraction is 0.53, 0.40, 0.13, and 0.10 for the scalar, label-aligned, 16×16, and 8×8 variants (seed 42), the order of their accuracy. Over three seeds the label-aligned variant's value is 0.41 ± 0.02 against 0.57 ± 0.04 for the scalar variant. Along the label-aligned variant's own trajectory (epochs 10 to 60, first 640 test frames), the fraction stays between 0.42 and 0.46, below one and below the scalar variant's own value on the same frames (0.56). The sufficient condition of Proposition 1(ii) therefore holds on every measured batch along the trajectory.

On real frames, the discarded gradient and the accuracy loss rise together at fixed granularity. Within each DDD17 test frame, the fraction of the kernel gradient a coarse variant discards rises with the frame's boundary density (Spearman 0.58 for the scalar variant). The per-frame discarded fraction is rank-correlated with the per-frame mIoU drop in all nine (granularity, seed) cells (0.10–0.45,



(a) Discarded fraction of the kernel gradient.

(b) Drop in mIoU below the per-pixel variant.

Figure 4: Where coarse modulation loses accuracy, frame by frame. DDD17 test frames in quintiles of boundary density (3,890 frames, three seeds, bars ± 1 std). The share of its kernel gradient a coarse variant discards (a) and its mIoU below the per-pixel variant on the same frames (b) both rise with boundary density, most for uniform modulation. The per-pixel variant's own gap to BPTT varies by 1.4 pp across the same quintiles.

n=3,890 frames each). The correlation is weak because per-frame mIoU is noisy and the fraction varies little inside one granularity. Figure 4 bins the same frames by boundary density.

A specificity control separates the coarse variants' loss from BPTT's advantage. The per-pixel variant's per-frame gap to BPTT over the same quintiles changes by $+1.4$ pp on average (0.0, $+1.6$, and $+2.5$ for seeds 42, 0, and 1) against $+6.1$ for the scalar variant. Per-frame mIoU averages over the classes present in a frame, on which PPM-3F and BPTT are level, while BPTT's dataset-level advantage comes from the rare classes accumulated over the set.

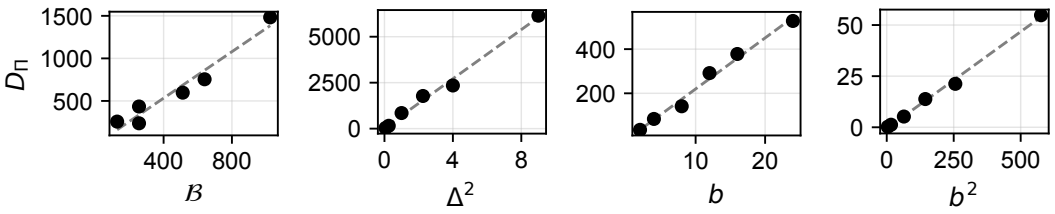
Proposition 2 predicts that the discarded energy splits into a smooth term $\propto G^2 b^2$ and a boundary term $\propto \Delta^2 \mathcal{B}\, b$. We test the scaling law in isolation. On controlled synthetic modulation fields, we hold everything fixed and vary one quantity at a time, measuring the discarded energy $D_\Pi = \|\mathbf{M} - \mathsf{A}_\Pi \mathbf{M}\|^2$ directly. The projector A_Π is exactly the block-average of Section 5.2, averaged over boundary phase so jumps do not align with the block grid. Figure 5 shows the bound is tight in every argument: at fixed block size, D_Π is linear in boundary length $\mathcal{B}$ (R^2=0.947) and in squared jump Δ^2 (R^2=0.991). For a piecewise-constant field, it grows linearly in b (R^2=0.987, the $\Delta^2 \mathcal{B}\, b$ term), whereas for a smooth field it grows quadratically in b (R^2=0.997, the $G^2 b^2$ term). The separation hypothesis of Proposition 2 excludes junctions, and the Voronoi fields have them, so the law holds beyond the hypothesis.

The accuracy benefit follows the same scaling law. On the synthetic task of Section E.3, sweeping the Voronoi cell count so the boundary fraction grows from 0.10 to 0.36 widens the per-pixel$-$scalar mIoU gap monotonically from $+1.1$ to $+6.5$ pp (five seeds per point). The benefit itself, not only the discarded energy, tracks discontinuity mass.

The same ordering holds on the real label fields. At 128×128, the boundary-pixel fraction is 0.020 for DDD17 against 0.038 for DSEC (one-hot targets have constant jump, so discontinuity mass orders by boundary length), matching the benefit ordering from $+9.7$ to $+15.9$ pp. Holding the class count fixed gives a third point. Merging the eleven DSEC classes into the six DDD17 categories lowers the boundary fraction to 0.029. The merged categories are flat (road and sidewalk), construction (sky, building, fence, and wall), object (pole and sign), nature (vegetation), human (person), and vehicle (car). The PPM$-$Uniform gap is then $+13.7 \pm 0.3$ pp over seeds $\{42, 0, 1\}$ (50.7 ± 0.4 against 37.0 ± 0.6, per seed $+13.5$, $+14.0$, and $+13.7$), between the DDD17 gap ($+9.2$) and the DSEC-11 gap ($+15.3$) of seed 42.

C Cross-Entropy and the Pool-then-Classify Readout

Proposition 3 states the exact potential identity for squared loss, whereas the method we run uses softmax cross-entropy and DECOLLE pools *rates* before classifying rather than averaging the gradient. We show here that the two reduce to the same quantity up to a controlled second-order term. For a linear readout with logits $\mathbf{z}_p = R\,\mathbf{r}_p$, readout matrix $R \in \mathbb{R}^{K \times C}$, and a softmax, the



(a) $D_\Pi \propto \mathcal{B}$. (b) $D_\Pi \propto \Delta^2$. (c) Boundary: $\propto b$. (d) Smooth: $\propto b^2$.

Figure 5: Tightness of Proposition 2 on controlled synthetic fields. Each panel varies one quantity, all others fixed, and measures D_Π. Dashed lines are least-squares fits. The smooth (G^2b^2) and boundary ($\Delta^2\mathcal{B}\,b$) terms scale as the bound predicts ($R^2 = 0.947, 0.991, 0.987, 0.997$ for (a)–(d)).

per-position modulation is $\mathbf{M}_p = R^\top(\mathrm{softmax}(\mathbf{z}_p) - \mathbf{y}_p)$ with one-hot label $\mathbf{y}_p$. Averaging this field per block (Equation (4)) thus averages the probability residual, whereas the pool-then-classify readout uses $R^\top(\mathrm{softmax}(\bar{\mathbf{z}}_B) - \bar{\mathbf{y}}_B)$.

Lemma 1 (Jensen gap of the two collapses)**.** *Fix a block B with logits $\{\mathbf{z}_p\}_{p\in B}$ and mean $\bar{\mathbf{z}}_B$, and let $\kappa := \sup_{\mathbf{z}} \sup_{\|\mathbf{u}\|=1} \|\nabla^2 \mathrm{softmax}(\mathbf{z})[\mathbf{u},\mathbf{u}]\|_2 < \infty$ (finite, since every second derivative of* softmax *is a bounded polynomial in the probabilities). A second-order Taylor expansion about $\bar{\mathbf{z}}_B$ gives*

$$\left\| \tfrac{1}{|B|} \textstyle\sum_{p\in B} \mathrm{softmax}(\mathbf{z}_p) \;-\; \mathrm{softmax}(\bar{\mathbf{z}}_B) \right\| \;\le\; \tfrac{\kappa}{2}\, \mathrm{Var}_B(\mathbf{z}),$$

where

$$\mathrm{Var}_B(\mathbf{z}) := \tfrac{1}{|B|} \textstyle\sum_{p\in B} \|\mathbf{z}_p - \bar{\mathbf{z}}_B\|^2,$$

and the two collapsed modulation vectors therefore differ by at most $\frac{\kappa}{2}\|R\|_2 \mathrm{Var}_B(\mathbf{z})$. This difference is second-order in the within-block logit spread and zero exactly when the block's logits agree.

Proof. Taylor's theorem with integral remainder gives $\mathrm{softmax}(\mathbf{z}_p) = \mathrm{softmax}(\bar{\mathbf{z}}_B) + \nabla\mathrm{softmax}(\bar{\mathbf{z}}_B)(\mathbf{z}_p - \bar{\mathbf{z}}_B) + \mathbf{r}_p$ with $\|\mathbf{r}_p\| \le \frac{\kappa}{2}\|\mathbf{z}_p - \bar{\mathbf{z}}_B\|^2$. The linear terms average to zero over the block, and averaging the remainders gives the bound. □

Measured on the trained network (DDD17 test set), the gap between the two collapses averages 7% of the residual norm at b=8 and 12% at b=16, rising to 34% at scalar granularity. The two collapses agree to second order at fine blocks and separate at the scalar end, which is why every comparison uses the stronger Uniform-3F control (Section 5.1).

The *exact* potential identity of Proposition 3 is squared-loss-specific. Under cross-entropy the averaged field is not a gradient: the per-position Hessians $R^\top(\mathrm{diag}(\mathbf{s}_p) - \mathbf{s}_p\mathbf{s}_p^\top)R$, with R the $K\times C$ readout matrix and $\mathbf{s}_p = \mathrm{softmax}(\mathbf{z}_p)$, differ across positions, so $\mathsf{A}_\Pi\nabla^2\mathcal{L}$ is not symmetric and no scalar potential exists. For the same reason no loss-gap bound of the form of Corollary 2 follows for it. The empirical claims rest on Corollary 1, Proposition 1(i), Proposition 2, and Proposition 4, which use only the modulation field $\mathbf{M}$ and the linearity of $\mathcal{J}$ and are therefore loss-agnostic. They carry over unchanged up to this second-order term.

D Implementation Details

Every comparison in the paper shares one configuration, which this appendix records in full (Table 6). The methods differ only in the backward path (Section 5.1). DDD17 has 15,950/3,890 train/test frames (14,630 after the held-out selection recording) and DSEC-Semantic 4,021/1,397. Frames are resized anisotropically to 128×128 (voxels bilinear, labels nearest-neighbor). The block grid of the 16×16 and 8×8 variants is fixed to the image, and the synthetic test of Section B randomizes its phase. Each SNN timestep receives one voxel bin together with the event-density and coordinate channels. A 60-epoch run takes 0.8–1.4 h on one NVIDIA RTX PRO 6000 GPU.

Evaluation uses the last layer's readout. Per-layer readout mIoU is monotone in depth (32.7, 36.5, and 39.9 from the first to the third layer). At each coarse variant's own checkpoint, the discarded fractions of Section 5.2 are 0.92, 0.47, and 0.31 for the scalar, 16×16, and 8×8 variants (last layer). Reported aggregates and quoted differences are computed from unrounded values and rounded independently for display.

Table 6: Configuration shared by all methods.

Component	Setting
Datasets	DDD17 (6 classes) and DSEC-Semantic (11 classes) at 128×128
Backbone	3 ASPP layers, each with three dilated 3×3 branches that share its $C=64$ output channels (21, 21, and 22) at dilation rates $(1,3,5)$, $(1,5,9)$, and $(1,7,13)$, 77K params in total
Neuron	LIF, leak $\alpha=0.9$, threshold $\vartheta=0.5$, arctangent surrogate gradient
Input	5-bin voxel grid + event-density + 2 coordinate channels (normalized x and y), one bin per timestep ($T=5$)
Loss	per-pixel cross-entropy, inverse square-root frequency class weights
Optimizer	AdamW, learning rate 10^{-3}, cosine-annealed, weight decay 10^{-4}, batch 32, 60 epochs
Augmentation	random horizontal flip + Gaussian input noise ($\sigma=0.03$ on half the batches, SNR ≈ 2.9), with flip-averaged evaluation
Seeds	$\{42, 0, 1\}$ for main results (mean$\pm$std) and seed 42 for ablations, all aggregates from unrounded values
Selection	epoch of best validation mIoU on held-out training data excluded from the training set (one DDD17 recording, 1,320 of the 15,950 frames, and 391 of the 4,021 DSEC training frames), never on test, for every run reported, ablations and sweeps included
Tuning	one shared recipe for all variants, with the backward path as the only manipulated variable
Pool variant	pooled logits broadcast to every position and scored against the per-pixel labels, with the same 1×1 readout applied per position at test
DECOLLE readout	as published: a fixed random projection of the flattened rate field to one K-vector per frame, broadcast to every position and scored per pixel, so the backbone receives the rank-K field it induces and no dense readout is trained

Algorithm 1 PPM-3F training (one mini-batch). Each layer updates independently. No inter-layer communication in the backward pass.

Require: voxels $\mathbf{V}$, labels $\mathbf{y}^*$, layers $\{(\mathbf{w}^{(\ell)}, f^{(\ell)}_{\text{read}})\}_{\ell=1}^{L}$
for $t = 1, \dots, T$ **do** ▷ SNN timesteps
  $\mathbf{s}^{(0)} \leftarrow \mathbf{V}$
  for $\ell = 1, \dots, L$ **do**
    $\mathbf{u}^{(\ell)}, \mathbf{s}^{(\ell)} \leftarrow \text{LIF}\big(\mathbf{w}^{(\ell)}, \text{sg}(\mathbf{s}^{(\ell-1)}), \text{sg}(\mathbf{u}^{(\ell)})\big)$ ▷ sg: stop-gradient, on the input spikes and on the membrane carried from $t-1$
    $\mathbf{r}^{(\ell)} \mathrel{+}= \mathbf{s}^{(\ell)}/T$ ▷ accumulate rates
  end for
end for
for $\ell = 1, \dots, L$ **do** ▷ independent per-layer updates
  $\hat{\mathbf{y}}^{(\ell)} \leftarrow f^{(\ell)}_{\text{read}}(\mathbf{r}^{(\ell)})$ ▷ per-pixel logits
  $\mathcal{L}^{(\ell)} \leftarrow \text{CE}(\hat{\mathbf{y}}^{(\ell)}, \mathbf{y}^*)$ ▷ per-pixel loss
  $\mathbf{w}^{(\ell)} \leftarrow \text{AdamW}\big(\mathbf{w}^{(\ell)}, \nabla_{\mathbf{w}^{(\ell)}} \mathcal{L}^{(\ell)}\big)$ ▷ local update
end for

D.1 Training Algorithm

Algorithm 1 details the PPM-3F training loop summarized in Section 4. Each layer accumulates spike rates over the T timesteps and then updates its own weights from its own per-pixel readout loss. The spikes passed between layers and the membrane carried between timesteps both pass through a stop-gradient, so the backward pass crosses neither layers nor timesteps, and peak activation memory is one layer.

E Additional Results

This appendix collects the per-seed, control, and robustness measurements that the body summarizes. Each subsection is referenced from the sentence it supports.

Table 7: Operating-point ablations (DDD17, mIoU %, seed 42, and * = default used throughout). Plain SGD is the update of Equation (1) without momentum or weight decay.

Depth L (PPM-3F)	1	2	3*	4	5	8
mIoU	32.9	36.5	39.9	40.8	41.2	41.5
Depth L, all variants	3*		5		8	
mIoU, Uniform-3F	30.6		28.9		29.0	
mIoU, PPM-3F	39.9		41.2		41.5	
mIoU, PPM-3F, gradient across layers	40.7		43.6		44.0	
mIoU, BPTT-SNN	43.3		47.7		47.7	
Granularity share of Uniform→BPTT	73%		66%		66%	
Width C (PPM-3F)	32: 37.3		64*: 39.9		128: 41.4	
Update rule	AdamW*		plain SGD			
mIoU, PPM-3F	39.9		36.0			
mIoU, Uniform-3F	30.6		27.0			
mIoU, BPTT-SNN	43.3		39.6			

E.1 Ablations

Table 7 sweeps each axis of the operating point, and the gains from moving away from the defaults lie within the seed spread. Returns shrink with depth ($+3.4$, $+0.9$, and $+0.4$ pp for the steps from two to five layers), consistent with local learning: without cross-layer gradients, extra layers add parallel feature maps rather than a feature hierarchy. The improvement beyond the defaults (≤ 1.6 pp) is within the seed-to-seed range of Table 8. At $C{=}128$, the three variants repeat the pattern, 32.3, 41.4, and 45.2 mIoU for Uniform-3F, PPM-3F, and BPTT-SNN. The granularity gap (9.1 pp against 9.2 at $C{=}64$) and the gap to BPTT-SNN (3.7 pp against 3.4) are both stable as width doubles. The default $T{=}5$ follows the input's five voxel bins.

The ordering does not depend on the optimizer or the learning rate. Under the plain update of Equation (1) run with plain SGD, it persists, 27.0, 36.0, and 39.6 mIoU for Uniform-3F, PPM-3F, and BPTT-SNN. Plain SGD here means no momentum or weight decay, learning rate 0.1 for all three variants, and the layer's normalization unchanged, so its batch statistics stay in the within-layer backward pass. The granularity gap is $+9.0$ pp and the gap to BPTT-SNN 3.6 pp, against 3.4 pp under AdamW at this seed. Under the shared AdamW recipe, a learning-rate search over $\{0.5, 2, 4\}\times$ the shared rate, each variant's rate selected on the validation recording, changes no conclusion. Validation keeps the shared rate for PPM-3F and BPTT-SNN (39.9 and 43.3% on test) and picks $2\times$ for Uniform-3F (31.4%), 8.5 pp below PPM-3F.

Depth and training budget widen the gap to BPTT-SNN without changing the ordering. At $L{=}5$, the three reach 28.9, 41.2, and 47.7 for Uniform-3F, PPM-3F, and BPTT-SNN. The gap to BPTT-SNN is then 6.5 pp against 3.4 at $L{=}3$ on the same seed, and the granularity gap is 12.3 pp against 9.2. At $L{=}8$, they reach 29.0, 41.5, and 47.7, and granularity keeps 66% of the distance at both depths (Table 7). Letting the gradient cross layers recovers 2.4 pp at $L{=}5$ and 2.5 pp at $L{=}8$ (43.6 against 41.2 and 44.0 against 41.5 mIoU). At $5\times$ the epochs, the three variants hold their ordering, 31.1, 40.7, and 44.8% for Uniform-3F, PPM-3F, and BPTT-SNN, so granularity keeps 70% of the distance.

Two frozen-backbone controls locate the uniform variant. A frozen random backbone with trained per-pixel readouts reaches $28.3{\pm}0.04\%$ (three seeds). Uniform modulation lifts the backbone only 2.2 pp beyond untrained features, while per-pixel modulation adds 11.9 pp on the same architecture, so the gain comes from training the backbone. The backbone trained under DECOLLE's published readout is little better: a per-pixel readout trained on it, frozen, reaches $29.1{\pm}0.7\%$ (three seeds), between the random frozen backbone (28.3) and Uniform-3F (30.5).

E.2 Per-Seed Results

Table 8 lists per-seed numbers for the multi-seed results summarized in the body. The PPM–Uniform gap is strictly positive on every seed of both datasets (gap rows of Table 8), and the rankings, PPM-3F above Uniform-3F and BPTT above PPM-3F, hold on all three seeds of both datasets.

Table 8: Per-seed results (mIoU, %, and seeds $\{42, 0, 1\}$). The S-TLLR variant keeps the official layer's trace constants, one dilation rate per layer, and weight standardization in place of batch normalization. Aggregates and the differences quoted in the text are computed from unrounded per-seed values and rounded independently for display. The granularity sweep is monotone on every seed.

Dataset / task	Method	seed 42	seed 0	seed 1	mean±std
DDD17 (mIoU ↑)	BPTT-SNN	43.26	42.97	42.26	42.8±0.5
	Uniform-3F	30.62	31.78	29.01	30.5±1.4
	16×16-3F	36.78	38.10	36.00	37.0±1.1
	8×8-3F	37.80	38.57	38.08	38.2±0.4
	PPM-3F	39.87	38.94	41.57	**40.1**±1.3
	PPM-3F (random $\mathbf{Q}$)	39.46	40.09	39.06	39.5±0.5
	DECOLLE readout (as published)	13.81	13.21	13.90	13.6±0.4
	Pool-then-classify	23.85	22.62	15.41	20.6±4.6
	Label-aligned modulation	32.05	32.32	31.70	32.0±0.3
	Random backbone, frozen, per-pixel readouts	28.28	28.20	28.27	28.3±0.04
	Published-rule backbone, frozen, per-pixel readouts	29.96	28.75	28.70	29.1±0.7
	PPM-3F, gradient across layers	40.69	40.81	42.01	41.2±0.7
	Uniform-3F, gradient across layers	30.69	29.60	28.05	29.4±1.3
	PPM-3F − Uniform-3F (gap)	9.25	7.16	12.56	9.7±2.7
DSEC (mIoU ↑)	BPTT-SNN	40.06	40.15	40.62	40.3±0.3
	Uniform-3F	22.93	21.21	21.92	22.0±0.9
	16×16-3F	34.27	34.04	34.24	34.2±0.1
	8×8-3F	35.77	35.25	36.21	35.7±0.5
	PPM-3F	38.24	37.85	37.58	**37.9**±0.3
	PPM-3F − Uniform-3F (gap)	15.31	16.63	15.67	15.9±0.7
S-TLLR, readout per layer (mIoU ↑)	Pooled	13.89	13.84	13.20	13.6±0.4
	Uniform	27.70	30.26	29.52	29.2±1.3
	Per-pixel	36.41	36.44	36.79	**36.5**±0.2

E.3 Dense Modulation Without Weight Transport: a Controlled Check

This subsection checks that the benefit of dense modulation survives without weight transport. The per-pixel modulation $\mathbf{M}_p = R^\top(\mathrm{softmax}(\mathbf{z}_p) - \mathbf{y}_p)$ uses the readout transpose $R^\top$, an intra-readout symmetric-weight access. Here $R \in \mathbb{R}^{K\times C}$ is the readout matrix, $\mathbf{z}_p = R\,\mathbf{r}_p$ the logits, and $\mathbf{y}_p$ the one-hot label. Proposition 1 and the surrounding analysis depend only on the modulation *field* and the linearity of $\mathcal{J}$, not on $\mathbf{M}$ being the exact readout gradient. A *fixed random* feedback $\mathbf{Q}$ in place of $R^\top$ should therefore retain the benefit while removing the transpose. Here $\mathbf{Q}$ is a fixed random $K\times C$ map per layer, shared across positions, and the readout itself still trains on the dense gradient.

Fixed feedback retains the benefit on both tasks. We check this on a controlled synthetic dense task (Voronoi class regions on 20×20 grids, $K{=}3$, fixed per-class embeddings plus noise as input, two layer-local convolutional layers, 5 seeds). Three variants differ only in the backward modulation path: scalar, per-pixel via $R^\top$, and per-pixel via a fixed random $\mathbf{Q}$. Fixed feedback matches the exact per-pixel gradient: $95.8\pm0.1\%$ mIoU (random feedback) versus $95.8\pm0.0\%$ ($R^\top$), both above scalar modulation ($94.4\pm0.2\%$, with non-overlapping spreads at five seeds). On the real task, the ordering persists: on DDD17 (main protocol), replacing $R^\top$ with a fixed random $\mathbf{Q}$ reaches $39.5\pm0.5\%$ mIoU (three seeds), against 30.5 ± 1.4 for Uniform-3F and 40.1 ± 1.3 for the $R^\top$ form. The granularity benefit therefore does not require weight transport, even within the readout.

E.4 Disparity

The regression task of H4 is DSEC stereo disparity at 128×128, scored by mean absolute error in pixels over labeled pixels (lower is better). It uses the backbone and rule of Section 4 with a one-channel readout per layer, and the epoch is chosen on a held-out slice of the training set. The constant predictor is the training median. Blocks of 16×16 recover about 70% of the per-pixel gain, as on segmentation. The within-image correlation between prediction and target is 0.11 for uniform and 0.30 for per-pixel modulation. Table 9 lists the values per seed.

E.5 Qualitative Results

Figure 6 shows where each variant errs, the visual counterpart of the boundary analysis in Section 5.4. Uniform-3F's failures are not spread evenly. It recovers the large, flat classes about as well as the other rules (on construction+sky it is 3 pp ahead of PPM-3F) and loses the small, boundary-dense

Table 9: Disparity (MAE, px).

Variant	seed 42	seed 0	seed 1	mean
constant predictor	1.70	1.70	1.70	1.70
uniform	1.67	1.63	1.67	1.66
16×16 blocks	1.55	1.52	1.53	1.54
per-pixel	1.48	1.48	1.48	1.48

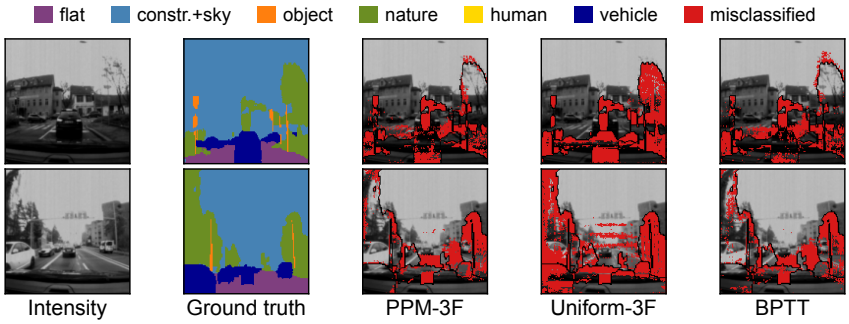


Figure 6: Where each variant errs (two DDD17 test scenes, with misclassified pixels in red, ground-truth class boundaries in black, and the intensity frame for scene context). Uniform-3F loses whole vehicles, poles, and vegetation to the class that surrounds them, while PPM-3F resolves them and tracks BPTT closely.

Table 10: Corruption indexed by $\mathrm{SNR} = \sigma_x/\sigma$, with σ_x=0.086 the standard deviation of the DDD17 voxel grid. All three variants use the same forward-only BN re-estimation, so feature quality alone determines how well it works. BPTT is the off-chip reference, and its +TTA column re-estimates a network trained off-chip. Frozen, Uniform-3F scores higher at the two lowest SNRs, which is what re-estimation repairs. Entries are mean±std over seeds $\{42, 0, 1\}$. **Bold**: best on-chip-trainable.

SNR	PPM-3F		Uniform-3F		BPTT (off-chip trained)	
	frozen	+TTA	frozen	+TTA	frozen	+TTA
∞	**40.1**±1.3	39.8±0.9	30.5±1.4	30.5±0.6	42.8±0.5	41.9±0.3
10	**40.3**±1.7	39.7±1.3	30.6±1.2	30.4±1.1	41.8±0.4	41.6±0.6
5	39.3±1.3	**39.7**±1.4	30.8±1.2	30.3±1.1	41.4±0.2	41.4±0.2
3	32.9±2.6	**40.5**±1.3	29.4±0.6	29.8±1.3	39.2±1.7	41.2±0.2
2	21.1±1.4	**41.0**±0.4	27.2±1.5	28.8±1.6	30.6±5.0	41.4±0.6
1	13.3±1.8	**35.2**±0.9	15.8±5.0	21.5±4.3	15.5±6.4	38.2±1.0

ones. Whole vehicles and poles are assigned to whatever surrounds them (vehicle IoU 16.6 against 39.2, human 0.0 against 17.5, three-seed means).

E.6 Test-Time Adaptation: Full Noise Sweep

This subsection tests two claims: that forward-only BN re-estimation, the same mechanism for every variant, restores dense local features under shift, and that the granularity ordering survives it. After re-estimation, PPM-3F exceeds Uniform-3F at every SNR (+9.3 to +13.8 pp, three-seed means). Re-estimation returns the dense features to their clean accuracy under Gaussian input noise (Hendrycks & Dietterich, 2019): at SNR 2, from 21.1 to 41.0%, within the seed spread of the clean value (the training recipe includes Gaussian input noise, Section D). BPTT's +TTA column applies the same re-estimation to the network trained off-chip. Table 10 reports the full sweep. One pass over the unlabeled corrupted stream refreshes the BN moments, and the adapted model is then scored on that stream.

F Accuracy at the ASIC Design's Precision

Section F reports the accuracy of PPM-3F trained at the ASIC design's precision. Table 11 is the measurement behind the precision sentence of Section 6. The design's forward path is 8-bit and its backward path 32-bit, so we train with weights, the membrane register, activations, and the readout

Table 11: PPM-3F on DDD17 with an 8-bit forward path and a 32-bit backward path (mIoU, %). The paired difference over seeds, -0.7 ± 2.6 pp (Table 11a), is not distinguishable from zero at three seeds, and no single component costs accuracy at seed 42 (Table 11b).

seed	main recipe	8-bit forward	diff
42	39.9	39.9	$+0.0$
0	38.9	40.3	$+1.4$
1	41.6	37.9	-3.6
mean	$40.1_{\pm 1.3}$	$39.4_{\pm 1.3}$	$-0.7_{\pm 2.6}$

(a) Full datapath, per seed.

quantized	mIoU
weights	40.1
+ membrane	39.7
+ activations	39.9
+ readout	39.9

(b) Cumulative components, seed 42.

quantized to 8 bits and gradients at 32 bits, and compare against the main recipe of Table 1 at the same three seeds.