Augmented Equivariant Mesh Networks for Anatomical Mesh Segmentation

Daniel Saragih [1]

Abstract

Anatomical mesh segmentation requires models that operate directly on irregular surface geometry while remaining robust to arbitrary patient pose and mesh resolution variation. Existing task-specific mesh and point-cloud methods are not equivariant, and can degrade sharply under test-time perturbation, for example dropping by 25–26 IoU points on intraoral scan segmentation at 40° tilt. We present EAMS, an **E**quivariant **A**natomical **M**esh **S**egmentor built on Equivariant Mesh Neural Networks (EMNN), and evaluate it across four clinically distinct tasks spanning edge-, vertex-, and face-level supervision. We combine intrinsic mesh descriptors with anatomy-aware priors, including PCA-derived frames for dental arches and liver surfaces, and augment message passing with soft regional aggregators and virtual nodes to capture multi-scale geometric context. Across intracranial aneurysm and intraoral segmentation, EAMS variants are competitive with specialized baselines on unperturbed inputs while remaining stable under geometric perturbations, and on liver surfaces they expose a favorable trade-off between canonical-pose accuracy and rotation robustness. These results position equivariant mesh models as a useful substrate for assistive scientific workflows on geometric biomedical data, where reproducible behavior under arbitrary orientation matters as much as raw accuracy.

1. Introduction

Scientific AI systems are increasingly expected to operate on structured geometric data rather than only on images or tabular measurements. Anatomical surface meshes are one such regime: they are the native output of many clinical reconstruction pipelines, and any assistive workflow built on top of them must remain reliable under arbitrary pose, variable tessellation, and limited annotation budgets.

[1]Department of Pathology and Molecular Medicine, Queen's University, Kingston, ON, Canada. Correspondence to: Daniel Saragih <danielg.saragih@gmail.com>.

Accepted to the AI for Science workshop (ICML 2026).

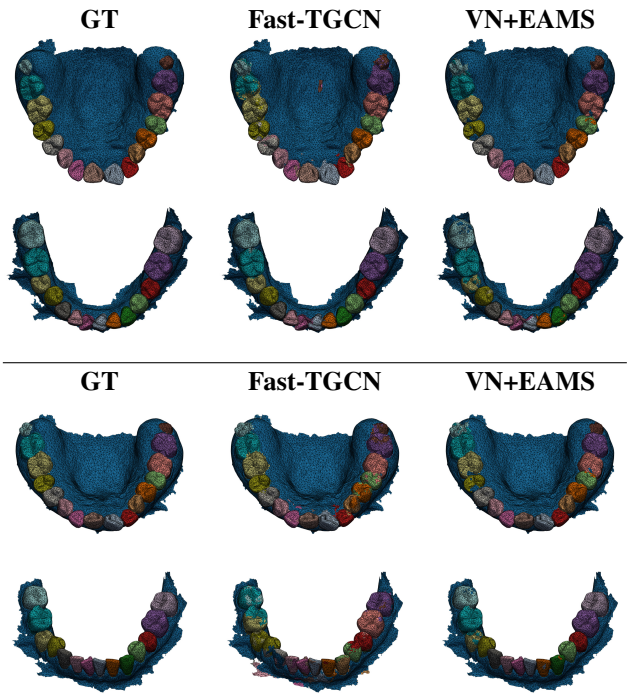


Figure 1. Qualitative tooth-segmentation comparisons. The top half shows the canonical orientation and the bottom half shows a 40° z-axis rotation. Fast-TGCN is competitive at baseline but degrades near posterior teeth and gingival boundaries under rotation, whereas VN+EAMS remains stable across both settings.

Anatomical mesh segmentation increasingly relies on 3D surface meshes rather than volumetric grids (Martin et al., 2015; Koo et al., 2017; Yang et al., 2020): meshes are compact, topology-aware representations and the native output of many clinical reconstruction pipelines. Tasks such as liver-surface labelling, intraoral tooth segmentation, and intracranial aneurysm delineation require methods that can operate directly on triangle-mesh geometry with varying resolution, noise, and pose.

Challenges. Effective mesh segmentation must satisfy two competing demands. First, predictions should be invariant to rigid-body transformations: a model that relabels the same anatomy differently depending on patient orientation provides no reliable clinical signal. Second, mesh resolution in medical datasets varies dramatically (e.g. 3,000–20,000 edges for liver surfaces), requiring feature representations

that are robust to sampling density while still capturing local geometry. Existing point-cloud methods (Qi et al., 2017; Wang et al., 2019; Wu et al., 2019; 2024) are natively resolution-agnostic but do not exploit the topological structure of meshes; current segmentation methods that operate on meshes (Yang et al., 2020; Ben-Hamadou et al., 2023; Li et al., 2024; Xi et al., 2025; Zhang et al., 2025) are not invariant and thus vulnerable to spurious correlations with patient pose.

This work. We address these challenges with EAMS, an anatomical mesh segmentor built on Equivariant Mesh Neural Networks (EMNN) (Trang et al., 2024), and evaluate it across four clinically distinct tasks spanning edge-, vertex-, and face-level supervision. We claim that not only does equivariance improve robustness, but that geometric priors and lightweight multi-scale context can make better mesh-based scientific inference when data arrive in arbitrary orientations.

Our contributions are:

- **Robust multi-task anatomical mesh benchmark.** We evaluate equivariant mesh segmentation on liver surfaces, intraoral scans, and intracranial aneurysms, covering edge-, vertex-, and face-level tasks. This unifies disparate datasets and segmentation paradigms under a lightweight ($<$ 2M parameters) framework that is decisively stronger than prior task-specific baselines in high-perturbation regimes while remaining competitive on unperturbed inputs.
- **Anatomy-aware featurisation.** We propose and evaluate anatomy-aware featurization, including PCA-derived anatomical frames whose canonical axes transform equivariantly with the mesh under SE(3) motions. These features, combined with efficient shape descriptors, consistently improve performance across all tasks.
- **Augmented message-passing.** We propose and evaluate two ways to extend EMNN's receptive field: soft regional aggregators, which impose a structured regional bottleneck through soft clustering, and virtual nodes, which add a lightweight global memory bank for flexible long-range exchange. These modules provide a simple form of multi-scale geometric modelling and, with regularization losses, consistently improve performance across all tasks.

2. Background and Related Work

Triangle meshes A triangle mesh is a tuple $\mathcal{M} = (\mathcal{V}, \mathcal{E}, \mathcal{F})$, where $\mathcal{V} = \{1, \ldots, n\}$ indexes the vertices, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the undirected edge set, and $\mathcal{F} \subseteq \mathcal{V}^3$ is the set of triangular faces, each an ordered triplet of vertices. Each vertex $i \in \mathcal{V}$ carries a coordinate $x_i \in \mathbb{R}^3$. A valid triangle mesh is a manifold: every edge belongs to at most two faces, no two distinct faces intersect except along a shared edge or vertex, and the faces around every vertex form a topological disk. We write $N(i) = \{j : (i,j) \in \mathcal{E}\}$ for the one-ring neighbourhood of vertex i and $\tau(i) = \{(j,k) : (i,j,k) \in \mathcal{F}\}$ for its adjacent faces.

Geometric features Every face $(i,j,k) \in \mathcal{F}$ induces a normal vector $n_{ijk} = (x_j - x_i) \times (x_k - x_i) \in \mathbb{R}^3$, whose magnitude equals twice the face area: $a_{ijk} = \|n_{ijk}\|/2$. Vertex-level normals can be obtained by aggregating over adjacent faces; an area-weighted average gives $n_i = \sum_{(j,k)\in\tau(i)} a_{ijk} n_{ijk} \,/\, \| \sum_{(j,k)\in\tau(i)} a_{ijk} n_{ijk} \|$. These quantities play a central role in our model: the cross product $(x_j - x_i) \times (x_k - x_i)$ appears directly in the surface-aware face messages (Eq. 1) and in the equivariant coordinate update (Eq. 3), encoding face geometry in a way that is invariant to translation and equivariant to rotation.

Deep learning on meshes Working with mesh data requires methods that incorporates the graph topology and geometric features of the mesh. Equivariant methods commonly rely on the invariants of their input features, which are then updated with learnable functions: prominent examples include E(n)-equivariant graph neural networks (Satorras et al., 2021), their extension to meshes EMNN (Trang et al., 2024), and simplicial variants (Eijkelboom et al., 2023). In this study we adopt EMNN as the mesh-native equivariant backbone, following the original EMNN paper's stronger mesh results relative to EGNN while keeping the architecture simple compared with convolutional (Lim et al., 2018; Hanocka et al., 2019; Gong et al., 2019; Feng et al., 2019; De Haan et al., 2020) and attention-based methods (Basu et al., 2022). A direct EGNN-versus-EMNN comparison is left to future work so we can isolate equivariance per se from the contribution of explicit mesh structure.

3. Methods

3.1. Datasets

Liver surfaces We retrieve a dataset of 200 manually annotated liver surface meshes (Zhang et al., 2025) drawn from three public CT datasets. The task is edge-level segmentation of two anatomical regions, the falciform ligament and the liver ridge, reviewed by clinical experts. Mesh resolution varies substantially (3,000–20,000 edges) due to natural morphological variation across subjects, posing a primary robustness challenge for equivariant segmentation.

Intraoral scans We use two intraoral scan datasets with annotated faces: 3DTeethSeg (Ben-Hamadou et al., 2023), comprising 1,800 scans from 900 patients covering upper and lower jaws separately, and 3D-IOSSeg (Li et al., 2024),

comprising 180 scans with fine-grained teeth annotations. A key distinction is that 3DTeethSeg (and most prior open-source datasets) was acquired by *indirect* scanning via a plaster base, whereas 3D-IOSSeg uses *direct* intraoral scanning, yielding richer surrounding oral tissue and a distribution closer to the real clinical environment. Consequently, 3D-IOSSeg contains substantially more abnormal-tooth cases, raising additional segmentation challenges (see Figure 5 in the appendix). As in previous work (Li et al., 2024; Xi et al., 2025), we downsample the mesh to 16,000 faces.

Intracranial aneurysm segments We use the IntrA dataset (Yang et al., 2020) which contains 103 intracranial aneurysm surface meshes. Each mesh is derived from reconstructing scanned 2D magnetic resonance angiography (MRA) images, and its vertices are annotated with a binary segmentation of the aneurysm sac. Full preprocessing details, including mesh cleanup, coordinate normalisation, spectral caching, and dataset-specific annotation conversions, are provided in Appendix A.

3.2. Mesh featurization

We construct a three-tier per-vertex and per-edge feature set, combining intrinsic shape descriptors, local mesh structure, and dataset-specific global coordinate priors. A central design requirement is that features should respect the symmetry group appropriate to each task. Intracranial aneurysm meshes arrive in arbitrary orientation, so we target invariance to the full E(3) group (rotations, reflections, and translations). Liver and dental meshes have clinically meaningful chirality: left versus right anatomy, so we target SE(3) invariance (rotations and translations only, no reflections). Full feature dimensions and per-dataset configurations are listed in Appendix B.

Intrinsic node features. The primary shape descriptor is the Heat Kernel Signature (HKS) (Sun et al., 2009), computed from the Laplace–Beltrami spectrum and sampled at 8 logarithmically spaced time scales, yielding an 8-dimensional per-vertex feature. As an intrinsic quantity, HKS is invariant to the full E(3) group. We additionally include pointwise area, the mean area of incident triangles, as a single-dimensional local surface-scale cue, which is likewise E(3)-invariant.

Edge features. Each directed edge is assigned a dihedral angle between the normals of its two adjacent face pairs, encoding local surface curvature. Dihedral angles are E(3)-invariant scalars. Degree-normalised and precomputed edge weights (dataset-specific) are also included to modulate message aggregation.

Global coordinate priors. Intrinsic descriptors capture local geometry but not global position within the organ, information that can be essential for tasks with strong spatial label priors. For the dental datasets (3DTeethSeg and 3D-IOSSeg), we compute a PCA-derived anatomical frame from the dental arch and express vertex positions in cylindrical form (r, θ, z) (see Algorithm 1 and the equivariance argument in Appendix B). Because the canonical frame is constructed from the mesh shape itself, its PCA axes rotate consistently with the mesh under SE(3) transformations; the only non-exact step is the dataset-specific sign-disambiguation heuristic used to choose a consistent anatomical direction. For liver segmentation, we do the same construction, but use a different sign resolution strategy, and area-weighted quantities to alleviate tesselation sensitivity; see Appendix B for details.

3.3. Equivariant mesh encoder

The EMNN architecture (Trang et al., 2024) extends EGNN (Satorras et al., 2021) to triangle meshes by exploiting two kinds of geometric quantities: *invariant* scalars—squared edge lengths $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2$ and face areas $\|(\boldsymbol{x}_j - \boldsymbol{x}_i) \times (\boldsymbol{x}_k - \boldsymbol{x}_i)\|$—that serve as safe inputs to scalar feature updates, and *equivariant* vectors—edge displacements $(\boldsymbol{x}_i - \boldsymbol{x}_j)$ and face normals $(\boldsymbol{x}_j - \boldsymbol{x}_i) \times (\boldsymbol{x}_k - \boldsymbol{x}_i)$—that appear exclusively in the coordinate update. The edge-message update retains the EGNN form: $\boldsymbol{m}_{ij} = \phi_e(\boldsymbol{h}_i, \boldsymbol{h}_j, \boldsymbol{e}_{ij}, \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2)$. A surface-aware message from face (i, j, k) to vertex i is additionally defined as

$$\boldsymbol{m}_{ijk} = \phi_f(\boldsymbol{h}_i, \boldsymbol{h}_j + \boldsymbol{h}_k, \|(\boldsymbol{x}_j - \boldsymbol{x}_i) \times (\boldsymbol{x}_k - \boldsymbol{x}_i)\|). \quad (1)$$

The vertex feature update aggregates messages from neighbouring vertices and faces:

$$\boldsymbol{h}_i = \phi_h\left(\boldsymbol{h}_i, \sum_{j \in N(i)} \boldsymbol{m}_{ij}, \sum_{(j,k) \in \tau(i)} \boldsymbol{m}_{ijk}\right). \quad (2)$$

The coordinate update couples the equivariant edge displacements and face normals:

$$\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i + \sum_{j \in N(i)} (\boldsymbol{x}_i - \boldsymbol{x}_j)\, \phi_x(\boldsymbol{m}_{ij}) + \sum_{(j,k) \in \tau(i)} \big((\boldsymbol{x}_j - \boldsymbol{x}_i) \times (\boldsymbol{x}_k - \boldsymbol{x}_i)\big)\, \phi_n(\boldsymbol{m}_{ijk}). \quad (3)$$

The first term propagates pairwise displacement information, while the second injects oriented local surface geometry through face normals without breaking equivariance.[1] All

[1]For the face-normal branch, our E(3) claim follows the oriented-surface convention used in EMNN: each triangular face is ordered so that $(\boldsymbol{x}_j - \boldsymbol{x}_i) \times (\boldsymbol{x}_k - \boldsymbol{x}_i)$ points outward. Under a reflection, the embedded geometry changes handedness, so the reflected face must be rewound by swapping j and k to preserve the outward normal.

EAMS-family encoders add learned graph-level aggregation (i.e. a global node).

3.4. Augmented message passing

Local message passing in an EMNN layer aggregates information only within a vertex's one-ring neighbourhood, limiting the model's receptive field to at most a few hops per layer. To provide longer-range context without adding many layers, we augment the encoder with learnable tokens that are not part of the surface mesh but can exchange information with all real vertices. We study two variants that differ in how strongly they constrain those global summaries: SRAs bias the model toward a small set of semantically distinct regional summaries, whereas virtual nodes provide a more flexible graph-level memory that can relay diffuse long-range context without enforcing a partition.

Soft regional aggregators. The soft regional aggregators (SRAs) partition the mesh into K learnable region prototypes using a differentiable soft assignment. At each forward pass, each vertex i computes an assignment vector $\boldsymbol{a}_i = \mathrm{softmax}(\phi_a(\boldsymbol{h}_i)) \in \mathbb{R}^K$; these stack row-wise into the *assignment matrix* $\boldsymbol{A} \in \mathbb{R}^{N\times K}$ with $[\boldsymbol{A}]_{ik} = [\boldsymbol{a}_i]_k$. The k-th region token is then formed by weighted pooling: $\boldsymbol{r}_k = \sum_i [\boldsymbol{A}]_{ik}\, \boldsymbol{h}_i$, The region tokens are then mixed by a small transformer encoder before being scattered back to nodes (the full configuration is given in Appendix B): $\boldsymbol{h}_i \leftarrow \boldsymbol{h}_i + \alpha \sum_k [\boldsymbol{A}]_{ik}\, \phi_{\mathrm{proj}}(\hat{\boldsymbol{r}}_k)$, where $\hat{\boldsymbol{r}}_k$ denotes the transformer output for region k and α is a learned residual weight. This soft partition creates a structured bottleneck: different vertices attend to different region mixtures, encouraging the network to compress the shape into a small number of semantically meaningful summaries. This bias is most natural when the relevant anatomy can be decomposed into a few coherent subregions and we want the global pathway itself to stay interpretable and spatially organised. The full update procedure is given in Algorithm 2 (Appendix B).

Virtual nodes. The virtual nodes (Zhang et al., 2026, VN), takes a different approach: V virtual-node feature vectors $\{\boldsymbol{v}_k\}$ are initialised from a shared learned seed and V virtual-node coordinates $\{\boldsymbol{u}_k\}$ are initialised at the graph centroid. At each EMNN layer, real nodes exchange messages with virtual nodes as if they were additional graph neighbours, and virtual nodes aggregate information from all real nodes. Unlike SRAs, virtual nodes impose no explicit partition structure and act instead as a small per-graph global memory bank, instead of aggregating existing features. This makes them a better fit when useful context is spatially diffuse or when forcing vertices into a small number of regions would be unnecessarily restrictive. Algorithm 3 (Appendix B) gives the full layer update, showing the virtual-node exchange in the context of the complete EMNN block with edge and face-area branches; relative to the formulation in (Zhang et al., 2026), our model additionally includes the mesh-specific face-area coordinate update.

3.5. Regularization objectives

The augmented modules introduce additional latent structure, so we regularise them to keep their global summaries informative and stable during training.

Boundary aware losses Our supervised objective combines a prediction loss with a boundary-aware embedding loss. Specifically, we use the boundary difference over union (Sun et al., 2023, bDoU) loss for the segmentation predictions and add a contrastive boundary objective (Tang et al., 2022, CBL) on the latent embeddings, encouraging nearby points across class boundaries to become more discriminative. We write this shared task loss as $\mathcal{L}_{\text{task}} = \mathcal{L}_{\text{pred}} + \lambda_{\text{cbl}}\, \mathcal{L}_{\text{cbl}}$.

Regional assignment losses For SRAs, we regularise the assignment matrix $\boldsymbol{A} \in \mathbb{R}^{N\times K}$ (whose rows are the per-vertex assignment vectors $\boldsymbol{a}_i$) with two unsupervised objectives. A *diversity loss* penalises off-diagonal entries of the Gram matrix of the column-normalised assignments: $\mathcal{L}_{\text{div}} = \|\tilde{\boldsymbol{A}}^\top \tilde{\boldsymbol{A}} - \boldsymbol{I}\|_F^2$, where $\tilde{\boldsymbol{A}}$ denotes column-wise ℓ_2-normalised $\boldsymbol{A}$. An *equipartition loss* penalises imbalance in the total mass assigned to each region: $\mathcal{L}_{\text{eq}} = \mathrm{Var}(\boldsymbol{1}^\top \boldsymbol{A})/(\bar{m}^2 + \epsilon)$, where $\bar{m}$ is the mean column mass. The combined loss $\mathcal{L}_{\text{SRA}} = \lambda_{\text{div}}\, \mathcal{L}_{\text{div}} + \lambda_{\text{eq}}\, \mathcal{L}_{\text{eq}}$ is averaged per mesh. Together these terms preserve the intended structured bottleneck by discouraging degenerate reuse of the same region token and preventing all vertices from collapsing into only one or two regions.

Virtual node losses For virtual nodes, we use a kernel-based energy objective on the virtual-node coordinates $\{\boldsymbol{u}_k\}$, following (Zhang et al., 2026). Let $k(\boldsymbol{p}, \boldsymbol{q}) = \exp(-\|\boldsymbol{p} - \boldsymbol{q}\|^2/2\sigma^2)$ be a Gaussian kernel, which is E(3)-invariant. We compute a virtual–virtual repulsion term k_{vv} as the mean off-diagonal pairwise kernel among $\{\boldsymbol{u}_k\}$, and a real–virtual attraction term k_{rv} as the mean kernel between a random subsample of $\{\boldsymbol{x}_i\}$ and $\{\boldsymbol{u}_k\}$. The loss $\mathcal{L}_{\text{VN}} = w_{vv}\, k_{vv} - w_{rv}\, k_{rv}$ simultaneously discourages virtual nodes from collapsing together and encourages them to remain close to the mesh surface. This preserves the virtual nodes' role as flexible relays: they spread out enough to cover different parts of the shape, but remain anchored near the geometry they summarize.

Liver local continuity loss Segmentation of landmarks on the liver surface is particularly challenging due to the subtle geometric cues needed to identify the folds defining the ligament. Moreover, the lack of absolute coordinates

Table 1. Intracranial aneurysm segmentation on the IntrA dataset (Yang et al., 2020). Results are mean ± std over five cross-validation folds. **Bold**: best per column (all rows); underline: second-best. EAMS, SRA+EAMS, and VN+EAMS are E(3)-invariant; their perturbation scores equal their baseline values and are marked †. Full IoU under perturbation is in Table 9.

	Parent vessel					Aneurysm				
	Baseline		Perturbation Dice (%) ↑			Baseline		Perturbation Dice (%) ↑		
Method	Dice (%) ↑	IoU (%) ↑	Rot-z 15°	Rot-z 40°	Refl-x	Dice (%) ↑	IoU (%) ↑	Rot-z 15°	Rot-z 40°	Refl-x
DGCNN (Wang et al., 2019)	97.11 ± 1.35	94.62 ± 2.38	96.75 ± 1.44	94.34 ± 0.88	95.10 ± 0.22	87.77 ± 5.33	80.79 ± 7.45	86.24 ± 5.95	76.61 ± 2.85	77.18 ± 1.32
PTv3 (Wu et al., 2024)	94.86 ± 1.17	90.54 ± 2.11	94.25 ± 1.32	93.06 ± 1.33	93.02 ± 1.52	79.80 ± 4.63	70.59 ± 5.76	78.51 ± 3.93	72.67 ± 4.23	72.68 ± 4.44
EAMS	98.31 ± 1.38	96.87 ± 2.59	98.31 ± 1.38†	98.31 ± 1.38†	98.31 ± 1.38†	93.59 ± 4.71	90.33 ± 7.72	93.59 ± 4.71†	93.59 ± 4.71†	93.59 ± 4.71†
SRA+EAMS	98.69 ± 0.88	97.71 ± 1.68	98.69 ± 0.88†	98.69 ± 0.88†	98.69 ± 0.88†	**95.81 ± 2.90**	**93.32 ± 5.18**	**95.81 ± 2.90**†	**95.81 ± 2.90**†	**95.81 ± 2.90**†
VN+EAMS	**98.90 ± 0.98**	**97.95 ± 1.84**	**98.90 ± 0.98**†	**98.90 ± 0.98**†	**98.90 ± 0.98**†	95.31 ± 4.55	93.15 ± 6.54	95.31 ± 4.55†	95.31 ± 4.55†	95.31 ± 4.55†

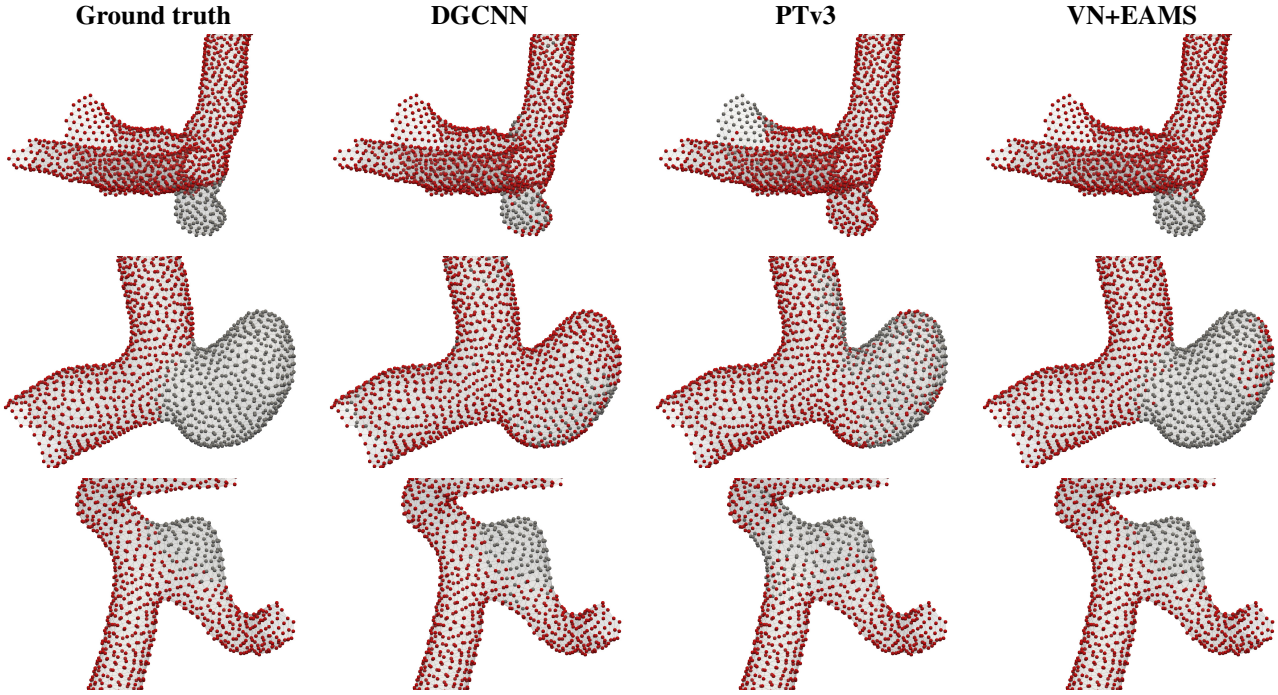


Figure 2. Qualitative IntrA comparisons on representative meshes in the canonical orientation. DGCNN and PTv3 are competitive on easy cases, but the invariant VN+EAMS predictions are consistently cleaner around aneurysm boundaries across the examples shown here.

makes it challenging to identify the "front" of the liver (where both landmarks are located), the "bottom" of the liver (where the ridge is located), and the "middle" of the liver (where the ligament is located). To address this, we introduce a local continuity loss that encourages adjacent edges to have similar predictions. Let $\mathcal{A}$ denote the set of adjacent edge pairs. The loss $\mathcal{L}_{\text{cont}} = \frac{1}{|\mathcal{A}|} \sum_{(e_i,e_j)\in\mathcal{A}} \|\hat{y}_{e_i} - \hat{y}_{e_j}\|^2$ encourages smoothness in the predicted labels across neighboring edges.

Final training objective The full objective is obtained by adding the task-specific auxiliary terms to $\mathcal{L}_{\text{task}}$. For the base EAMS model, we optimise $\mathcal{L}_{\text{task}}$, and on liver meshes we additionally add the continuity term $\lambda_{\text{cont}}\,\mathcal{L}_{\text{cont}}$. For SRA+EAMS, we optimise $\mathcal{L}_{\text{task}} + \mathcal{L}_{\text{SRA}}$, while for VN+EAMS we optimise $\mathcal{L}_{\text{task}} + \mathcal{L}_{\text{VN}}$. All loss weights are given in Appendix C.

4. Experiments

In this section, we report results on three anatomical segmentation tasks. Standard deviations are reported over five cross-validation folds except on Teeth3DS, which is trained once on its designated train/test split.

4.1. Intracranial aneurysm segmentation

We evaluate on the IntrA dataset (Yang et al., 2020), which comprises 103 intracranial aneurysm surface meshes with per-vertex binary labels separating the aneurysm sac from the parent vessel wall. We compare against PointTransformerV3 (PTv3) (Wu et al., 2024) and DGCNN (Wang et al., 2019) as baseline methods. Both baselines operate natively on point clouds and are provided the same vertex coordinates as input; our EAMS variants operate directly on the mesh graph. All results are averaged over five cross-validation folds.

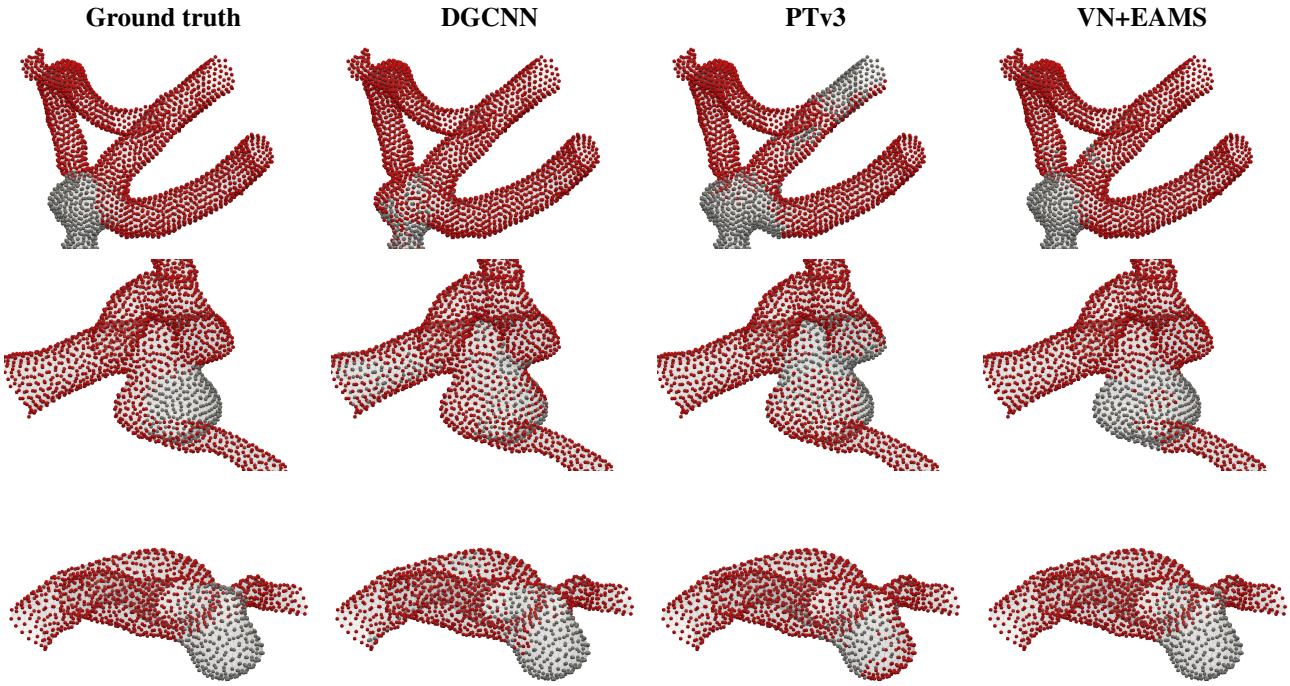


Figure 3. Qualitative IntrA comparisons on representative meshes after a 40° z-axis rotation. Under rotation, DGCNN and PTv3 show visibly noisier aneurysm boundaries and more leakage into the parent vessel, whereas VN+EAMS remains visually consistent across all three cases. Appendix Figure 7 adds the omitted EAMS and SRA+EAMS predictions for the same rotated examples.

Table 1 reports Dice and IoU for both segmentation targets on unperturbed meshes alongside Dice under test-time geometric perturbations. All three EAMS variants improve consistently over both baselines across every metric, especially upon rotation perturbations. Full IoU results under all perturbations are provided in Table 9 in the appendix, and Figures 2 and 3 show that this robustness is also visible qualitatively on representative baseline and 40°-rotated cases.

4.2. Intraoral scan tooth segmentation

We evaluate on two intraoral scan benchmarks sharing the same FDI labelling scheme. **3D-IOSSeg** (Li et al., 2024) comprises 180 direct intraoral scans with per-face labels spanning the gingiva and 16 tooth classes, with results averaged over five cross-validation folds. **Teeth3DS** (Ben-Hamadou et al., 2023) is a larger benchmark of 1800 upper and lower jaw scans from the 3DTeethSeg'22 challenge; owing to its size we train on a single fold on its designated split of 1200 training and 600 test scans. Table 2 reports Average IoU for both datasets: we average IoU over the classes present in each mesh and then average those per-mesh values over the evaluation set, rather than averaging the final per-class IoU summaries. Following common practice we squash mirrored FDI pair labels (e.g. T11/31). We compare against DGCNN (Wang et al., 2019), PTv3 (Wu et al., 2024), and Fast-TGCN (Li et al., 2024); on Teeth3DS we retain Fast-TGCN as the strongest baseline. Per-class breakdowns are provided in Tables 10–13.

On 3D-IOSSeg and Teeth3DS, VN+EAMS approaches the best baseline Fast-TGCN despite being nearly **10× smaller in parameter count** (1.9M vs. 19.8M parameters). Performance on the rearmost wisdom-tooth classes (T18/38 and T28/48) is the main driver of variance across methods, owing to greater morphological variation and fewer training examples. The robustness gap under rotation is stark on both benchmarks, which matters clinically because patient pose and scanner orientation are not guaranteed to match between training and deployment. Figure 1 shows the same trend qualitatively: Fast-TGCN is competitive at baseline but deteriorates under rotation, whereas VN+EAMS remains stable. We also tested an EAMS variant without the dental frame coordinates and observed a **more than 50% drop**, with frequent left/right tooth confusions, confirming the value of this anatomy-aware feature design.

4.3. Liver surface segmentation

We evaluate on the liver surface segmentation dataset of (Zhang et al., 2025), which comprises surface meshes of the liver with per-edge labels spanning three anatomical classes: background, ligament, and ridge. We compare against PointNet++ (Qi et al., 2017), DGCNN (Wang et al., 2019), and MeshGraphCNN (Zhang et al., 2025) as baselines. All three

Table 2. Intraoral scan tooth segmentation on 3D-IOSSeg (Li et al., 2024) and Teeth3DS (Ben-Hamadou et al., 2023). Metric is Average IoU (%), obtained by averaging each mesh over its ground-truth-present classes and then averaging over meshes, not by averaging the final per-class IoU summaries. 3D-IOSSeg results are mean ± std over five cross-validation folds; Teeth3DS results are from a single fold (no std). **Bold**: best per column (all rows); underline: second-best. The PCA-derived canonical frames are SE(3)-equivariant, but the final sign-resolution heuristic can introduce a small residual sensitivity in practice, so we report empirical robustness under all perturbations. Per-class results are in Tables 10 and 11.

	3D-IOSSeg (Li et al., 2024)			Teeth3DS (Ben-Hamadou et al., 2023)		
Method	Baseline ↑	Rot-z 15° ↑	Rot-z 40° ↑	Baseline ↑	Rot-z 15° ↑	Rot-z 40° ↑
DGCNN (Wang et al., 2019)	67.12 ± 0.36	63.25 ± 0.81	42.02 ± 2.46	—	—	—
PTv3 (Wu et al., 2024)	80.08 ± 0.44	75.87 ± 0.45	39.05 ± 2.35	—	—	—
Fast-TGCN (Li et al., 2024)	**81.05 ± 0.74**	76.45 ± 1.14	52.33 ± 1.62	**84.31**	81.84	58.53
EAMS	71.43 ± 1.64	71.71 ± 1.53	72.05 ± 1.60	80.54	80.94	81.11
SRA+EAMS	75.28 ± 1.21	75.49 ± 1.44	75.98 ± 1.23	82.31	82.98	83.05
VN+EAMS	79.74 ± 0.86	**79.92 ± 0.98**	**80.09 ± 1.01**	83.16	**83.33**	**83.40**

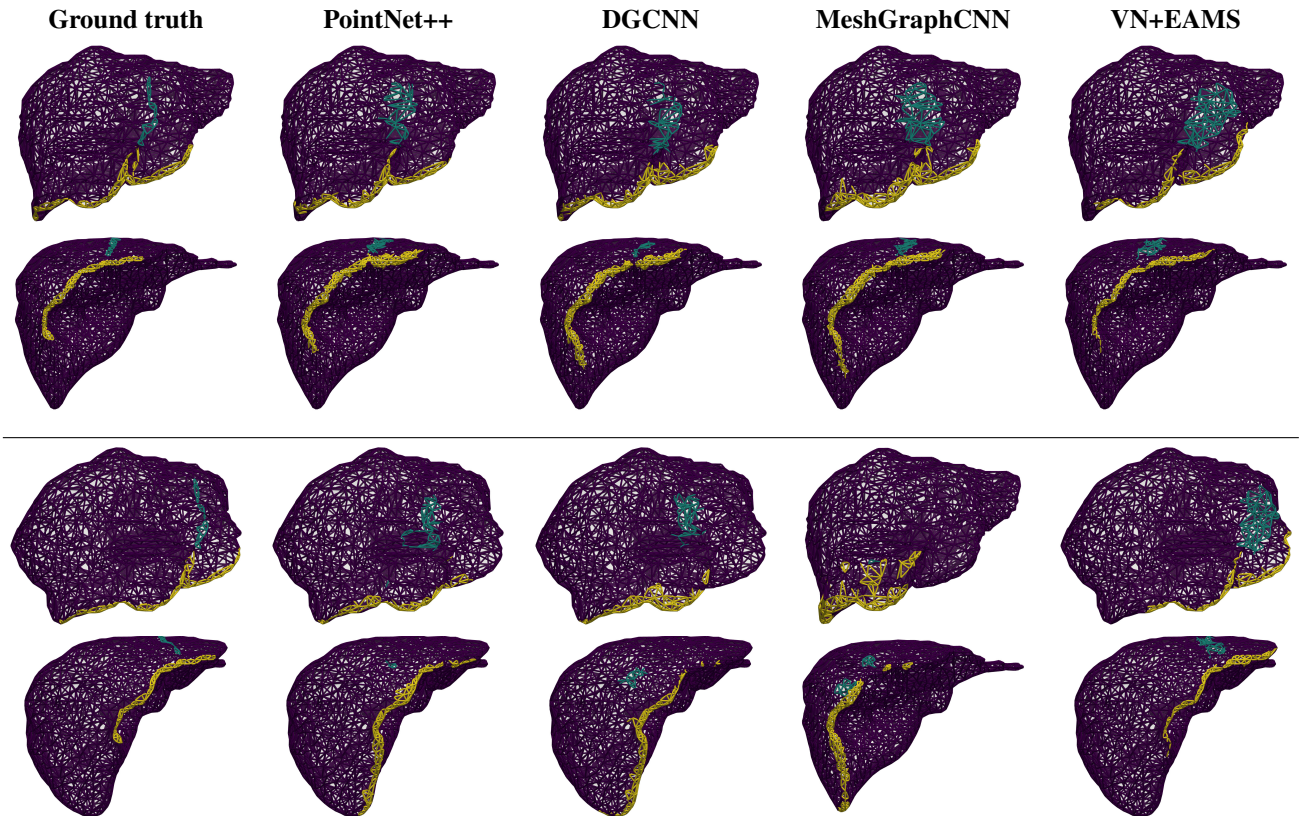


Figure 4. Qualitative liver-surface comparisons on two representative meshes, with the top half in the canonical orientation and the bottom half after a 40° z-axis rotation. MeshGraphCNN uses a slightly different liver mesh because of its preprocessing requirements (Appendix A). All three baselines degrade substantially under rotation, especially on the thin ligament regions. Appendix Figure 8 qualitatively compares three EAMS-family columns.

EAMS variants operate directly on the mesh graph and target SE(3)-invariance; for liver, SRA+EAMS uses $K = 32$ regional tokens and VN+EAMS uses $V = 8$ virtual nodes. This task highlights a balanced trade-off between canonical-pose accuracy and rotation robustness: the baselines can exploit the near-canonical scan alignment in the training set as an orientation prior, whereas EAMS must identify ligament and ridge regions from intrinsic geometry alone.

Table 3 focuses on the main segmentation metric, Dice, together with Chamfer distance (CD) for the ligament and ridge classes on unperturbed meshes and under a 40° z-axis rotation. We report CD after multiplying by 100 for readability. EAMS, SRA+EAMS, and VN+EAMS remain

Table 3. Liver surface segmentation on the dataset of (Zhang et al., 2025). Results are mean ± std over five cross-validation folds. CD is reported after multiplying by 100. **Bold**: best per column (all rows); underline: second-best. †: MeshGraphCNN results are our own reproduction. Full IoU, CD×100, and HD results are in Table 14.

	Ligament				Ridge			
	Baseline		Rot-z 40°		Baseline		Rot-z 40°	
Method	Dice (%) ↑	CD×100 ↓	Dice (%) ↑	CD×100 ↓	Dice (%) ↑	CD×100 ↓	Dice (%) ↑	CD×100 ↓
PointNet++ (Qi et al., 2017)	35.14 ± 2.38	**0.202 ± 0.051**	6.27 ± 1.97	2.842 ± 0.947	60.61 ± 0.91	**0.067 ± 0.007**	44.73 ± 2.53	1.352 ± 0.235
DGCNN (Wang et al., 2019)	34.25 ± 3.92	0.293 ± 0.032	2.77 ± 0.62	4.331 ± 1.134	60.31 ± 1.17	0.080 ± 0.021	42.11 ± 1.10	1.619 ± 0.126
MeshGraphCNN† (Zhang et al., 2025)	**47.00 ± 0.75**	0.677 ± 0.046	4.66 ± 0.66	7.677 ± 0.865	**60.92 ± 1.11**	0.253 ± 0.056	39.64 ± 0.80	2.484 ± 0.073
EAMS	20.46 ± 1.25	3.928 ± 0.668	20.47 ± 1.26	4.307 ± 0.193	56.36 ± 0.82	0.873 ± 0.064	56.35 ± 0.82	0.876 ± 0.065
SRA+EAMS	26.33 ± 2.16	1.722 ± 0.651	**26.32 ± 2.16**	**1.722 ± 0.650**	54.54 ± 1.11	1.071 ± 0.372	54.52 ± 1.14	1.069 ± 0.371
VN+EAMS	20.98 ± 1.38	2.013 ± 0.592	20.99 ± 1.38	2.012 ± 0.591	58.50 ± 2.07	0.610 ± 0.160	**58.51 ± 2.08**	**0.609 ± 0.159**

Table 4. Single-fold IntrA ablations using IoU only. PA = parent vessel, AN = aneurysm.

Config.	PA IoU (%) ↑	AN IoU (%) ↑
EAMS	93.39	79.90
- no global node	92.29	76.90
- no dihedrals	93.05	77.12
- no dihed. & HKS	78.89	49.86
SRA+EAMS ($K = 32$)	94.77	84.12
- no reg. geom.	94.06	83.22
- $K = 8$	94.42	83.52
- $K = 16$	94.55	83.59
- $K = 64$	94.14	82.95
VN+EAMS ($V = 16$)	95.02	83.05
- $V = 8$	94.67	81.62
- $V = 32$	93.31	80.40
- $V = 64$	92.01	75.38

nearly unchanged under rotation, while the non-equivariant baselines degrade sharply. Full IoU, CD×100, and Hausdorff distance (HD) results for the baseline, 15°, and 40° conditions are given in Table 14, and Figure 4 shows representative baseline and 40°-rotated cases.

Accordingly, the EAMS family trails the strongest baselines on unperturbed meshes—particularly for ligament Dice—but remains nearly unchanged under rotation, unlike all three baselines. The ligament and ridge occupy anatomically consistent frontal regions of the liver, so orientation-aware models can use pose as a strong cue, whereas an invariant model must recover them from intrinsic surface geometry alone. Even so, Figure 4 shows competitive qualitative structure. Among the EAMS variants, SRA+EAMS gives the strongest rotated ligament Dice and CD, while VN+EAMS gives the best rotated ridge Dice and CD; Table 14 shows the same stability pattern across the full baseline, 15°, and 40° evaluation suite.

4.4. Ablation studies

Table 4 reports single-fold IntrA ablations using IoU only, isolating the shared global node, the SRA geometry term, and the number of learned global tokens. For base EAMS, removing the global node already hurts both PA and AN IoU, and removing both dihedrals and HKS causes the largest collapse, indicating that HKS remains the dominant non-coordinate cue. SRA+EAMS is stable across the tested region counts, while VN+EAMS is most sensitive to the number of virtual nodes.

Table 5. Simple liver ablation for the base EAMS encoder.

Class	Config.	Dice ↑	CD×100 ↓
Ligament	Base	12.76	5.78
Ligament	+ frame	18.78	4.38
Ligament	+ frame + cont	19.51	3.81
Ridge	Base	51.06	1.70
Ridge	+ frame	56.17	0.81
Ridge	+ frame + cont	56.08	0.88

Table 5 gives a simple liver ablation for the base ("all dataset" config in Table 7) encoder. Adding the anatomical frame gives the main gain on both classes, and the continuity loss provides a smaller follow-up gain, mainly on the ligament metrics.

5. Discussion

EAMS shows that equivariant mesh segmentation is practical across clinically distinct anatomical tasks. For the clinical setting, the main lesson is that geometric priors can make inference on mesh-structured biomedical data generalizable: once orientation is no longer a hidden shortcut, the model behaves more predictably across acquisition settings, and the SRA/VN augmentations provide a lightweight form of multi-scale geometric modelling rather than task-specific engineering. The two augmentations should be read as different architectural biases: SRA imposes a structured bottleneck that is well matched to anatomy that can be compressed into a few coherent regional summaries, whereas virtual nodes provide a more flexible graph-level memory when relevant context is spatially diffuse.

The main limitation is the liver trade-off: strict SE(3) invari-

ance improves robustness but forfeits the global orientation prior that non-invariant methods exploit when training and test meshes share a canonical pose. The augmented variants also incur a modest runtime cost relative to the base encoder. Future work should therefore focus on richer intrinsic descriptors, more principled sign-disambiguation for canonical frames, learned pose-aware aggregation that preserves equivariance, and broader evaluation on cross-dataset anatomical mesh tasks where robustness and generalizability matter for assistive scientific workflows.

Acknowledgements

The author acknowledges funding from the Vector Institute as part of the research internship program. The research was enabled in part by the use of computing resources provided by the Digital Research Alliance of Canada (`alliancecan.ca`).

Impact Statement

This work studies robust anatomical mesh segmentation as a decision-support component for clinical and research workflows. More stable predictions under pose variation could reduce manual annotation burden, improve consistency across acquisition settings, and make downstream geometric analysis less sensitive to scanner orientation, but incorrect segmentations could still propagate harmful errors into measurements, visualisation, or treatment-planning systems. Any deployment should therefore remain clinician-supervised, validated for the target anatomy and acquisition pipeline, and paired with task-specific quality control rather than treated as a fully autonomous system.

References

Basu, S., Gallego-Posada, J., Viganò, F., Rowbottom, J., and Cohen, T. Equivariant mesh attention networks. *arXiv preprint arXiv:2205.10662*, 2022.

Ben-Hamadou, A., Smaoui, O., Rekik, A., Pujades, S., Boyer, E., Lim, H., Kim, M., Lee, M., Chung, M., Shin, Y.-G., et al. 3dteethseg'22: 3d teeth scan segmentation and labeling challenge. *arXiv preprint arXiv:2305.18277*, 2023.

De Haan, P., Weiler, M., Cohen, T., and Welling, M. Gauge equivariant mesh cnns: Anisotropic convolutions on geometric graphs. *arXiv preprint arXiv:2003.05425*, 2020.

Eijkelboom, F., Hesselink, R., and Bekkers, E. J. E (n) equivariant message passing simplicial networks. In *International Conference on Machine Learning*, pp. 9071–9081. PMLR, 2023.

Feng, Y., Feng, Y., You, H., Zhao, X., and Gao, Y. Meshnet: Mesh neural network for 3d shape representation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 8279–8286, 2019.

Gong, S., Chen, L., Bronstein, M., and Zafeiriou, S. Spiralnet++: A fast and highly efficient mesh convolution operator. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, pp. 0–0, 2019.

Hanocka, R., Hertz, A., Fish, N., Giryes, R., Fleishman, S., and Cohen-Or, D. Meshcnn: a network with an edge. *ACM Transactions on Graphics (ToG)*, 38(4):1–12, 2019.

Koo, B., Özgür, E., Le Roy, B., Buc, E., and Bartoli, A. Deformable registration of a preoperative 3d liver volume to a laparoscopy image using contour and shading cues. In *International conference on medical image computing and computer-assisted intervention*, pp. 326–334. Springer, 2017.

Levy, D., Kaba, S.-O., Gonzales, C., Miret, S., and Ravanbakhsh, S. Using multiple vector channels improves e (n)-equivariant graph neural networks. *arXiv preprint arXiv:2309.03139*, 2023.

Li, J., Cheng, B., Niu, N., Gao, G., Ying, S., Shi, J., and Zeng, T. A fine-grained orthodontics segmentation model for 3d intraoral scan data. *Computers in Biology and Medicine*, 168:107821, 2024.

Lim, I., Dielen, A., Campen, M., and Kobbelt, L. A simple approach to intrinsic correspondence learning on unstructured 3d meshes. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pp. 0–0, 2018.

Martin, C. B., Chalmers, E. V., McIntyre, G. T., Cochrane, H., and Mossey, P. A. Orthodontic scanners: what's available? *Journal of orthodontics*, 42(2):136–143, 2015.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

Satorras, V. G., Hoogeboom, E., and Welling, M. E (n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.

Sun, F., Luo, Z., and Li, S. Boundary difference over union loss for medical image segmentation. In *International conference on medical image computing and computer-assisted intervention*, pp. 292–301. Springer, 2023.

Sun, J., Ovsjanikov, M., and Guibas, L. J. A concise and provably informative multi-scale signature based on heat diffusion. In *Computer Graphics Forum*, volume 28, pp. 1383–1392. Wiley Online Library, 2009.

Tang, L., Zhan, Y., Chen, Z., Yu, B., and Tao, D. Contrastive boundary learning for point cloud segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8489–8499, 2022.

Trang, T. A., Ngo, N. K., Levy, D. T., Vo, T. N., Ravanbakhsh, S., and Hy, T. S. E (3)-equivariant mesh neural networks. In *International Conference on Artificial Intelligence and Statistics*, pp. 748–756. PMLR, 2024.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., and Solomon, J. M. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)*, 38(5):1–12, 2019.

Wu, T., Pan, L., Zhang, J., Wang, T., Liu, Z., and Lin, D. Balanced chamfer distance as a comprehensive metric for point cloud completion. *Advances in Neural Information Processing Systems*, 34:29088–29100, 2021.

Wu, W., Qi, Z., and Fuxin, L. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pp. 9621–9630, 2019.

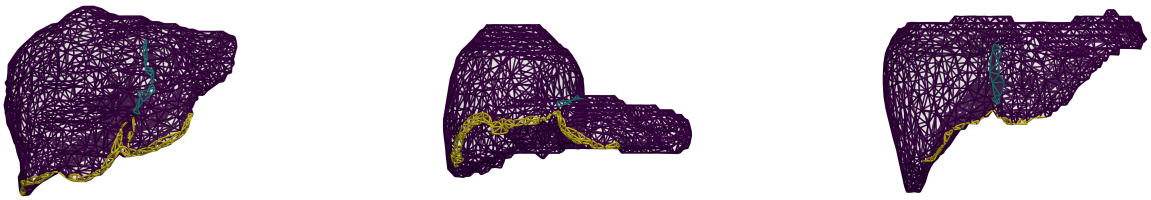
Wu, X., Jiang, L., Wang, P.-S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., and Zhao, H. Point transformer v3: Simpler faster stronger. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4840–4851, 2024.

Xi, S., Liu, Z., Chang, J., Wu, H., Wang, X., and Hao, A. 3d dental model segmentation with geometrical boundary preserving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10476–10485, 2025.

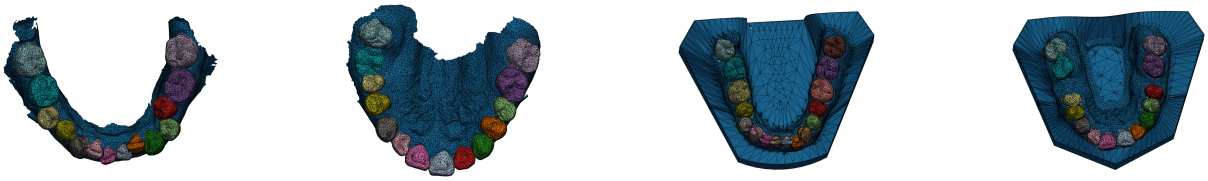
Yang, X., Xia, D., Kin, T., and Igarashi, T. Intra: 3d intracranial aneurysm dataset for deep learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2656–2666, 2020.

Zhang, X., Feng, J., Liu, P., Han, M., Kang, Y., Zhu, J., Wang, L., Wang, X., Ali, S., and Zhang, L. Nested resolution mesh-graph cnn for automated extraction of liver surface anatomical landmarks. *Medical Image Analysis*, pp. 103825, 2025.

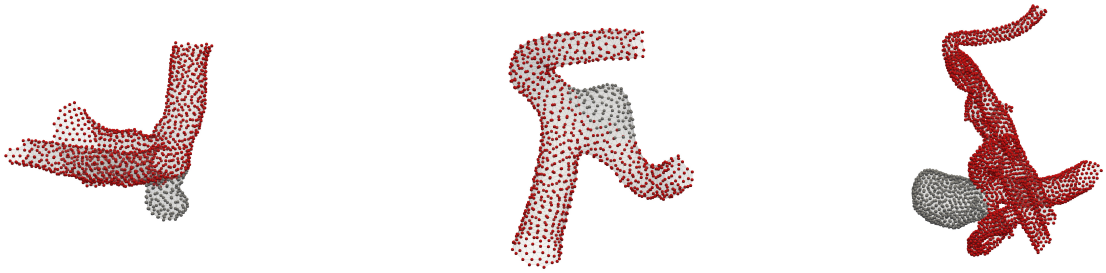
Zhang, Y., Cen, J., Han, J., and Huang, W. Fast and distributed equivariant graph neural networks by virtual node learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.



(a) Liver surfaces (3Dircadb, LiTS, Amos). Edge colours denote the falciform ligament (blue) and liver ridge (yellow) annotation regions.



(b) Intraoral scans. Left pair: 3D-IOSSeg (direct scan, richer oral tissue). Right pair: 3DTeethSeg (indirect scan via plaster base).



(c) Intracranial aneurysm surfaces from IntrA. Vertex colours indicate aneurysm vs. vessel wall labels.

Figure 5. Representative ground-truth segmentations from the three benchmark datasets used in our evaluation.

A. Data processing

Pipeline overview. The data pipeline separates three concerns: raw mesh ingestion, reusable geometric preprocessing, and late feature assembly. Meshes are first converted into graph-structured objects with their supervision targets. Expensive geometric quantities—surface normals, adjacency structures, and the Laplace–Beltrami spectral basis—are computed once and stored in per-mesh caches. Model input tensors are assembled only after batching, so the same cached geometry can support multiple feature configurations without rerunning the full pipeline. The per-dataset feature configurations are listed in Appendix B; the pre-transforms applied before training are described in Appendix C.

Mesh cleanup. Before any geometric operator is computed, each mesh undergoes a defensive cleanup pass[1]. Non-finite vertex coordinates are discarded; faces that reference any removed vertex are also removed. Degenerate triangles (those with repeated vertex indices) and near-zero-area faces are filtered out, and any vertices left unreferenced after face removal are pruned. Connectivity is then rebuilt on the cleaned mesh. This pass is not merely cosmetic: the discrete stiffness and mass matrices used for the Laplace–Beltrami eigensystem are sensitive to degenerate elements, and zero-area faces or isolated vertices can produce near-singular Laplacians that destabilise the spectral decomposition. Cleanup therefore acts as a numerical regularization pass on the mesh geometry itself. Crucially, it does not perform remeshing or vertex merging, so label alignment between the original annotation and the processed mesh is preserved.

[1]This cleanup is the main source of discrepancy between our liver meshes and those in MeshGraphCNN (Zhang et al., 2025), as seen in Figure 4.

Coordinate normalisation. After cleanup, vertex coordinates $\boldsymbol{x}_i \in \mathbb{R}^3$ are centred and isotropically scaled:

$$\boldsymbol{x}_i^{\text{norm}} = \frac{\boldsymbol{x}_i - \bar{\boldsymbol{x}}}{s + \varepsilon}, \qquad \bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i, \tag{4}$$

where s is the overall mesh extent and ε is a small constant for numerical stability. Normalisation makes meshes approximately comparable in magnitude, reduces sensitivity to absolute acquisition scale, and stabilises both the downstream spectral decomposition and network optimisation.

Spectral preprocessing and HKS caching. The most expensive per-mesh step is the spectral decomposition of the discrete Laplace–Beltrami operator. On the normalised mesh, the pipeline solves the generalised eigenproblem

$$\boldsymbol{W}\phi_k = \lambda_k \boldsymbol{M}\phi_k, \tag{5}$$

where $\boldsymbol{W}$ and $\boldsymbol{M}$ are the discrete stiffness and mass matrices respectively. The resulting eigenpairs $\{(\lambda_k, \phi_k)\}$ are stored in the per-mesh cache. Heat Kernel Signatures (Sun et al., 2009) are then evaluated from this cached basis as

$$\text{HKS}(i,t) = \sum_k e^{-\lambda_k t}\, \phi_k(i)^2, \tag{6}$$

at logarithmically spaced time scales, after which a fixed subset of channels is selected and per-mesh normalised to emphasise relative surface variation. Caching the eigensystem avoids repeating the spectral decomposition on every training run; the lightweight HKS channel assembly is deferred to the feature stage.

Late feature assembly. Node and edge features are assembled from the cached mesh objects only after batching. Denoting the cached mesh as $\mathcal{M}$, the model inputs are

$$\boldsymbol{X} = f_{\text{node}}(\mathcal{M}), \qquad \boldsymbol{E} = f_{\text{edge}}(\mathcal{M}), \tag{7}$$

where f_{node} and f_{edge} produce the node- and edge-feature matrices respectively. Because $\mathcal{M}$ is independent of the feature functions, the same cached dataset supports multiple feature configurations without rerunning preprocessing. Stochastic rigid transformation is applied as augmentation during teeth and liver training for the default EAMS runs: the effect is minimal because the canonical-frame features are built from an equivariant PCA construction, with only a small residual sensitivity coming from the final sign-resolution heuristic.

Dataset-specific annotation formats. The four benchmark datasets share the same geometric preprocessing pipeline but differ in how their raw annotations are stored, requiring dataset-specific conversion to per-vertex or per-edge supervision targets.

Intraoral scans (Teeth3DS and 3D-IOSSeg). Teeth3DS provides per-vertex tooth labels directly in the FDI World Dental Federation numbering system (Figure 6). 3D-IOSSeg instead encodes class information as face colours; face classes are decoded from colour values and per-vertex labels are obtained by majority vote over each vertex's incident faces. Following the protocol of (Xi et al., 2025), meshes for both dental datasets are downsampled to 16,000 faces, and tooth labels are projected back onto the simplified geometry after downsampling.

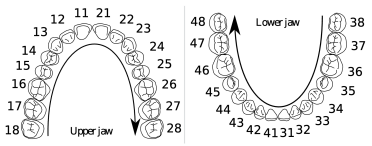


Figure 6. FDI World Dental Federation numbering system for tooth labelling, adapted from (Ben-Hamadou et al., 2023) under CC BY-SA 4.0.

Liver surfaces. The liver dataset (Zhang et al., 2025) provides supervision at the edge level. An external edge graph is supplied alongside each mesh, with per-edge labels covering three anatomical classes: background, falciform ligament, and liver ridge. When geodesic edge weights are used, they follow an inverse-distance scheme $w_{ij} \propto (d_{ij} + \varepsilon)^{-1}$, so geodesically nearby neighbours receive larger influence during message passing.

Intracranial aneurysms (IntrA). IntrA (Yang et al., 2020) provides per-vertex binary labels separating the aneurysm sac from the surrounding vessel wall. When the annotation row count and the mesh vertex count do not match exactly, a nearest-neighbour remapping assigns each mesh vertex the label of its closest annotated point.

B. Mesh featurization

Table 6 lists all features used in our experiments with their dimensionalities and computation methods. Table 7 summarises the per-dataset configuration.

Table 6. Feature definitions. All features are computed after PyG batching. Node features are concatenated into $\boldsymbol{h}_i^{(0)}$; edge features into $\boldsymbol{e}_{ij}$.

Feature	Type	Dim	Description
`pointwise_area`	node	1	Mean area of incident triangles per vertex, computed via scatter aggregation over the vertex-to-face mapping.
`hks`	node	8	Heat Kernel Signature (Sun et al., 2009) computed from the Laplace–Beltrami spectrum via `trimesh`; 8 normalised time scales are concatenated.
`dental_frame_cylindrical`	node	3	Cylindrical coordinates (r, θ, z) of each vertex in a PCA-derived anatomical dental frame. The mesh is centred, PCA axes are estimated, an anterior-posterior sign heuristic selects one of the two equivalent axis orientations, and coordinates are converted to cylindrical form. The frame construction itself is SE(3)-equivariant. Used for Teeth3DS and 3D-IOSSeg only.
`canonical_frame_cylindrical`	node	3	Cylindrical coordinates (r, θ, z) of each vertex in a PCA-derived anatomical liver frame. The mesh is centred using an area-weighted centre of mass, PCA axes are estimated from an area-weighted covariance, an anterior-posterior sign heuristic selects one of the two equivalent axis orientations, and coordinates are converted to cylindrical form. The frame construction itself is SE(3)-equivariant. Used for liver only.
`com_fps_anchor_distances`	node	4	Distances from each node to the centroid and three furthest nodes, computed via a single-pass Furthest Point Sampling (FPS) algorithm.
`degree_weight`	edge	1	Reciprocal of the target-node degree for each directed edge.
`copy_weight`	edge	1	Pre-computed edge weights from the data pipeline when available; falls back to a small constant otherwise.
`dihedrals`	edge	1	Dihedral angle between the normals of the two face pairs sharing the edge; defaults to π for boundary edges with no adjacent face pair.

Equivariance of the dental frame. Under a rigid-body transformation $\boldsymbol{x}_i \mapsto \boldsymbol{R}\boldsymbol{x}_i + \boldsymbol{t}$ with $\boldsymbol{R} \in \mathrm{SO}(3)$, the centred coordinates become $\boldsymbol{R}(\boldsymbol{x}_i - \boldsymbol{c})$, so the covariance transforms as $\boldsymbol{S} \mapsto \boldsymbol{R}\boldsymbol{S}\boldsymbol{R}^\top$ and the PCA eigenvectors as $\boldsymbol{U} \mapsto \boldsymbol{R}\boldsymbol{U}$. Thus the canonical PCA frame itself is exactly SE(3)-equivariant; the only heuristic element is line 3, which chooses between the two sign-equivalent orientations of the leading axis. The AP sign resolution depends only on the projected tooth-arch shape, not on the external orientation. Consequently the cylindrical coordinates (r_i, θ_i, z_i) are *identical before and after any SE(3) transformation*. Reflections (improper rotations) are not compensated: they flip $\det(\boldsymbol{U})$ and would swap left and right arch sides, which is intentional since dental anatomy has left–right chirality.

Liver frame: area-weighted PCA. The liver uses the same overall PCA-to-cylindrical construction as the dental frame, but replaces the uniform centroid and covariance with area-weighted quantities to reduce sensitivity to irregular tessellation. Specifically, if A_i denotes the local vertex area associated with vertex i, we compute the area-weighted centre of mass

$$\boldsymbol{\mu} = \frac{1}{\sum_{i=1}^{N} A_i} \sum_{i=1}^{N} A_i \boldsymbol{x}_i \tag{8}$$

Table 7. Per-dataset default feature configurations and input dimensionalities. The coordinate channels column counts the number of $\mathbb{R}^3$ vector channels propagated alongside the scalar node features.

Dataset	Node features	Edge features	Node dim	Edge dim	Coord. channels
All datasets	`area`, `hks`	`deg`, `copy`, `dihedrals`	9	3	2 (pos. + normal)

Liver	`canonical_cyl`
	`canonical_cart`
	`com_fps_anchor_distances`
		—	19	3	2

Teeth3DS	`dental_cyl`	—	12	3	2
3D-IOSSeg	`dental_cyl`	—	12	3	2
IntrA	—	—	9	3	2

Algorithm 1 Dental cylindrical frame

Require: Vertex coordinates $\{\boldsymbol{x}_i\}$
1: Centre: $\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i - \frac{1}{N}\sum_j \boldsymbol{x}_j$
2: PCA: $[\boldsymbol{U}, \boldsymbol{\Lambda}] \leftarrow \mathrm{eig}(\sum_i \boldsymbol{x}_i \boldsymbol{x}_i^\top)$
3: **Resolve AP sign:** project vertices as $y_i \leftarrow [\boldsymbol{U}^\top \boldsymbol{x}_i]_1$; fit a quadratic $y \mapsto p(y)$ to the arch; if p is minimised at the positive end, negate column 1 of $\boldsymbol{U}$
4: Rotate: $\boldsymbol{y}_i \leftarrow \boldsymbol{U}^\top \boldsymbol{x}_i$
5: Shift posterior baseline: $[\boldsymbol{y}_i]_2 \leftarrow [\boldsymbol{y}_i]_2 - \min_j [\boldsymbol{y}_j]_2$
6: Convert to cylindrical: $(r_i, \theta_i, z_i) \leftarrow \mathrm{cart2cyl}(\boldsymbol{y}_i)$
Ensure: Per-vertex (r_i, θ_i, z_i)

and the area-weighted covariance

$$\boldsymbol{C} = \frac{1}{\sum_{i=1}^N A_i} \sum_{i=1}^N A_i (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^\top. \tag{9}$$

This change matters more for liver than for teeth because liver meshes exhibit much larger inter-patient variation in lobe shape and are also more vulnerable to local remeshing artifacts near vessels or thin folds. Weighting by area makes the canonical frame depend on the organ's physical surface distribution rather than on raw vertex density: densely triangulated nuisance regions contribute little because their associated A_i are small, whereas the dominant right-lobe versus tapering left-lobe mass distribution remains stable. In practice, this yields a more anatomically faithful centre and principal axes, making the liver cylindrical coordinates less sensitive to mesh-resolution artifacts while preserving the same SE(3)-equivariance argument as above. As with the dental frame, the approximation is only in the heuristic sign disambiguation, not in the PCA frame construction itself.

Algorithm 2 Soft regional aggregator (SRA) update

Require: Node features $\{\boldsymbol{h}_i\}$, coordinates $\{\boldsymbol{x}_i\}$, number of regions K, learned residual weight α
1: **for** each node i **do**
2: $\quad \boldsymbol{a}_i \leftarrow \mathrm{softmax}(\phi_a(\boldsymbol{h}_i)) \in \mathbb{R}^K$ (optionally augmented with dist. to CoM $\boldsymbol{u}_i = \|\boldsymbol{x}_i - \bar{\boldsymbol{x}}\|$)
3: **end for**
4: Stack into assignment matrix: $[\boldsymbol{A}]_{ik} \leftarrow [\boldsymbol{a}_i]_k$, so $\boldsymbol{A} \in \mathbb{R}^{N \times K}$
5: **for** each region k **do**
6: $\quad \boldsymbol{r}_k \leftarrow \sum_i [\boldsymbol{A}]_{ik} \boldsymbol{h}_i$ (optionally augmented with geom. feature $\boldsymbol{u}_k^r = \sum_i [\boldsymbol{A}]_{ik} \boldsymbol{u}_i$)
7: **end for**
8: Mix: $[\hat{\boldsymbol{r}}_1, \ldots, \hat{\boldsymbol{r}}_K] \leftarrow \mathrm{TransformerEncoder}([\boldsymbol{r}_1, \ldots, \boldsymbol{r}_K])$
9: **for** each node i **do**
10: $\quad \boldsymbol{h}_i \leftarrow \boldsymbol{h}_i + \alpha \sum_k [\boldsymbol{A}]_{ik} \phi_{\mathrm{proj}}(\hat{\boldsymbol{r}}_k)$
11: **end for**

Algorithm 3 Virtual node update (per EMNN layer)

Require: Node features $\{\boldsymbol{h}_i\}$, coordinates $\{\boldsymbol{x}_i\}$, mesh edges E, faces $\mathcal{F}$; virtual features $\{\boldsymbol{v}_k\}_{k=1}^{V}$, virtual coordinates $\{\boldsymbol{u}_k\}$

Compute standard EMNN edge messages $\boldsymbol{m}_i^{\text{edge}}$ and coordinate updates $\Delta\boldsymbol{x}_i^{\text{edge}}$
Compute face-area messages $\boldsymbol{m}_i^{\text{face}}$ and coordinate updates $\Delta\boldsymbol{x}_i^{\text{face}}$
For each graph, compute the mean node coordinate $\bar{\boldsymbol{x}}$.
$\boldsymbol{m}_k^{v} \leftarrow \|\boldsymbol{u}_k - \bar{\boldsymbol{x}}\|^2$ (correlation feature for virtual node k)
for each node i and each virtual node k in its graph **do**
  $\Delta\boldsymbol{x}_{i\to k} \leftarrow \boldsymbol{u}_k - \boldsymbol{x}_i$
  $\boldsymbol{m}_{ik}^{\text{virt}} \leftarrow \phi_{rv}(\boldsymbol{h}_i,\, \boldsymbol{v}_k,\, \|\Delta\boldsymbol{x}_{i\to k}\|^2,\, \boldsymbol{m}_k^{v})$
end for
Aggregate toward nodes:
  $\boldsymbol{m}_i^{\text{virt}} \leftarrow \frac{1}{V}\sum_{k=1}^{V} \boldsymbol{m}_{ik}^{\text{virt}}$
  $\Delta\boldsymbol{x}_i^{\text{virt}} \leftarrow \frac{1}{V}\sum_{k=1}^{V} \phi_{\text{virt}\to\text{node}}(\boldsymbol{m}_{ik}^{\text{virt}})(\boldsymbol{x}_i - \boldsymbol{u}_k)$
Aggregate toward virtual nodes:
  $\bar{\boldsymbol{m}}_k^{\text{virt}} \leftarrow \frac{1}{N_g}\sum_{i\in g} \boldsymbol{m}_{ik}^{\text{virt}}$
  $\Delta\boldsymbol{u}_k \leftarrow \frac{1}{N_g}\sum_{i\in g} \phi_{\text{node}\to\text{virt}}(\boldsymbol{m}_{ik}^{\text{virt}})(\boldsymbol{u}_k - \boldsymbol{x}_i)$
Update virtual states: $\boldsymbol{v}_k \leftarrow \boldsymbol{v}_k + \phi_v(\boldsymbol{v}_k,\, \bar{\boldsymbol{m}}_k^{\text{virt}})$
Optionally project or collapse coordinate channels before adding the propagated updates
Update node coordinates: $\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i + \Delta\boldsymbol{x}_i^{\text{edge}} + \Delta\boldsymbol{x}_i^{\text{face}} + \Delta\boldsymbol{x}_i^{\text{virt}}$
Update virtual coordinates: $\boldsymbol{u}_k \leftarrow \boldsymbol{u}_k + \Delta\boldsymbol{u}_k$
Update node features:
  $\boldsymbol{h}_i \leftarrow \phi_h([\boldsymbol{h}_i,\, \boldsymbol{m}_i^{\text{edge}},\, \boldsymbol{m}_i^{\text{face}},\, \boldsymbol{m}_i^{\text{virt}}])$

Ensure: Updated $\{\boldsymbol{h}_i, \boldsymbol{x}_i, \boldsymbol{v}_k, \boldsymbol{u}_k\}$

C. Training and implementation details

Optimiser and learning rate schedule. All models are trained with AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 0.01). We use a plateau-based learning rate schedule: initial LR 10^{-3}, reduction factor 0.6, minimum LR 10^{-5}. Gradient norms are clipped to 1.0. Training runs for 100 epochs with validation every 5 epochs; all inputs are 32-bit floating point.

Batch size and cross-validation. All experiments use batch size 1, except liver segmentation which uses batch size 8. We perform 5-fold cross-validation for the liver, 3D-IOSSeg, and IntrA datasets, and a single train/test split for Teeth3DS following the protocol of (Ben-Hamadou et al., 2023).

Loss weights. The prediction loss coefficient is set to 1.0 for all tasks. The contrastive boundary loss coefficient is set to 10.0 for all tasks, except for liver segmentation where it is set to 1.0. For liver segmentation, the local continuity loss uses $\lambda_{\text{cont}} = 1.0$. For SRA runs, the assignment regularization losses use $\lambda_{\text{div}} = \lambda_{\text{eq}} = 1.0$. For virtual-node runs, the virtual-node coordinate loss uses $w_{vv} = w_{rv} = 1.0$.

Global-token configuration. For the liver experiments, SRA+EAMS uses $K = 32$ soft regional aggregators and VN+EAMS uses $V = 8$ virtual nodes.

Encoder architecture. The encoder is an augmented EMNN with hidden dimensions $(128, 128, 128)$, SiLU activations, and batch normalization. We use a multi-vector channel (Levy et al., 2023; Trang et al., 2024), which replaces vector features $\boldsymbol{x}_i \in \mathbb{R}^3$ with $\boldsymbol{X}_i \in \mathbb{R}^{3C}$ for some small C; we typically set $C = 2$ where the extra channel comes from the normal vector.

Decoder architecture. For all datasets except liver segmentation, predictions are produced by an MLP decoder applied independently at each node, $\hat{\boldsymbol{y}}_i = f_{\text{dec}}(\boldsymbol{h}_i)$, where f_{dec} has hidden dimensions $(128, 128, 128)$, SiLU activations, and dropout 0.1. For liver segmentation, supervision is defined on directed edges, so we instead decode edgewise messages and

scatter them back to vertices. For an edge (j, i), the decoder forms

$$\boldsymbol{z}_{ji} = [\boldsymbol{h}_j \,\|\, \boldsymbol{h}_i \,\|\, \boldsymbol{e}_{ji} \,\|\, \psi(\|\boldsymbol{x}_j - \boldsymbol{x}_i\|_2)],$$

predicts an edge label $\hat{\boldsymbol{y}}_{ji} = f_{\text{edge}}(\boldsymbol{z}_{ji})$, and aggregates incoming edge predictions as $\hat{\boldsymbol{y}}_i = \sum_{j \in \mathcal{N}(i)} \hat{\boldsymbol{y}}_{ji}$. Here f_{edge} uses the same hidden dimensions, activation, and dropout as the nodewise MLP decoder, while the distance embedding ψ is a one-hidden-layer MLP with hidden width 128 and output dimension 16.

Pre-transforms. Meshes are pre-processed once before training with the following deterministic transforms: coordinate normalisation, vertex-normal estimation, face-adjacency structures, vertex-to-edge mapping, and cached HKS computation. When stochastic augmentation is applied, random rigid transformations are sampled with rotation angles uniformly distributed in $[-\pi * 0.15, \pi * 0.15]$ and translation magnitudes uniformly distributed in $[-6, 6] \times [-8, 8] \times [-5, 5]$ before normalisation.

Metrics. We report Intersection over Union (IoU) across all tasks. For liver segmentation we additionally report Dice, Chamfer distance (Wu et al., 2021), and Hausdorff distance. For IntrA we additionally report Dice.

Training runtimes. All experiments are run on a single H100 GPU. Table 8 reports per-epoch wall-clock runtimes for the base EAMS model. Relative to EAMS, SRA+EAMS increases runtime by a factor of $1.45\times$ and VN+EAMS by $1.64\times$.

Table 8. Per-epoch runtime for EAMS.

Dataset	Runtime / epoch
Liver	7 s
3D-IOSSeg	9.2 s
Teeth3DS	72 s
IntrA	1.2 s

D. Broader impacts

This appendix is intentionally omitted because the required Impact Statement now appears in the main paper before the references.

E. Additional experimental results

E.1. Intracranial aneurysm segmentation: full robustness results

Table 9 extends Table 1 with IoU scores for both segmentation targets across all test-time geometric perturbations. EAMS, SRA+EAMS, and VN+EAMS are E(3)-invariant, so their scores are identical to the unperturbed results in Table 1 and are omitted here.

E.2. Intraoral scan tooth segmentation: per-class results

Tables 10–13 extend Table 2 with per-class IoU (%) for the left and right jaw halves of both intraoral benchmarks. These appendix rows are class-wise summaries, whereas the main-table Average IoU averages each mesh over its ground-truth-present classes and then averages over meshes. Column headers use squashed FDI notation: Txy/x′y′ averages the mirrored upper- and lower-jaw classes. For 3D-IOSSeg (Tables 10–11), EAMS methods are listed at Baseline only. For Teeth3DS (Tables 12–13), all conditions are listed for all methods owing to PCA-induced imperfect invariance; no standard deviation is reported (single fold).

E.3. Liver surface segmentation: full robustness and distance results

Table 14 extends Table 3 with IoU and Hausdorff distance (HD), and reports Chamfer distance after multiplying by 100 for readability, across all listed test-time conditions. The EAMS-family rows show only minor variation across rotations, in contrast to the large degradation of the non-equivariant baselines. Among the EAMS variants, SRA+EAMS gives the strongest ligament distance metrics, while VN+EAMS gives the strongest ridge distance metrics.

Table 9. Full robustness results for intracranial aneurysm segmentation on the IntrA dataset. Results are mean ± std over five cross-validation folds. EAMS, SRA+EAMS, and VN+EAMS are E(3)-invariant; their scores are identical to Table 1 under all listed transformations and are omitted.

		Parent vessel		Aneurysm	
Method	Condition	Dice (%) ↑	IoU (%) ↑	Dice (%) ↑	IoU (%) ↑
DGCNN (Wang et al., 2019)	Baseline	97.11 ± 1.35	94.62 ± 2.38	87.77 ± 5.33	80.79 ± 7.45
	Rot-z 15°	96.75 ± 1.44	93.95 ± 2.52	86.24 ± 5.95	78.42 ± 7.88
	Rot-z 40°	94.34 ± 0.88	89.59 ± 1.53	76.61 ± 2.85	65.20 ± 3.73
	Refl-x	95.10 ± 0.22	90.90 ± 0.32	77.18 ± 1.32	66.77 ± 1.17
PTv3 (Wu et al., 2024)	Baseline	94.86 ± 1.17	90.54 ± 2.11	79.80 ± 4.63	70.59 ± 5.76
	Rot-z 15°	94.25 ± 1.32	89.49 ± 2.33	78.51 ± 3.93	68.54 ± 5.18
	Rot-z 40°	93.06 ± 1.33	87.40 ± 2.29	72.67 ± 4.23	62.18 ± 5.05
	Refl-x	93.02 ± 1.52	87.40 ± 2.56	72.68 ± 4.44	62.61 ± 5.29

Table 10. Per-class IoU (%) for the left jaw half on 3D-IOSSeg (Li et al., 2024) (Gingiva and T11/31–T18/38). Results are mean ± std over five cross-validation folds. Baseline scores are mostly unchanged under rotation.

Method	Condition	Gingiva ↑	T11/31 ↑	T12/32 ↑	T13/33 ↑	T14/34 ↑	T15/35 ↑	T16/36 ↑	T17/37 ↑	T18/38 ↑
DGCNN (Wang et al., 2019)	Baseline	81.69 ± 0.68	60.02 ± 0.41	60.00 ± 0.67	71.68 ± 0.82	71.55 ± 0.40	64.33 ± 0.32	74.95 ± 0.36	75.33 ± 1.68	50.93 ± 1.10
	Rot-z 15°	79.36 ± 0.32	54.56 ± 1.24	52.68 ± 1.53	67.77 ± 0.71	65.95 ± 1.60	57.08 ± 0.56	73.71 ± 0.89	74.89 ± 1.83	47.29 ± 1.68
	Rot-z 40°	72.41 ± 1.43	29.86 ± 2.96	28.00 ± 3.09	46.15 ± 4.36	42.99 ± 2.40	27.02 ± 0.74	53.82 ± 2.38	41.85 ± 5.00	33.36 ± 5.50
PTv3 (Wu et al., 2024)	Baseline	91.09 ± 0.22	79.95 ± 0.62	77.55 ± 0.45	83.99 ± 0.64	81.31 ± 0.57	80.88 ± 1.51	83.77 ± 0.70	83.33 ± 0.97	54.23 ± 2.71
	Rot-z 15°	90.50 ± 0.20	76.48 ± 0.98	71.93 ± 1.55	80.42 ± 0.86	77.25 ± 0.46	73.87 ± 1.25	82.05 ± 0.98	81.90 ± 1.13	34.27 ± 6.71
	Rot-z 40°	80.92 ± 0.61	54.92 ± 2.23	28.71 ± 2.46	53.94 ± 4.05	28.53 ± 3.75	21.91 ± 6.41	34.62 ± 5.91	35.23 ± 3.65	10.79 ± 3.65
Fast-TGCN (Li et al., 2024)	Baseline	94.80 ± 0.66	83.96 ± 0.93	77.78 ± 2.07	84.70 ± 0.42	77.58 ± 1.79	80.20 ± 1.06	82.09 ± 0.93	84.80 ± 0.87	56.12 ± 3.26
	Rot-z 15°	94.00 ± 0.78	81.52 ± 1.66	72.83 ± 1.80	79.39 ± 0.61	71.63 ± 2.85	75.02 ± 1.87	79.96 ± 1.22	82.70 ± 1.10	52.54 ± 1.79
	Rot-z 40°	84.90 ± 2.32	54.67 ± 3.50	38.50 ± 3.15	46.09 ± 2.71	42.71 ± 1.42	49.19 ± 0.76	63.91 ± 2.41	67.19 ± 2.51	43.16 ± 4.78
EAMS (ours)	Baseline	93.15 ± 0.75	66.18 ± 3.22	69.83 ± 1.93	73.80 ± 2.51	71.46 ± 2.22	67.08 ± 2.06	76.23 ± 1.75	74.93 ± 2.71	26.04 ± 5.75
SRA+EAMS (ours)	Baseline	94.29 ± 0.38	77.19 ± 2.03	75.33 ± 1.07	78.78 ± 1.28	72.40 ± 1.86	71.99 ± 1.50	75.01 ± 1.40	77.00 ± 1.65	19.52 ± 4.43
VN+EAMS (ours)	Baseline	93.87 ± 0.71	80.16 ± 1.35	80.00 ± 0.68	84.85 ± 0.90	77.94 ± 0.91	78.20 ± 0.82	82.90 ± 0.71	82.17 ± 1.36	44.90 ± 7.71

Table 11. Per-class IoU (%) for the right jaw half on 3D-IOSSeg (Li et al., 2024) (T21/41–T28/48). Results are mean ± std over five cross-validation folds. Baseline scores are mostly unchanged under rotation.

Method	Condition	T21/41 ↑	T22/42 ↑	T23/43 ↑	T24/44 ↑	T25/45 ↑	T26/46 ↑	T27/47 ↑	T28/48 ↑
DGCNN (Wang et al., 2019)	Baseline	61.89 ± 1.22	60.57 ± 1.79	70.44 ± 1.05	70.10 ± 0.28	66.39 ± 0.48	70.69 ± 0.39	69.18 ± 1.18	55.80 ± 0.96
	Rot-z 15°	54.77 ± 1.37	55.35 ± 1.25	65.21 ± 0.46	66.75 ± 0.22	63.95 ± 0.28	68.53 ± 0.38	68.94 ± 1.52	53.08 ± 2.62
	Rot-z 40°	28.21 ± 1.41	29.98 ± 3.45	46.87 ± 1.92	55.17 ± 1.96	49.64 ± 1.56	52.65 ± 3.06	48.98 ± 4.36	24.86 ± 8.75
PTv3 (Wu et al., 2024)	Baseline	80.04 ± 0.90	77.58 ± 0.84	83.15 ± 0.38	81.32 ± 0.59	78.34 ± 0.98	79.24 ± 1.52	78.97 ± 1.22	59.51 ± 3.31
	Rot-z 15°	74.91 ± 1.16	73.93 ± 0.90	79.77 ± 1.05	77.48 ± 0.77	74.84 ± 1.27	74.70 ± 1.53	75.54 ± 1.07	48.60 ± 8.30
	Rot-z 40°	53.85 ± 1.48	44.78 ± 1.32	46.37 ± 3.93	35.08 ± 3.54	23.32 ± 8.13	30.94 ± 7.96	28.99 ± 4.76	24.90 ± 2.15
Fast-TGCN (Li et al., 2024)	Baseline	85.53 ± 0.88	82.43 ± 1.41	87.28 ± 1.02	82.70 ± 0.68	74.43 ± 1.67	76.35 ± 0.73	78.14 ± 2.28	46.50 ± 9.52
	Rot-z 15°	81.77 ± 1.42	76.38 ± 2.04	77.91 ± 0.76	71.81 ± 1.39	66.08 ± 1.69	75.08 ± 1.23	78.09 ± 2.97	53.29 ± 6.24
	Rot-z 40°	57.53 ± 3.55	45.88 ± 4.40	41.52 ± 2.48	38.50 ± 1.54	41.32 ± 3.51	57.72 ± 2.94	64.30 ± 3.02	49.42 ± 6.71
EAMS (ours)	Baseline	68.50 ± 3.20	67.63 ± 2.31	73.54 ± 2.04	72.32 ± 1.89	67.50 ± 1.59	72.28 ± 1.31	73.61 ± 1.96	36.78 ± 3.26
SRA+EAMS (ours)	Baseline	79.57 ± 1.77	77.09 ± 1.69	77.78 ± 1.57	76.06 ± 1.38	68.19 ± 1.88	71.39 ± 1.84	77.80 ± 2.29	27.73 ± 8.92
VN+EAMS (ours)	Baseline	80.87 ± 2.94	78.84 ± 2.25	82.66 ± 1.08	82.39 ± 1.28	75.29 ± 1.04	77.55 ± 1.52	77.68 ± 0.71	43.31 ± 5.77

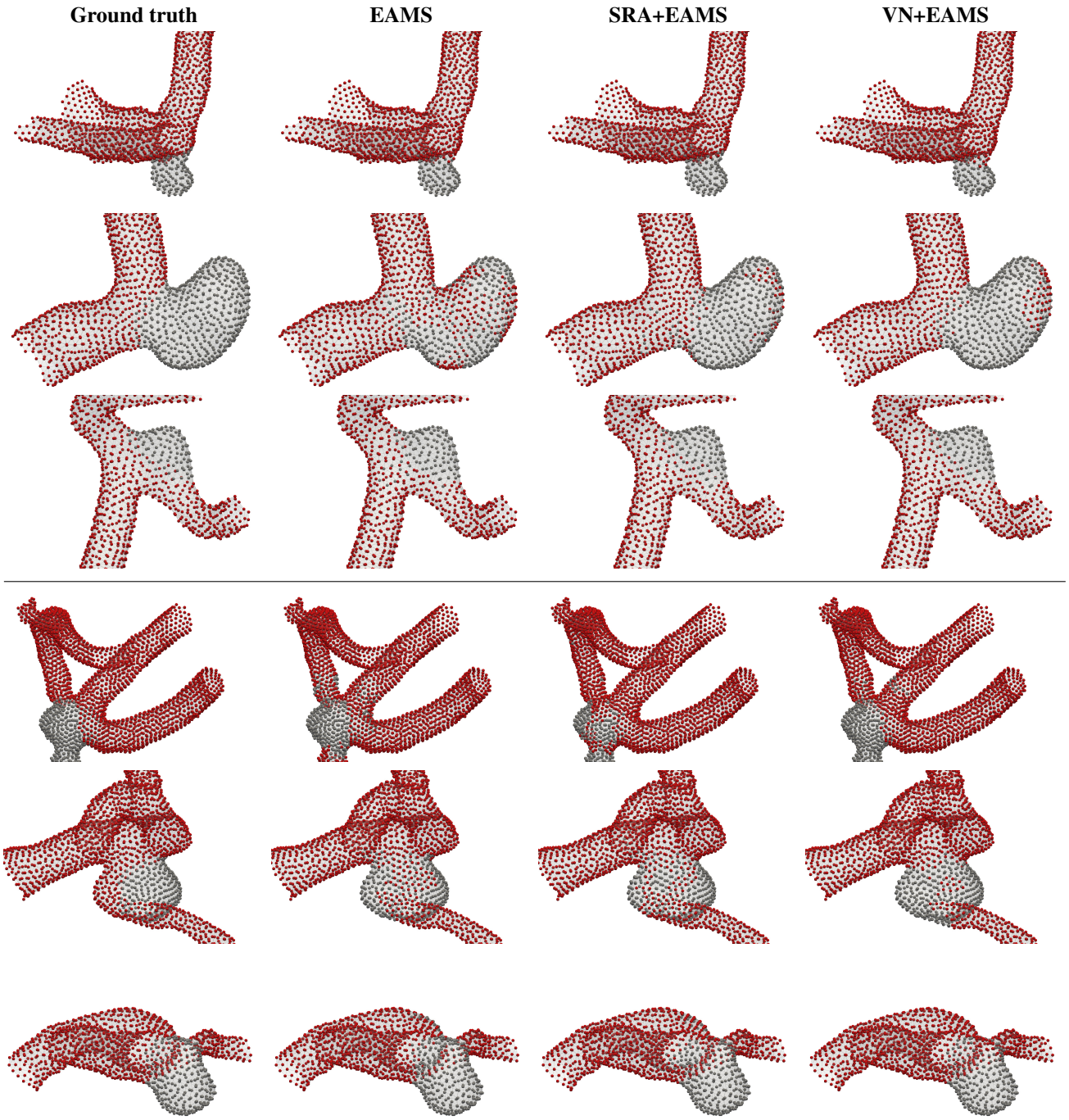


Figure 7. Additional qualitative IntrA comparison for the invariant mesh variants on the same cases used in Figures 2 and 3, with the top half showing the canonical orientation and the bottom half showing a 40° z-axis rotation. This appendix-only figure restores the omitted EAMS and SRA+EAMS predictions, showing that all three mesh-based variants remain visually stable under 40° rotation, with SRA+EAMS and VN+EAMS typically producing the cleanest aneurysm boundaries.

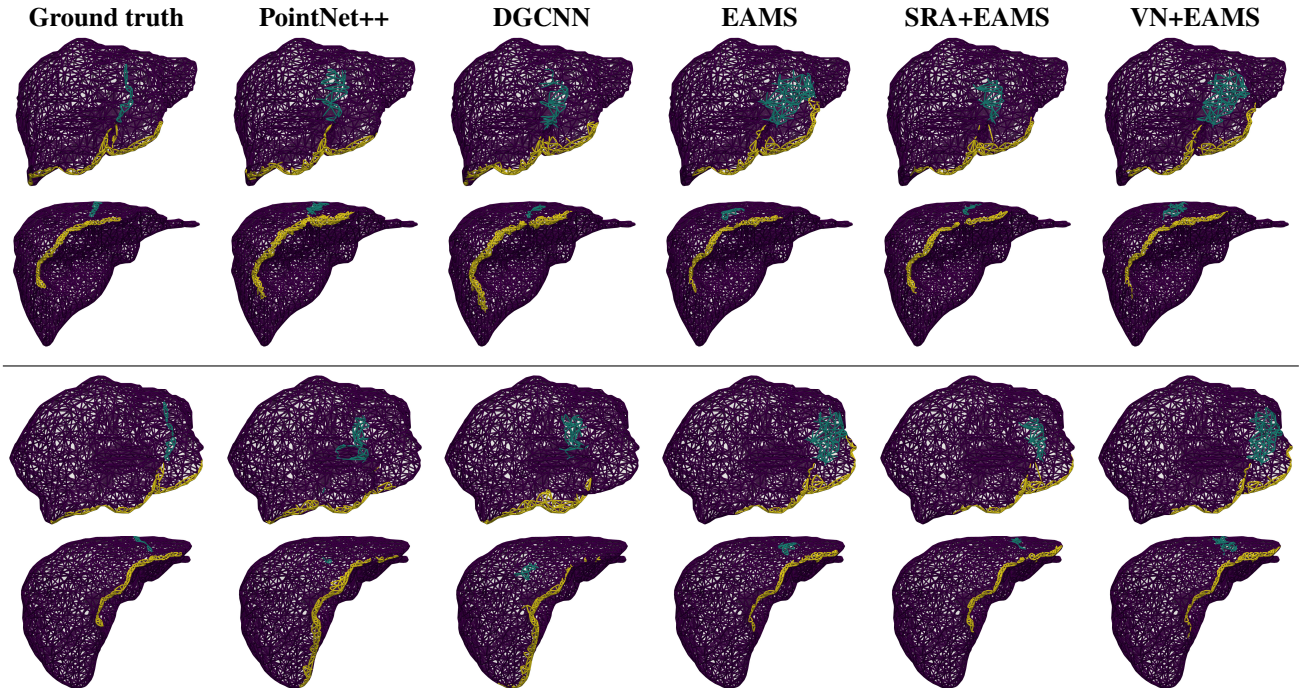


Figure 8. Additional qualitative liver results beyond the main-text figure. The extra columns add PointNet++, DGCNN, and the full EAMS family on representative cases from the liver task, with the top half in the canonical orientation and the bottom half after a 40° z-axis rotation. These extra results make the same robustness pattern explicit across both non-equivariant point baselines and invariant mesh variants.

Table 12. Per-class IoU (%) for the left jaw half on Teeth3DS (Ben-Hamadou et al., 2023) (Gingiva and T11/31–T18/38). Single-fold results; no standard deviation reported. All conditions listed for all methods due to PCA-induced imperfect SE(3)-invariance.

Method	Condition	Gingiva ↑	T11/31 ↑	T12/32 ↑	T13/33 ↑	T14/34 ↑	T15/35 ↑	T16/36 ↑	T17/37 ↑	T18/38 ↑
Fast-TGCN (Li et al., 2024)	Baseline	90.63	85.73	84.02	82.55	87.25	85.90	85.55	75.35	55.69
	Rot-z 15°	90.37	83.27	81.32	79.72	84.16	82.17	81.73	71.64	36.88
	Rot-z 40°	86.20	57.21	55.35	56.87	50.97	53.54	65.15	55.06	19.28
EAMS (ours)	Baseline	90.53	80.80	80.55	82.28	82.41	78.22	79.81	75.27	62.87
	Rot-z 15°	90.68	81.17	81.33	82.97	82.98	78.55	79.86	75.48	70.10
	Rot-z 40°	90.73	81.48	81.74	83.37	83.39	78.75	79.88	75.52	70.11
SRA+EAMS (ours)	Baseline	90.42	84.34	83.17	83.78	85.41	82.39	82.15	73.64	57.64
	Rot-z 15°	90.74	84.32	83.58	84.50	85.67	82.65	83.01	75.86	69.54
	Rot-z 40°	90.76	84.42	83.58	84.46	85.69	82.70	83.09	75.89	69.51
VN+EAMS (ours)	Baseline	91.18	85.11	84.07	83.88	84.99	81.79	82.04	76.32	36.04
	Rot-z 15°	91.26	84.89	83.79	83.52	85.03	81.73	82.19	76.93	48.14
	Rot-z 40°	91.28	84.99	83.95	83.75	85.15	81.86	82.28	77.05	48.14

Table 13. Per-class IoU (%) for the right jaw half on Teeth3DS (Ben-Hamadou et al., 2023) (T21/41–T28/48). Single-fold results; no standard deviation reported. All conditions listed for all methods due to PCA-induced imperfect SE(3)-invariance.

Method	Condition	T21/41 ↑	T22/42 ↑	T23/43 ↑	T24/44 ↑	T25/45 ↑	T26/46 ↑	T27/47 ↑	T28/48 ↑
Fast-TGCN (Li et al., 2024)	Baseline	84.62	84.04	83.12	87.66	87.66	85.78	75.22	57.28
	Rot-z 15°	83.68	81.76	82.40	86.02	83.74	82.83	73.26	61.85
	Rot-z 40°	61.81	54.16	56.86	66.17	54.32	49.73	57.37	46.00
EAMS (ours)	Baseline	80.04	79.40	81.66	81.49	80.33	81.83	75.27	63.04
	Rot-z 15°	80.17	79.51	81.64	81.96	81.15	82.53	75.73	68.62
	Rot-z 40°	80.41	79.69	81.70	81.96	81.10	82.55	75.83	68.58
SRA+EAMS (ours)	Baseline	83.36	82.53	83.37	84.41	82.46	81.26	72.71	58.28
	Rot-z 15°	83.67	83.02	83.75	84.86	82.92	82.38	74.76	68.98
	Rot-z 40°	83.78	83.07	83.67	84.87	82.93	82.59	75.32	69.00
VN+EAMS (ours)	Baseline	85.05	84.63	84.72	84.84	82.59	82.01	75.19	44.90
	Rot-z 15°	84.86	84.78	85.00	85.07	82.84	82.41	76.28	57.18
	Rot-z 40°	85.02	84.80	85.00	85.06	82.81	82.41	76.18	57.16

Table 14. Full robustness results for liver surface segmentation on the dataset of (Zhang et al., 2025). Results are mean ± std over five cross-validation folds. CD is reported after multiplying by 100. Table 3 shows the corresponding Dice and CD summaries for the baseline and 40° conditions.

		Ligament			Ridge		
Method	Condition	IoU (%) ↑	CD×100 ↓	HD ↓	IoU (%) ↑	CD×100 ↓	HD ↓
PointNet++ (Qi et al., 2017)	Baseline	23.65 ± 1.76	0.202 ± 0.051	0.0536 ± 0.0041	44.65 ± 0.94	0.067 ± 0.007	0.0498 ± 0.0028
	Rot-z 15°	14.36 ± 1.38	0.444 ± 0.099	0.0717 ± 0.0061	43.24 ± 0.39	0.176 ± 0.037	0.0704 ± 0.0077
	Rot-z 40°	3.94 ± 1.40	2.842 ± 0.947	0.1490 ± 0.0173	29.51 ± 2.14	1.352 ± 0.235	0.2037 ± 0.0169
DGCNN (Wang et al., 2019)	Baseline	22.60 ± 2.80	0.293 ± 0.032	0.0629 ± 0.0032	44.42 ± 1.10	0.080 ± 0.021	0.0534 ± 0.0055
	Rot-z 15°	17.31 ± 1.48	0.735 ± 0.177	0.0943 ± 0.0033	42.70 ± 1.06	0.189 ± 0.075	0.0826 ± 0.0129
	Rot-z 40°	1.58 ± 0.41	4.331 ± 1.134	0.1929 ± 0.0163	27.37 ± 0.93	1.619 ± 0.126	0.2229 ± 0.0116
MeshGraphCNN (Zhang et al., 2025)	Baseline	34.57 ± 0.99	0.677 ± 0.046	0.1127 ± 0.0066	46.96 ± 0.87	0.253 ± 0.056	0.0760 ± 0.0104
	Rot-z 15°	20.59 ± 0.31	1.424 ± 0.036	0.1431 ± 0.0063	41.32 ± 0.85	0.507 ± 0.053	0.1143 ± 0.0069
	Rot-z 40°	2.59 ± 0.38	7.677 ± 0.865	0.2810 ± 0.0083	25.52 ± 0.55	2.484 ± 0.073	0.2840 ± 0.0056
EAMS (ours)	Baseline	13.02 ± 0.75	3.928 ± 0.668	0.1353 ± 0.0153	41.12 ± 0.96	0.873 ± 0.064	0.1510 ± 0.0046
	Rot-z 15°	13.02 ± 0.75	4.118 ± 0.527	0.1376 ± 0.0129	41.12 ± 0.94	0.874 ± 0.065	0.1512 ± 0.0044
	Rot-z 40°	13.03 ± 0.75	4.307 ± 0.193	0.1399 ± 0.0094	41.11 ± 0.96	0.876 ± 0.065	0.1511 ± 0.0044
SRA+EAMS (ours)	Baseline	17.10 ± 1.54	1.722 ± 0.651	0.0953 ± 0.0051	39.35 ± 0.63	1.071 ± 0.372	0.1416 ± 0.0267
	Rot-z 15°	17.13 ± 1.52	1.722 ± 0.650	0.0954 ± 0.0051	39.35 ± 0.65	1.070 ± 0.371	0.1416 ± 0.0267
	Rot-z 40°	17.10 ± 1.54	1.722 ± 0.650	0.0953 ± 0.0051	39.34 ± 0.66	1.069 ± 0.371	0.1416 ± 0.0267
VN+EAMS (ours)	Baseline	13.42 ± 0.76	2.013 ± 0.592	0.1161 ± 0.0260	42.74 ± 2.15	0.610 ± 0.160	0.1283 ± 0.0168
	Rot-z 15°	13.43 ± 0.77	2.014 ± 0.593	0.1161 ± 0.0260	42.75 ± 2.16	0.608 ± 0.159	0.1282 ± 0.0166
	Rot-z 40°	13.43 ± 0.77	2.012 ± 0.591	0.1161 ± 0.0260	42.75 ± 2.16	0.609 ± 0.159	0.1282 ± 0.0168