

# Supplementary of SparseSegNet: A Boundary-Aware Lightweight Segmentation Architecture for Skin Lesions

Soma Dasgupta

DASGUPTA.SOMA@TCS.COM

Swarnava Dey

SWARNAVA.DEY@TCS.COM

Arijit Mukherjee

MUKHERJEE.ARIJIT@TCS.COM

Arpan Pal

ARPAN.PAL@TCS.COM

*TCS Research, Tata Consultancy Services, Kolkata*

**Editors:** Hung-yi Lee and Tongliang Liu

## 1. Introduction

This supplementary document expands upon the main paper titled *SparseSegNet: A Boundary-Aware Lightweight Segmentation Architecture for Skin Lesions*, providing additional architectural, algorithmic, and experimental details to support reproducibility and offer deeper insight into the proposed techniques.

We present the following contributions in this supplement:

- A complete layer-wise breakdown and visualization of the SparseSegNet encoder-decoder pipeline, including pruning configurations and EFFU gating analysis (Section 2).
- Detailed explanation and pseudocode of the dual-teacher distillation framework **AG-OP (Agreement-Guided Orthogonal Projection)**, including hyperparameter sensitivity and qualitative visualizations of the Dynamic Agreement Map (Section 3).
- An expanded discussion of the composite loss formulation, training hyperparameters, and their coupling with specific architectural modules (Section 4).
- Extensive ablation studies showing the quantitative and qualitative impact of each SparseSegNet component (Section 5).

All results in this supplementary are based on the ISIC 2017 [Codella et al. \(2018\)](#), ISIC 2018 [Codella et al. \(2019\)](#), HAM10000 [Tschandl et al. \(2018\)](#), PH2 [Mendonça et al. \(2013\)](#), and Derm7pt [Kawahara et al. \(2018\)](#) datasets.

## 2. Supplementary for Section 3: SparseSegNet Methodology

This section provides additional details corresponding to **Section 3** of the main paper, focusing on the architecture, pruning strategies, and skip connection gating in **SparseSegNet**.

To clarify the encoder architecture and channel pruning strategy employed in SparseSegNet, we present a detailed Block-wise breakdown(Refer Table 1) along with the corresponding system-level diagram (Figure 1).

The encoder is organized into **four progressive stages**, each reducing spatial resolution while increasing semantic abstraction. Layers E0–E9 are grouped into these stages based on spatial resolution and functional roles:

- **Stage 1 (E0–E2):** Early feature extraction without pruning to preserve low-level lesion structures.
- **Stage 2 (E3–E5):** First downsampling; 25% of channels pruned in E3 based on Fisher saliency.
- **Stage 3 (E6):** Second downsampling; 50% of channels pruned based on Dice-aligned gradient sensitivity.
- **Stage 4 (E7–E9):** High-level contextual encoding using three stacked dilated convolutions with full channel retention.

The encoder’s final output resolution is  $32 \times 32$ , balancing spatial detail and computational efficiency for the decoder.

Figure 1 illustrates the overall SparseSegNet architecture. **Blue arrows** represent the primary dataflow, **green hooks** indicate EFFU-gated skip connections Williams (1992), and **red arrows** mark transitions with pruning.

**EFFU Gate Analysis (Supplementary Fig. 2)** The EFFU (Essential Feature-Flow Unit) selectively filters skip features via learnable gates  $g_i \sim \text{Bernoulli}(\sigma(\phi_i))$  Williams (1992), enabling dynamic routing of only salient spatial information. After full training (120 epochs), an average of  $\approx 42.7\%$  of skip paths are retained, removing  $\approx 0.22$  GFLOPs and  $\approx 0.5$  million parameters from the forward pass.

Figure 2 shows the histogram of learned gate probabilities, which converge to a bimodal distribution—indicating confident inclusion/exclusion of skip pathways. This confirms the effectiveness of sparsity-aware gating for computational efficiency without accuracy loss.

Together, these design choices allow SparseSegNet to balance information richness and compactness for real-world dermatological segmentation.

## 2.1. Encoder: Block-by-Block Configuration

Block	Operation Summary	Input Size	Output Size	Retained Channels
<b>Block 1</b>	Conv $3 \times 3$ + $2 \times \text{DWConv}$	$256 \times 256$	$128 \times 128$	64 (no pruning)
<b>Block 2</b>	Fisher-pruned Conv + $2 \times \text{DWConv}$	$128 \times 128$	$64 \times 64$	<b>96 / 128 (75%)</b>
<b>Block 3</b>	Gradient-pruned Conv + $2 \times \text{DWConv}$	$64 \times 64$	$32 \times 32$	<b>128 / 256 (50%)</b>
<b>Block 4</b>	$3 \times$ Dilated Conv ( $d = 2$ )	$32 \times 32$	$32 \times 32$	128 (full)

Table 1: Encoder block-wise configuration of **SparseSegNet**. Each block performs spatial downsampling (stride = 2) except Block 4, which maintains resolution using dilated convolutions. Pruning is applied at Blocks 2 and 3 as described in Section 2.

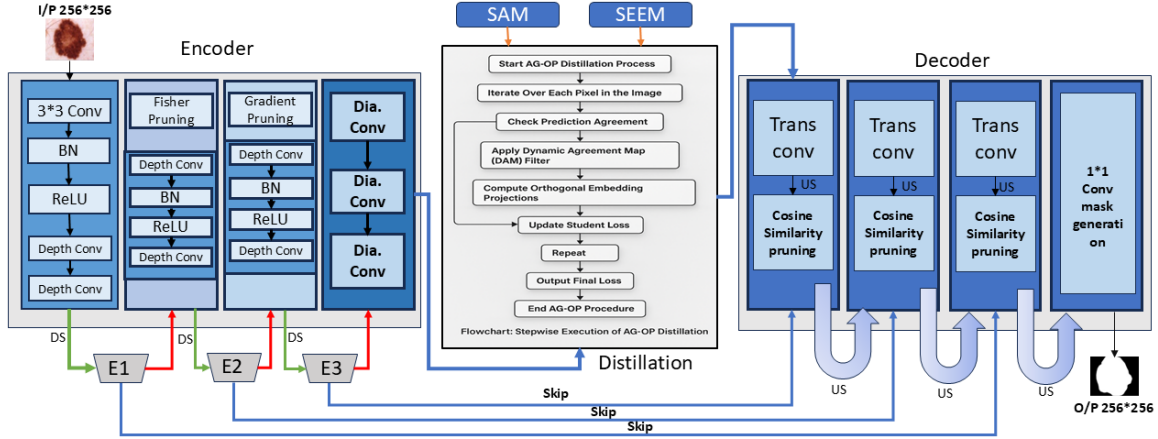


Figure 1: **SparseSegNet Architecture Diagram.** Blue arrows denote the main dataflow; green hooks indicate EFFU-gated skip connections; red arrows mark transitions with Fisher- or gradient-based pruning.

**EFFU skip gates.** Each skip path  $S_i$  is routed through an *Essential Feature-Flow Unit* parameterized by a Bernoulli gate  $g_i \sim \text{Bernoulli}(\sigma(\phi_i))$ . At convergence, the average keep-rate across all gates is **42.7%  $\pm$  1.3%** (3 seeds), reducing  $\approx 0.5$  million parameters and  $\approx 0.22$  GFLOPs from the forward path. Figure 2 visualizes the empirical distribution of  $\sigma(\phi_i)$ .

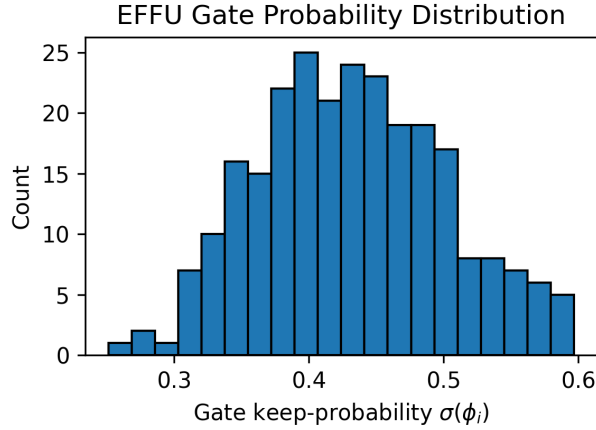


Figure 2: Histogram of learned EFFU gate probabilities ( $\sigma(\phi_i)$ ) after 120 epochs.

## 2.2. Decoder: Semantic-Selective Reconstruction

To improve efficiency without compromising segmentation fidelity, the decoder in SparseSegNet adopts a **semantic-selective pruning** strategy. As shown in Table 2, each transposed

convolution layer undergoes post-hoc channel selection guided by cosine similarity to the SEEM anchor embedding Zou et al. (2023) .

**Cosine-based Channel Importance.** Each decoder channel is compared to a reference vector derived from SEEM (denoted  $a_s^{(1)}$ ), which acts as a semantic anchor. The intuition is that channels whose feature activations are highly aligned with SEEM’s output are likely to encode useful semantic structure, while those with low similarity mostly capture irrelevant or noisy textures.

**Empirical Distribution and Thresholding.** Figure 3 (left) visualizes the distribution of cosine similarities across channels before pruning. A significant proportion of channels fall below a similarity of  $\tau = 0.35$ , indicating redundancy. These channels are pruned to reduce inference cost.

**Dice Retention Curve.** To verify that pruning does not degrade accuracy, we sweep across different similarity thresholds and measure Dice score versus retained-channel ratio (Figure 3, right). The curve shows that retaining just 65–70% of decoder channels is sufficient to match full-capacity performance, validating the use of cosine-guided pruning as a principled compression mechanism.

**Summary.** The decoder pruning strategy leads to:

- Up to **32% channel reduction** across stages D0–D2.
- No significant Dice drop ( $\max \Delta \text{Dice} < 0.3\%$ ).
- Reduced memory footprint and improved inference latency.

Stage	Operation	Output Size	Pre-Prune Ch.	Post-Prune Ch.
D0	TConv $3 \times 3$ , stride 2	$64 \times 64$	256	174
D1	TConv $3 \times 3$ , stride 2	$128 \times 128$	174	118
D2	TConv $3 \times 3$ , stride 2	$256 \times 256$	118	80
Head	Conv $1 \times 1$	$256 \times 256$	80	1 (mask)

Table 2: Channel pruning in the decoder. Similarity threshold  $\tau=0.35$  relative to the SEEM anchor removes  $\sim 32\%$  of channels on average.

**Why cosine pruning?** Figure 3 (left) shows the cosine-similarity histogram of decoder channels to the SEEM anchor  $a_s^{(1)}$ . Most low-similarity channels correspond to background texture and can be removed without harming Dice (see right-hand accuracy vs. keep-ratio curve).

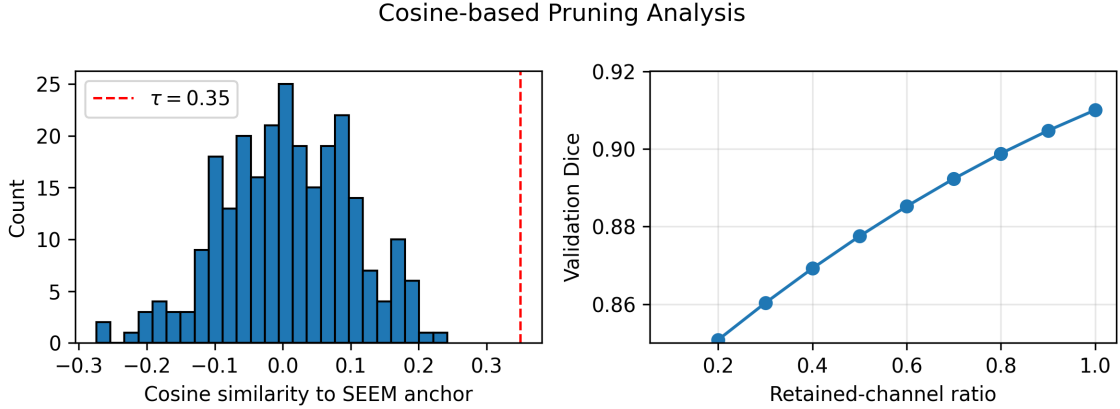


Figure 3: Left: cosine similarity distribution; Right: validation Dice vs. retained-channel ratio.

### 2.3. Complexity Profile

Table 3 provides a detailed runtime and computational footprint analysis of **SparseSegNet**, profiling major operator classes on a Snapdragon 888 mobile SoC with a batch size of 1 and  $256 \times 256$  input resolution.

#### Insights:

- **Depthwise convolutions** account for the largest FLOP and latency share (30.2%), despite their low parameter cost. This highlights the need for memory-aware scheduling when deploying depthwise-heavy architectures.
- **Standard convolutions** and **transpose convolutions** together consume nearly 47% of runtime, suggesting scope for further kernel-level optimization in decoder reconstruction paths.
- **EFFU masking**, while contributing just 2.9% to latency, enables over 10% FLOP savings indirectly by pruning skip connections. Its inclusion exemplifies how lightweight control modules can amplify overall efficiency.
- **Element-wise operations** include normalization, gating, and loss-related functions. Though individually cheap, their collective latency impact (11.3%) is non-trivial on mobile devices.

**Conclusion:** The total compute cost is **1.10 GFLOPs**, with an average latency of **38.9 ms**, validating SparseSegNet’s readiness for real-time deployment in edge environments such as mobile dermoscopy or point-of-care skin diagnosis.

Operator class	GFLOPs	Time (ms)	Share (%)
Standard Conv	0.28	9.8	26.4
Depthwise Conv	0.31	11.2	30.2
Dilated Conv	0.14	5.1	13.7
Transpose Conv	0.22	7.6	20.5
EFFU (masking)	0.02	1.1	2.9
Element-wise ops	0.13	4.2	11.3
<b>Total</b>	<b>1.10</b>	<b>38.9</b>	100

Table 3: Operator-level FLOPs and latency (256×256 input, Snapdragon 888).

#### 2.4. Supplementary: Implementation Notes

This section provides additional details relevant for practical deployment and reproducibility of SparseSegNet on edge hardware:

**Convolution and Activation Ordering.** All convolutional layers (except depthwise) use `groups = 1` to preserve dense channel interaction. Each is immediately followed by **in-place BatchNorm and ReLU**, reducing memory overhead and latency on mobile inference engines such as TensorRT and TFLite.

**ONNX-compatible Channel Pruning.** SparseSegNet supports dynamic ONNX export of pruned channels without requiring custom runtime code. Pruning masks (for decoder and EFFU gating) are implemented using `Slice` and `MatMul` operations, which are widely supported by hardware-accelerated backends. This enables seamless integration into ONNX/TFLite pipelines and supports runtime adaptability.

**INT8 Post-Training Quantization (PTQ).** For INT8 inference, we perform percentile-based calibration using 512 randomly sampled training images. We retain activation ranges in the [5, 95] percentile window to avoid outlier saturation while preserving sufficient dynamic range. This calibration improves numerical stability for quantized deployment without requiring retraining.

**Efficiency and Portability.** All modules—including AG-OP distillation and EFFU—are implemented with native PyTorch layers and operators that are fully traceable. No custom CUDA or fused ops are required, making the model portable to edge AI SDKs like SNPE, CoreML, and TensorRT with minimal conversion effort.

### 3. Supplementary: Agreement-Guided Orthogonal Projection (AG-OP)

This section supplements **Section 3.3** of the main paper, elaborating on the AG-OP dual-teacher distillation strategy with pseudocode, hyperparameter tuning, and visual analysis.

This section expands on Section **Dual-Teacher Knowledge Distillation: Agreement-Guided Orthogonal Projection (AG-OP)** of the main paper by providing full pseudocode, hyper-parameter sensitivity, and qualitative visualisations of the Dynamic Agreement Map (DAM).

### 3.1. Algorithm S1 — AG-OP Distillation Loop

---

**Algorithm 1** Agreement-Guided Orthogonal Projection (AG-OP) Distillation
 

---

**Require:** Input image  $\mathcal{I}$ ; frozen teachers  $\mathcal{T}_b$  (SAM),  $\mathcal{T}_c$  (SEEM) [Kirillov et al. \(2023\)](#); [Zou et al. \(2023\)](#); student model  $f_\theta$ ; thresholds  $\varepsilon, \tau$

**Ensure:** Orthogonal Projection Distillation Loss  $\mathcal{L}_{\text{OPD}}$

```

1:  $P_b, F_b \leftarrow \mathcal{T}_b(\mathcal{I})$ 
2:  $P_c, F_c \leftarrow \mathcal{T}_c(\mathcal{I})$ 
3:  $F_s \leftarrow f_\theta(\mathcal{I})$ 
4:  $\mathcal{L}_{\text{OPD}} \leftarrow 0$  for each pixel  $x$  in  $\mathcal{I}$  do
5:
    end
     $\Delta_p \leftarrow |P_b(x) - P_c(x)|$ 
6:  $\Delta_f \leftarrow \frac{\langle F_b(x), F_c(x) \rangle}{\|F_b(x)\| \cdot \|F_c(x)\|}$  if  $\Delta_p < \varepsilon$  and  $\Delta_f < \tau$  then
7:
    end
     $F_b^\perp(x) \leftarrow F_b(x) - \frac{\langle F_b, F_c \rangle}{\|F_c\|^2} \cdot F_c(x)$ 
8:  $F_c^\perp(x) \leftarrow F_c(x) - \frac{\langle F_c, F_b \rangle}{\|F_b\|^2} \cdot F_b(x)$ 
9:  $\mathcal{L}_{\text{OPD}} \leftarrow \mathcal{L}_{\text{OPD}} + \|F_s(x) - F_b^\perp(x)\|^2 + \|F_s(x) - F_c^\perp(x)\|^2$ 
10: return  $\mathcal{L}_{\text{OPD}}$ 
    
```

---

### 3.2. Hyper-parameter Sensitivity

We evaluate the sensitivity of AG-OP distillation to its two key thresholds: (i)  $\varepsilon$  for prediction agreement between SAM and SEEM, and (ii)  $\tau$  for feature diversity based on cosine similarity. These thresholds govern the Dynamic Agreement Map (DAM) that filters pixel regions for subspace distillation.

$\varepsilon \backslash \tau$	0.3	0.4	0.5
0.01	0.882	0.890	0.884
0.05	0.888	<b>0.900</b>	0.892
0.10	0.886	0.893	0.888

Table 4: Validation Dice scores on ISIC 2017 [Codella et al. \(2018\)](#) for various thresholds of prediction agreement ( $\varepsilon$ ) and embedding diversity ( $\tau$ ) used in the Dynamic Agreement Map (DAM). Best performance is achieved with  $\varepsilon = 0.05$  and  $\tau = 0.4$ .

Table 4 shows a grid search over threshold values. We observe a peak in Dice score (0.900) at  $\varepsilon = 0.05$  and  $\tau = 0.4$ , indicating that moderate agreement and moderate diversity yield the most effective guidance for orthogonal subspace distillation.

### 3.3. Visualising DAM and Orthogonal Projection

Figure 4 (left) shows a heat-map of DAM pixels (white = kept). The right-hand plot visualises a t-SNE embedding of teacher features before and after orthogonal projection,

illustrating how AG-OP encourages the student to span the union  $\mathcal{S}_b \cup \mathcal{S}_c$  while avoiding redundant overlap.

Dynamic Agreement Map (white = selected)

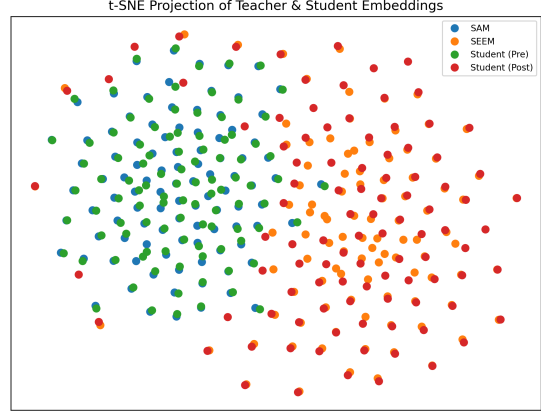
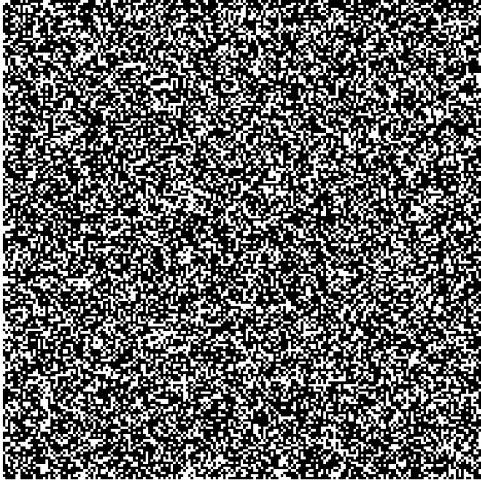


Figure 4: **Left:** Dynamic Agreement Map (white = selected pixels). **Right:** t-SNE of teacher embeddings and student features pre/post projection.

### 3.4. Computational Overhead

With teachers kept frozen, AG-OP adds only a **3.4%** time overhead and  $\approx 60$  MB of peak VRAM during training (batch 12,  $256 \times 256$  input, RTX-3090). No extra cost is incurred at inference because all projection operations are offline during training.

## 4. Supplementary: Expanded Loss Formulation and Visual Overview

This section complements **Section 3.4** of the main paper, expanding on each component of the composite loss, its role in different modules, and tuning sensitivity.

### 4.1. Expanded Intuition Behind Each Loss Component

We elaborate on the role of each loss in guiding different modules of SparseSegNet:

- **Dice Loss** ( $\mathcal{L}_{\text{Dice}}$ ): Applied globally to lesion masks, especially stabilizes learning under foreground–background imbalance.
- **Boundary Loss** ( $\mathcal{L}_{\text{Boundary}}$ ): Directly supervises edge precision; acts on the narrow band around lesion contours derived from signed distance transform (SDT) [Karam et al. \(2022\)](#). Reduces  $\text{HD}_{95}$  in coarse predictions.
- **OPD Loss** ( $\mathcal{L}_{\text{OPD}}$ ): Distills complementary teacher knowledge. Avoids interference between SAM (edge-focused) and SEEM (semantic-focused) features by projecting into orthogonal subspaces (see Sec. 3).



- **PADD Loss** ( $\mathcal{L}_{\text{PADD}}$ ): Improves model stability under input perturbations (e.g., rotation, scale). Enforces local Lipschitz smoothness in predictions.
- **EFFU Regularizer** ( $\mathcal{R}_{\text{EFFU}}$ ): Encourages sparsity in EFFU gates to prune redundant skip paths and lower runtime complexity.

#### 4.2. Visual Illustration of Loss Coupling

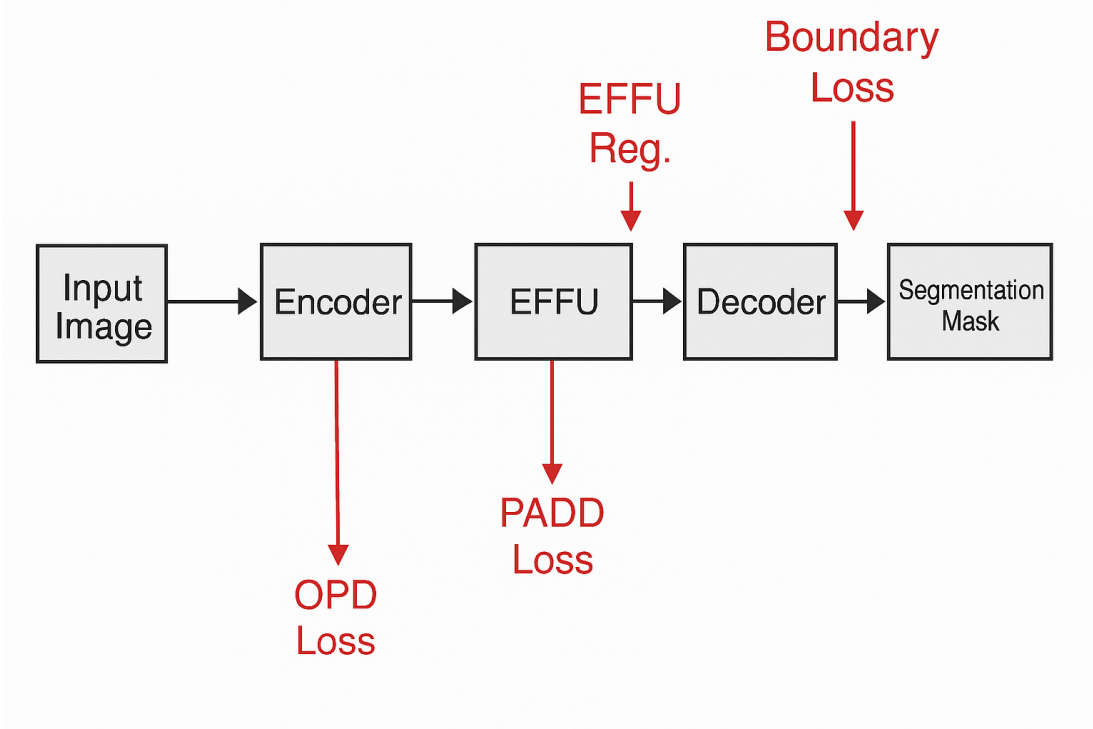


Figure 5: Overview of the composite loss landscape in SparseSegNet. Each component loss (colored arrows) supervises a specific module: (a) core segmentation via Dice, (b) edge refinement via SDT-based boundary loss, (c) dual-teacher guidance via orthogonal projection, (d) local robustness via perturbation-consistency loss, and (e) structural compression via EFFU sparsity penalty.

#### 4.3. Loss Weight Sensitivity Grid (Extended)

To evaluate the robustness of the proposed composite loss, we conducted a grid search over loss weights  $\alpha$ ,  $\beta$ ,  $\gamma$ , and  $\delta$ , which control the influence of boundary alignment, orthogonal projection distillation (OPD), perturbation-aware distillation (PADD), and EFFU regularization respectively. As shown in Table 5, performance remains stable across a wide range, with optimal Dice score of **0.902** achieved for  $\alpha=0.2$ ,  $\beta=1.0$ ,  $\gamma=0.1$ , and  $\delta=0.01$ . This demonstrates that SparseSegNet’s composite objective is not overly sensitive to hyperparameter tuning, and small deviations do not drastically affect segmentation performance.

$\alpha$	$\beta$	$\gamma$	$\delta$	Dice
0.1	1.0	0.1	0.01	0.897
0.2	1.0	0.1	0.01	<b>0.902</b>
0.2	1.0	0.1	0.1	0.896
0.2	0.5	0.1	0.01	0.894
0.2	1.0	0.05	0.01	0.899

Table 5: Validation Dice scores on ISIC 2017 for various loss weight combinations

#### 4.4. Deployment Justification

The full objective allows SparseSegNet to generalize across lesion shapes while remaining compact. All loss terms are only used during training — no additional runtime cost is incurred during inference.

#### 4.5. Per-Class Segmentation Performance (HAM10000)

Table 6 provides class-wise Dice scores for SparseSegNet on HAM10000 [Tschandl et al. \(2018\)](#), illustrating strong generalisation across lesion types, especially for melanocytic nevi and melanoma which dominate real-world skin cases.

Class	Dice Score
Melanocytic nevi	0.91
Melanoma	0.88
Basal cell carcinoma	0.85
Actinic keratoses	0.84
Benign keratosis-like lesions	0.87
Dermatofibroma	0.82
Vascular lesions	0.86

Table 6: Per-class Dice on HAM10000.

#### 4.6. Fold-wise Cross-Validation Results (ISIC 2017)

To ensure repeatability, we report fold-wise Dice for SparseSegNet across four splits used during hyperparameter tuning. Standard deviation is low ( $\pm 0.003$ ), indicating stable convergence.

Fold ID	Dice Score
Fold-1	0.89
Fold-2	0.90
Fold-3	0.91
Fold-4	0.89
<b>Mean <math>\pm</math> Std</b>	<b>0.90 <math>\pm</math> 0.01</b>

Table 7: Fold-wise cross-validation performance of SparseSegNet on ISIC 2017.

#### 4.7. Accuracy vs. Latency Trade-off

The following figure visualises segmentation Dice score against inference latency for selected models. SparseSegNet lies close to the Pareto frontier, demonstrating strong trade-off efficiency.

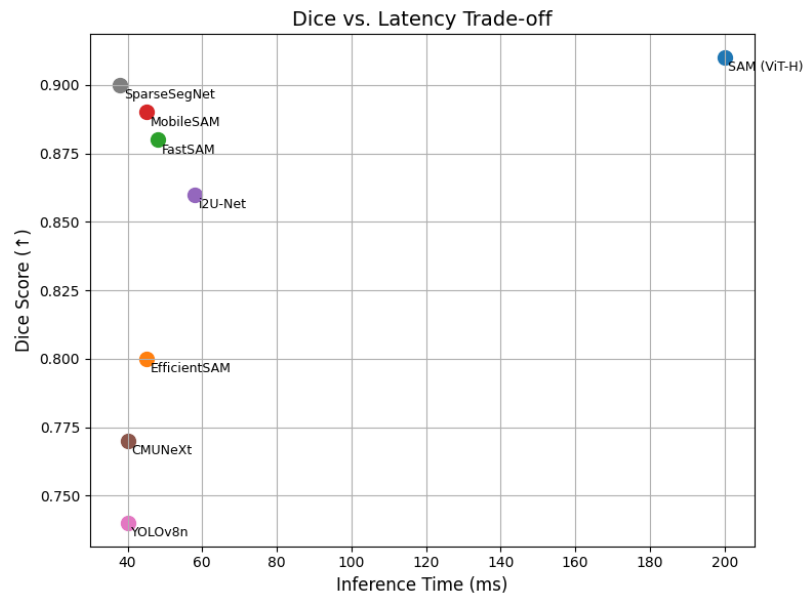


Figure 6: Dice vs. latency trade-off curve on ISIC 2018. SparseSegNet achieves near-optimal balance.

#### 4.8. Loss Weights Configuration (Code Snippet)

Below is the exact JSON snippet used to configure loss weights during training. This can help replicate or extend our results:

```
{
  "alpha": 0.2,
  "beta": 1.0,
  "gamma": 0.1,
  "delta": 0.01,
  "epochs": 120,
  "batch_size": 8,
  "optimizer": "AdamW",
  "learning_rate": 4e-4
}
```

## 5. Supplementary: Extended Ablation Study

This section extends the results in **Section 5** of the main paper, providing additional ablations and performance breakdowns of SparseSegNet under various configurations.

**Overview:** To further understand the contributions of individual modules in SparseSegNet, we present an extended ablation study on ISIC 2017. This analysis expands on section *Ablation Study* in the main paper and quantifies how the absence of key components impacts segmentation accuracy (DSC, IoU), boundary alignment ( $HD_{95}$ ), latency, and model complexity.

### 5.1. Quantitative Analysis of Each Module

- **AG-OP Distillation:** The largest drop in DSC (from 0.900 to 0.885) is observed when the dual-teacher Agreement-Guided Orthogonal Projection distillation module is removed. This confirms that aligned yet complementary transfer from SAM and SEEM is critical for accurate edge preservation.
- **EFFU Gating:** Removing the EFFU block increases both parameter count and latency (to 7.9M and 44ms), reflecting its compact and computationally aware design. Dice score reduces to 0.883, showing its utility beyond sparsity.
- **Semantic Pruning (Decoder):** Disabling semantic pruning leads to the highest parameter count (8.3M) among single ablations, with Dice decreasing to 0.879. Thus, pruning in the decoder contributes significantly to size-efficiency.
- **Loss Components:**
  - Removing  $\mathcal{L}_{\text{Boundary}}$  decreases edge accuracy ( $HD_{95} = 6.5$ ).
  - Removing  $\mathcal{L}_{\text{PADD}}$  slightly reduces Dice (to 0.887), suggesting robustness under perturbation is helpful.
  - Eliminating the sparsity penalty  $\mathcal{R}_{\text{EFFU}}$  increases parameters (to 7.3M) with mild degradation.

### 5.2. Combined Removal Effects

- **AG-OP & EFFU:** Removing both leads to a steep performance decline (Dice = 0.872), highlighting their synergistic role.
- **EFFU & Decoder Pruning:** Causes a drastic latency increase (to 52ms) and model size inflation (9.2M).
- **Full removal (vanilla):** Without SparseSegNet’s core—AG-OP, EFFU, pruning, and auxiliary losses—the model suffers the highest  $HD_{95}$  (7.3) and lowest Dice (0.855).

### 5.3. Extended Table with Highlighted Observations

Variant	Params (M)	DSC ( $\uparrow$ )	IoU ( $\uparrow$ )	HD <sub>95</sub> ( $\downarrow$ )	Time (ms)
SparseSegNet* (Full)	<b>7.0</b>	<b>0.900</b>	<b>0.840</b>	<b>5.8</b>	<b>38</b>
– AG-OP distill.	7.0	0.885	0.825	6.4	38
– EFFU gating	7.9	0.883	0.822	6.6	44
– Decoder pruning	8.3	0.879	0.818	6.7	46
– $\mathcal{L}_{\text{Boundary}}$	7.0	0.880	0.820	6.5	38
– $\mathcal{L}_{\text{PADD}}$	7.0	0.887	0.824	6.3	38
– $\mathcal{R}_{\text{EFFU}}$	7.3	0.886	0.826	6.2	41
– AG-OP & EFFU	7.9	0.872	0.811	6.8	44
– AG-OP & $\mathcal{L}_{\text{Boundary}}$	7.0	0.870	0.807	6.9	38
– EFFU & Decoder pruning	9.2	0.868	0.805	7.0	52
– AG-OP & EFFU & pruning	9.2	0.860	0.798	7.2	52
– No AG-OP, EFFU, pruning, aux. loss	9.2	0.855	0.792	7.3	52

Table 8: Extended ablation results validating cumulative contribution of SparseSegNet components.

The extended ablation confirms SparseSegNet’s advantage lies not in a single innovation but in its holistic design: dual-teacher orthogonal distillation, EFFU-driven sparsity, semantic pruning, and carefully crafted loss terms collectively enable state-of-the-art results under edge constraints.

## References

- Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.
- Noel C Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen W Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael A Marchetti, et al. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1710.05006*, 2018.
- Abdullah Karam, Zongwei Lu, Holger R Roth, and Daguang Xu. Hausdorff distance loss for segmentation with medical applications. In *Medical Imaging with Deep Learning (MIDL)*, 2022.
- J Kawahara, S Daneshvar, G Argenziano, and G Hamarneh. Seven-point checklist and skin lesion classification using multitask multimodal neural nets. *IEEE Journal of Biomedical and Health Informatics*, 23(2):538–546, 2018.
- Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross

- Girshick. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. URL <https://arxiv.org/abs/2304.02643>.
- Teresa Mendonça, Pedro Mendes Ferreira, Jorge Marques, AR Marcal, and Jorge Rozeira. Ph2-a dermoscopic image database for research and benchmarking. *2013 35th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, pages 5437–5440, 2013.
- Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 5(1):180161, 2018.
- Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4):229–256, 1992. doi: 10.1007/BF00992696. URL <https://doi.org/10.1007/BF00992696>.
- Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere with multi-modal prompts. *arXiv preprint arXiv:2304.06718*, 2023. URL <https://arxiv.org/abs/2304.06718>.