Beyond Gene Counts: Self-Supervised Cell Representations from Individual Molecules

Anonymous authors
Paper under double-blind review

Abstract

Image-based spatial transcriptomics records the position of every mRNA molecule inside each cell, yet representation learning for these data still reduces each cell to a vector of gene counts, discarding the subcellular organization of these molecules. Where transcripts sit within a cell, near the nucleus, along the membrane or in protrusions, varies between cells in ways a count vector cannot record. We formulate subcellular representation learning as self-supervised learning over point clouds of molecules, each labeled with one of thousands of genes, observed in an arbitrary orientation and with incomplete detection. We present moments of a learned measure (Molm), the first self-supervised model for this setting, which represents a cell as a hierarchy of molecules, the localization programs they form, and the cell as a whole. At every layer, a transformer groups molecules into programs, and the geometry of each program is computed exactly, respecting rotations and reflections by construction. We also introduce a benchmark with known subcellular ground truth, built on real cells whose molecule positions are resampled from known localization programs, and use it to profile performance of unsupervised discovery of subcellular structures. Molm recovers a cell subtype defined only by molecular arrangement at 0.780 balanced accuracy against 0.426 for the best existing method, and localization patterns at 0.448 against 0.259. On real data it leads all label-free methods in cell-type prediction while additionally capturing cell shape, which count-based and foundation-model embeddings do not encode.

1 Introduction

Image-based spatial transcriptomics is used to map where cells sit in a tissue and which genes they express. These platforms, such as MERFISH, CosMx and Xenium, do not measure cells directly. They detect individual mRNA molecules and record where each one is (He et al., 2022; Janesick et al., 2023). Cells are defined afterwards, by segmentation, and the standard analysis then sums each cell's molecules into a vector of gene counts. That summary keeps which genes a cell expresses, but discards where inside the cell each transcript was found.

This discarded information carries biology. Transcripts gather at the periphery, around the nucleus or in protrusions, and when many molecules shift together toward one compartment, the shift can mark a different cell state (Moor et al., 2017). Existing representation learning methods for spatial transcriptomics mostly ignore it. Spatial methods describe each cell by its count vector, combined with its position in the tissue and its interactions with neighboring cells (Singhal et al., 2024; Tejada-Lapuerta et al., 2025), and single-cell and foundation models start from the same count vector (Lopez et al., 2018; Hao et al., 2024; Cui et al., 2024; Theodoris et al., 2023; Wen et al., 2024). As a result, two cells with identical counts but different internal arrangements receive the same representation.

We therefore learn representations of cells directly from their molecules. We treat each cell as a set of points in the plane, where each point is one molecule tagged with its gene (Section 2). Three features of this data shape our design. First, a cell's orientation on the slide is arbitrary: rotating or flipping the cell does not change it, so its representation should not change either, while still capturing directions within the cell, such as polarity. Second, not every molecule is detected. Two random subsets of a cell's molecules are therefore two views of the same cell, which gives a natural signal for self-supervision. Third, there are thousands of genes, and the patterns that matter involve

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.

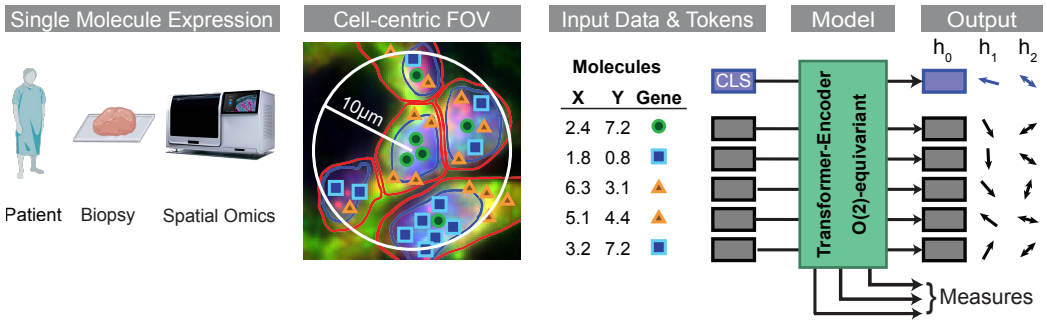


Figure 1: MOLM in outline. Molecules in a 10 µm field of view around each cell become tokens carrying their gene and position, alongside a CLS token. At every layer the network learns which molecules belong together, grouping them into programs (the *measures*), and then computes, rather than learns, the geometry of each program around every molecule: how many of its molecules lie nearby, in which direction, and how they are spread. The output has a scalar stream h^0 and two directional streams h^1 and h^2 (arrows) that rotate with the cell. Because the grouping is recomputed at every layer, the programs grow more abstract with depth.

groups of genes that localize together. These groups must be learned, rather than each gene treated separately.

Related work. Beyond the count-based methods above, which take the cell as the smallest unit, a few methods work below the cell. Tools for single-molecule imaging quantify localization gene by gene (Imbert et al., 2021), and SubCellSpace (Wouters et al., 2026) learns localization patterns at scale, but again separately for each gene, so structure shared across genes is out of reach. FOCUS (Lu et al., 2025) connects nearby transcripts into a graph and trains a shallow graph network with cell-type labels, which discards the relative geometry that carries localization. pyTrance (Strenger et al., 2026) uses a single-layer graph network to detect pairs of co-localizing transcripts, rather than to represent the cell. None of these is self-supervised over all of a cell's molecules, and none represents directional geometry in a way that is unaffected by rotating or flipping the cell.

Our approach. We present MOLM, a self-supervised model that represents a cell as a hierarchy: individual molecules, the localization programs they form, and the cell as a whole (Figure 1). MOLM is a transformer (Vaswani et al., 2017) over molecule tokens, whose attention takes the distances between molecules into account. At every layer, it learns which molecules belong together by giving each molecule a weight in a set of programs, which we call *measures*. A program might, for instance, collect a group of genes whose transcripts line the membrane. MOLM then computes the geometry of each program exactly, rather than learning it: around each molecule, how dense the program's molecules are, in which direction they concentrate, and whether they spread evenly, line up along a membrane, or form a cluster (Section 3). Because this geometry is computed, the network does not need to discover it from data, each quantity keeps a fixed meaning, and all of them rotate and reflect with the cell by construction. Classical geometric descriptors use fixed weights (Mallat, 2012; Willatt et al., 2018); here the weights are learned and recomputed at every layer, so the programs grow more abstract with depth (Section 4). Training combines self-distillation with a second objective: the geometry computed from two random subsamples of the same cell should agree (Section 4.2).

Real data never reveal the true localization programs, so we also build a synthetic benchmark. It keeps real cells, with their boundaries, genes and counts, and resamples only where their molecules sit, from known programs. It includes a hidden cell subtype that differs only in molecule positions, which no count vector can distinguish (Appendix E). MOLM recovers this subtype at 0.780 balanced accuracy, against 0.426 for the best subcellular baseline, and the localization pattern at 0.448 against 0.259 (Table 2). On three real datasets, it leads all label-free methods in cell-type prediction and additionally predicts cell shape (Table 1).

Our contributions are:

- We formulate, for the first time, subcellular representation learning as self-supervised learning over the molecules of a cell, each labeled with its gene, and identify the challenges that define it (Section 2).
- We present Molm, the first self-supervised method for this setting, which learns a hierarchy of localization programs and computes their geometry exactly and independently of orientation (Sections 3, 4 and 4.2).
- We introduce a benchmark with known subcellular ground truth, built on real cells, and use it to profile performance of unsupervised discovery of subcellular structures (Appendices E and G).
- We demonstrate that Molm outperforms existing approaches in recovering subcellular organization, while leading all label-free methods in predicting cell-type and morphological features (Section 5).

2 Problem setting

Data. A cell imaged at subcellular resolution is a set of detected molecules,

$$C = \{(p_j, e_j)\}_{j=1}^{L}, \qquad p_j \in \mathbb{R}^2, \quad e_j \in \{1, \ldots, G\}, \tag{1}$$

where p_j is the position of molecule j in a frame centered on the cell and e_j is its gene. Typically a cell has $L \sim 10^3$ molecules drawn from $G \sim 10^4$ genes. The molecules have no natural order, their number varies from cell to cell, and each carries a gene label; formally, such data is a realization of a *marked point process*.

Goal. We seek an encoder f, trained without labels, that maps a cell to representations at two levels: one vector $z_j = f(C)_j$ for each molecule, describing that molecule in the context of its cell, and one vector z_C for the cell as a whole. Between these levels, the encoder should form intermediate groupings of molecules, localization programs, from which both are built.

Constraints from how the data are acquired. Three properties of the data constrain f. First, a cell's orientation reflects how the tissue was cut and mounted, not the cell itself, and a section can be mounted face up or face down. Cell-level features must therefore be unchanged by rotations and reflections, that is, invariant to the orthogonal group $O(2)$, while features that describe directions within the cell, such as a polarity axis, must rotate with it, that is, be equivariant. Second, each transcript is detected with probability less than one: detection is a random *thinning*, so a random subsample of a cell's molecules is another observation of the same cell rather than a corrupted one, and its representation should agree with that of the full cell. Third, with thousands of genes, the structures of interest, such as sets of genes that co-localize in a compartment, are defined by groups of genes and depend on what else the cell contains, so programs must be computed from a contextual representation rather than from gene identity alone. Molm satisfies the first constraint by construction (Section 3), uses the second as its self-supervised signal (Section 4.2), and meets the third through its architecture (Section 4).

What the representation should capture. We ask three things of the learned representations, and Section 5 tests each with frozen linear probes. *(i) Cell identity.* Reading positions must not cost the expression information that count vectors already carry, so z_C should predict cell type at least as well as count-based representations. *(ii) Subcellular organization.* Two cells that differ only in where their molecules sit should receive different representations, and each molecule's representation should reflect where in the cell it sits. Because real data never reveal the true localization programs, we test this on a benchmark in which they are known: at the cell level, by recovering a hidden subtype defined only by molecular arrangement, and at the molecule level, by recovering the compartment each molecule was drawn from (Appendix E). *(iii) Cell geometry.* Since molecule positions trace out the cell, z_C should also carry the cell's size and shape, which count vectors can reflect only through transcript number.

3 Geometry as moments of a measure

Molm computes local geometry exactly, as moments of learned measures. Let x_j be the encoder's representation of molecule j and B a learned map giving weights $b_j = B(x_j) \in \mathbb{R}$. Each cell then carries a signed measure $\mu = \sum_j b_j \delta_{p_j}$, in which a positive b_j counts molecule j toward a structure and a negative one toward its complement. We use its positive and negative parts (its Jordan decomposition), $\mu^{\pm} = \sum_j b_j^{\pm} \delta_{p_j}$ with $b^{\pm} = \max(\pm b, 0)$, together with the counting measure $\sum_j \delta_{p_j}$.

3.1 Local moments

Identify $\mathbb{R}^2$ with $\mathbb{C}$, writing $z_j = p_{j,x} + i\, p_{j,y}$, and let a_{ij} be Gaussian proximity weights of width σ, normalized over the other molecules,

$$a_{ij} \;=\; \frac{\exp\!\big(-\|p_i - p_j\|^2/2\sigma^2\big)}{\sum_{k\neq i} \exp\!\big(-\|p_i - p_k\|^2/2\sigma^2\big)} \quad \text{for } j \neq i, \qquad a_{ii} = 0. \tag{2}$$

For each of the three measures $\nu = \sum_j \nu_j \delta_{p_j}$, the local moments at molecule i through second order are

$$\begin{aligned} M_i^0 &= \textstyle\sum_j a_{ij}\nu_j, & D_i^1 &= \textstyle\sum_j a_{ij}\nu_j (z_j - z_i), \\ D_i^2 &= \textstyle\sum_j a_{ij}\nu_j (z_j - z_i)^2, & N_i &= \textstyle\sum_j a_{ij}\nu_j |z_j - z_i|^2. \end{aligned} \tag{3}$$

Up to normalization, D^1 and D^2 are the gradient and traceless Hessian of the smoothed measure at p_i (Lindeberg, 1994). Under a rotation $z \mapsto e^{i\theta} z$, M^0 and N are invariant, D^1 rotates by $e^{i\theta}$ and D^2 by $e^{2i\theta}$; we call these types 0, 1 and 2. All four are computed in one pass from uncentered aggregates (Appendix A).

3.2 Geometric invariants

Write $M_i^{0,\pm}$ for the mass of $\mu^{\pm}$ at p_i and $Z_i = \max\{M_i^{0,+} + M_i^{0,-}, \varepsilon\}$ for their combined local mass, floored by a small $\varepsilon > 0$. We form four invariants,

$$d_i = \frac{N_i}{\sigma^2 Z_i}, \qquad \kappa_i = \frac{|D_i^2|}{\sigma^2 Z_i}, \qquad |H_i| = \frac{2|D_i^1|}{\sigma^2 Z_i}, \qquad \alpha_i = \frac{\mathrm{Re}\big[\big(\overline{D_i^1}\big)^2 D_i^2\big]}{|D_i^1|^2\, |D_i^2|}, \tag{4}$$

which are, respectively, local intrinsic dimension, anisotropy, mean curvature, and the alignment between the gradient and the tangent axis (Appendix C). The first two are standard point-cloud estimators (Demantké et al., 2011) and separate the basic local structures: a point mass at the query gives $(d, \kappa) = (0, 0)$, a line through it $(1, 1)$, and a filled plane $(2, 0)$.

4 The architecture

Molm is a stack of pre-norm transformer encoder layers (Vaswani et al., 2017; Xiong et al., 2020) over a scalar stream h^0. Each layer has content attention, a feedforward and, last, the measurement, which is the only block that touches the equivariant streams. A cell is tokenized as its molecules plus a cls token, and each molecule is embedded from its gene alone. Position enters through the measurement and through a distance term in the attention logit,

$$s_{ij} \;=\; \frac{q_i \cdot k_j}{\sqrt{d_{qk}}} \;-\; \frac{\|p_i - p_j\|^2}{2\tau^2}, \tag{5}$$

with a learned length scale τ per head. The distance term is absorbed into the query and key features, so fused attention computes Equation (5) exactly (Wu et al., 2026). We use eight layers of width 512, $C = 64$ measures and $\sigma = 2.5\,\mu\mathrm{m}$ for the real data task and for the architecture ablation. The loss and lengthscale ablations use four layers at width 128, $C = 32$.

4.1 The measurement layer

With $x_j = \mathrm{LN}(h_j^0)$, one layer computes C measures and their moments:

$$b_j = B\,x_j \in \mathbb{R}^C, \qquad C \text{ measures} \tag{6}$$

$$\nu_j = \left(1,\ b_j^+,\ b_j^-\right) \in \mathbb{R}^W, \qquad W = 1 + 2C \tag{7}$$

$$V_j = \nu_j \otimes \left(1,\ z_j,\ z_j^2,\ |z_j|^2\right), \qquad 6W \text{ real columns} \tag{8}$$

$$A_i = \textstyle\sum_j a_{ij} V_j, \qquad a_{ij} \text{ as in Equation (2)} \tag{9}$$

Here ν_j holds molecule j's mass in the counting measure and in the two halves $\mu^{\pm}$ of each of the C measures, and the blocks of A_i give the moments of Equation (3) (Appendix A); since V_j does not depend on i, this is a single matrix product. The layer *defines* the equivariant streams as these moments, $h_i^1 = \{D_i^1\}$ and $h_i^2 = \{D_i^2\}$, rather than carrying them as learned state, and nothing else writes into them. Because B reads the contextual representation rather than the gene embedding, a measure can combine genes whose membership depends on the rest of the cell: at the first layer it is close to a fixed gene list, while deeper layers form measures that are not functions of the gene alone.

Geometry returns to h^0 at one site. The invariants of Section 3.2 and a learnable invariant Gram readout of each stream condition a zero-initialized FiLM residual on the scalar stream (Appendix B). FiLM lets geometry scale token features as well as shift them, and the zero initialization starts training from a plain transformer.

Cell-level tasks generally utilize as representation the h^0 stream of the CLS token. Where the h^k streams are used, we utilize their Gram components. Molecule-level tasks use measures of intermediate layers.

4.2 Objectives

The scalar stream is trained by self-distillation in the form of DINO and iBOT on the cls and masked molecule tokens (Caron et al., 2021; Zhou et al., 2022), with KOLEO terms (Sablayrolles et al., 2019) spreading the cls token's streams against collapse. The equivariant streams are trained to agree across two random subsamples of a cell, which by Section 2 are both observations of it. Agreement is measured by a cosine per measure and per order (Equation (14)). A second term on those streams scores learnable Gram features against a prototype vocabulary, reaching the complete $O(2)$-invariant content of a stream. An expression decoder supervises the cls token, and the cosine extends to the intermediate layers' streams, so the moments of every layer are supervised. Appendix D gives the full objective.

5 Experiments

Data. We evaluate on three datasets, each from a different platform and panel. Colon is mouse colon imaged with MERFISH (Cadinu et al., 2024); we use the ten healthy day-0 sections of a colitis time course, with a 943-gene panel after removing 47 control probes. Its cell-type labels are hierarchical clusters at three resolutions, with 8, 66 and 72 classes; we report the coarsest and give all three in Table 11. Lung is human non-small-cell lung carcinoma imaged with CosMx over a 960-gene panel, with 22 annotated cell types (Bruker Spatial Biology, 2022; He et al., 2022). DCIS is human breast carcinoma imaged with Xenium over a 313-gene panel, with 19 annotated cell types (Janesick et al., 2023). Each dataset is split by section, so no section contributes cells to both training and test: six sections train and four test on Colon, and one serial section trains and its neighbor tests on Lung and DCIS. The splits therefore measure generalization across sections of the same tissue rather than across individuals. We discard cells with fewer than 50 transcripts and cells whose center lies within 10 µm of a section boundary, so that every retained cell is seen in full. Appendix F gives split sizes.

Baselines. We compare against nine methods, all run by us on the same splits, gene panels and filtered cells (Appendix G). Two represent a cell by its counts alone: Seurat, the standard log-normalization and PCA pipeline (Hao et al., 2024), and scVI, a variational model of the counts (Lopez et al., 2018). BANKSY augments each cell's counts with those of its spatial neighbors (Singhal et al.,

Table 1: Cell-type probing and morphology regression on three real datasets. Cell type is balanced accuracy from a logistic probe on each embedding; size and shape are ridge R^2, averaged over the descriptors of that group. Colon's cell-type column reports Tier 1 of its three-level hierarchy. Rows are grouped by what each method uses. Seurat is the standard log-normalization and PCA pipeline. The Molm rows are its three outputs: the scalar stream h^0, and the two directional streams probed through their rotation-invariant Gram features (Appendix H). scVI, FOCUS and Molm are averaged over three training seeds; standard errors are given in Tables 11 to 13. Bold marks the best entry in each column and any entry within one standard error of it. *FOCUS trains on the cell-type labels, so it is excluded from bolding in the cell-type columns.

	Cell type			Size			Shape		
Method	Lung	DCIS	Colon	Lung	DCIS	Colon	Lung	DCIS	Colon
Counts only									
Seurat (PCA)	0.750	0.800	0.935	0.199	0.607	0.445	0.032	0.038	0.077
scVI	0.753(1)	0.823(1)	0.879(4)	0.153(4)	0.413(8)	0.450(6)	0.025	0.019(1)	0.123(1)
Counts with tissue context									
BANKSY	0.723	0.800	0.949	0.415	0.661	0.466	0.024	0.049	0.081
Foundation models, zero-shot									
CellPLM	0.617	0.790	0.870	0.299	0.468	0.341	0.028	0.027	0.057
scGPT	0.638	0.810	0.882	0.263	0.607	0.627	0.035	0.042	0.159
Geneformer	0.652	0.813	0.902	0.331	0.677	0.751	0.035	0.051	0.174
Nicheformer	0.643	0.812	0.894	0.311	0.645	0.643	0.035	0.046	0.166
Subcellular positions, trained on cell-type labels									
FOCUS*	0.732(4)	0.837(1)	0.934(1)	0.639(20)	0.730(11)	0.811(5)	0.048(1)	0.061(1)	0.167(2)
Subcellular positions, self-supervised (ours)									
Molm h^0	**0.758(3)**	**0.840(1)**	**0.955(1)**	**0.905(3)**	**0.915**	**0.952(1)**	0.690(5)	0.520(8)	0.692
Molm h^1	0.669(1)	0.800(1)	0.833(8)	0.888(1)	0.843(5)	0.893(2)	0.677(5)	0.436(7)	0.527(4)
Molm h^2	0.699(2)	0.808(1)	0.912(8)	**0.906(3)**	0.863(5)	0.922(2)	**0.819(3)**	**0.618(11)**	**0.732(4)**

2024). Four transcriptomic foundation models are used zero-shot: CellPLM, scGPT, Geneformer and Nicheformer (Wen et al., 2024; Cui et al., 2024; Theodoris et al., 2023; Tejada-Lapuerta et al., 2025). Two read subcellular positions: FOCUS (Lu et al., 2025), which is semi-supervised and trains on the cell-type labels, and pyTrance (Strenger et al., 2026), which we evaluate on the synthetic benchmark only. Baselines with a choice of width are reported at 50 dimensions, fixed before the results were seen, and Table 11 gives every width.

Tasks. The tasks below test the three requirements of Section 2: cell type tests (i), cell subtype and localization pattern test (ii), and size and shape test (iii). Every method produces a frozen embedding, per cell for the first four tasks and per molecule for the last, which we read with the same linear probes, fit on the training section and scored on the held-out one (Appendix H). On real data there are three tasks. *Cell type* predicts the annotated cell type with a logistic probe, scored by balanced accuracy. *Size* regresses the area and perimeter of the convex hull of the cell's transcripts, and *shape* their circularity and eccentricity, each with a ridge probe scored by R^2. On synthetic cells, where the generating factors are known, two further tasks ask where molecules sit: *cell subtype* predicts the hidden subtype jointly with the cell type (57 classes), and *localization pattern* predicts, for each molecule, which of the seven compartments of Table 7 it was drawn from, with a separate probe fit within each gene.

Cell type. The scalar stream leads every method that did not see the labels, on all three datasets, though narrowly (Table 1). This is expected: cell type is largely an expression phenotype, and a count vector already carries most of it. The test is whether an encoder built on positions keeps that signal, and it does. Giving every baseline its best embedding width changes one column: on Lung, Seurat at 30 dimensions reaches 0.761, within the seed variation of our 0.758 (Table 11). Our lead widens at the finer Colon label levels, from 0.006 at Tier 1 to 0.043 at Tier 2.

Seurat beats all four transcriptomic foundation models on two of the three datasets. FOCUS trains on these labels and is excluded from the comparison. Class counts differ across datasets, so a column

Table 2: Recovery of the generating factors on synthetic DCIS, balanced accuracy from a logistic probe, three seeds per method. Columns are the 19 cell types, the 57 cell type × subtype combinations, and the seven localization patterns. Molm (measure, L4) reads the layer-4 measures directly. The pattern column is scored on the 126 genes every method admits and on a different cell set from the first two columns, so it is not directly comparable to them. *FOCUS trains on cell-type labels and is excluded from bolding in the two label columns. Conventions as in Appendix I.

Method	Cell type	Cell subtype	Pattern
Molm (h^0)	**0.894(2)**	**0.780(3)**	0.402(2)
Molm (measure, L4)	–	–	**0.448(2)**
pyTrance	0.532(2)	0.409(2)	0.259(2)
FOCUS*	0.857	0.426(2)	0.248(3)

compares down its length and not across. Table 11 repeats the comparison at every embedding width and at all three Colon tiers, and the geometric claim is tested on synthetic cells, where the generating factors are known (Table 2).

Cell size and shape. Both groups of targets are computed from the transcript cloud that Molm reads, so size mainly checks that spatial extent survives the encoder. It does: Molm is highest on all six size descriptors (Table 12), and FOCUS, the only baseline that reads positions inside the cell, is the strongest baseline. Shape is the harder test. Count-based methods predict size reasonably, because size tracks transcript count, but they do not recover shape: no baseline exceeds $R^2 = 0.22$ on any shape descriptor, while the best Molm readout reaches 0.58 to 0.92 (Table 13). FOCUS reads positions yet stays below 0.21, so access to positions is not enough on its own. The three readouts divide as the construction predicts. The type-2 stream is the strongest on five of the six shape descriptors, reaching 0.915 on Lung eccentricity against 0.736 for the scalar stream, and its advantage is largest on eccentricity, a function of the second moment that h^2 carries. Circularity depends on the cell boundary, which low-order moments do not determine.

Subcellular structure on synthetic cells. Real data never reveal the true localization programs, so we built a benchmark in which they are known (Appendix E). It keeps the real cells of two DCIS sections, with their cell and nucleus boundaries, gene panel, annotations and per-cell counts, and resamples only the molecule positions. Each molecule is placed in one of seven compartments built from its own cell's boundaries: random, intranuclear, extranuclear, nuclear edge, perinuclear, cell edge and pericellular, following the pattern definitions of Imbert et al. (2021). How a gene's molecules are spread over these compartments is set by eight localization programs shared across genes, so genes that load the same program move together. Within each cell type, cells are assigned at random to one of three subtypes that differ only in how active each program is. The subtype is therefore independent of expression, shape and position by construction, and only the arrangement of molecules inside the cell can reveal it.

Molm recovers the hidden subtype at 0.780(3) balanced accuracy against 0.426(2) for the best baseline, and its layer-4 measures recover the localization pattern at 0.448(2) against 0.259(2) (Table 2), margins far above the seed variation. The measures also beat the embedding they feed, 0.448(2) against 0.402(2) for the pooled scalar stream, and the layer matters: recovery rises through the encoder, peaks at an interior layer and falls again (Figure 2A, Table 14). The errors are the expected ones (Figure 2B): compartments with a sharp boundary, such as the nucleus, the nuclear edge and the cell edge, are recovered best, while the random and extranuclear patterns, which spread across the whole cytoplasm, are confused most. Two differences between methods are not controlled. Training budgets are unequal, with Molm trained for 215 epochs against 50 for pyTrance and 10 for FOCUS, and FOCUS's embedding is dominated by cell size, carrying transcript count at $R^2 \approx 0.99$.

Ablations. Table 3 removes parts of the geometric pathway at full size. Removing the measurement costs 0.063 on cell subtype and 0.147 on scalar-stream shape, and removing the type-1 moments costs nearly as much. Removing only the type-2 moments leaves cell subtype unchanged but costs 0.069 on shape, consistent with shape depending on the second moment. One arm without the measurement, which instead embeds each molecule's distance from the cell center, recovers part of the shape signal on the scalar stream (0.454 against 0.476), but the full model reaches 0.550 from

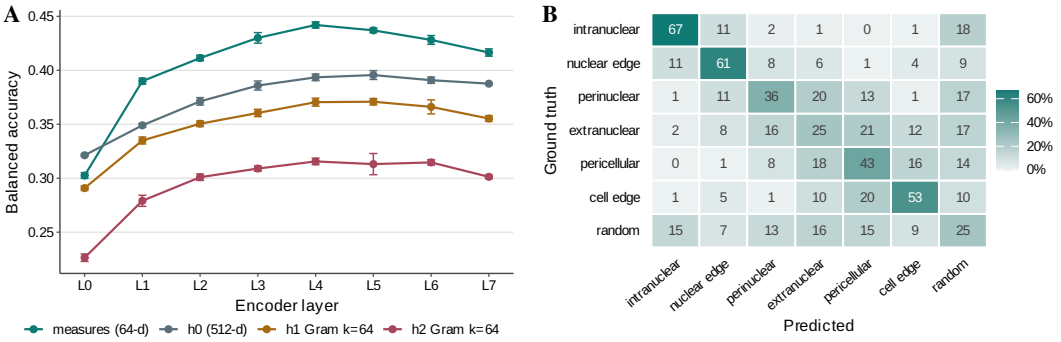


Figure 2: Layer-wise localization signal and classification errors. **(A)** Pattern recovery from each layer's measures, read at each molecule, against the scalar stream for reference; h^1 and h^2 are probed through their Gram features, as in Table 1. Recovery rises through the encoder, peaks at an interior layer and falls again. **(B)** Confusion matrix of the seven localization patterns predicted from the layer-4 measures; rows are the true compartment, columns the predicted one.

Table 3: Ablations of the geometric pathway on synthetic DCIS, at $d = 512$ and eight layers. The first row is the full model; indentation shows nesting, and − and + mark removals and additions relative to the parent row. Shape is given for all three readouts and the other columns for the scalar stream. Blank cells are arms that have no such stream to read. Every row is three seeds. Conventions as in Appendix I.

Ablation	Cell type	Cell subtype	Size	Shape		
	h^0	h^0	h^0	h^0	h^1	h^2
Molm (full)	0.896(1)	0.787(2)	0.932(1)	0.476	0.461(5)	0.550(2)
− type-2 moments and losses	0.896(1)	0.789(2)	0.923(3)	0.407(6)	0.404(6)	–
− type-1 moments and losses	0.875	0.743(5)	0.889(1)	0.329(3)	–	–
− measurement	0.873	0.724(7)	0.905(2)	0.329(4)	–	–
+ radial embedding	0.869(1)	0.712(2)	0.949(1)	0.454(5)	–	–
+ xy embedding, − equivariance	0.869(1)	0.697(2)	0.949(1)	0.390(13)	–	–
− attention proximity term	0.857(1)	0.507(21)	0.942(3)	0.300(26)	–	–
− attention proximity term	0.885(1)	0.747(2)	0.937(3)	0.508(2)	0.488(3)	0.554(3)

its type-2 stream, which that arm does not have. The distance term in attention matters little while the geometry is intact (0.040 on cell subtype) and a great deal once equivariance is removed (0.190). Arms train for equal steps rather than equal compute.

Table 4 varies the length scale at the smaller configuration. The scale trades cell subtype against shape: the smallest scale, 1.25 µm, is best on cell subtype (0.742) and worst on shape (0.354), while larger, multi-scale and learned scales reach 0.45 to 0.50 on shape at a cost of 0.01 to 0.04 on cell subtype relative to 2.5 µm. Cell type varies by less than 0.01 across all arms, and size improves modestly with scale. The fixed 2.5 µm scale used elsewhere sits between these, second on cell subtype but behind the larger scales on shape; we screened the scale at this configuration and did not repeat the sweep at full size. Among the objectives (Table 15), masking matters most, costing 0.148 on cell subtype and 0.222 on shape when removed; the invariant cross-entropy costs 0.030 on cell subtype, and no other single removal separates from the reference (Appendix J).

6 Discussion

We have argued that the molecule, not the count vector, is the natural unit of image-based spatial transcriptomics, and presented Molm, a self-supervised model that represents a cell as a hierarchy of molecules, localization programs and the whole cell, with the geometry of each program computed exactly. On a benchmark with known ground truth, Molm recovers subcellular organization that existing methods miss, and on three real platforms it does so while matching or exceeding count-based and foundation-model embeddings on cell identity.

Table 4: Length-scale ablations on synthetic DCIS, at $d = 128$ and four layers, with 32 measures. Each row uses one fixed scale σ, eight fixed scales spanning the stated range, or learned scales from the stated initialization. Lengths are in µm; the fixed 2.5 µm arm is the scale used elsewhere. Every row is three seeds. Conventions as in Appendix I.

Ablation	Cell type	Cell subtype	Size	Shape
1 fixed, $\sigma = 1.25$	0.869	0.742(3)	0.898(3)	0.354(5)
1 fixed, $\sigma = 2.5$	0.864	0.683(9)	0.904(10)	0.403(18)
1 fixed, $\sigma = 5$	0.862(1)	0.663(5)	0.916(2)	0.463(2)
1 fixed, $\sigma = 10$	0.862(1)	0.659(9)	0.925(2)	0.448(10)
8 fixed, 1.25–10	0.866(1)	0.675(9)	0.927(5)	0.481(11)
1 learned, init 2.5	0.860	0.648(6)	0.920(2)	0.500(4)
1 learned, init 5	0.862	0.657(4)	0.928(8)	0.469(3)
8 learned, init 1.25–10	0.864(1)	0.661(7)	0.921(6)	0.487(6)

What the results show. Three findings carry the argument. First, reading positions does not cost expression information: an encoder that sees each molecule rather than a count vector still leads on cell type, the task on which count vectors are strongest. Second, arrangement alone is learnable without labels: a subtype that differs only in where molecules sit, and is invisible to every count-based method by construction, is recovered from a self-supervised representation, and so is the compartment of individual molecules. Third, the structure of the model is visible in what it learns. Shape is carried best by the type-2 stream, which holds the second moments that determine elongation, and localization is carried best by the measures of interior layers rather than by the input or the final embedding, as a hierarchy of increasingly abstract programs predicts.

Limitations. Several limitations qualify these results. Molm works in the plane of the section and ignores depth, although the platforms it reads increasingly resolve molecules in three dimensions. It inherits the cell segmentation: which molecules belong to a cell is decided before the model sees them, so segmentation errors enter the representation. Our splits separate sections rather than donors, and therefore measure generalization across sections of the same tissue rather than across individuals. The morphology targets on real data are computed from the transcript cloud that Molm reads, which is why we treat size as a check and shape as the stronger test. The molecule-level and arrangement-only tasks are evaluated on synthetic cells only, because real data carry no ground truth for them. The synthetic benchmark encodes one model of localization, designed by us, with seven compartments and a fixed number of programs; results on it show that Molm can recover such structure, not that real cells are organized this way. Linear probes measure only what is linearly available in an embedding, and we fixed their settings for all methods rather than tuning them. Finally, training budgets were not matched across methods, and the length scale was screened at a smaller configuration and not re-swept at full size.

Future work. Several extensions follow directly. The Colon dataset is a colitis time course of which we use only the healthy arm; probing disease stage within a cell type would test, on real data, whether subcellular organization tracks cell state. The directional streams can be tested directly by predicting polarity vectors, such as the offset from a cell's center to its nucleus, which no count-based representation can express. The learned programs should be compared against known compartment markers, which would turn the geometric readouts into biological annotations. Training and testing on different donors, which the Lung dataset allows, would measure generalization across individuals. Pretraining a single model across many sections, panels and platforms would test whether the learned programs transfer between gene panels, a step toward a foundation model for subcellular organization rather than for counts alone. Because the measures assign every molecule a weight in each program, they could also help delineate cells and compartments directly from molecules, easing the dependence on segmentation noted above. And because the measurement is defined for any point set, the same construction extends to three dimensions and, beyond single cells, to groups of neighboring cells, where the same moments would describe tissue rather than subcellular organization.

AI use statement

In this work, we used generative AI tools to:

1. **Design and give feedback on methodology and experiments.** The research questions and the design are ours. We used agents to sharpen what each experiment could establish: to work out which ablation arms isolate a single mechanism and which unavoidably remove several at once, and to write the expected outcome of a run down in advance so that a disagreement became a finding rather than a reason to adjust the reading.
2. **Implement methods.** The model, the training pipeline and the evaluation are written with coding agents under human supervision. We have reviewed the code and take full responsibility for its correctness.
3. **Generate synthetic data sets.** The benchmark of Appendix E is agent-implemented to our specification. Its ground truth is known by construction, so its correctness is checkable, and we checked it.
4. **Clean and reformat datasets.** Agents wrote the evaluation pipeline and the scripts that turn its output into the tables reported here.
5. **Interpret results.** We used agents to read the measurements against the claims, including to argue that a result we had published internally did not support the mechanism we had attached to it. That claim was withdrawn.
6. **Propose and refine hypotheses.** Several of the paper's claims were narrowed or dropped after an agent argued the evidence did not carry them.
7. **Identify relevant literature.** Agents searched beyond our own reading, including for prior art that forecloses framings we had considered.
8. **Write and edit the paper.** This goes beyond correcting grammar. Agents drafted prose, tables and captions throughout, under a convention that every drafting decision is recorded with its reasoning and accepted, deferred or rejected by an author. The argument, the claims and every number are ours, and we stand behind the text.

We have not used generative AI tools to:

1. Support qualitative and thematic data analysis.
2. Help develop theoretical models or conceptual frameworks.
3. Formulate mathematical claims.
4. Provide critical ingredients for proving mathematical claims.
5. Assist in the writing of proofs.

This required disclosure task is not applicable to this work:

1. Assist with translation.

Reproducibility statement

Anonymized source code for the model, training, the synthetic data generator and the evaluation is provided with the submission, and the appendices specify the architecture, the objective and the generator in enough detail to reimplement them. All three datasets are public; Appendix F gives the panels, the sections used to train and to test, the split sizes and what the section-level split does and does not control. Every embedding in the paper, ours and each baseline's, is frozen and read by the same probes at library defaults, tuned for no method; Appendix H gives that protocol in full.

References

Bruker Spatial Biology. CosMx SMI NSCLC FFPE dataset. Public reference dataset, 2022. URL https://brukerspatialbiology.com/products/cosmx-spatial-molecular-imager/ffpe-dataset/nsclc-ffpe-dataset/.

Paolo Cadinu, Kisha N. Sivanathan, Aditya Misra, Rosalind J. Xu, Davide Mangani, Evan Yang, Joseph M. Rone, Katherine Tooley, Yoon-Chul Kye, Lloyd Bod, Ludwig Geistlinger, Tyrone Lee, Randall T. Mertens, Noriaki Ono, Gang Wang, Liliana Sanmarco, Francisco J. Quintana, Ana C. Anderson, Vijay K. Kuchroo, Jeffrey R. Moffitt, and Roni Nowarski. Charting the cellular biogeography in colitis reveals fibroblast trajectories and coordinated spatial remodeling. *Cell*, 187(8):2010–2028.e30, 2024.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9912–9924, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scGPT: Toward building a foundation model for single-cell multi-omics using generative AI. *Nature Methods*, 21(8):1470–1480, 2024.

Jérôme Demantké, Clément Mallet, Nicolas David, and Bruno Vallet. Dimensionality based scale selection in 3D LiDAR point clouds. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVIII-5/W12:97–102, 2011.

Nadav Dym and Steven J Gortler. Low-dimensional invariant embeddings for universal geometric learning. *Foundations of Computational Mathematics*, 25(2):375–415, 2025.

Yuhan Hao, Tim Stuart, Madeline H. Kowalski, Saket Choudhary, Paul Hoffman, Austin Hartman, Avi Srivastava, Gesmira Molla, Shaista Madad, Carlos Fernandez-Granda, and Rahul Satija. Dictionary learning for integrative, multimodal and scalable single-cell analysis. *Nature Biotechnology*, 42(2):293–304, 2024.

Shanshan He, Ruchir Bhatt, Carl Brown, Emily A. Brown, Derek L. Buhr, Kan Chantranuvatana, Patrick Danaher, Dwayne Dunaway, Ryan G. Garrison, Gary Geiss, Mark T. Gregory, Margaret L. Hoang, Rustem Khafizov, Emily E. Killingbeck, Dae Kim, Tae Kyung Kim, Youngmi Kim, Andrew Klock, Mithra Korukonda, Alecksandr Kutchma, Zachary R. Lewis, Yan Liang, Jeffrey S. Nelson, Giang T. Ong, Evan P. Perillo, Joseph C. Phan, Tien Phan-Everson, Erin Piazza, Tushar Rane, Zachary Reitz, Michael Rhodes, Alyssa Rosenbloom, David Ross, Hiromi Sato, Aster W. Wardhani, Corey A. Williams-Wietzikoski, Lidan Wu, and Joseph M. Beechem. High-plex imaging of RNA and proteins at subcellular resolution in fixed tissue by spatial molecular imaging. *Nature Biotechnology*, 40(12):1794–1806, 2022.

Arthur Imbert, Wei Ouyang, Adham Safieddine, Emeline Coleno, Christophe Zimmer, Edouard Bertrand, Thomas Walter, and Florian Mueller. FISH-quant v2: a scalable and modular analysis tool for smFISH image analysis. *bioRxiv*, 2021. doi: 10.1101/2021.07.20.453024.

Amanda Janesick, Robert Shelansky, Andrew D. Gottscho, Florian Wagner, Stephen R. Williams, Morgane Rouault, Ghezal Beliakoff, Carolyn A. Morrison, Michelli F. Oliveira, Jordan T. Sicherman, Andrew Kohlway, Jawad Abousoud, Tingsheng Yu Drennon, Seayar H. Mohabbat, and Sarah E. B. Taylor. High resolution mapping of the tumor microenvironment using integrated single-cell, spatial and in situ analysis. *Nature Communications*, 14:8353, 2023.

Yunyang Li, Lin Huang, Zhihao Ding, Xinran Wei, Chu Wang, Han Yang, Zun Wang, Chang Liu, Yu Shi, Peiran Jin, et al. E2former: An efficient and equivariant transformer with linear-scaling tensor products. *Advances in Neural Information Processing Systems*, 38:23593–23630, 2026.

Tony Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, 21(2):225–270, 1994.

Romain Lopez, Jeffrey Regier, Michael B. Cole, Michael I. Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15(12):1053–1058, 2018.

Qiaolin Lu, Jiayuan Ding, Lingxiao Li, and Yi Chang. Graph contrastive learning of subcellular-resolution spatial transcriptomics improves cell type annotation and reveals critical molecular pathways. *Briefings in Bioinformatics*, 26(1):bbaf020, 2025. doi: 10.1093/bib/bbaf020.

Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, 2012. doi: 10.1002/cpa.21413.

Andreas E Moor, Matan Golan, Efi E Massasa, Doron Lemze, Tomer Weizman, Rom Shenhav, Shaked Baydatch, Orel Mizrahi, Roni Winkler, Ofra Golani, et al. Global mrna polarization regulates translation efficiency in the intestinal epithelium. *Science*, 357(6357):1299–1303, 2017.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Spreading vectors for similarity search. In *International Conference on Learning Representations*, 2019.

Kristof T. Schütt, Oliver T. Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *PMLR*, pp. 9377–9388, 2021.

Vipul Singhal, Nigel Chou, Joseph Lee, Yifei Yue, Jinyue Liu, Wan Kee Chock, Li Lin, Yun-Ching Chang, Erica Mei Ling Teo, Jonathan Aow, Hwee Kuan Lee, Kok Hao Chen, and Shyam Prabhakar. BANKSY unifies cell typing and tissue domain segmentation for scalable spatial omics data analysis. *Nature Genetics*, 56(3):431–441, 2024.

Leon Strenger, Cledi Alicia Cerda-Jara, Nikos Karaiskos, and Nikolaus Rajewsky. pyTrance finds co-localizing RNAs in subcellular spatial transcriptomics data. *bioRxiv*, 2026. doi: 10.64898/2026.05.07.723470.

Alejandro Tejada-Lapuerta, Anna C Schaar, Robert Gutgesell, Giovanni Palla, Lennard Halle, Mariia Minaeva, Larsen Vornholz, Leander Dony, Francesca Drummer, Till Richter, et al. Nicheformer: a foundation model for single-cell and spatial omics. *Nature Methods*, 22(12):2525–2538, 2025.

Christina V. Theodoris, Ling Xiao, Anant Chopra, Mark D. Chaffin, Zeina R. Al Sayed, Matthew C. Hill, Helene Mantineo, Elizabeth M. Brydon, Zexian Zeng, X. Shirley Liu, and Patrick T. Ellinor. Transfer learning enables predictions in network biology. *Nature*, 618(7965):616–624, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pp. 5998–6008, 2017.

Soledad Villar, David W. Hogg, Kate Storey-Fisher, Weichi Yao, and Ben Blum-Smith. Scalars are universal: Equivariant machine learning, structured like classical physics. In *Advances in Neural Information Processing Systems*, volume 34, pp. 28848–28863, 2021.

Hongzhi Wen, Wenzhuo Tang, Xinnan Dai, Jiayuan Ding, Wei Jin, Yuying Xie, and Jiliang Tang. CellPLM: Pre-training of cell language model beyond single cells. In *International Conference on Learning Representations*, 2024.

Hermann Weyl. *The Classical Groups: Their Invariants and Representations*. Princeton University Press, 1946.

Michael J. Willatt, Félix Musil, and Michele Ceriotti. Feature optimization for atomistic machine learning yields a data-driven construction of the periodic table of the elements. *Physical Chemistry Chemical Physics*, 20(47):29661–29668, 2018. doi: 10.1039/C8CP05921G.

David Wouters, Jose Ignacio Alvira Larizgoitia, Nynke Tilkema, Paulien Van Minsel, Ceyhun Alar, Nick Seeuws, Ilias Theodoros Koulalis, Minkyoung Lee, Niels Vandermeulen, Srivathsan Adivarahan, Sandrine Da Cruz, Katy Vandereyken, Andreas Moor, Bernard Thienpont, Thierry Voet, and Alejandro Sifrim. SubCellSpace: Automated characterization of subcellular mRNA localization patterns in spatial transcriptomics. *bioRxiv*, 2026. doi: 10.64898/2026.04.28.720613.

Haixu Wu, Minghao Guo, Yuezhou Ma, Yuanxu Sun, Jianmin Wang, Wojciech Matusik, and Mingsheng Long. FlashBias: fast computation of attention with bias. *Advances in Neural Information Processing Systems*, 38:20729–20756, 2026.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 10524–10533, 2020.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. iBOT: Image BERT pre-training with online tokenizer. In *International Conference on Learning Representations*, 2022.

A Implementation of the measurement block

Computing the moments. Write $A_i[f] = \sum_j a_{ij}\, v_j\, f(z_j)$. The binomial expansion of Li et al. (2026, Thm. 3.2) gives the centered moments of Equation (3) from uncentered aggregates,

$$M_i^0 = A_i[1], \qquad D_i^1 = A_i[z] - z_i A_i[1],$$
$$D_i^2 = A_i[z^2] - 2z_i A_i[z] + z_i^2 A_i[1], \qquad N_i = A_i[|z|^2] - 2\,\mathrm{Re}\big(\bar{z}_i A_i[z]\big) + |z_i|^2 A_i[1],$$

so the four blocks $A_i[1]$, $A_i[z]$, $A_i[z^2]$ and $A_i[|z|^2]$ of Equation (9) serve every query. Being centered, the moments are translation invariant. No normalization, nonlinearity or residual writes into h^1 or h^2: a normalization would divide out the magnitudes d and κ are built from, and a nonlinearity would break these identities.

The counting measure divides by a constant. For the counting measure the proximity weights sum to one by construction, so $M_i^0 = Z_i = 1$ exactly and $d_i = N_i/\sigma^2$ divides by a constant rather than by an estimate. For $\mu^{\pm}$ the normalizer is itself estimated from whatever molecules fall inside the kernel, so the same readout is correspondingly noisier — which bounds where the in-layer dimension reading can be claimed at all.

Degenerate rows. A query excluded from the key set attends nothing, so its moments are identically zero and every readout is $0/\varepsilon = 0$ — the correct reading of "this token has no moment" rather than a division to guard against. The floor in Z_i exists for this case, and for it alone.

One denominator per measure. Both Jordan halves of a measure must share the total variation of *that* measure. A single half can be identically zero over a whole cell — $M_i^{0,+}$ vanishes wherever $b \le 0$ across the key set — so a half cannot normalize itself; and with several measures per scale, pairing adjacent channels rather than each measure with its own opposite half divides one measure by another's mass.

B Computing the invariant readout G

Coupling to the scalar stream. The per-measure invariants of Equation (4) and the readout G of each stream are pooled, normalized because they span orders of magnitude, and used to condition a FiLM residual together with x_i:

$$u_i = \mathrm{LN}\big[\,\text{readouts}_i,\ G(h_i^1),\ G(h_i^2)\,\big], \qquad g_i = \mathrm{SiLU}\big(W_p\,[\,u_i,\ x_i\,]\big), \tag{10}$$
$$h_i^0 \mathrel{+}= \gamma(g_i) \odot W_0\, x_i + \beta(g_i), \qquad W_0, \beta \text{ zero-initialized.} \tag{11}$$

Conditioning on x_i lets each token select which measures' geometry is relevant to it. The rest of this appendix describes G.

The entries of the Gram $hh^\top$, the pairwise inner products $\langle h_c, h_{c'}\rangle$, generate every $O(2)$-invariant polynomial of an equivariant stream — the first fundamental theorem for the orthogonal group (Weyl, 1946; Villar et al., 2021). Since only invariants may enter h^0, that Gram is the whole of what a stream can contribute there, and the named readouts of Equation (4) draw on a small part of it; G supplies the rest without choosing which contractions to take. Forming $hh^\top$ outright would cost $W(W+1)/2$ numbers per token and stream to carry the $2W-1$ degrees of freedom counted below, so it is never formed.

Let h be an equivariant type-k stream of Section 4.1, written under the identification of $\mathbb{R}^2$ with $\mathbb{C}$ of Section 3.1 as W real two-vectors, $h \in \mathbb{R}^{W\times 2}$. Take two *independent* equivariant projections $A, B \in \mathbb{R}^{m\times W}$, each acting on the channel index alone and identically on both spatial components, $(Ah)_a = \sum_c A_{ac} h_c \in \mathbb{R}^2$, and contract them elementwise:

$$G(h)_a \;=\; \big\langle (Ah)_a,\ (Bh)_a \big\rangle_{\mathbb{R}^2}, \qquad a = 1, \dots, m. \tag{12}$$

A contraction of this form is the scalar-update term of PaiNN (Schütt et al., 2021), which pairs it there with a learned gate.

Expanding the projections shows what this computes. Writing $A_{a\cdot}$ for the a-th row,

$$G(h)_a \;=\; \sum_{c,c'} A_{ac} B_{ac'} \,\langle h_c, h_{c'} \rangle \;=\; \left\langle\, A_{a\cdot}^{\top} B_{a\cdot},\; hh^{\top} \,\right\rangle_F, \tag{13}$$

so each output is a rank-one linear functional of the full Gram $hh^{\top}$ — a Gram functional with no Gram in it. The $W \times W$ matrix is never materialized, and m invariants cost m projected rows rather than the $W(W+1)/2$ entries of an explicit Gram.

Invariance. Under $h \mapsto hR$ with $R \in O(2)$ both projected two-vectors rotate together, so their inner product is unchanged. This holds for reflections as well as rotations: a real inner product of two co-rotating two-vectors is by construction a functional of $\mathrm{Re}(hh^{\top})$ and cannot produce the Im part, which is the component that flips sign under the mirror. The $O(2)$ commitment of Section 2 is therefore structural here rather than a property to be checked at each call site — the same distinction that excludes the circulation invariant from Equation (4).

Width. A type-k stream in two dimensions admits no linear invariant, and its Gram has rank at most two, so its orbit space has dimension $2W - 1$. The realized degrees of freedom are $\min(m,\, 2W-1)$, so $m = 2W - 1$ is the point at which the readout stops losing them. Saturating that count is not all further invariants could buy. Generic separation — telling apart almost every pair of configurations unrelated by a rotation or reflection — arrives one invariant later, at $2W$, and separation everywhere at about $4W - 1$ (Dym & Gortler, 2025; Villar et al., 2021). We set $m = 2W - 1$, on the view that functional independence is the operative requirement here and that injectivity on orbits does not repay twice the width. Keeping A and B untied matters: a single shared projection on both sides would force the functional's matrix in Equation (13) to be symmetric positive semi-definite.

Streams, and why they are read separately. The two streams are read separately and concatenated, $G(h_i) = [\, G(h_i^1);\, G(h_i^2)\,]$. No cross-stream term appears, because no quadratic form couples weight 1 to weight 2: such a term carries weight ± 1 and cannot be invariant. The coupling is irreducibly cubic, and it is α_i of Equation (4) that supplies it.

That α_i is needed at all is a counting matter. Read separately the streams span $2(2W-1) = 4W - 2$ dimensions, where the joint configuration has $4W - 1$: separate readouts are invariant under independent rotations of the two streams, while the configuration is defined only up to one common rotation. The shortfall is exactly one, and it is the orientation of the type-1 frame relative to the type-2 frame. Dropping α_i would leave every per-stream statistic intact and lose precisely that.

Rank, and the output projection. Taken bare each output is rank-one in the Gram, and the projection W_p of Equation (10) lifts that restriction: output j becomes $\langle \sum_a (W_p)_{ja} A_{a\cdot}^{\top} B_{a\cdot},\, hh^{\top} \rangle$, whose matrix is a sum of m rank-one terms and so has rank up to $\min(m, W)$ — full rank at $m = 2W - 1$. What remains is a dictionary restriction rather than a rank one — all outputs share a single m-dimensional subspace of the symmetric space — and an explicit dense Gram readout is the $m = W(W+1)/2$ limit of the same family.

Numerics. G is quadratic in feature magnitude, so its inputs are RMS-normalized beforehand; after normalization the mean per-channel squared norm is approximately one and the outputs are $O(1)$. This is separate from the normalization of the pool itself in Equation (10).

Relative to PaiNN. PaiNN applies Equation (12) to vector features and leaves the number of contractions a hyperparameter. What differs here is that m is derived rather than chosen, at the orbit dimension, bracketed above by the separating count of Dym & Gortler (2025). Both counts belong to the plane; in n dimensions the orbit dimension is $nW - n(n-1)/2$.

C Moment identities and readouts

This appendix derives the four readouts of Equation (4).

Local intrinsic dimension. Suppose μ is supported on an affine subspace of dimension m passing through p_i. Restricted to that subspace the kernel is an m-dimensional Gaussian of width σ, whose second moment is $\mathbb{E}\|\Delta p\|^2 = m\sigma^2$. Hence $N_i = m\sigma^2 Z_i$ and $d_i = m$ exactly, giving the three entries tabulated in Section 3.2: 0 for a point mass at the query, 1 for a line through it, 2 for a filled plane.

Anisotropy. D_i^2 is the traceless part of the second moment, so it vanishes on any neighborhood isotropic about the query and saturates on one confined to a line. On a line through p_i with direction $e^{i\phi}$, displacements are $\Delta z = te^{i\phi}$ with $\mathbb{E}[t^2] = \sigma^2$, so $D_i^2 = \sigma^2 e^{2i\phi} Z_i$ and $\kappa_i = 1$; on the filled plane the phase averages to zero and $\kappa_i = 0$.

Mean curvature. No pairing of types is required. For a measure concentrated near a curve, $\nabla\rho_\sigma = \rho_\sigma H/2 + O(\sigma^2)$, so the mean-curvature vector is available from the first moment and the mass alone — which is why $|H_i|$ reads D^1 rather than D^2.

Why the alignment invariant is cubic. In two dimensions there is *no* invariant pairing of a type-1 with a type-2 quantity: under $p \mapsto R_\theta p$ both $D^1 D^2$ and $\overline{D^1} D^2$ retain a phase, carrying $e^{3i\theta}$ and $e^{i\theta}$ respectively. Writing an inner product $\langle D^1, D^2 \rangle$ into an invariant pool would therefore break equivariance. The lowest-order invariant is cubic, $\mathrm{Re}\big(\overline{D^1}^{\,2} D^2\big)$, and α_i is its normalized form — equivalently $\cos(\arg D_i^2 - 2\arg D_i^1)$, the cosine of the angle between the tangent axis and twice the gradient direction. The normalization is not cosmetic: being cubic in the moments, the unnormalized quantity is the noisiest in the pool, scaling as the cube of the local mass.

D Objectives

All models reported here train under the same objective configuration, at both widths and on every dataset; the arms of Table 15 vary it as their rows state. The terms act at two token sites and read one of the three streams, as set out in Table 5.

Scalar stream. Two cross-entropy terms follow the usual teacher–student form, one on the CLS token and one on the masked molecule tokens (Caron et al., 2021; Zhou et al., 2022). Both score against a bank of prototypes; student logits are taken at temperature τ_s and the teacher's at τ_t, and the teacher's CLS assignments are normalized by Sinkhorn–Knopp over a queue of recent teacher features (Caron et al., 2020; Oquab et al., 2024). KOLEO spreads the CLS features and counters collapse (Sablayrolles et al., 2019). A negative-binomial head on the CLS token predicts the crop's molecule histogram, normalized over the gene panel, following Lopez et al. (2018). The molecule tokens carry no KOLEO term at any type.

Equivariant streams. h^1 and h^2 receive two families of term at both sites. The consistency term compares the student's moments to the teacher's by a cosine taken per channel,

$$\mathcal{L}_{\text{cons}} \;=\; \frac{1}{|C|} \sum_{c \in C} \left(1 - \frac{s_c \cdot t_c}{\|s_c\| \, \|t_c\|} \right), \tag{14}$$

where s_c and $t_c \in \mathbb{R}^2$ are the student and teacher values of channel c. This is the per-measure, per-order form of Section 4.2. Normalizing each channel before the cosine gives every channel equal weight; a single cosine over the concatenation would weight channels by squared norm, under which the coarsest scales dominate. C contains the learned channels only. The density weightings hold a fixed kernel smoother of position with no learned parameter, so student and teacher agree on them by construction. The two views entering Equation (14) are subsamples of one cell, which makes the term the thinning invariance of Section 2.

The invariant cross-entropy reads the learned Gram features of Equation (12). Each stream is normalized, passed through a readout of that form with its own projections, layer-normalized, and then through a prototype head of the same construction as the scalar stream's, against its own smaller bank; the teacher's assignments are Sinkhorn–Knopp normalized over a queue, as at the CLS token. The prototypes must be invariant and may not rotate with the data; otherwise heads would assign two views of one cell to different prototypes. The Gram supplies that invariance, and it bounds the term's

Table 5: The objective set by token site and stream. h^1 and h^2 receive identical treatment and share a row; their KOLEO terms are weighted separately but set to the same value. Equation (15) gives the weights and how the terms assemble.

Stream	CLS token	Molecule tokens
h^0	teacher–student cross-entropy; KOLEO; expression likelihood	teacher–student cross-entropy on masked tokens
h^1, h^2	consistency; invariant cross-entropy; KOLEO	consistency, at every layer; invariant cross-entropy

Table 6: Terms of Equation (15) removed by each row of Table 15. Four rows remove more than one term, because the coefficient reaching the named term also governs others.

Row of Table 15	Terms removed
no deep supervision	$\mathcal{L}^{\text{mol}}_{\text{deep}}$, the earlier layers' moments; the last layer's consistency remains
no expression head	$\mathcal{L}_{\text{expr}}$
no KOLEO	all three of $\mathcal{L}^{\text{CLS}}_{\text{ko}}$, on h^0, h^1 and h^2
no masking	$\mathcal{L}^{\text{mol}}_{\text{ce}}$, which is left no masked tokens to act on; the geometric terms select visible tokens and continue
no per-molecule geometry	$\mathcal{L}^{\text{mol}}_{\text{cons}}$, $\mathcal{L}^{\text{mol}}_{\text{deep}}$ and $\mathcal{L}^{\text{mol}}_{\text{inv}}$, every geometric term at the molecule site
no type-k consistency	$\mathcal{L}^{\text{CLS}}_{\text{cons}}$, $\mathcal{L}^{\text{mol}}_{\text{cons}}$ and, sharing their coefficient, $\mathcal{L}^{\text{mol}}_{\text{deep}}$
no type-k invariant CE	$\mathcal{L}^{\text{CLS}}_{\text{inv}}$ and $\mathcal{L}^{\text{mol}}_{\text{inv}}$

reach. A type-k feature splits into its orbit label, the $2W - 1$ invariants the Gram carries, and one remaining degree of freedom, the rotation of that pattern. This head reaches the label completely, and Equation (14) supervises the rotation. Since G depends on h^k through the Gram alone, and $\partial G/\partial h^k$ vanishes along the orbit tangent, the two gradients are orthogonal: neither term substitutes for the other, and neither works against it.

At the molecule site both families select the visible tokens.

Deep supervision. The molecule consistency term extends to the moments of every layer. A loss on layer L alone reaches that layer's measurement and no earlier one, since each earlier module map is reachable only through its own zero-initialized coupling. The extension shares the weight of the term it extends.

Weights. The total assembles as

$$\begin{aligned}\mathcal{L} \;&= \mathcal{L}^{\text{CLS}}_{\text{ce}} + \lambda_{\text{ko}}\mathcal{L}^{\text{CLS}}_{\text{ko}} + \lambda_{\text{cons}}\mathcal{L}^{\text{CLS}}_{\text{cons}} + \lambda_{\text{inv}}\mathcal{L}^{\text{CLS}}_{\text{inv}} + \lambda_{\text{expr}}\mathcal{L}_{\text{expr}} \\ &\quad + \lambda_{\text{mol}}\Big(\mathcal{L}^{\text{mol}}_{\text{ce}} + \mu_{\text{cons}}\big(\mathcal{L}^{\text{mol}}_{\text{cons}} + \mathcal{L}^{\text{mol}}_{\text{deep}}\big) + \mu_{\text{inv}}\mathcal{L}^{\text{mol}}_{\text{inv}}\Big).\end{aligned} \tag{15}$$

Every coefficient is 1 except the three KOLEO terms, which are 0.1. The molecule branch sits inside λ_{mol} while the CLS branch does not, so equal coefficients at the two sites do not correspond to equal contributions, and the two sites are not directly comparable in Table 15. A gene-prediction head, a per-cell expression head and a co-membership term are implemented but disabled throughout.

Ablation arms. Since λ_{mol} is an outer factor, not every term can be removed on its own: setting it to zero removes the whole molecule branch, and zeroing the inner coefficients leaves the masked cross-entropy in place. The rows of Table 15 are the removals a single setting reaches, and four of them remove more than one term. Table 6 gives the correspondence.

E Synthetic Data Generation

Our synthetic subcellular position generator behaves as a benchmark in which the true subcellular localization pattern of every transcript, gene and cell is known by construction.

What is inherited and what is synthesized. For every retained cell, the generator preserves the real section's gene panel, cell and nucleus boundary polygons, per-cell per-gene molecule counts, cell coordinates, and cell-type annotation exactly, and resynthesizes only the transcript positions. It additionally records synthetic ground-truth cell-subtype and per-molecule compartment labels. The one cell-level edit is a filter: cells whose annotation is a junk or unassigned class are dropped rather than generated as a spurious cell type, which removes 5.1% and 2.9% of the two sections used here.

Cell identity. Each cell carries two labels. Its *cell type* t is the inherited real annotation, visible from expression and morphology. Its *cell subtype* $s \in \{1, \dots, S\}$ is synthetic: a partition *within* each cell type whose members differ only in their localization mixtures $\theta_{t,s,\cdot}$. Cell subtype is assigned uniformly at random within each cell type, using no information about the cell's expression, shape or position, so $s \perp$ (expression, shape) by construction; it also carries no spatial structure across the section, so a cell's neighbors say nothing about it. Intracellular arrangement is thus the only channel carrying s, and recovering it is a clean probe of whether a method reads subcellular organization. We use $S = 3$ with uniform within-type assignment probabilities.

Relation to sim-fish. Seven map-based pattern semantics are adapted from FISH-quant v2 / sim-fish (Imbert et al., 2021); Table 7. We replace the remaining machinery: probability maps are rebuilt in two dimensions from real cell and nucleus polygons rather than from three-dimensional HeLa templates, coordinates are drawn with replacement plus sub-pixel jitter into continuous micrometers rather than without replacement onto integer pixels, and every molecule's pattern is an independent draw from a hierarchical population mixture over the whole seven-compartment simplex at the real per-cell per-gene counts, rather than one named pattern applied at a chosen strength and spot count.

Compartments. For each cell, its known transcript counts are assigned locations, or "compartments", within its nuclear and cell membrane polygons following $K = 7$ patterns: random, intranuclear, extranuclear, nuclear edge, perinuclear, cell edge, and pericellular. These compartments are represented in normalized probability maps over the cell's pixel grid, with each supported map summing to one over the pixels; Table 7 gives each map before that normalization, in the canonical index order the module heads below select from. A compartment with no support in a given cell — a thin band in a very small one, say — is instead left all-zero and places nothing. Both sections reported here empirically reproduce every input molecule count and also have complete nucleus coverage. A gene's localization is a mixture over these compartments, an element of the simplex Δ^{K-1} of probability vectors over the K compartments.

The per-molecule draw. Each of the $n_{c,g}$ real molecules of gene g in a cell of type t and subtype s draws a compartment $k \sim \text{Categorical}(\theta_{t,s,g,\cdot})$ and then a position from $\text{map}_k(\cdot \mid c)$. The vector $\theta_{t,s,g} \in \Delta^{K-1}$ is therefore the gene's localization proportion vector, with the diffuse case the random component of the same simplex. All cells of a (t, s) cell-subtype set share one θ, so realized per-cell proportions differ from it only by multinomial noise at the cell's real depth. Because a gene loads $L \geq 2$ modules (below), $\theta_{t,s,g}$ is a mixture of near-vertex module programs rather than near-vertex itself: at the settings used here its largest component has median 0.71 and 10th percentile 0.47, and 30% of (t, s, g) rows carry a second component above 0.25.

Hierarchical, low-rank localization proportions. θ is drawn top-down over three nested levels and low-rank over genes. The low rank is what makes a cell subtype's deviation coherent across a set of genes: with per-gene independent deviations ~95% of the deviation cancels under gene marginalization, capping any readout that pools genes. The generative units are $M \ll G$ modules, each a localization program φ_m over compartments together with the genes loading it; what differs between cell-subtype sets is module activity, not per-gene mixtures,

$$\varphi_m \in \Delta^{K-1}, \quad \Lambda_{g,m} \geq 0, \quad \pi_0 \sim \text{Dir}(\kappa_0 \mathbf{1}_M), \quad \pi_t \sim \text{Dir}(\kappa_1 \pi_0), \quad \pi_{t,s} \sim \text{Dir}(\kappa_2 \pi_t), \tag{16}$$

Table 7: The seven localization compartments, as unnormalized maps on the cell's own pixel grid; each supported map is divided by its sum over those pixels before use. Rows are given in the canonical compartment index order $k = 0, \ldots, K-1$, which is fixed by the implementation and is what the module heads m mod K select. The cell and nucleus pixel sets are the real polygons rasterized at 0.2125 µm/px (the Xenium pixel), the nucleus intersected with the cell; $\mathbb{1}[\cdot]$ is the indicator of a pixel set; d_{cell} is the Euclidean distance transform of the cell mask, measured inward from the membrane and zero outside the cell, and d_{nuc} that of the nucleus complement, measured outward from the nuclear envelope and zero inside the nucleus; and $b = 2$ px $= 0.425$ µm is the band width. Every map is built from the cell's real polygons rather than from a template: the two band compartments select a boundary — the strict $0 < d_{\text{nuc}}$ excluding the nucleus interior, where d_{nuc} vanishes — and the two quadratic ones ramp across the cytoplasm, `perinuclear` growing with distance from the membrane and `pericellular` with distance from the nucleus.

k	compartment	unnormalized map
0	`random`	$\mathbb{1}[\text{cell}]$
1	`intranuclear`	$\mathbb{1}[\text{nucleus}]$
2	`extranuclear`	$\mathbb{1}[\text{cell} \wedge \neg\,\text{nucleus}]$
3	`nuclear_edge`	$\mathbb{1}[0 < d_{\text{nuc}} \le b] \cdot \mathbb{1}[\text{cell}]$
4	`perinuclear`	$d_{\text{cell}}^2 \cdot \mathbb{1}[\text{cell} \wedge \neg\,\text{nucleus}]$
5	`cell_edge`	$\mathbb{1}[d_{\text{cell}} \le b] \cdot \mathbb{1}[\text{cell} \wedge \neg\,\text{nucleus}]$
6	`pericellular`	$d_{\text{nuc}}^2 \cdot \mathbb{1}[\text{cell} \wedge \neg\,\text{nucleus}]$

$$\beta_g \propto \sum_m \Lambda_{g,m}\pi_{0,m}\varphi_m, \qquad \theta_{t,g} \propto \sum_m \Lambda_{g,m}\pi_{t,m}\varphi_m, \qquad \theta_{t,s,g} \propto \sum_m \Lambda_{g,m}\pi_{t,s,m}\varphi_m, \tag{17}$$

each normalized over compartments. Here G is the number of genes in the panel, φ_m is module m's localization program over the K compartments, and $\Lambda_{g,m}$ is gene g's loading on that module — nonzero on exactly the L modules the gene loads, and summing to one over them. The activities $\pi \in \Delta^{M-1}$ say how strongly each module is active, one vector per level, each drawn about the level above it: π_0 over the population, π_t for cell type t, and $\pi_{t,s}$ for subtype s within it, so $\pi_{t,s,m}$ is the activity of module m in that cell-subtype set. Weighting the programs by activity and loading gives the localization proportions at the same three levels, β_g, $\theta_{t,g}$ and $\theta_{t,s,g}$: the last is what the per-molecule draw above uses, and the two coarser ones are the same construction at the population and cell-type levels, recorded alongside it as ground truth.

Modules and loadings. Programs φ and loadings Λ are fixed across the section; only the activities vary by level, so one Dirichlet draw moves every gene in a module together. Module m's dominant compartment is the one at index m mod K of Table 7, which guarantees that every compartment heads a module once $M \ge K$ (assigning heads at random leaves one compartment unheaded with probability 0.98 at $M = 8$ and 0.50 at $M = 16$). That head receives mass 0.85 on top of its share of a $\text{Dir}(0.35\,\mathbf{1}_K)$ remainder; every component is then floored at 0.01 and the profile renormalized, which leaves the realized head mass at 0.83–0.89. Each gene loads $L \sim \text{Uniform}\{2, 3\}$ distinct modules with $\text{Dir}(\lambda\mathbf{1}_L)$ weights.

Constraints. Two constraints are enforced. First, $L \ge 2$: at $L = 1$ the activity cancels in the normalization and $\theta_{t,s,g} = \theta_{t,g} = \varphi_m$ leaves the gene signal-free. Second, Λ is a function of gene index alone, never of expression, so the localization modules are not deliberately aligned with expression abundance or marker-gene structure. Independence of subcell type and counts instead follows from random subtype assignment and the unchanged count vectors.

Concentrations, floors and seeding. κ_1 and κ_2 are total concentrations, so per component they are $\approx \kappa/M$; κ_0 and λ are per-component instead. Drawn Dirichlet weights are floored at 10^{-4}, since Gamma underflow can otherwise zero a gene's entire contraction. The draw uses a seed stream independent of the cell count, so one θ is shared across sections of different size and calibration on a subsample is exact.

Parameter settings. Table 8 lists the parameters used. M is the dominant coherence knob and is set per gene panel: fewer modules make more genes move together.

Table 8: Generation parameters. The weight floor is a numerical safeguard against underflow; the profile floor is a support constraint that keeps every compartment accessible. Both are reported for reproducibility.

Parameter	Value	Role
Structure		
M (modules)	8	Number of shared localization programs
L (loads per gene)	2–3	Modules per gene; $L \geq 2$ enforced
S (cell subtypes)	3	Synthetic partition within each cell type
K (compartments)	7	Table 7
Genes per module	97.8	Expected from G, L, M; realized 99.5
Concentrations		
κ_1	64.0	Cell-type activity about the population; total
κ_2	6.0	Subtype about its cell type; total
κ_0	10.0	Population activity; per component
λ (loadings)	1.0	A gene's weights over its modules; per component
Module profiles		
Dominant mass	0.85	Mass added to a module's dominant compartment
Tail concentration	0.35	Spread of the remaining mass
Numerical and spatial		
Profile floor	10^{-2}	Keeps every compartment positive
Weight floor	10^{-4}	Guards Gamma underflow in activity and loading draws
Rasterization	0.2125 µm/px	Xenium pixel pitch
Random seed	7	Shared by both sections

F Datasets

Table 9 gives the three datasets and Table 10 the split sizes.

Colon is the healthy arm of a DSS-induced colitis time course, sampled at days 0, 3, 9, 21 and 35 (Cadinu et al., 2024). We use the ten day-0 sections alone, so every Colon figure reported here and in Table 1 describes that subset rather than the full course. Its panel is 990-plex as assayed; removing the 47 `Blank-` control probes leaves the 943 of Table 9, and the minimum-cell gene filter drops none of them. Lung and DCIS carry no control probes, so their panels pass through at 960 and 313. Molm and every baseline that fits a basis therefore see the same gene set on each dataset.

Lung is the public CosMx reference dataset for non-small-cell lung carcinoma (Bruker Spatial Biology, 2022; He et al., 2022), eight sections from five donors. We use two: `Lung5_Rep2` to train and `Lung5_Rep3` to test. DCIS is two adjacent sections of one breast block (Janesick et al., 2023), `Rep1` to train and `Rep2` to test.

No split separates donors. The two Lung sections are serial sections of one donor, the two DCIS sections are serial sections of one block, and the Colon split divides slices of the same three mice. Each pair therefore shares tissue architecture, and these splits measure generalization across sections rather than across individuals.

Table 9: The three datasets. Panel is the number of genes assayed. For Colon the labels are hierarchical Leiden clusters at three resolutions; Table 1 reports Tier 1.

Dataset	Species	Tissue	Platform	Panel	Labels
Colon	Mouse	Colon	MERFISH	943	Tier 1: 8, Tier 2: 66, Tier 3: 72
Lung	Human	Lung, NSCLC	CosMx	960	22 cell types
DCIS	Human	Breast cancer	Xenium	313	19 cell types

Table 10: Split sizes. Each dataset is split by section. Filtered counts are after the two quality filters described in Section 5.

	Sections		Cells		Cells, filtered	
Dataset	Train	Test	Train	Test	Train	Test
Colon	6	4	149 861	89 589	103 193	60 102
Lung	1	1	106 660	100 264	95 892	90 235
DCIS	1	1	167 780	118 752	151 909	111 122

G Baselines

We run every baseline ourselves from a shared codebase, and probe every embedding by the protocol below. We remove control probes first. Their identifiers are platform-specific: genes prefixed `Blank-` in Colon, of which there are 47, and transcripts flagged `is_gene=False` in DCIS, which covers deprecated codewords and all negative-control probe types. Lung needs no such rule: its panel carries no control probes under any of the usual names. Colon is supplied log-normalized, to 10 000 counts followed by log1p. Neither step reaches the foundation models: their gene vocabularies are fixed and contain no control probes, and they normalize internally.

Every method that fits a basis fits it on training cells alone. Gene filtering, variable-gene selection, scaling statistics and components are all estimated on the training split, and every cell is then projected into that basis, so no test information enters any fit. Methods are deterministic and run once unless noted.

Several baselines expose a range of embedding widths. Table 1 uses 50 dimensions throughout, in each method's default variant. That choice was fixed before the results were seen, rather than taking each method's best width, because selecting on the reported metric would lift every baseline and leave ours untouched.

Seurat (Hao et al., 2024). Log-normalization, 2000 variable features, per-gene scaling, and PCA by truncated SVD (irlba, seed 42), at 10 to 50 components. Test cells are projected through the training variable-gene list, the per-gene means and standard deviations, and the loadings.

scVI (Lopez et al., 2018). Raw integer counts, a negative-binomial likelihood, 10 to 50 latent dimensions, trained for up to 400 epochs with early stopping on a validation fraction drawn from the training cells. Test cells are encoded by the fitted model. Colon uses a sample-level batch covariate. Training is stochastic, so we run three replicates per latent size and report their spread. The first replicate ran without a fixed seed, so it is reproducible only to the extent that the others bound it.

BANKSY (Singhal et al., 2024). Each profile is augmented with its spatial neighborhood ($\lambda = 0.2$, cell-type mode, 15 spatial neighbors, azimuthal Gabor filter term) and reduced by PCA to 10 to 50 components, with the PCA fitted on training cells. Neighborhoods are confined to spatial partitions, one tissue section for Colon and DCIS and section $\times$ field of view for Lung, whose FOV identifiers repeat across sections. We enforce this by offsetting partition coordinates, so no cross-partition pair can be neighbors. The augmentation itself is computed over all cells, since spatial neighbors are physical and cannot be restricted to a split.

scGPT (Cui et al., 2024). Embeddings are extracted zero-shot from the whole-human pretrained checkpoint. Gene symbols are matched against the model vocabulary, with mouse genes uppercased before lookup. The output is 512-dimensional.

Geneformer (Theodoris et al., 2023). Raw integer counts are tokenized with the Geneformer-V2-104M tokenizer, model input size 4096, gene symbols mapped to Ensembl IDs via `mygene`. Embeddings are extracted zero-shot from the pretrained 104M-parameter model. The output is 768-dimensional.

CellPLM (Wen et al., 2024). Embeddings are extracted zero-shot from the `20231027_85M` checkpoint via `CellEmbeddingPipeline`. The output is 512-dimensional.

Nicheformer (Tejada-Lapuerta et al., 2025). Gene symbols are mapped to Ensembl IDs. The tokenizer normalizes counts to 10 000 and applies log1p internally, then selects the top 1497 expressed genes per cell. Embeddings are extracted zero-shot from the pretrained `theislab/Nicheformer` model. The output is 512-dimensional.

FOCUS (Lu et al., 2025). FOCUS represents each cell as a transcript-level gene neighborhood network, connecting transcripts by spatial proximity within the cell. We build transcript graphs from the transcript coordinates with a 2.0 µm radius threshold. On Colon every transcript is assigned to the cytoplasm, because the platform gives no nuclear-overlap annotation; on DCIS the nuclear and cytoplasmic labels come from the platform's `overlaps_nucleus` flag. The semi-supervised embeddings come from three independently trained FOCUS3 ResGCN checkpoints using tier-specific cell-type labels. The output is 128-dimensional. We also ran a contrastive-only variant with the classification loss weight set to $\alpha = 0$, and do not report it: that configuration is ours rather than the authors', and on Lung two of its three runs show metrics declining with epoch, so it is not stably trained.

pyTrance (Strenger et al., 2026). pyTrance is a single-layer graph network over transcripts, built to find co-localizing transcript pairs. We take the per-transcript representations and pool them to the cell by summation, giving a 128-dimensional embedding trained for 50 epochs. It appears in Table 2 and in no other table.

H Evaluation protocol

Every embedding in this paper is scored the same way, ours and the baselines alike.

Probe protocol. A linear probe reads features that are already present, so anything it recovers was linearly available in the embedding. We freeze every embedding and apply the same probes throughout, at scikit-learn 1.8 defaults. Nothing below is tuned, by us or per method.

Each cell's vector is scaled to unit L_2 norm. Features are otherwise left as they are: no centering, no per-feature scaling, no dimensionality reduction. Scaling by cell discards the embedding's overall magnitude, which for some methods carries information. For the h^1 and h^2 keys the Gram is built first and then normalized the same way.

We probe cell type with logistic regression (L2 penalty, $C = 1$, lbfgs, `class_weight="balanced"`, `max_iter=1000`) and report balanced accuracy, the macro-average recall over classes. We probe morphology with ridge regression at $\alpha = 1$ with a fitted intercept, targets in their original units and winsorized at the 1st and 99th percentiles, and report R^2. We fit on the training section and score on the held-out one, following the split of Table 10.

Holding α fixed has one consequence worth stating. Every feature vector has unit norm, but their widths differ: 50 for scVI, 512 for h^0, 8385 for the h^2 Gram. A common α therefore penalizes a wide representation more, because each of its coordinates is smaller. That works against our equivariant streams, which are the widest embeddings in the comparison.

Probe inputs. Tables 1 and 11 to 13 each report three readouts of one model. We probe the scalar stream h^0 directly, at width 512. A probe on h^1 or h^2 would read a quantity that rotates with the frame, so we probe those two through the raw Gram of the stream. Write $H \in \mathbb{R}^{W\times 2}$ for a stream in the real basis, with $W = 1 + 2C = 129$ at $C = 64$ measures. The probe input is the upper triangle of $HH^\top$, of width $W(W+1)/2 = 8385$. Per cell that matrix has rank at most two, since H has two columns, so the features are highly dependent.

These features are the plain $O(2)$-invariant quadratic form of the stream, and carry no learned parameters. The implicit Gram readout of Appendix B is a different object: two learned projections contracted elementwise, sized to the degrees of freedom, and feeding the pool of Equation (10). Using the plain form here leaves nothing trained between the stream and the probe.

I Reporting conventions

Every table in this paper follows one convention, stated here rather than in each caption.

Uncertainties. A parenthetical digit is the standard error over training seeds, on the trailing digits shown: 0.780(3) means 0.780 ± 0.003. It is always seed variability of the printed number, never spread across the features averaged into it. Where a table's seed coverage is uneven it carries an n column giving the count; where every row has the same count, the caption states it. A replicated entry printed without a parenthetical has a standard error below the displayed precision, not a deterministic one. Single-seed entries carry no uncertainty, and we claim nothing from them.

Marks. Bold marks every entry a column does not separate from its best, at best$-v \le \sqrt{se_{\text{best}}^2 + se_v^2}$, so a column can carry several bold entries or, where the best is isolated, one. An em dash marks a configuration that was not run.

J Additional results

Cell type at every width and label depth. Three things in Table 11 bear on how Table 1 should be read. The scalar stream leads all five columns even when every baseline is given its best width, so the comparison does not depend on the 50-dimension rule; the only entry not separated from it is FOCUS, which trains on the labels. Every method falls sharply as the Colon label set refines, and the ordering is largely preserved. And our lead over the best baseline widens with the label set, from 0.005 at Tier 1 to 0.049 at Tier 2.

Morphology per descriptor. Table 1 collapses each morphology category to a mean over its descriptors. The two tables below show the components. Bolding and variant selection follow Table 1, but the parenthetical is not the same quantity: there it is the seed standard error of a category mean, here of a single descriptor, and a mean is steadier than its parts, so these are generally larger.

Which layer's measures to read. Each layer's 64 measures were scored on their own, everything else held fixed (Table 14). The rise from L0 to L4 and the fall from L4 to L7 are both monotone, and the peak sits inside the network rather than at either end.

Eight layers were tested and the best was picked on the split that scores it, so an isolated maximum would deserve suspicion. The shape of the curve rules that out. L3 and L5 sit within 0.003 of L4 while L6 and L7 fall away by 0.015 and 0.020, and the profile is monotone in both directions with no exceptions. A selection artifact would be a lone spike; this is a plateau, corroborated on both sides. The three plateau layers are not separable from one another: L3 against L4 is −0.003 with a 95% interval of [−0.012, +0.005], and the other two pairings are the same. L6, the layer read before this sweep, sits 0.015 below.

Width does not substitute for choosing the layer. One layer's 64 measures beat every wider set, including all eight concatenated: 0.440 at 64 dimensions, against 0.424 for L3–L5 at 192, 0.411 for L2–L7 at 384 and 0.405 for all eight at 512.

Objective ablations. Two removals separate from the reference in Table 15. Masking is much the larger: without it cell subtype falls 0.148 and shape 0.222, and it is the only removal that costs every column. The type-k invariant cross-entropy costs 0.030 on cell subtype, about three standard errors. No other single removal separates from the reference on that column, and cell type spans only 0.016 across all eight rows.

Three smaller results are worth recording without being claimed. Dropping KOLEO raises size to 0.913(2) and lowers shape to 0.322(8), against the reference's 0.904(3) and 0.357(14); dropping the per-molecule geometry term lowers size to 0.891(4). Both sit near two and a half standard errors. And removing the expression head, the only wholly non-geometric term besides masking, changes nothing measurable in any column. These are leave-one-out removals at one configuration, so each bounds what a term contributes alone and not what the objective needs. A term whose signal is carried elsewhere looks free when it is the only one removed.

Table 11: Cell-type linear probing, full results. Balanced accuracy from a logistic probe. Table 1 shows one embedding width and Colon's Tier 1 alone; this table adds the width sweep and the two finer Colon tiers (8 classes at Tier 1, 66 at Tier 2, 72 at Tier 3). d is the embedding width, and for the equivariant streams the shape of the stream: 129 channels by the two-dimensional $O(2)$ irrep. The probe reads the stream's invariant Gram, which is wider (Appendix H). Rows are grouped as in Table 1, by what each method uses. scVI, FOCUS and Molm are averaged over three training seeds and carry a standard error; the rest are single runs. *FOCUS is semi-supervised and trains on ground-truth cell-type labels; it is excluded from bolding. Conventions as in Appendix I, with the bolding applied across every width rather than within one.

				Colon		
Method	d	Lung	DCIS	Tier 1	Tier 2	Tier 3
Counts only						
Seurat (PCA)	10	0.686	0.745	0.824	0.444	0.425
	20	0.747	0.795	0.939	0.576	0.553
	30	**0.761**	0.799	0.932	0.629	0.604
	40	0.749	0.800	0.934	0.657	0.640
	50	0.750	0.800	0.935	0.665	0.660
scVI	10	0.714(2)	0.812(1)	0.833(7)	0.438(6)	0.436(4)
	20	0.747(2)	0.820	0.853(4)	0.434(8)	0.431(8)
	30	0.752(3)	0.822	0.861(6)	0.446(4)	0.440(3)
	40	0.753(1)	0.823	0.871(4)	0.457(2)	0.451(3)
	50	0.753(1)	0.823(1)	0.879(4)	0.473(3)	0.469(3)
Counts with tissue context						
BANKSY	10	0.619	0.723	0.854	0.424	0.395
	20	0.698	0.791	0.942	0.590	0.570
	30	0.712	0.799	0.933	0.639	0.615
	40	0.713	0.799	0.936	0.656	0.645
	50	0.723	0.800	0.949	0.663	0.662
Foundation models, zero-shot						
CellPLM	512	0.617	0.790	0.870	0.478	0.480
scGPT	512	0.638	0.810	0.882	0.513	0.516
Geneformer	768	0.652	0.813	0.902	0.525	0.541
Nicheformer	512	0.643	0.812	0.894	0.557	0.546
Subcellular positions, trained on cell-type labels						
FOCUS*	128	0.732(4)	0.837(1)	0.934(1)	0.669(10)	0.660(11)
Subcellular positions, self-supervised (ours)						
Molm h^0	512	**0.758(3)**	**0.840(1)**	**0.955(1)**	**0.708(6)**	**0.709(6)**
Molm h^1	129×2	0.669(1)	0.800(1)	0.833(8)	0.544(5)	0.518(7)
Molm h^2	129×2	0.699(2)	0.808(1)	0.912(8)	0.615(9)	0.593(11)

Table 12: Regression of cell size descriptors, ridge R^2. Columns are the area and the perimeter of the convex hull of the transcript cloud, on each dataset. Table 1 averages the two into its size block.

	Lung		DCIS		Colon	
Method	Hull	Perim.	Hull	Perim.	Hull	Perim.
Counts only						
Seurat (PCA)	0.202	0.195	0.573	0.640	0.451	0.438
scVI	0.148(4)	0.159(4)	0.395(7)	0.432(8)	0.444(6)	0.456(6)
Counts with tissue context						
BANKSY	0.414	0.415	0.629	0.693	0.470	0.461
Foundation models, zero-shot						
CellPLM	0.294	0.305	0.425	0.511	0.338	0.344
scGPT	0.270	0.255	0.574	0.640	0.621	0.632
Geneformer	0.345	0.318	0.646	0.708	0.747	0.754
Nicheformer	0.319	0.303	0.616	0.675	0.637	0.649
Subcellular positions, trained on cell-type labels						
FOCUS	0.641(22)	0.637(18)	0.706(13)	0.753(9)	0.810(5)	0.813(5)
Subcellular positions, self-supervised (ours)						
Molm h^0	0.890(3)	**0.919(2)**	**0.887**	**0.942(1)**	**0.949(1)**	**0.956(1)**
Molm h^1	0.882	0.895(1)	0.820(6)	0.866(5)	0.897(3)	0.890(2)
Molm h^2	**0.898(2)**	0.913(3)	0.845(5)	0.882(5)	0.923(3)	0.920(2)

Table 13: Regression of cell shape descriptors, ridge R^2. Columns are circularity and eccentricity, on each dataset. Table 1 averages the two into its shape block.

	Lung		DCIS		Colon	
Method	Circ.	Ecc.	Circ.	Ecc.	Circ.	Ecc.
Counts only						
Seurat (PCA)	0.022	0.041	0.058	0.018	0.094	0.059
scVI	0.010	0.039	0.025(2)	0.014	0.150	0.096(1)
Counts with tissue context						
BANKSY	0.011	0.036	0.075	0.022	0.101	0.062
Foundation models, zero-shot						
CellPLM	0.022	0.034	0.038	0.016	0.071	0.043
scGPT	0.027	0.043	0.062	0.023	0.196	0.123
Geneformer	0.028	0.043	0.075	0.027	0.215	0.134
Nicheformer	0.028	0.043	0.069	0.024	0.204	0.128
Subcellular positions, trained on cell-type labels						
FOCUS	0.046(1)	0.049(1)	0.091(2)	0.031	0.204(3)	0.129(2)
Subcellular positions, self-supervised (ours)						
Molm h^0	0.645(5)	0.736(6)	0.540(3)	0.501(13)	**0.659(4)**	0.726(4)
Molm h^1	0.611(10)	0.744(2)	0.440(9)	0.433(6)	0.509(9)	0.545(7)
Molm h^2	**0.723(5)**	**0.915(3)**	**0.580(13)**	**0.655(8)**	0.632(6)	**0.831(2)**

Table 14: Pattern recovery from each layer's measures, on synthetic DCIS, three seeds. Each row scores that layer's 64 measures alone, pooled to the cell; h^0 is the pooled scalar stream, given for reference. Δ is against L4, the layer Table 2 reports. **These are own-set figures**, on the 167 genes this comparison admits, and are not comparable to Table 2, which scores the 126 genes every method admits and where L4 reads 0.448.

Layer	d	Bal. acc.	95% CI	Δ vs L4
h^0	512	0.396	[0.385, 0.408]	−0.044
L0	64	0.304	[0.296, 0.311]	−0.137
L1	64	0.392	[0.383, 0.401]	−0.048
L2	64	0.420	[0.410, 0.429]	−0.020
L3	64	0.437	[0.428, 0.446]	−0.003
L4	64	**0.440**	[0.431, 0.450]	–
L5	64	0.439	[0.430, 0.448]	−0.001
L6	64	0.425	[0.416, 0.435]	−0.015
L7	64	0.420	[0.411, 0.430]	−0.020

Table 15: Leave-one-out objective ablations on synthetic DCIS. Every row trains at $d = 128$, four layers and a single fixed scale of 2.5 µm. The first row keeps the whole objective and is the reference; each other row removes from it the terms Table 6 lists. That reference is the same run as the 2.5 µm row of Table 4, which shares this configuration and varies only the scale. Cell type and cell subtype are balanced accuracy from a logistic probe; size and shape are ridge R^2. Every row is three seeds. Conventions as in Appendix I.

Ablation	Cell type	Cell subtype	Size	Shape
all loss terms	0.864	0.683(9)	0.904(10)	0.403(18)
no deep supervision	0.864	0.687(7)	0.916(3)	0.406(20)
no expression head	0.863(2)	0.686(11)	0.917(1)	0.412(6)
no KOLEO	0.860(1)	0.672(7)	0.914(7)	0.368(16)
no masking	0.851(1)	0.535(1)	0.852(2)	0.181(11)
no per-molecule geometry	0.867(1)	0.690(7)	0.894(4)	0.396(4)
no type-k consistency	0.865(1)	0.692(3)	0.917(1)	0.410(14)
no type-k invariant CE	0.863(1)	0.653(6)	0.885(2)	0.409(25)