ϵ-Seg: Sparsely Supervised Semantic Segmentation of Microscopy Data

Sheida Rahnamai Kordasiabi[1,2], **Damian Dalle Nogare**[1], **Florian Jug**[1]
[1]Human Technopole, Milan, Italy
[2]Technical University of Dresden, Germany

Abstract

Semantic segmentation of electron microscopy (EM) images of biological samples remains a challenge in the life sciences. EM data captures details of biological structures, sometimes with such complexity that even human observers can find it overwhelming. We introduce ϵ-Seg, a method based on hierarchical variational autoencoders (HVAEs), employing center-region masking, sparse label contrastive learning (CL), a Gaussian mixture model (GMM) prior, and clustering-free label prediction. Center-region masking and the inpainting loss encourage the model to learn robust and representative embeddings to distinguish the desired classes, even if training labels are sparse (0.05% of the total image data or less). For optimal performance, we employ CL and a GMM prior to shape the latent space of the HVAE such that encoded input patches tend to cluster w.r.t. the semantic classes we wish to distinguish. Finally, instead of clustering latent embeddings for semantic segmentation, we propose a MLP semantic segmentation head to directly predict class labels from latent embeddings. We show empirical results of ϵ-Seg and baseline methods on 2 dense EM datasets of biological tissues and demonstrate the applicability of our method also on fluorescence microscopy data. Our results show that ϵ-Seg is capable of achieving competitive sparsely-supervised segmentation results on complex biological image data, even if only limited amounts of training labels are available. Code available at https://github.com/juglab/eps-Seg.

1 Introduction

Electron Microscopy (EM) comes in multiple flavors and is without doubt the tool of choice for high-resolution investigations of biological samples [12]. Today, microscopists can capture fine cellular structures at nanometer resolution [22, 3]. Although this opens unprecedented possibilities for studying the very fabric of life, it also means that such microscopes are producing an unfathomable amount of raw image data that then are available to be analyzed [36].

A key module of nearly every analysis pipeline is the segmentation step, where specific structures of interest must be found in the entire body of captured image data. Performing this step manually, is typically not feasible as it takes an impossibly long time [16, 36, 22]. Unfortunately, even semantic segmentation of EM data of biological samples remains a challenge [3, 31].

Ideally, methods for segmenting EM data should (i) lead to sufficiently good segmentation results for the downstream analysis tasks at hand with as few training labels as possible, (ii) generalize well to different imaging conditions and image tissue types and/or be able to fine-tune on moderate amounts of new training data [9], (iii) be able to benefit from sparse labeled data via supervised contrastive learning approaches, and if possible (iv) operate on a hierarchy of spatial scales to distinguish objects not only by either detailed textures or larger scale shapes, but both.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

With this in mind, we introduce ϵ-Seg, a novel and sparsely supervised semantic segmentation framework for EM images that reduces the 'hunger' for labeled data by using a powerful hierarchical VAE (HVAE) [28, 21] with a GMM prior instead of a regular Gaussian one. Furthermore, our method uses center-region inpainting and contrastive learning to enhance feature consistency and segmentation robustness, even when training data is scarce. Hence, ϵ-Seg learns structured latent space representations with effective feature separation for the semantic classes of interest. Once such features are learned, they can be clustered to obtain meaningful semantic segmentations. However, since this process is computationally intensive, we integrate a dedicated semantic segmentation head that directly produces segmentation labels, improving both accuracy and runtime.

2 Related Work

Sparse Supervision. Deep learning has transformed microscopy image segmentation. The U-Net [26] has long been a standard architecture, achieving strong results when trained in a fully supervised setting. However, such approaches rely on dense annotations, which are costly and time-consuming to obtain. At the other extreme, self-supervised methods such as MAESTER [34] learn directly from raw data without labels, offering excellent scalability but typically at the cost of reduced segmentation accuracy compared to fully supervised approaches. Between these extremes lies a growing body of work on sparse or weak supervision, which seeks to achieve label efficiency while maintaining good performance. We aim to surpass self-supervised methods in accuracy while requiring only a fraction of the annotations needed by fully supervised methods. Comprehensive reviews on segmentation methods in large-scale EM with deep learning are available [3], with representative examples including slice-wise pseudo-label propagation for neuronal membranes (4S) [30] , or domain adaptation variants of U-Net designed for limited-annotation settings [4].

Hierarchical Variational Autoencoders. Hierarchical architectures, like HVAEs [28, 21, 32, 7, 24], appear to be an interesting choice for segmenting biological microscopy data. Based on variational autoencoders [20], these powerful models learn a full approximate posterior, but are limited by the typically used Gaussian prior, making us wonder if a Gaussian mixture would not be a more suitable choice for the semantic segmentation task at hand. While the above-mentioned methods pursue label efficiency through different strategies, they do not explicitly enforce semantically disentangled latent representations. In contrast, we explicitly enforce semantically disentangled latent representations by combining a GMM prior with contrastive learning, ensuring that each latent component aligns with a distinct object class. This motivates our focus on HVAEs, which progressively encode features from fine to coarse across network layers. As higher-level semantic structure emerges in deeper layers, the latent space can be disentangled and aligned with semantic classes, enabling efficient segmentation and downstream biological analysis.

Gaussian Mixture Models (GMMs). GMMs have been extensively used to model multimodal distributions and are a key component for many clustering methods [8, 27, 5, 10]. Many approaches integrate GMMs within autoencoder-based architectures, either explicitly as a clustering module [5] or by enforcing multimodal latent structure through a GMM prior [8, 10]. In VAEs, GMM priors enable structured latent spaces where each mixture component represents a distinct cluster or class [10, 8]. Some methods employ direct optimization of GMM objectives alongside autoencoders [5], while others leverage categorical latent variables within GMVAE frameworks, using discrete reparameterization techniques such as the Gumbel-Softmax [19] relaxation to improve scalability [8]. These techniques effectively combine deep generative models with Gaussian mixture priors, enhancing unsupervised representation learning and clustering performance in high-dimensional data spaces.

Contrastive Learning (CL). CL has gained attention for its ability to refine feature representations by maximizing similarities between related samples and minimizing them between unrelated ones. Methods like SimCLR [6] and MoCo [15] demonstrated their effectiveness in many applications. In the context of EM segmentation, CL enables better alignment of latent representations with subcellular structures. We will use CL to ensure that each GMM component corresponds to a distinct semantic class, not just in the highest level of the hierarchy we learn.

Next, we present our proposed method, which integrates hierarchical variational autoencoders with GMM-based priors and contrastive learning to achieve accurate and label-efficient EM segmentation.

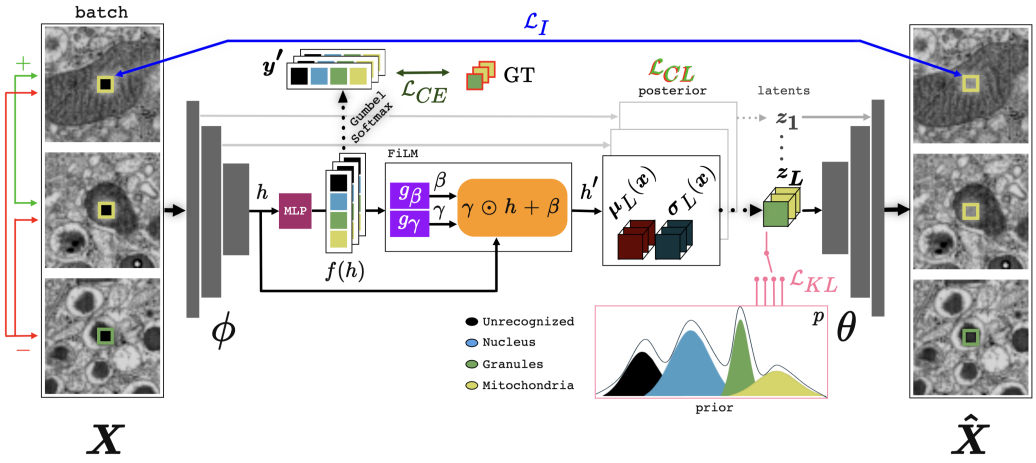

Figure 1: The overall pipeline of ϵ-Seg which is trained on an inpainting task (of center-region masked inputs). ϕ and θ are encoder and decoder of the network, respectively. Dotted arrows show sampling from a distribution (Gumbel-Softmax (Categorical-like distribution) for segmentation head and Normal distribution for conditional posterior). h is an intermediate feature embedding of input x coming from the encoder ϕ. $f(h)$ is a logit vector and $|f(h)| = C$ with C being the number of different classes/GMM prior components (equal to 4 for "BetaSeg" [22]). β and γ are feature-wise linear modulation (FiLM's [23]) parameters (shifting and scaling factors) of features h. h' are the posterior distribution's parameters and are divided into two chunks shown as $\boldsymbol{\mu_L}(\boldsymbol{x})$ and $\boldsymbol{\sigma_L}(\boldsymbol{x})$ by c being the corresponding label of the masked center region of each input patch x in the batch. z_L is a sample from $\mathcal{N}(\boldsymbol{\mu_L}(\boldsymbol{x}), \boldsymbol{\sigma_L^2}(\boldsymbol{x}))$. $\boldsymbol{y'}$ is a differentiable sample from a Gumbel-Softmax [19] distribution. Green arrow shows positive pair of patches having similar labels, and red arrows show negative pairs of patches having dissimilar labels. $\mathcal{L}_{CL}$ is then computed on $\boldsymbol{\mu}s$ (further explanation can be found in section 3). For $\mathcal{L}_I$ inpainting loss, $\mathcal{L}_{CL}$ contrastive loss, $\mathcal{L}_{CE}$ cross-entropy loss and $\mathcal{L}_{KL}$ refer to Equations 1, 16, 14 and 15 respectively.

3 Methods

The method we propose is based on a Hierarchical VAE (HVAE) backbone similar to the ones described in [28, 24]. We modify the standard HVAE setup by (i) using a Gaussian mixture model (GMM) instead of the default Gaussian, so every semantic class we want to distinguish has its own predetermined Gaussian region, and by (ii) adding a contrastive loss (CL), we further ensure that latent encodings are grouped by their semantic similarity through all hierarchy levels.

As the basis for our work, we used the openly available HVAE backbone of Hierarchical DivNoising (HDN) [24]. HVAEs, as introduced elsewhere [28, 32, 21, 24], consist of a bottom-up path (encoder) and a top-down path (decoder) with trainable parameters ϕ and θ, respectively. The encoder extracts features from a given input $\boldsymbol{x}$ at progressively coarser scales, creating a hierarchical latent encoding $\boldsymbol{z}$ that splits into sub-spaces $\boldsymbol{z}_i, i = 1 \dots L$, with L being the number of hierarchy levels, or latent layers, in the HVAE. The decoder network in regular HVAEs reconstructs $\boldsymbol{x}$, starting from the topmost latent variables $\boldsymbol{z}_L$. Here, we first switch from reconstructing $\boldsymbol{x}$ to inpainting a masked central region in $\boldsymbol{x}$, as described next.

Autoencoding vs. Inpainting. In contrast to regular VAEs and HVAEs that use a reconstruction loss on full input patches $\boldsymbol{x}$, we are using masked autoencoding instead [18]. Since our aim is to learn semantic features that can be used for pixel-level semantic segmentation, the zero-masking we employed asks the network to only reconstruct the masked region, effectively learning features that best represent the masked semantic class. We conducted experiments with masked regions of various sizes and have always ensured that all masked pixels were from the same semantic class, see Table 8.

The model is trained to reconstruct the masked center pixel(s) using an MSE-based inpainting loss on $\boldsymbol{X}$, a training batch of inputs, of size B, as

$$\mathcal{L}_{\text{I}} = \frac{1}{B} \sum_{\boldsymbol{x} \in \boldsymbol{X}} \left(\boldsymbol{x}^{\text{mask}} - \hat{\boldsymbol{x}}^{\text{mask}} \right)^2, \tag{1}$$

where $\hat{\boldsymbol{x}}^{\text{mask}}$ is the inpainted masked region the decoder predicted, and $\boldsymbol{x}^{\text{mask}}$ is the mask region of the respective input patch prior to zero-masking.

HVAEs with Gaussian Priors. The Gaussian prior of regular VAEs only applies to the topmost hierarchy level in HVAEs, where it remains $\mathcal{N}(0, I)$ as depicted in Figure S1.

The latent variables $\boldsymbol{z}$ of a HVAE are split into L layers $\boldsymbol{z}_i, i \in [1, \ldots, L]$ so that

$$p_\theta(\boldsymbol{z}) = p_\theta(\boldsymbol{z}_L) \prod_{i=1}^{L-1} p_\theta(\boldsymbol{z}_i | \boldsymbol{z}_{i+1}), \tag{2}$$

$$p_\theta(\boldsymbol{z}_L) = \mathcal{N}(\boldsymbol{z}_L | \boldsymbol{0}, \boldsymbol{I}), \tag{3}$$

$$p_\theta(\boldsymbol{z}_i | \boldsymbol{z}_{i+1}) = \mathcal{N}(\boldsymbol{z}_i | \mu_{p,i}(\boldsymbol{z}_{i+1}), \sigma^2_{p,i}(\boldsymbol{z}_{i+1})) \text{ and} \tag{4}$$

$$p_\theta(\boldsymbol{x} | \boldsymbol{z}_1) = \mathcal{N}(\boldsymbol{x} | \mu_{p,0}(\boldsymbol{z}_1), \sigma^2_{p,0}(\boldsymbol{z}_1)), \tag{5}$$

where $\mu_\theta(\boldsymbol{z}_i)$ and $\sigma^2_\theta(\boldsymbol{z}_i)$ represent the mean and the variance of the latent encoding, parameterized by θ.

For each layer i, the approximate posterior $q_\phi(\boldsymbol{z}_i | \boldsymbol{x}, \boldsymbol{z}_{<i})$, computed by the encoder, is defined as

$$q_\phi(\boldsymbol{z}_i | \boldsymbol{x}, \boldsymbol{z}_{<i}) = \mathcal{N}(\boldsymbol{z}_i; \mu_\phi(\boldsymbol{x}, \boldsymbol{z}_{<i}), \sigma^2_\phi(\boldsymbol{x}, \boldsymbol{z}_{<i})), \tag{6}$$

where $\mu_\phi(\boldsymbol{x}, \boldsymbol{z}_{<i})$ and $\sigma_\phi(\boldsymbol{x}, \boldsymbol{z}_{<i})$ are functions parameterized by ϕ, and are the mean and variance conditioned on the input $\boldsymbol{x}$ and the latent variables from lower layers $j < i$, denoted by $\boldsymbol{z}_{<i}$.

The KL divergence term for each layer in the Evidence Lower Bound (ELBO) is

$$\mathbb{E}_{q_\phi(\boldsymbol{z}_{>i} | \boldsymbol{x})} \left[\text{KL} \left(q_\phi(\boldsymbol{z}_i | \boldsymbol{x}, \boldsymbol{z}_{<i}) \,\|\, p_\theta(\boldsymbol{z}_i | \boldsymbol{z}_{i+1}) \right) \right], \tag{7}$$

where $\boldsymbol{z}_{>i}$ are all $\boldsymbol{z}_j$ for $j > i$.

HVAEs with a GMM Prior. When replacing the topmost prior $p_\theta(\boldsymbol{z}_L)$ in an HVAE with a Gaussian mixture model (GMM), the prior becomes a weighted sum of Gaussians

$$p_\theta(\boldsymbol{z}_L) = \sum_{c=1}^{C} \pi_c \mathcal{N}(\boldsymbol{z}_L; \mu_c, \sigma^2_c), \tag{8}$$

where C is the total number of Gaussian components and also the number of semantic classes we want to distinguish, π_c are the mixing coefficients of the GMM with $\sum_{c=1}^{C} \pi_c = 1$, and $\mathcal{N}(\boldsymbol{z}_L; \mu_c, \sigma^2_c)$ is a Gaussian component with mean μ_c and standard deviation σ_c.

Note that there is a one-to-one correspondence between Gaussian components of the GMM and the semantic classes ϵ-Seg is supposed to distinguish. This would ensure that the latent variable follows a categorical distribution over the semantic classes; we ideally want the mixture assignment $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_C)$ to act as a one-hot vector, *i.e.* one π_c should be 1, and the rest should be 0.

However, in practice, learning a fully discrete $\boldsymbol{\pi}$ is challenging because the standard VAE framework with a GMM prior typically results in soft assignments [10]. To encourage hard assignments, one could (i) use a Gumbel-Softmax [19] trick to approximate categorical sampling while maintaining differentiability [8], (ii) introduce an entropy loss to encourage π_c values to be closer to either 0 or 1. In our experiments, we used the Gumbel-Softmax during training, while reverting to the standard softmax at inference time. We also introduced an entropy loss term as a form of self-supervision, which yielded moderate improvements in the Gumbel-Softmax-based results (see Supplementary Material), but did not lead to significant gains w.r.t. the best-performing softmax configuration. We therefore report the softmax-based results as our main findings, without the additional training phase using the entropy loss. In future work, we plan to investigate alternative self-supervision strategies to

further enhance the segmentation performance, leveraging the vast amount of available unlabeled data, within the proposed framework.

The approximate posterior for the topmost latent $\boldsymbol{z}_L$, can now be expressed as

$$q_\phi(\boldsymbol{z}_L|\boldsymbol{x}) = \sum_{l=1}^{C} q_\phi(c = l|\boldsymbol{x})\, q_\phi(\boldsymbol{z}_L|\boldsymbol{x}, c = l), \tag{9}$$

where $q_\phi(c|\boldsymbol{x})$ is the approximate posterior probability of the GMM component c set to label l given input $\boldsymbol{x}$ and $q_\phi(\boldsymbol{z}_L|\boldsymbol{x}, c)$ is the topmost approximate posterior conditioned on $\boldsymbol{x}$ and component c. We model $q_\phi(\boldsymbol{z}_L|\boldsymbol{x}, c)$ over all possible labels itself with a Gaussian

$$q_\phi(\boldsymbol{z}_L \mid \boldsymbol{x}, c) = \mathcal{N}(\boldsymbol{z}_L; \mu_{\boldsymbol{L}}(\boldsymbol{x}), \sigma_{\boldsymbol{L}}(\boldsymbol{x})), \tag{10}$$

by predicting $\mu_{\boldsymbol{L}}(\boldsymbol{x})$ and $\sigma_{\boldsymbol{L}}(\boldsymbol{x})$ (see boxes labeled with "posterior" in Figure 1). In practice, the parameters $\mu_{\boldsymbol{L}}(\boldsymbol{x})$ and $\sigma_{\boldsymbol{L}}(\boldsymbol{x})$ are computed once from the FiLM-conditioned encoder output and are shared across all components l. As a result, the mixture in Equation 9 reduces to

$$q_\phi(\boldsymbol{z}_L|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}_L; \mu_{\boldsymbol{L}}(\boldsymbol{x}), \sigma_{\boldsymbol{L}}(\boldsymbol{x})), \tag{11}$$

as depicted in Figure 1. In order to predict $\mu_{\boldsymbol{L}}(\boldsymbol{x})$ and $\sigma_{\boldsymbol{L}}(\boldsymbol{x})$), we must compute the conditional posterior.

Computing the Conditional Posterior. In this section, we describe the main backbone of our method leading from a given input patch $\boldsymbol{x} \in \boldsymbol{X}$ to the computed posteriors $q_\phi = \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{x}), \boldsymbol{\sigma}^2(\boldsymbol{x}))$. Figure 1 illustrates the overall pipeline of ϵ-Seg.

The encoder, parametrized by ϕ, processes $\boldsymbol{x}$, leading to intermediate features h in the topmost hierarchy level L. These features are then passed through an MLP classifier (rouge box in Figure 1), producing a vector of logits $f(h)$ with dimensionality C, coinciding with the number of classes ϵ-Seg is tasked to distinguish.

Instead of directly using h as our posterior distribution parameters, as done in our Vanilla HVAE baseline, we are using $f(h)$, fed through two additional MLPs, g_γ and g_β (see violet boxes in Figure 1), to compute parameters, γ and β such that $\gamma = g_\gamma(f(h))$ and $\beta = g_\beta(f(h))$.

Those MLPs are mapping logits $f(h)$ into feature-wise scaling and shifting factors. In this way, the encoded features h are modulated via these FiLM [23] parameters γ and β into h' via computing $h' = \gamma \odot h + \beta$, where $\odot$ denotes the Hadamard product (element-wise multiplication). The modulated feature representation h' is then chunked into two parts, $\boldsymbol{\mu}_L(\boldsymbol{x})$ and $\boldsymbol{\sigma}_L(\boldsymbol{x})$, and used to parameterize the conditional Gaussian posterior in Equation 11.

The Latent Sematic Segmentation Head. To avoid computationally costly downstream latent space clustering to perform the semantic segmentation task (as done in Xie et al. [34] and Han et al. [14] using K-Means clustering), we are introducing a segmentation head tasked to perform the semantic pixel classification tasks directly from the computed logits $f(h)$.

To compute $q_\phi(c|\boldsymbol{x})$ of Equation 9, we use a categorical reparameterization trick via Gumbel-Softmax [19].

The standard Gumbel-Softmax formula using the class probabilities π_i is

$$y'_i = \frac{\exp((\log \pi_i + g_i)/\tau)}{\sum_{j=1}^{C} \exp((\log \pi_j + g_j)/\tau)}, \tag{12}$$

where $g_i \sim \text{Gumbel}(0, 1)$ are Gumbel noise samples. Instead of probabilities π_i, we work with logits $f(h)$ (raw scores before softmax). The equivalent formula becomes

$$y'_i = \frac{\exp((f_i(h) + g_i)/\tau)}{\sum_{j=1}^{C} \exp((f_j(h) + g_j)/\tau)}. \tag{13}$$

The temperature parameter τ in the Gumbel-Softmax distribution plays a crucial role in controlling the degree of discreteness in the sampled values. During training, τ is often annealed from a higher value to a lower one, gradually transitioning from a smooth approximation to a discrete categorical distribution.

In ϵ-Seg, we use a typical annealing schedule $\tau = \max(\tau_{\min}, \exp(-rt))$, where $r = 0.999$ is the decay rate, $\tau_{\min} = 0.5$, and t is the training step. Therefore, Gumbel enables the differentiable sampling of categorical variables, improving gradient estimation, and semi-supervised classification [19].

Next, we draw a vector $\boldsymbol{y'}$, representing the class assignment (segmentation prediction) for an input patch $\boldsymbol{x}^{(i)}$ in the batch $\boldsymbol{X}$, by sampling from the Gumbel-Softmax distribution parameterized by logits $f(h)$ with temperature τ.

For input patches $\boldsymbol{x}^{(i)} \in \boldsymbol{X}$ for which we know the class label l_i, we want to ensure that $y_l'^{(i)} \in \boldsymbol{y'}^{(i)}$ is the largest entry. We do so using the cross-entropy loss

$$\mathcal{L}_{CE} = -\sum_{\boldsymbol{x}^{(i)} \in \boldsymbol{X}} log\, y_l'^{(i)}. \tag{14}$$

Computing the Kullback Leibler Divergence. As it is commonly done in VAEs [20], the KL-divergence term is regularizing the parameters of our encoder, ϕ, such that the approximate posterior will be close to our prior $p_\theta(\boldsymbol{z})$. In HVAEs, KL is computed at each hierarchy level. Changing from a standard Gaussian prior at the highest hierarchy level L to using a GMM prior, as described earlier in this section, requires us to define a strategy to compute the KL-divergence appropriately.

Hershey and Olsen [17] address the challenge of efficiently approximating the KL divergence between two GMMs, and Durrieu et al. [13] propose lower and upper bounds to estimate this divergence. While these approaches can be needed in practical setups [10, 8], we only need to compute the KL divergence between the posterior $q_\phi(\boldsymbol{z}_L|\boldsymbol{x})$ (Equation 11) and the l-th GMM component, where l is either the known class label for an input patch $\boldsymbol{x}^{(i)}$, or $l = \underset{y'^{(j)} \in \boldsymbol{y'}^{(j)}}{\arg\max}\ y'^{(j)}$ for a patch $\boldsymbol{x}^{(j)}$ for which we do not have a ground truth class label.

Hence, Equation 8 becomes $p_{\theta,c}(\boldsymbol{z}_L) = \mathcal{N}(\boldsymbol{z}_L; \mu_l, \sigma_l^2)$, and $\mathcal{L}_{KL}$ is therefore still computed as the divergence between two normal distributions. The KL loss over all hierarchy levels is therefore

$$\mathcal{L}_{KL} = -(\mathrm{KL}(q_\phi(\boldsymbol{z}_1|\boldsymbol{x}) \parallel p_\theta(\boldsymbol{z}_1|\boldsymbol{z}_2)) + \textstyle\sum_{i=2}^{L-1} \mathrm{KL}(q_\phi(\boldsymbol{z}_i|\boldsymbol{z}_{i-1}) \parallel p_\theta(\boldsymbol{z}_i|\boldsymbol{z}_{i+1})) + \mathrm{KL}(q_\phi(\boldsymbol{z}_L|\boldsymbol{z}_{L-1}, c) \parallel p_{\theta,c}(\boldsymbol{z}_L))). \tag{15}$$

Contrastive Loss. The contrastive loss consists of two terms, positive pair loss $\mathcal{L}_+$, which encourages proximity between samples belonging to the same class, and negative pair loss $\mathcal{L}_-$, that penalizes proximity between samples of different classes, ensuring inter-class separation. We define boolean matrices P and N for positive pairs and negative pairs, respectively, as $P_{ij} = \begin{cases} 1 & \text{if } l_i = l_j \text{ and } i \neq j, \\ 0 & \text{otherwise} \end{cases}$ and $N_{ij} = \begin{cases} 1 & \text{if } l_i \neq l_j, \\ 0 & \text{otherwise,} \end{cases}$ with l_i and l_j being the labels of patches i and j, respectively. These loss terms then become $\mathcal{L}_+ = \frac{1}{\sum_{i,j} P_{ij}} \sum_{i,j} P_{ij} \cdot \mathcal{D}(\boldsymbol{\mu}^{(i)}, \boldsymbol{\mu}^{(j)})$ and $\mathcal{L}_- = \sum_{i,j} Nij \cdot \ell_-(\mathcal{D}(\boldsymbol{\mu}^{(i)}, \boldsymbol{\mu}^{(j)}))$, with $\boldsymbol{\mu}^{(i)}$ being the predicted means of the posterior distribution over all hierarchy levels for a patch i in batch $\boldsymbol{X}$, and $\mathcal{D}(\boldsymbol{\mu}^{(i)}, \boldsymbol{\mu}^{(j)})$ a distance function. In our experiments, we used the Euclidean distance. Note that for $\mathcal{L}_-$ we define the penalty function $\ell_-(d) = \begin{cases} 0 & \text{if } d \geq m, \\ (m-d)^2 & \text{otherwise} \end{cases}$, with m being the so-called *margin*, a hyperparameter that must be set appropriately, *e.g.* using grid-search.

The full contrastive loss term is finally defined as

$$\mathcal{L}_{CL} = \lambda \mathcal{L}_+ + (1 - \lambda)\mathcal{L}_-, \tag{16}$$

with λ being a hyperparameter that balances the positive and negative pair loss with each other.

Readers might wonder why a contrastive loss is useful when a GMM prior is used, where for each structure to be classified (*i.e.* for each label) we have defined a Gaussian component in its own right. The main reason is that the GMM prior only takes effect at the uppermost hierarchy level L. At all levels $i < L$, $\mathcal{L}_{CL}$ is taking care of the desired label-wise segregation of latent encodings.

The Overall Loss of ϵ-Seg. Taken all together, the overall loss of ϵ-Seg is

$$\mathcal{L} = \mathcal{L}_I + \alpha_1 \mathcal{L}_{CE} + \alpha_2 \mathcal{L}_{KL} + \alpha_3 \mathcal{L}_{CL}, \tag{17}$$

Learning Paradigm	Model	U	N	G	M	Avg DSC
Self-Supervised	Vanilla HVAE* [24]	0.44	0.55	0.34	0.13	0.37
	Han et al.* [14]	–	–	–	–	0.66
	MAESTER* [34]	0.84	0.95	0.56	0.79	0.79
Sparsely Supervised	Labkit [2]	0.85	0.44	0.68	0.61	0.65
	U-Net	0.90	0.96	0.78	0.66	0.83
	ϵ-Seg (ours)	0.91	0.96	0.82	0.86	0.89
Fully Supervised	Vanilla ViT [11]	0.91	0.98	0.77	0.87	0.88
	Segmenter [29]	0.91	0.99	0.86	0.90	0.92
	U-Net [26]	0.94	0.99	0.90	0.87	0.93

Table 1: Dice similarity coefficient per class and average across all classes on the "BetaSeg" dataset [22]. Methods marked with an asterisk use K-Means clustering on latent features to conduct semantic segmentation (see Section 3). U: Unrecognized, N: Nucleus, G: Granules, M: Mitochondria.

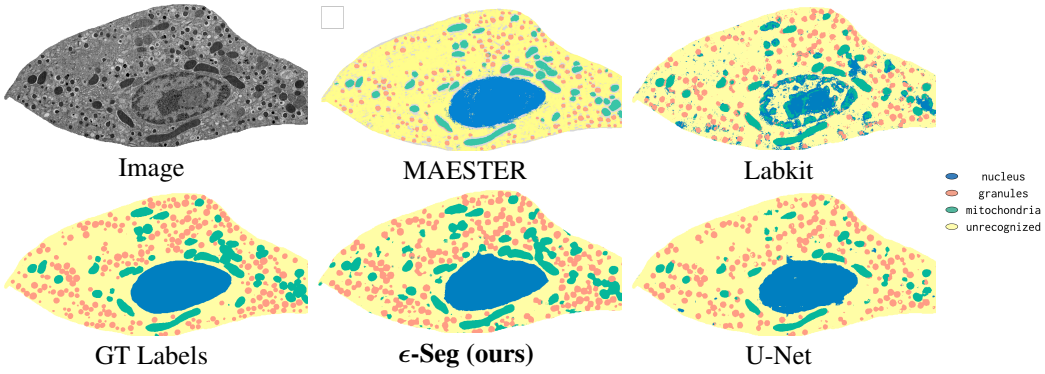

Figure 2: Qualitative segmentation result on part of the test image stack (here we show section 627 of $high\_c4$ of the "BetaSeg" dataset [22]).

where α_i's are hyperparameters to adjust the contribution of each loss with each other. We tuned those hyperparameters using grid search and by manual tuning.

Next, we show empirical results we obtained using ϵ-Seg and comparisons to several baseline methods on two dense EM datasets and one fluorescence microscopy dataset.

4 Experiments and Results

Datasets. One of the datasets used in this study is the "BetaSeg" [22] dataset from OpenOrganelle [16], a public repository of high-resolution cellular imaging data. Acquired via Focused Ion Beam Scanning Electron Microscopy (FIB-SEM), the dataset focuses on primary mouse pancreatic islet β cells from a high-glucose-dosage group, chosen for comparison with prior works. It underwent preprocessing, including rescaling each stack to form 4×4×4 nm isotropic voxels, which can be viewed in any arbitrary orientations, and generating reference segmentations through human annotation or manually corrected deep learning models. The final dataset consists of four cell volumes with binary segmentation masks for seven subcellular structures, centrioles, nucleus, plasma membrane, microtubules, golgi body, granules, and mitochondria, along with an eighth "unrecognized" category. Notably, the nucleus, granules, mitochondria, and unrecognized regions dominate the dataset. For evaluation, cells 1, 2, and 3 were used for training, while cell 4 served as an independent test set.

Next, We used "liver FIBSEM" dataset that samples were fresh needle biopsies fixed with 4%PFA and 2%GA in phosphate buffer. High contrast staining was performed with reduced osmium and Waltons lead aspartate stain [33] and embedded in Epon. Sample preparation and imaging was done on a ZEISS GeminiSEM according to prior reports [35]. The final dataset consists of one cell volume with 11 crops that have been extracted from a cell volume, annotated manually and used for training, validation and testing. The segmentation masks consist of six subcellular structures, mitochondria,

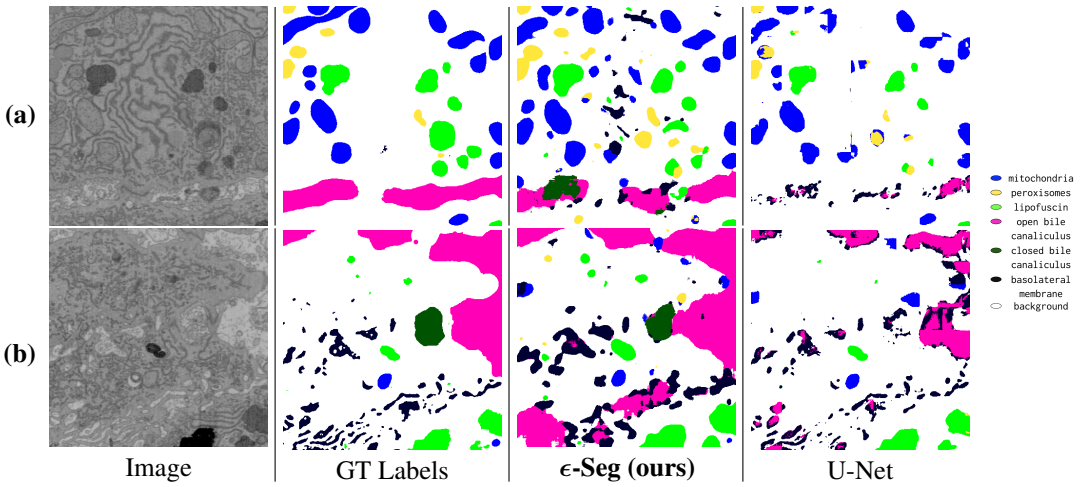

Figure 3: Qualitative segmentation result on two crops of the whole 3D volume. (a) and (b) are section 80 and 26 of crop00 and crop10 in "liver FIBSEM" dataset respectively. The U-Net is sparsely-supervised (for the fully-supervised U-Net result, see Figure S4).

Model	B	M	P	L	BM	OBC	CBC	Avg DSC
U-net [26]-Fully Supervised	0.97	0.95	0.85	0.79	0.52	0.87	0.90	0.84
U-net-Sparsely Supervised	0.94	0.81	0.68	0.81	0.49	0.39	0.00	0.59
ϵ-Seg-Sparsely Supervised	0.91	0.82	0.63	0.81	0.39	0.70	0.46	0.67

Table 4: Dice similarity coefficient per class and average across all classes comparing our model with baselines for "liver FIBSEM" dataset. B: Background, M: Mitochondria, P: Peroxisomes, L: Lipofuscin, BM: Basolateral Membrane, OBC: Open Bile Canaliculus, CBC: Closed Bile Canaliculus.

peroxisomes, lipofuscin, basolateral membrane, open bile canaliculus and closed bile canaliculus, along with a seventh "background" category.

Per-Class Dice Coefficient			Avg
Background	Cytoplasm	Nuclei	DSC
0.94	0.86	0.90	0.90

Table 2: Dice similarity coefficient per class and average for "Aitslab-bioimaging" datasets.

Trained on	Per-Class Dice Coefficient				Avg
	U	N	G	M	DSC
high_c1	**0.85**	0.38	**0.68**	**0.61**	0.63
high_c2	0.80	0.33	0.58	0.56	0.57
high_c3	0.82	**0.44**	0.63	0.42	0.58

Table 3: Labkit results. Due to different image sizes, Labkit was trained on individual volumes. U: Unrecognized, N: Nucleus, G: Granules, M: Mitochondria.

RLF	Per-Class Dice Coefficient				Avg
	U	N	G	M	DSC
20	0.89	0.98	0.81	0.83	0.88
15	0.88	0.98	0.81	0.78	0.86
10	0.86	0.98	0.80	0.75	0.85
5	0.85	0.96	0.77	0.76	0.84
1	0.79	0.95	0.69	0.69	0.78

Table 5: DSC per class and average across all classes. The "RLF" column (Relative Labeling Factor) specify a scaling factor where 20 corresponds to 0.05% and 1 as small as 0.0025% of the total labeles available. U: Unrecognized, N:Nucleus, G:Granules, M:Mitochondria.

Entropy Loss	Per-Class Dice Coefficient				Avg
	U	N	G	M	DSC
✗	0.81	0.97	0.74	0.71	0.81
✓	0.86	0.98	0.80	0.75	**0.85**

Table 6: Effect of entropy loss: The best checkpoint of a sparsely supervised model was further trained using batches with 50% unlabeled data. U: Unrecognized, N:Nucleus, G:Granules, M:Mitochondria.

While it is true that FIB-SEM datasets like "BetaSeg" [22] offer isotropic resolution suitable for 3D processing, this is not always the case in EM imaging, where data often comes in 2D slices (especially in higher-throughput screens).

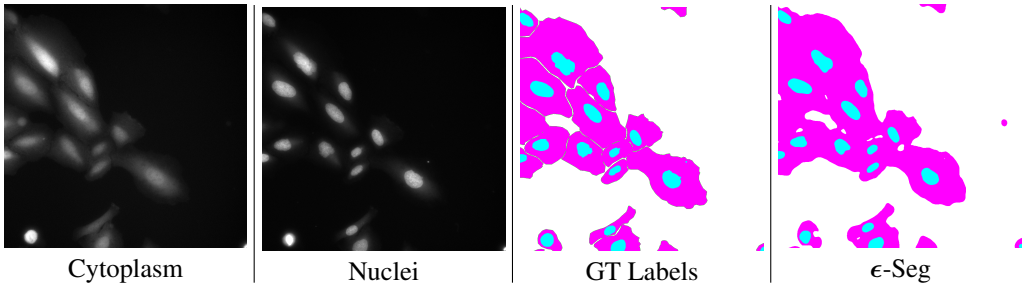

Figure 4: Qualitative results on a representative 2-channel image from the overlapping subset of the "Aitslab-bioimaging1" and "Aitslab-bioimaging2" datasets. The first two panels show the fluorescence microscopy channels: EGFP-Galectin-3-labeled cytoplasm (left) and Hoechst 33342-stained nuclei (center-left). The center-right panel (GT) displays the ground truth semantic segmentation with nuclei (cyan) and cytoplasm (magenta). The rightmost panel (ϵ-Seg) shows the prediction from our method.

Furthermore, we conducted an experiment on overlapping subset of two datasets Aitslab-bioimaging1 [1] and Aitslab-bioimaging2 [25]. The Aitslab-bioimaging1 dataset is a benchmarking fluorescence microscopy dataset containing 50 images of Hoechst 33342-stained U2OS osteosarcoma cell nuclei, including annotations for nuclei, nuclear fragments, and micronuclei, designed for training and evaluating neural networks for instance and semantic segmentation and the Aitslab-bioimaging2 dataset is a fluorescence microscopy dataset containing 60 images of EGFP-Galectin-3 labeled U2OS osteosarcoma cells with hand-annotated cell outlines, designed for training and benchmarking neural networks for instance and semantic segmentation, with over 2200 annotated cell objects and compatibility with object detection tasks. The overlapping subset of them contains 30, 2-channel images for training and 10 for testing.

Evaluation Metrics. We used Dice Similarity Coefficient (DSC) to evaluate the segmentation performance. DSC is a widely used metric in image segmentation and measures the similarity between the predicted and actual segmentation masks.

Let A and B be two sets representing the binary segmentation masks of the ground truth and the predicted segmentation. The Dice coefficient is defined as $Dice(A, B) = \frac{2|A \cap B|}{|A|+|B|}$, where $|A \cap B|$, the number of overlapping pixels between the predicted and ground truth masks, $|A|$, the number of pixels in the ground truth mask, and $|B|$, the number of pixels in the predicted mask.

Experiments. We use an architecture similar to the one used in the HDN work [24]. For all hyperparameters we have introduced, we used grid searches to find a good balance between performance and stability. We first evaluate our method on "BetaSeg" dataset [22] and compare its performance against baseline methods shown in Table S1. They demonstrate that our approach outperforms existing baselines in terms of DSC (F1-score). For the Labkit baseline we trained per cell and show the results in Table 3 and report the best class-wise performance in Table S1. Quantitative segmentation result are shown in Figure 2 (complete Figure S3).

To further validate the robustness of our method, we conduct experiments on the "liver FIBSEM" dataset, comparing it with U-Net baselines (fully and sparsely-supervised). Quantitative and qualitative results are shown in Table 4 and Figure 3, respectively (complete Figure S4). Additionally, we show ϵ-Seg results also on a fluorescent microscopy dataset (see Table 2 and Figure 4).

Model Ablations. We strip our model down to a vanilla HVAE and then re-introduce one component at a time, showing how each of the modules we have introduced above contributes to the overall performance we report. These results on the "BetaSeg" dataset are shown in Table 7.

Additionally, we evaluate how the quality of the results depends on the amount of available training labels. To this end, we are starting from 0.05% of the total image data available in the "BetaSeg" dataset and gradually decreasing the used training labels down to 0.0025%. The results of these experiments can be found in Table 5. As discussed in Section 3, $\mathcal{L}_H$ helps us to gain additional performance also from the unlabeled data, which we measure and report in Table 6. Finally, we measured the effect of differently sized masking regions in Table 8.

Loss			Prior	Per-Class Dice Coefficient				Avg
KL	CL	CE	Distribution	U	N	G	M	DSC
✓	✗	✗	$\mathcal{N}$	0.44	0.55	0.34	0.13	0.37
✓	✓	✗	$\mathcal{N}$	0.83	0.95	0.69	0.76	0.81
✓	✗	✓	$\mathcal{N}$	0.81	0.97	0.80	0.75	0.83
✓	✓	✓	$\mathcal{N}$	0.81	0.97	0.73	0.72	0.81
✓	✗	✓	GMM	0.82	0.97	0.72	0.75	0.82
✓	✓	✓	GMM	0.86	0.98	0.80	0.75	**0.85**

Table 7: Loss components and prior distribution ablation on "BetaSeg" dataset. U: Unrecognized, N: Nucleus, G: Granules, M: Mitochondria.

Mask	Per-Class Dice Coefficient				Avg
Size	Unrecognized	Nucleus	Granules	Mitochondria	DSC
9x9	0.83	0.95	0.65	0.73	0.79
7x7	0.84	0.97	0.72	0.75	0.82
5x5	0.87	0.94	0.78	0.80	0.85
3x3	0.88	0.97	0.81	0.80	**0.87**
1x1	0.86	0.98	0.80	0.75	0.85

Table 8: Label consistency ablation on "BetaSeg" dataset. The "Mask Size" column indicates the size of the center-region mask, within which the pixel-wise ground truth labels are consistent.

Limitations. While ϵ-Seg achieves competitive segmentation results using only sparse supervision, several limitations do remain. First, all experiments we present are conducted on 2D images. Extending the presented framework to operate in full 3D is an important next step, especially for volume EM data analysis. Second, we noticed that the effectiveness of our entropy-based loss must be improved, *e.g.* by replacing it with a more adaptive or data-driven strategy. Finally, in the presented form, hyperparameters such as the contrastive loss margin still require manual tuning, which is not ideal for the ease of use by biological experts.

5 Discussion

Here we presented ϵ-Seg, a novel semantic segmentation approach that leverages the variational latent representation of hierarchical variational autoencoders (HVAEs) trained on a limited amount of pixel-labels in an inpainting setup. We used a GMM prior instead of the traditionally employed Gaussian prior and introduced a novel segmentation head that incorporates both a cross-entropy loss and an entropy loss to leverage available data for which no ground truth (GT) class-labels are available. The integration of contrastive loss, combined with the structural advantages of the GMM prior, provides a means to effectively distinguish biological structures directly from the latent space encoding.

Transformer-based architectures, as used in MAESTER [34], usually have a rather large number of trainable parameters (*i.e.* $328, 452, 352$ trainable parameters in MAESTER). This makes such approaches less applicable to life-scientists since they require rather powerful compute setups. Even our biggest network, in contrast, only employs $3, 800, 869$ trainable parameters (see Tables S2 and S3), making it fast to train and easy to use. Our experiments also highlight an interesting fact, namely that smaller mask sizes with consistent labels emerged as the best strategy. This stands in contrast to Transformer-based approaches where a relatively large fraction of the input images is masked during training [34].

By combining hierarchical representations with advanced regularization techniques such as contrastive learning, we have shown that we can achieve competitive segmentation performance on complex microscopy data, even with relatively small models and limited training data. The proposed approach tackles the challenge of label scarcity, enhances latent space representations tailored to structured biological data, and lays the groundwork for future exploration of semi-supervised learning techniques and adaptive latent priors.

Overall, this work bridges the gap between fully supervised and unsupervised methods by offering a scalable approach for large-scale biomedical semantic image data segmentation.

References

[1] Malou Arvidsson, Salma Kazemi Rashed, and Sonja Aits. An annotated high-content fluorescence microscopy dataset with hoechst 33342-stained nuclei and manually labelled outlines. *Data Brief*, 46: 108769, 2023.

[2] Matthias Arzt, Joran Deschamps, Christopher Schmied, Tobias Pietzsch, Deborah Schmidt, Pavel Tomancak, Robert Haase, and Florian Jug. Labkit: Labeling and segmentation toolkit for big image data. *Frontiers in Computer Science*, 4, 2022.

[3] Abhinav Aswath, Abdulrahman Alsahaf, Ben N. G. Giepmans, and George Azzopardi. Segmentation in large-scale cellular electron microscopy with deep learning: A literature survey. *Medical Image Analysis*, 89:102920, 2023.

[4] Róger Bermúdez-Chacón, Okan Altingövde, Carlos Becker, Mathieu Salzmann, and Pascal Fua. Visual correspondences for unsupervised domain adaptation on electron microscopy images. *IEEE Trans. Med. Imaging*, 39(4):1256–1267, 2020.

[5] Ahcène Boubekki, Michael Kampffmeyer, Robert Jenssen, and Ulf Brefeld. Joint optimization of an autoencoder for clustering and embedding. *Machine Learning*, 110(6):1901–1937, 2021.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pages 1597–1607. PMLR, 2020.

[7] Rewon Child. Very deep VAEs generalize autoregressive models and can outperform them on images. In *International Conference on Learning Representations (ICLR)*, 2021.

[8] Mark Collier and Hector Urdiales. Scalable deep unsupervised clustering with concrete GMVAEs. In *1st Workshop on Deep Continuous-Discrete Machine Learning, ECML*, 2019.

[9] Ryan Conrad and Kedar Narayan. CEM500K, a large-scale heterogeneous unlabeled cellular electron microscopy image dataset for deep learning. *eLife*, 10:e65894, 2021.

[10] Nat Dilokthanakul, Pedro A. M. Mediano, Marta Garnelo, Matthew C. H. Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016. Under review at ICLR 2017.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

[12] Damjana Drobne. 3D imaging of cells and tissues by focused ion beam/scanning electron microscopy (FIB/SEM). *Methods in Molecular Biology*, 950:275–292, 2013.

[13] Jean-Louis Durrieu, Jean-Philippe Thiran, and Francis Kelly. Lower and upper bounds for approximation of the kullback–leibler divergence between gaussian mixture models. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4833–4836, 2012.

[14] Hongqing Han, Mariia Dmitrieva, Alexander Sauer, Ka Chun Tam, and Jens Rittscher. Self-supervised voxel-level representation rediscovers subcellular structures in volume electron microscopy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2276–2285, 2022.

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735, 2020.

[16] Lars Heinrich, Daniel Bennett, David Ackerman, William Park, John Bogovic, Nico Eckstein, Alexander Petruncio, Joe Clements, Sharmistha Pang, Chao-Shun Xu, Jan Funke, Walter Korff, Harald F. Hess, Jennifer Lippincott-Schwartz, Stephan Saalfeld, Andrew V. Weigel, and COSEM Project Team. Whole-cell organelle segmentation in volume electron microscopy. *Nature*, 599(7883):141–146, 2021.

[17] John R. Hershey and Peder A. Olsen. Approximating the kullback–leibler divergence between gaussian mixture models. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages IV–317–IV–320, 2007.

[18] Zhicheng Huang, Xiaojie Jin, Chengze Lu, Qibin Hou, Ming-Ming Cheng, Dongmei Fu, Xiaohui Shen, and Jiashi Feng. Contrastive masked autoencoders are stronger vision learners. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2023.

[19] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations (ICLR)*, 2017.

[20] Durk P. Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2014.

[21] Lars Maaløe, Marco Fraccaro, Valentin Liévin, and Ole Winther. BIVA: a very deep hierarchy of latent variables for generative modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6548–6559. Curran Associates, Inc., 2019.

[22] Andreas Müller, Daniel Schmidt, Chao-Shun Xu, Sharmistha Pang, Justin V. D'Costa, Stefan Kretschmar, Christian Münster, Thorsten Kurth, Florian Jug, Martin Weigert, Harald F. Hess, and Michele Solimena. 3D FIB-SEM reconstruction of microtubule-organelle interaction in whole primary mouse β cells. *Journal of Cell Biology*, 220(2):e202010039, 2021.

[23] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 3942–3951, 2018.

[24] Mangal Prakash, Mauricio Delbracio, Peyman Milanfar, and Florian Jug. Interpretable unsupervised diversity denoising and artefact removal. In *International Conference on Learning Representations (ICLR)*, 2022.

[25] Salma Kazemi Rashed, Malou Arvidsson, Rafsan Ahmed, and Sonja Aits. An annotated high-content fluorescence microscopy dataset with EGFP-galectin-3-stained cells and manually labelled outlines. *Data Brief*, 58:111148, 2025.

[26] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241. Springer, 2015.

[27] Marek Śmieja, Maciej Wołczyk, Jacek Tabor, and Bernhard C. Geiger. SeGMA: Semi-supervised gaussian mixture autoencoder. *IEEE Trans. Neural Netw. Learn. Syst.*, 32(9):3930–3941, 2021.

[28] Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3745–3753. Curran Associates, Inc., 2016.

[29] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7262–7272, 2021.

[30] Eichi Takaya, Yusuke Takeichi, Mamiko Ozaki, and Satoshi Kurihara. Sequential semi-supervised segmentation for serial electron microscopy image with small number of labels. *J. Neurosci. Methods*, 351: 109066, 2021.

[31] Kai Philipp Treder, Chenyang Huang, Jinseok S. Kim, and Angus I. Kirkland. Applications of deep learning in electron microscopy. *Microscopy (Oxford)*, 71(Supplement_1):i100–i115, 2022.

[32] Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 19667–19679, 2020.

[33] J. Walton. Lead aspartate, an en bloc contrast stain particularly useful for ultrastructural enzymology. *J. Histochem. Cytochem.*, 27(10):1337–1342, 1979.

[34] Ronald Xie, Kuan Pang, Gary D. Bader, and Bo Wang. MAESTER: Masked autoencoder guided segmentation at pixel resolution for accurate, self-supervised subcellular structure recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17521–17531, 2023.

[35] C. Shan Xu, Kenneth J. Hayworth, Zhiyuan Lu, Peter Grob, Ana M. Hassan, José G. García-Cerdán, Krishna K. Niyogi, Eva Nogales, Richard J. Weinberg, and Harald F. Hess. Enhanced FIB-SEM systems for large-volume 3D imaging. *eLife*, 6:e25916, 2017.

[36] Chao-Shun Xu, Sharmistha Pang, Gleb Shtengel, Andreas Müller, Anna T. Ritter, Heather K. Hoffman, Shin-Ya Takemura, Zhipeng Lu, Helene A. Pasolli, Nikhil Iyer, Jihoon Chung, Daniel Bennett, Andrew V. Weigel, Michael Freeman, Sean B. van Engelenburg, Tobias C. Walther, Robert V. Farese Jr, Jennifer Lippincott-Schwartz, Ira Mellman, Michele Solimena, and Harald F. Hess. An open-access volume electron microscopy atlas of whole cells and tissues. *Nature*, 599(7883):147–151, 2021. Erratum in: *Nature*, vol. 599, no. 7885, p. E5, 2021, doi:10.1038/s41586-021-04132-8.

ϵ-Seg: Sparsely Supervised Semantic Segmentation of Microscopy Data

Supplementary Material

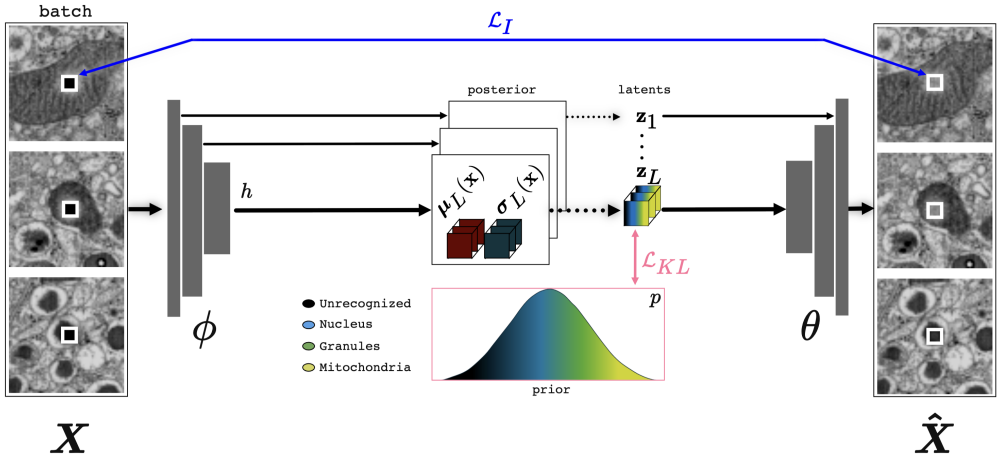

Figure S1: The overall pipeline of Vanilla HVAE in Table S1 (first row in Table 7), which is trained on an inpainting task (of the center-region masked inputs). ϕ and θ are encoder and decoder of the network, respectively. Dotted arrows show sampling from a distribution. h is an intermediate feature embedding of input $\boldsymbol{x}$ coming from the encoder ϕ and it is posterior distribution's parameters which is divided into two chunks shown as $\boldsymbol{\mu}_L$ and $\boldsymbol{\sigma}_L$. $\boldsymbol{z}_L$ is a sample from $\mathcal{N}(\boldsymbol{\mu}_L(\boldsymbol{x}), \boldsymbol{\sigma}_L^2(\boldsymbol{x}))$. For $\mathcal{L}_I$ inpainting loss and $\mathcal{L}_{KL}$ refer to Equations 1 and 7 respectively.

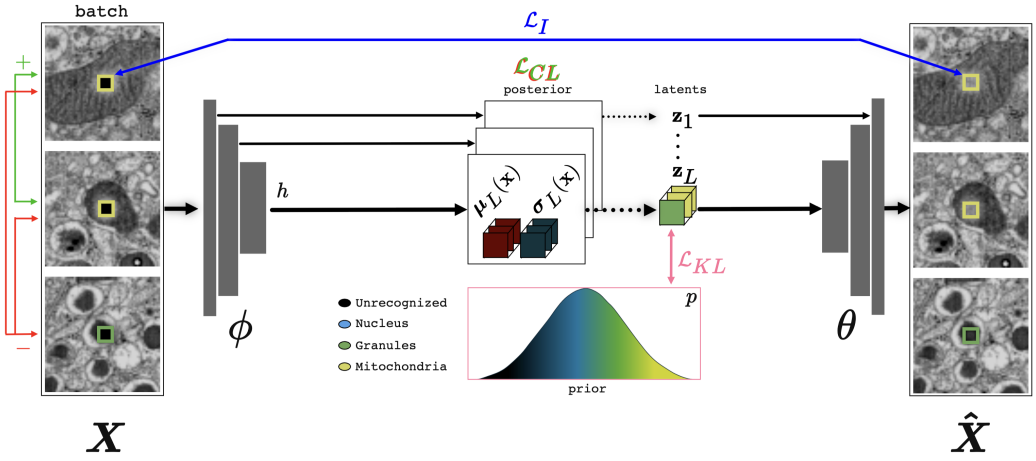

Figure S2: The overall pipeline of Vanilla HVAE with only CL added in the pipeline in the second row in Table 7, which is trained on an inpainting task (of the center-region masked inputs). Green and red arrows are showing positive and negative pair respectively, in a batch. ϕ and θ are encoder and decoder of the network, respectively. Dotted lines show sampling from a distribution. h is an intermediate feature embedding of input $\boldsymbol{x}$ coming from the encoder ϕ and it is posterior distribution's parameters which is divided into two chunks shown as $\boldsymbol{\mu}_L$ and $\boldsymbol{\sigma}_L$. $\boldsymbol{z}_L$ is a sample from $\mathcal{N}(\boldsymbol{\mu}_L(\boldsymbol{x}), \boldsymbol{\sigma}_L^2(\boldsymbol{x}))$. For $\mathcal{L}_I$ inpainting loss, $\mathcal{L}_{CL}$ contrastive loss and $\mathcal{L}_{KL}$ refer to Equations 1, 16 and 7 respectively.

Model	Learning Paradigm	U	N	G	M	Avg DSC
Vanilla HVAE* [24]	Self-Supervised	0.44	0.55	0.34	0.13	0.37
Labkit [2]	Sparsely Supervised	0.85	0.44	0.68	0.61	0.65
U-net [26]	Fully Supervised	0.94	0.99	0.90	0.87	0.93
U-net	Sparsely Supervised	0.90	0.96	0.78	0.66	0.83
Vanilla ViT [11]	Fully Supervised	0.91	0.98	0.77	0.87	0.88
Segmenter [29]	Fully Supervised	0.91	0.99	0.86	0.90	0.92
MAESTER* [34]	Self-Supervised	0.84	0.95	0.56	0.79	0.79
Han et al* [14]	Self-Supervised	-	-	-	-	0.66
ϵ-Seg (+$\mathcal{L}_H$)	Sparsely Supervised	0.89	0.98	0.81	0.83	0.88

Table S1: Dice similarity coefficient per class and average across all classes comparing our model with baselines on the "BetaSeg" dataset [22]. Methods marked with an asterisk use K-Means clustering on latent features to conduct semantic segmentation (more explanation can be found in Section 3). U: Unrecognized, N:Nucleus, G:Granules, M:Mitochondria.

# res. blocks	Per-Class Dice Coefficient				Avg DSC
	U	N	G	M	
5	0.86	0.98	0.80	0.75	0.85
4	0.85	0.97	0.80	0.74	0.84
3	0.88	0.96	0.81	0.80	0.86
2	0.87	0.97	0.81	0.77	0.86
1	0.85	0.97	0.80	0.72	0.84

Table S2: Residual blocks ablation (3 latent variables). U: Unrecognized, N: Nucleus, G: Granules, M: Mitochondria.

Entropy-based Loss. When the sample $\boldsymbol{y'}$ of the Gumbel-Softmax distribution is uniform, the network is maximally unsure about which class to predict for the current input patch. We noticed that this is commonly the case early during training, where the network has not yet seen a lot of patches for which ground truth labels are available.

To encourage the network not to predict a uniform $\boldsymbol{y'}$, we introduced an entropy loss for all patches $\boldsymbol{x}^{(j)} \in \boldsymbol{X}$ for which we do not have a ground truth class label.

$$\mathcal{L}_H = - \sum_{\boldsymbol{x}^{(j)} \in \boldsymbol{X}} \boldsymbol{y'}^{(j)} log(\boldsymbol{y'}^{(j)}). \tag{18}$$

# latent	Per-Class Dice Coefficient				Avg DSC
	U	N	G	M	
2	0.87	0.98	0.81	0.76	**0.86**
3	0.86	0.98	0.80	0.75	0.85

Table S3: Latent variables ablation (5 res. blocks/layer). U: Unrecognized, N:Nucleus, G:Granules, M:Mitochondria.

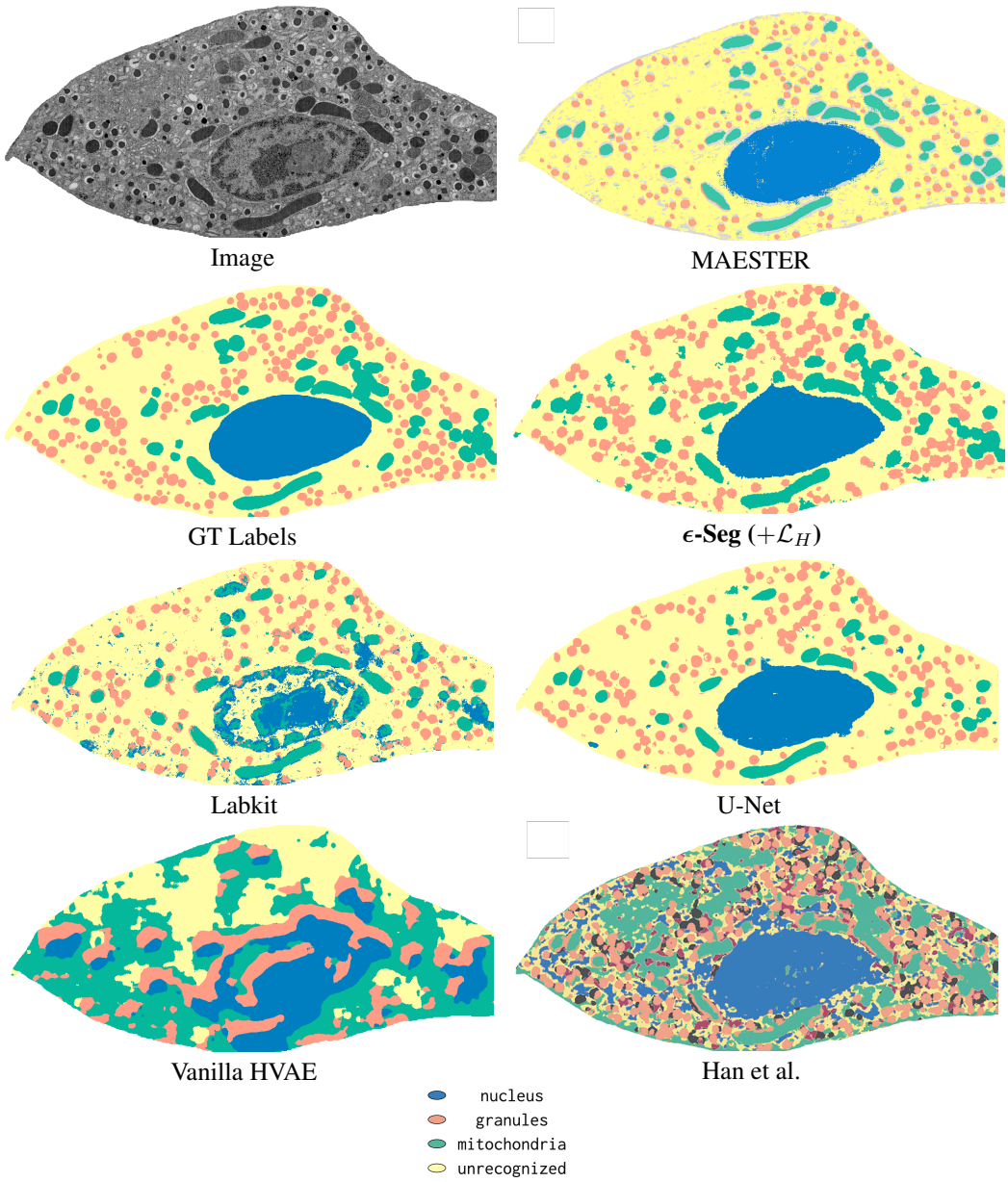

Figure S3: Qualitative segmentation result on part of the test image stack (section 627 of $high\_c4$ in "BetaSeg" dataset).

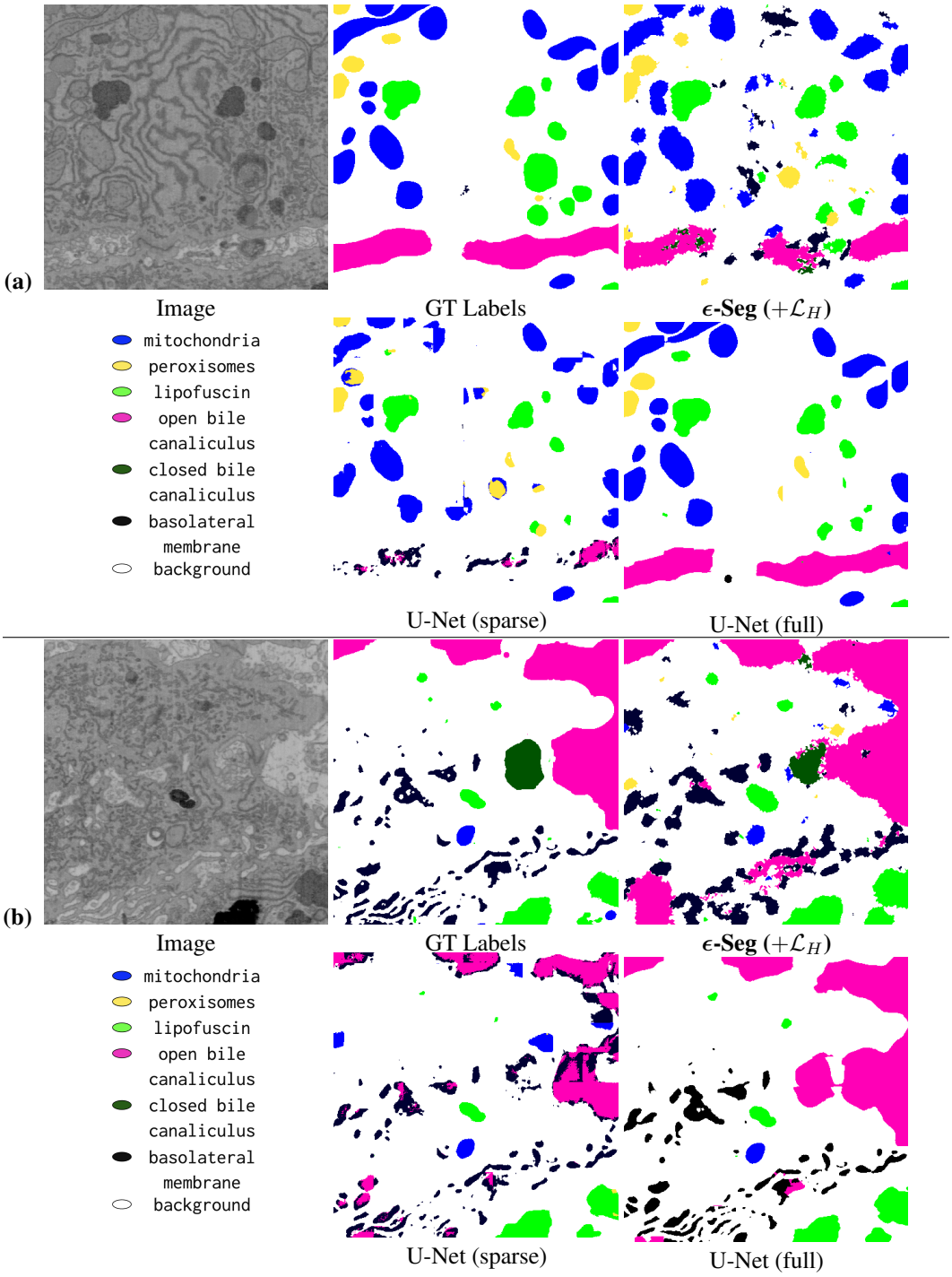

Figure S4: Qualitative segmentation result on two crops of the whole 3D volume. (a) and (b) are section 80 and 26 of crop00 and crop10 in "liver FIBSEM" dataset respectively. U-Net (sparse) and (full) is sparsely-supervised and fully-supervised respectively.

RLF	Model	U	N	G	M	Avg DSC
20	U-net	0.63	0.75	0.51	0.12	0.50
	ϵ-Seg	0.89	0.98	0.81	0.83	0.88
15	U-net	0.53	0.64	0.41	0.14	0.43
	ϵ-Seg	0.88	0.98	0.81	0.78	0.86
10	U-net	0.30	0.20	0.42	0.34	0.31
	ϵ-Seg	0.86	0.98	0.80	0.75	0.85
5	U-net	0.71	0.00	0.00	0.03	0.18
	ϵ-Seg	0.85	0.96	0.77	0.76	0.84
1	U-net	0.17	0.00	0.37	0.02	0.14
	ϵ-Seg	0.79	0.95	0.69	0.69	0.78

Table S4: Comparison between U-Net and ϵ-Seg on the "BetaSeg" dataset under varying label sparsity levels. "RLF" (Relative Labeling Factor) specifies the fraction of available labels, where 20 corresponds to 0.05% and 1 to 0.0025% of total labels. U: Unrecognized, N: Nucleus, G: Granules, M: Mitochondria. Although both models were trained with *balanced supervision*, using patches selected to include all classes, the U-Net still fails to segment the nucleus at very low labeling levels (RLF 1 and 5). This illustrates a key limitation of discriminative models such as U-Net, under extreme supervision sparsity, even balanced examples may not suffice to generalize fine-grained or context-sensitive structures like the nucleus. In contrast, ϵ-Seg benefits from its class-aware latent modeling via the GMM prior, which enables it to extract meaningful representations for different structures and distinguish them semantically. We note that the sparse U-Net reported earlier was trained on slice numbers 800, 600, and 500 of the "high_c1", "high_c2", and "high_c3" volumes of the "BetaSeg" dataset. For selecting the same amount of data used in ϵ-Seg, to train the 2D U-Net on, as reported in the table above, we extracted 64x64 patches where except background, different classes are approximately well balanced.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the paper's contributions, including the design of ϵ-Seg, an HVAE-based segmentation framework with a GMM prior, center-region inpainting, contrastive learning, and a dedicated semantic segmentation head. These claims are appropriately scoped and supported by the methodology and experiments presented in the rest of the paper. The text also specifies that the method works with extremely limited supervision and addresses common practical challenges in EM segmentation, which are demonstrated through empirical results.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We included a dedicated Limitations mini-headline at the end of the Experiments and Result (Section 4). There, we discuss that the current method is restricted to 2D data and would likely benefit from a 3D extension. We also note that the entropy-based loss could be further optimized, and that dataset-specific tuning is required for some hyperparameters, such as the contrastive loss margin.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: The paper does not include formal theoretical results or proofs (e.g., theorems or lemmas). However, it provides detailed derivations and explanations of the model components and loss functions (see Section 3), including the use of a GMM prior in the HVAE framework and KL divergence formulation.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides all necessary implementation details, including model architecture (Figure 1), training settings, dataset descriptions and evaluation metrics (Section 4). Loss terms, and component configurations are also disclosed to allow faithful reproduction of the reported results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example

(a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Two of the datasets used in our experiments are publicly available and referenced in the paper. The third dataset is private and cannot be shared due to data access restrictions. We will publicly release the code on GitHub along with detailed instructions to reproduce all experiments based on the public datasets.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide all relevant training and evaluation details, including data splits, optimizer type, learning rate, batch size, and other key hyperparameters. Where appropriate, we explain how hyperparameters were chosen, either based on prior work or grid search.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: While our main experiment (Table S1) includes 5-fold cross-validation to mitigate variability due to data splits, we did not report error bars or perform statistical significance tests. Given the limited size of our dataset and the exploratory nature of our work, our focus was on assessing the feasibility of the proposed method rather than establishing statistically significant performance differences.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In this work, we mentioned the number of parameters in our largest model and the efficiency of our approach. Our method improves upon previous techniques by eliminating the need for K-Means clustering, allowing the model to directly generate segmentation labels from the segmentation head. This change significantly accelerates the inference process, resulting in faster segmentation without sacrificing accuracy.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: Yes, the research conducted in this paper conforms with the NeurIPS Code of Ethics. We have adhered to all relevant ethical guidelines, ensuring transparency, fairness, and respect for privacy in our work.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: The focus of this paper is primarily on technical advancements in segmentation, and while it does not explicitly address societal impacts, the method may have positive implications in fields like medical imaging. However, the societal implications are, if at all, only indirect and we believe the answer 'NA' is most appropriate.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: This paper does not involve models or data with a high risk for misuse, and thus does not describe any specific safeguards related to their release.

Guidelines:

- The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Yes, all creators and original owners of assets used in this paper, including datasets, code, and models, have been properly credited. Additionally, the licenses and terms of use associated with these assets have been explicitly mentioned and respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Yes, the private dataset used in this paper is well documented, including details on its structure, size, and usage. However, due to privacy and confidentiality constraints, the dataset is not publicly available. Access to the dataset is restricted, but interested parties can contact the authors to be connected to the dataset owners.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This paper does not involve research with human subjects, and therefore, no IRB or equivalent approvals were required.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: No large language models (LLMs) were used as part of the core methods in this research.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.