

SaSi: A Self-augmented and Self-Interpreted Deep-Learning Approach for Few-shot Cryo-ET Particle Detection

Gokul Adethya T^{1*}, Bhanu Pratyush Mantha^{2*}, Tianyang Wang³, Xingjian Li^{4†}, Min Xu^{4†},

1 Department of Computer Science and Engineering, National Institute of Technology, Tiruchirappalli, Tamil Nadu, India

2 Department of Electronics and Communication Engineering, National Institute of Technology, Tiruchirappalli, Tamil Nadu, India

3 Department of Computer Science, University of Alabama at Birmingham, Birmingham, Alabama, United States

4 Computational Biology Department, Carnegie Mellon University, Pittsburgh, Pennsylvania, United States

* These authors contributed equally to this work.

† Corresponding authors.

Abstract

Cryo-electron tomography (cryo-ET) has emerged as a powerful technique for imaging macromolecular complexes in their near-native states. However, the localization of 3D particles in cellular environments still presents a significant challenge due to low signal-to-noise ratios and missing wedge artifacts. Deep learning approaches have shown great potential, but they need huge amounts of data, which can be a challenge in cryo-ET scenarios where labeled data is often scarce. In this paper, we propose a novel Self-augmented and Self-interpreted (SaSi) deep learning approach towards few-shot particle detection in 3D cryo-ET images. Our method builds upon self-augmentation techniques to further boost data utilization and introduces a self-interpreted segmentation strategy for alleviating dependency on labeled data, hence improving generalization and robustness. As demonstrated by experiments conducted on both simulated and real-world cryo-ET datasets, the SaSi approach significantly outperforms existing state-of-the-art methods for particle localization, demonstrating its compatibility with model architectures ranging from CNNs to Vision Transformers (ViT). This research increases understanding of how to detect particles with very few labels in cryo-ET and thus sets a new benchmark for few-shot learning in structural biology.

Author summary

This study addresses the challenge of identifying 3D particles in cryo-electron tomography (cryo-ET) images, which capture cellular structures in near-native conditions. Cryo-ET is a powerful tool in biology, but finding particles in these images is difficult because they are often noisy, and labelled data needed for training detection algorithms are scarce. Our approach, called SaSi, introduces a method to detect particles in cryo-ET images even when few labels are available. SaSi combines strategies to boost learning with limited data by augmenting existing images and teaching the

model to independently assess and refine its predictions. We tested SaSi on both simulated and real cryo-ET datasets and found that it outperformed existing methods, showing promise for advancing particle detection when data are limited. This work may assist researchers more effectively in studying complex cellular structures and biological processes, furthering advances in cellular biology and possible medical applications.

1 Introduction

Cell biological processes rely on complex networks of molecular assemblies, whose native structures and spatial distributions are crucial to understanding cellular mechanisms. Cryo-electron tomography (cryo-ET) has been gaining popularity in structural biology in recent years. Cryo-ET [1,2] is a powerful technique that provides three-dimensional (3D) visualization of macromolecular complexes in near-native states at sub-nanometre resolutions. This provides new insights into the understanding of cellular processes and actions of drugs. Cryo-ET has helped in the discovery of many important structures like SARS-Cov-2, which was responsible for the COVID-19 pandemic [3]. Deep learning based automated Cryo-ET image analysis [4,5] has received wide attention due to its high efficiency and low cost. However, these computational approaches still face challenges in the localization and classification of 3D particles in cellular environments. First, electrons in imaging interact strongly with biological samples, limiting the dose to prevent damaging samples during imaging. The limited dose limits the resolution of tomograms to about 5nm [6], which isn't enough to study the structures of macromolecular complexes. Second, imaging angles are typically limited to $\pm 60^\circ$ or $\pm 70^\circ$, because of sample thickness. This results in incomplete reconstruction with a missing wedge in Fourier space, making 3D particle picking very challenging compared to 2D particle picking in cryo-electron micrographs. With the detected particles, researchers can further apply subtomogram averaging [7] to enhance the resolution of macromolecular complexes by aligning and averaging copies of identical particles.

Several existing solutions have been proposed to address the problem of particle localization and classification. DeepFinder [8] achieved the best localization performance, and Multi-Cascade DS Net [9] achieved the best classification performance. Recently, DeepETPicker [10] achieved state-of-the-art (SOTA) performance in both classification and localization on the simulated SHREC2021 benchmark. However, given the highly noisy and large 3D tomograms, labeling thousands of particles is too laborious and usually unrealistic in practice.

This paper considers few-shot learning settings in the problem of particle localization. We find that existing SOTA solutions delivered sub-optimal performance when only tens of particles in a 3D tomogram are manually picked and labeled. This reveals several generalization risks of the existing frameworks, which mostly focus on model architecture and pipeline design.

To solve this problem, we introduce a Self-augmented and Self-interpreted (SaSi) deep learning approach to achieve few-shot particle detection in 3D cryo-ET images. Based on a general U-Net model architecture, we propose the following novel design, which has not been studied in existing literature.

(1) Self-augmented Data Utilization. The idea aims to fully exploit the value of the existing available tomogram data to enhance learning. One key challenge is the sparse distribution of particles over the 3D tomogram data, especially for those small structures of interest. This issue is even more severe given only a few particles are labeled. To address this, our first strategy is to increase the particle density. While popular data augmentation techniques only apply a transformation to the existing examples, here we adopt Augmix [11] to simultaneously increase existing particles' variants and volumetric occupancy without relying on external resources. To further

make use of the unlabeled tomogram regions, we consider self-supervised learning [12–14] to learn generalized features. Specifically, we apply contrastive learning. The representations of images are learnt by optimizing contrastive loss utilizing positive and negative pairs, where positive pairs are the pairs of images that are augmentations of the same image, and negative pairs are the augmentations of other images.

(2) *Self-interpreted Image Segmentation*. To further alleviate the dependency on labeled data, we design a novel self-interpreted segmentation approach enforcing the model to interpret and validate its own outputs by leveraging its inherent consistency. For each input example, an augmented version is created using spatial transformations. Instead of fitting the external ground truth, this approach regularizes the segmentation outputs to undergo the same transformation as their corresponding inputs. This component can be used on both supervised and self-supervised learning. This self-validation mechanism helps escape over-fitting to limited ground truth labels for supervised learning. For self-supervised learning complements the standard contrastive learning component by pre-training the decoder of our segmentation model.

Besides the above core techniques, we also improve loss function design and post-processing by considering the particular characteristics of the cryo-ET particle detection task. We use spherical masks instead of ground truth segmentation masks, as they are generally not available in the real world. So, this helps us to generalize better. We evaluate our method’s performance on both simulated and real-world datasets.

Our main contributions are summarized as follows:

- We identify the challenge of particle detection in cryo-ET analysis with label scarcity and sparse distribution. To the best of our knowledge, we are the first to explore few-shot learning settings for this challenging task.
- By analyzing the particular characteristics of our target data, we design a novel self-augmented and self-interpreted (SaSi) deep learning approach to solve the problem of few-shot particle detection in cryo-ET images.
- We demonstrate our method’s effectiveness on both simulated and real-world cryo-ET benchmark datasets. Compared with state-of-the-art baseline approaches, our method improves the localization performance by a significant margin.
- The core components in our SaSi approach are compatible with most of the popular model architectures, such as CNNs and Vision Transformers (ViT).

2 Related Works

Early works in cryo-ET for localizing macromolecules were based on Template Matching [15] and DoG (difference of Gaussians) [16]. In Template Matching, the position and orientation of a predefined template are determined by maximizing its cross-correlation with the tomogram., but this has several limitations, like strong dependence on a predefined template and the need for manual threshold tuning. DoG applies a band passband-pass filter that removes noisy high-frequency components and homogeneous low-frequency areas, obtaining borders of the structures. The DoG picks particles irrespective of the classes, and the performance depends heavily on the tuning of Gaussian filters. In recent years, machine-learning methods have been applied to cryo-ET, Chen et al. [17] used Support Vector Machines for detection and classification.

With the increase in cryo-ET data, Deep learning methods started gaining popularity. Che et al. [18] propose 3 models: DSRF3D-v2, which is a 3D version of VGGNet [19]; RB3D, which is a 3D variant of ResNet [20] and CB3D uses a 3D CNN based model for classification of macromolecules. Luengo et al. [21] proposed a

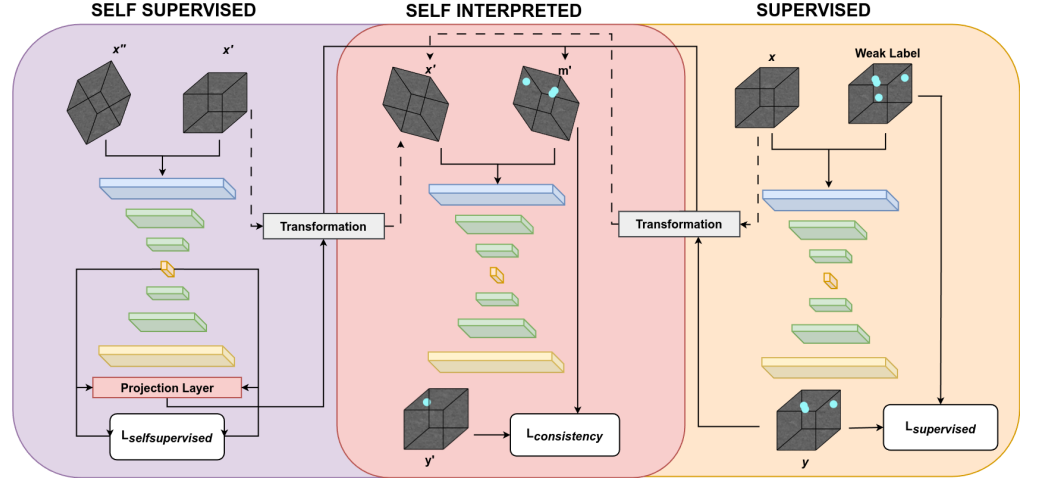


Fig 1. This figure illustrates the training phase incorporating self-supervised learning using augmented pair x', x'' , supervised learning with ground truth mask generated using weak labels, and self-interpreted using x' and predicted class m' from either of self-supervised and supervised learning phase.

supervised approach to classify voxels, but it required manually designed features or rules, which often have various limitations. Chen et al. [22] developed another supervised segmentation method, utilizing the excellent capabilities of CNN, but a separate CNN is trained for each type of structural feature. Li et al. [23] proposed a Faster-RCNN [24] based method for automatic identification and localization in a slice-by-slice fashion, but 3D information in adjacent slices was not properly utilized. MC-DS-Net uses a denoising and segmentation architecture; however, they use a real ground truth mask of macromolecules. DeepFinder [8] uses a 3D-UNet for generating segmentation voxels and then applies mean-shift clustering post-processing to find the positions of particles. It also uses spherical masks as weak labels instead of real ground truth masks of macromolecular particles, which are generally unavailable in the real world. DeepETPicker [10] utilizes a 3D ResUNet model, taking advantage of coordinated convolution multiscale image pyramid inputs to enhance localization performance with the help of weak labels. Mean-pooling non-maximum suppression (MP-NMS), post-processing is applied to the generated segmentation output masks to extract the positions of particles. ProtoNet-CE [25] applies a few-shot learning-based method for the task of subtomogram classification only. They combine task-specific embeddings with task-agnostic embeddings, and these combined embeddings are classified using the nearest neighbor classifier.

3 Methodology

In this section, we begin by outlining the problem setup and standard supervised training strategy for the few-shot particle detection task. From section 0.3, we introduce the four technical components used in our SaSi framework, which are the post-processing module (sec 0.3), occupancy improving with Augmix (sec 0.4), self-supervised feature learning with SimCLR (sec 0.5) and self-interpreted learning for segmentation (sec 0.6). An overall architecture of SaSi is illustrated in Figure 1.

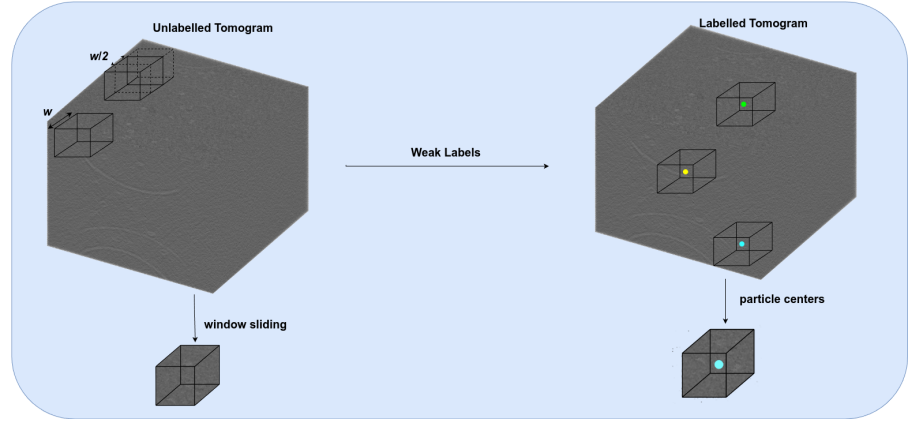


Fig 2. During testing and self-supervised learning, a sliding window approach with window size W and stride $W/2$ is applied, while particle center-based sampling is utilized for supervised learning.

3.1 Problem Setup

In cryo-ET data analysis, the task of particle detection differs from general object detection in computer vision in several key aspects. Firstly, the original labeling formation for training and evaluation is point annotations to mark particle positions rather than bounding boxes. The objective is to predict the central positions of the particles of interest. This approach is used because the radius is already known for most biological particles. Secondly, only a small number of 3D cryo-ET tomograms are typically collected for a given biological sample. Following standard practice, we use one tomogram for training and the others for testing. It is important to note that, although only a limited number of point annotations are provided for the particles, the entire training tomogram is considered the available dataset since a tomogram represents the smallest unit of data obtained from cryo-ET imaging.

By training a segmentation model with a post-processing step, our final goal is to predict the particle position for an unknown number of particles in the entire tomogram.

3.2 Supervised Learning

To achieve both particle localization and classification, we apply a supervised segmentation model U-Net, which gives voxel-wise predictions. For this part, we mostly follow the practice in existing SOTA approaches [8, 10].

Since tomogram data have a large spatial dimension, is divided into smaller subvolumes (samples) of size $W \times W \times W$, which is the input to the model. To alleviate the potential information loss caused by patch boundaries, we further perform window sliding during inference to obtain subtomograms of size $W \times W \times W$ using a kernel of size $W \times W \times W$ and a stride of $W/2$.

As illustrated in Figure 2, we choose the subtomogram centered on the particle's centroid. Thus, each batch will contain samples of an equal number of particle types. To increase model robustness and prevent loss of information, we use spatial transformation since any other augmentation that affects voxel values can be risky and could distort the underlying pattern. This risk is exceptionally high because the model has fewer examples for counterbalancing the distortion, making it more likely to overfit the altered data.

Using the weak point labels and minimum radius for each particle, we create pseudo-strong labels by generating a sphere of minimum radius filled centered at weak

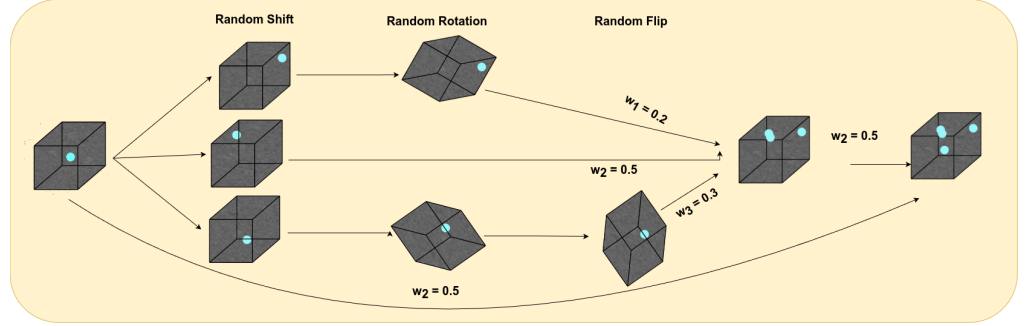


Fig 3. An illustration of the composition process for masks generated using weak labels during AugMix.

labels centroid. These generated spheres are used as a ground truth segmentation mask, converting the problem into a segmentation task. We adopt Focal Loss and Dice Loss as the learning objective as

$$L_{sup} = \lambda_{dice} L_{dice} + \lambda_{focal} L_{focal}, \quad (1)$$

where the coefficient is set to $\lambda_{dice} = 20$ and $\lambda_{focal} = 1$ according to empirical observations.

3.3 Post Processing

We apply the arg max operation on the model output to get a voxel-wise classification. Then we apply connected components in 3D (cc3d) [26] using 26 connected components, which uses a 3D variant of the two-pass method by Rosenfeld and Pflatz augmented with Union-Find and a decision tree based on the 2D 8-connected work of Wu, Otoo, and Suzuki on this sub tomogram to generate some arbitrary number of clusters. Compared against previous post-processing strategies Meanshift (in Deepfinder [8]) and MP-NMS (in DeepETpicker [10]), we find cc3d is much more reliable in the few-shot settings, and it requires no hyperparameters to be tuned. Therefore, in our experiments, we also integrate cc3d into Deepfinder and DeepETpicker as stronger SOTA baselines.

3.4 Augmix

Generating augmented examples is crucial in enhancing generalization when training examples are limited. Many particles occupy a very small portion of the volume in cryo-ET images. Therefore, we adopt augmix [11] to simultaneously generate augmented particles and increase the volumetric density, leading to more efficient learning from insufficient training data. To avoid potential information loss regarding structural changes, we limit the set of augmentation operations $\mathcal{T} = \{T_1, T_2, \dots, T_n\}$ to include only random shifting, rotation, and flipping. During the AugMix process, an input example x is augmented to several variants $\{x'_i\}_{i=1}^m$ by different augmentation chains. Each chain consists of a sequence of k augmentation operations selected randomly from the predefined set \mathcal{T} as $x'_i = T_{i_1} \circ T_{i_2} \circ \dots \circ T_{i_k}(x)$. The mixed augmented example is calculated by $x_{aug} = \sum_{i=1}^m w_i x'_i$, with each coefficient w_i randomly sampled from a Dirichlet distribution as $w_i \sim \text{Dirichlet}(\alpha, \alpha, \dots, \alpha)$. We then mix the result of the augmentation chain and the original image at "skip connection" to generate the final example as $x_{mix} = \beta x + (1 - \beta)x_{aug}$ with β sampled from a Beta distribution. We adopt the same mixing process for the annotation and use the argmax operation to get the final Augmix-generated ground truth mask. This process is illustrated in Figure 3.

3.5 Self-Supervised

We use the popular contrastive learning approach SimCLR and NT-Xent Loss on the training subvolumes to learn general cryo-ET image features. By passing two augmented variants $x_{mix}^{(1)}$ and $x_{mix}^{(2)}$, we get their embeddings from the encoder or equivalent block depending on the model architecture and squash them to 1D vectors $e^{(1)}, e^{(2)}$. Gupta et al. [27] has shown the non-linear projection layer to be effective, so we apply a ReLU activation on the squashed embeddings and pass it to a linear layer of output dimension 128 to generate $z^{(1)}, z^{(2)}$. For each input x , the NT-Xent Loss maximizes the similarity between its augmented pair of the projected embeddings $z^{(1)}, z^{(2)}$, where minimizes the similarity for different examples.

We have two configurations for self-supervision: Periodic Self-Supervision (PSS) and Initial Self-Supervision (ISS). In PSS, self-supervised learning is performed at regular intervals until half the total number of epochs is reached. In ISS, we continuously perform self-supervised learning during the first few epochs and after that, supervised learning continues. The self-supervised learning phase enhances the model’s ability to extract robust and meaningful feature representations from the limited available data. After certain epochs, we transition exclusively to supervised learning. This shift allows the model to fine-tune its understanding and adapt specifically to the downstream task, optimizing performance by focusing on task-specific patterns and nuances. This balanced approach ensures that the model benefits from both broad feature learning and targeted task adaptation, leading to improved generalization and efficiency in few-shot scenarios.

3.6 Self-interpreted Image Segmentation

Here we introduce the self-interpreted learning approach for training segmentation models without requiring ground truth segmentation masks. Our method relies on the intuition that a model should be able to explain its predictions instead of only fitting external ground truth labels in a blind manner. We achieve this goal by involving a transformation-aware consistency loss on predicted segmentation maps. Specifically, For each input example x , we generate an augmented version x' by applying a spatial transformation T as $x' = T(x)$. The segmentation model f predicts segmentation maps $m = f(x)$ and $m' = f(x')$ for the original and augmented inputs, respectively. Then we apply the same transformation T on the segmentation map m and use this transformed map $T(m)$ as a target to learn $f(x')$. To be specific, we apply the same loss function $L(m', T(m))$ as in Eq. (1).

The rationale behind this self-interpreted approach is that if a model truly understands the structure of the objects it segments, its segmentation results should remain consistent according to the spatial transformations applied to the input. We apply this self-interpreted segmentation loss on both self-supervised learning and supervised learning phases of the training as it requires no labels. In addition, unlike traditional contrastive learning approaches that focus on transformation-invariant feature learning only for the encoder part, our self-interpreted feature learning also covers the decoder.

4 Experiments

4.1 Datasets

4.1.1 Simulated Dataset

We use SHREC2021 [28], which contains ten 3D tomograms, the ground truth containing information about each macromolecule’s localization and class. It also contains weak labels in the form of the coordinates of the centroid of each macromolecular particle. Each 3D tomogram is of size $512 \times 512 \times 512$. There are a total of 12 protein classes, vesicles and gold fiducials.

We represent i^{th} tomogram using T_i . To better simulate the realistic low-resource scenario, we use only the tomogram T_0 for training. The SHREC2021 [28] dataset officially suggests T_9 as the test data. However, to perform a more reliable evaluation, we adopt three different subsets as the test set, which are (1) T_9 , (2) T_{8-9} and T_{6-9} . The additional tomograms involved in the test set are not specified by the paper, but here we create diverse settings for a more comprehensive comparison. We use the f1 score to evaluate the performance of particle localization.

4.1.2 Real Dataset

We use real experimental data [29] consisting of 3 tomograms with 3 annotated classes of macromolecular structures, namely RIBO, 26S, and TRIC. There are a total of 646 labeled macromolecules and the size of each tomogram is $375 \times 926 \times 926$. Since we have only 3 tomograms, we use 1 tomogram each for training (G4L3T1), validation (G3L6T1), and test (G4L3T2). Similar to the SHREC dataset, here we use both the official test split and the combination of validation and test set for evaluation. Since the labeled data are extremely scarce in the few-shot scenario, we apply a fixed set of empirical hyperparameters without using the validation set for hyperparameter tuning.

4.2 Baselines

We evaluate DeepFinder’s (DF) and DeepETPicker’s performance using the model architecture as illustrated in their respective papers. For DeepFinder, we tested two different post-processing methods, the first being mean-shift, as given in the paper, and the other being cc3d. Similarly, for DeepETPicker, we use MP-NMS and cc3d post-processing.

Additionally, we involve a Vision Transformer-based architecture VideoMAE (3D-MAE) [30]. To better fit our cryo-ET data, we implement a scaled-down version of VideoMAE and evaluate its performance. The original VideoMAE consisted of 100M parameters, our version contains 3.5M parameters as illustrated in Table 3.

Table 1. Localization F1 scores on different few-shot settings. N represents the number of labeled points for each class. DF refers to DeepFinder. SaSi* denotes the best performance achieved by SaSi among different configurations, including Augmix, SSL, and post-processing.

Method	$N = 3$			$N = 5$			$N = 10$		
	T_9	T_{8-9}	T_{6-9}	T_9	T_{8-9}	T_{6-9}	T_9	T_{8-9}	T_{6-9}
DF (meanshift)	0.129	0.065	0.071	0.062	0.031	0.057	0.146	0.073	0.090
DF (cc3d)	0.288	0.186	0.189	0.293	0.195	0.211	0.358	0.224	0.198
DeepETpicker (nmsv2)	0.024	0.024	0.024	0.023	0.031	0.027	0.023	0.023	0.024
DeepETpicker (cc3d)	0.387	0.213	0.201	0.302	0.152	0.165	0.403	0.213	0.212
SaSi* (cc3d)	0.378	0.262	0.273	0.372	0.362	0.315	0.359	0.410	0.356

Method	$N = 3$		$N = 5$		$N = 10$	
	Test	Avg	Test	Avg	Test	Avg
DF (cc3d)	0.050	0.066	0.045	0.076	0.132	0.123
SaSi (cc3d)	0.060	0.076	0.081	0.098	0.136	0.163

Table 2. Localization performance evaluated by F1 score on the real world dataset [29]. The test refers to the official test split, and Avg refers to using the combination of the test and validation tomogram as the test set.

Model	# Parameters	e_i size
DeepFinder	903,060	13,824
DeepETPicker	8,841,092	6,912
MAE	3,573,772	6,912

Table 3. Model’s parameter count and its corresponding size of squashed 1D vector encoded embedding e_i

4.3 Implementation details

All our experiments are conducted using PyTorch. We apply random rotation, random flip, and random shift augmentation to increase the data size artificially. In Random Shift, we shift the subtomogram center within the range of $\pm 50\%$ of the subtomogram size. For random rotation, we take a subtomogram of size up to $\sqrt{2} \times W$ to avoid filling the voxel with arbitrary values and to preserve the voxel information. For Augmix, we set $\alpha = 1$ in Dirichlet distributions. We use the Adam optimizer with a learning rate of 0.0001. The batch size used is 16, and we use a combination of dice loss and focal loss in the ratio 20:1. The subtomogram size is $24 \times 24 \times 24$ voxels. So the kernel size used is $24 \times 24 \times 24$, and the stride is 12. Considering the few-shot setting, we train each model for a total of 8000 epochs, which is equivalent to around 10,000 to 80,000 iterations for most configurations. The temperature parameter used for NT-Xent loss is 0.1. In PSS, self-supervised learning is periodically performed at intervals of 400 epochs up to the 4000th epoch. In ISS, we perform self-supervised learning once for 10 epochs, followed by supervised learning until training finishes. The self-interpreted learning loss is applied before the 4000th epoch as it achieved better empirical performance. For training, we use 8xA5000 GPUs.

5 Main Results

Table 1 reports the performance of baselines and our approach on the SHREC 2021 benchmark. It is shown that our SaSi approach outperforms existing SOTA baselines DeepFinder and DeepETPicker on the few-shot settings with significant margins. We observe that the baseline results are much better and more robust with cc3d post-processing. SaSi even surpasses those stronger SOTA baselines in most cases.

We observe ISS showing a substantial improvement compared to all the baseline models in all $N = 3, 5, 10$ particles. From Table 2, we also see similar improvements while applying SaSi in real-world datasets. Though all the results on this challenging real dataset are still low, SaSi shows clear improvements in most experimental groups.

Method	$N = 3$			$N = 5$			$N = 10$		
	T_9	T_{8-9}	T_{6-9}	T_9	T_{8-9}	T_{6-9}	T_9	T_{8-9}	T_{6-9}
DF cc3d	0.288	0.186	0.189	0.293	0.195	0.211	0.358	0.224	0.198
DF cc3d + Augmix	0.200	0.177	0.239	0.393	0.368	0.335	0.361	0.437	0.350
3D-MAE cc3d	0.417	0.299	0.247	0.387	0.215	0.200	0.429	0.431	0.321
3D-MAE cc3d + Augmix	0.263	0.261	0.272	0.484	0.442	0.403	0.444	0.517	0.458
DeepETPicker cc3d	0.387	0.213	0.201	0.302	0.152	0.165	0.403	0.213	0.212
DeepETPicker cc3d + Augmix	0.374	0.364	0.277	0.313	0.308	0.240	0.314	0.441	0.299

Table 4. Analysis of Augmix on the SHREC benchmark with different baseline model architectures.

6 Ablation Study

In Table 4, we study the impact of Augmix on the baselines. We can notice that tomogram 9, which is the test tomogram used for the SHREC2021 dataset, shows better results for non-augmix-based method, however, we also show that tomogram_9 alone is not a good criteria to judge the model’s performance, so we display the models performance on tomogram_8 and tomogram_9 represented by T_{8-9} and average of all 4 tomograms (6,7,8,9) represented by T_{6-9} . For $N = 3$, we note that the AugMix performs worse than its corresponding baselines across different models. This is mainly due to the extremely low pool of training samples from which the augmix is being generated, causing high uncertainty at the cost of robustness. Even though AugMix preserves better information, there aren’t enough samples to alleviate the little distortion of information. For $N = 5, 10$ we can see that massive improvements indicating the robustness of the method. For $N = 3$, we observe a performance decrease when using AugMix, both with and without SSL, following a consistent trend. However, for $N = 5, 10$, AugMix enhances SSL performance in most cases when focusing on T_{8-9} .

Table 5 compares the performance of SSL with PSS and ISS alongside AugMix. We have experimented with all SSL and AugMix combinations while running a few configurations for DeepETPicker and 3D-MAE to get an overview of performance across models. We observe that almost all combinations of SSL and AugMix show a substantial improvement over the corresponding baselines for each model, except for DeepETPicker. We find that DeepETPicker exhibits anomalous trends compared to the other models, likely due to its significantly higher parameter count, more than twice that of MAE and ten times that of DeepFinder leading to faster overfitting.

Table 5. Analysis of the combination of Augmix and self-supervised learning strategies on the SHREC benchmark.

Method	$N = 3$			$N = 5$			$N = 10$		
	T_9	T_{8-9}	T_{6-9}	T_9	T_{8-9}	T_{6-9}	T_9	T_{8-9}	T_{6-9}
DF cc3d PSS	0.375	0.195	0.239	0.318	0.189	0.173	0.468	0.238	0.222
DF cc3d PSS + Augmix	0.353	0.300	0.280	0.263	0.216	0.234	0.451	0.433	0.370
DF cc3d ISS + Augmix	0.378	0.262	0.273	0.372	0.362	0.315	0.359	0.410	0.356
DF cc3d ISS	0.468	0.522	0.386	0.457	0.314	0.284	0.459	0.304	0.263
DeepETPicker cc3d PSS	0.252	0.133	0.161	0.203	0.127	0.132	0.272	0.136	0.130
DeepETPicker cc3d ISS	0.452	0.264	0.247	0.155	0.092	0.128	0.197	0.101	0.121
3D-MAE cc3d PSS	0.435	0.451	0.341	0.387	0.388	0.304	0.404	0.215	0.209
3D-MAE cc3d PSS + Augmix	0.345	0.348	0.282	0.424	0.436	0.362	0.450	0.426	0.371

SSL improves robustness to model variations and helps in learning good representations as it relies on the intrinsic properties of the data, free from labelled biases. This is crucial when working with imbalanced classes. However, given our problem setup, the limited particle count can lead to a reduced sample size, increasing the chance of learning incorrect patterns and making overfitting more likely. PSS offers an advantage by integrating SSL and supervised learning, allowing them to complement each other. SSL mitigates overfitting by leveraging all tomograms. By contrast, ISS simplifies the process using a fixed SSL step count and generally improves with more SSL steps without needing fine-tuning. Although it does not prevent overfitting inherently, it relies on AugMix and consistency loss for robustness.

The main challenge with PSS is determining the right intervals between SSL and supervised learning to apply SSL before overfitting occurs. In contrast, ISS uses a more traditional pretraining method that avoids the need for hyperparameter adjustments like SSL intervals or steps, as performance generally improves with more SSL steps. This explains why ISS outperforms PSS significantly, as shown in Table 5. However, ISS does not inherently prevent overfitting and instead relies on AugMix and consistency

loss for robustness, which is also evident from the results (except for $N = 3$, due to the previously mentioned AugMix issue).

Table 6. Comparison between Self-Interpreted Learning and Baseline Methods. "Consistency" indicates that self-interpreted learning is applied.

Method	$N = 3$			$N = 5$			$N = 10$		
	T_9	T_{8-9}	T_{6-9}	T_9	T_{8-9}	T_{6-9}	T_9	T_{8-9}	T_{6-9}
DF cc3d	0.288	0.186	0.189	0.293	0.195	0.211	0.358	0.224	0.198
DF cc3d Consistency	0.198	0.170	0.213	0.072	0.358	0.260	0.327	0.200	0.225
DF cc3d + Augmix	0.200	0.177	0.239	0.393	0.368	0.335	0.361	0.437	0.350
DF cc3d Consistency + Augmix	0.409	0.269	0.297	0.513	0.331	0.343	0.484	0.393	0.349
DF cc3d PSS	0.375	0.195	0.239	0.318	0.189	0.173	0.468	0.238	0.222
DF cc3d Consistency PSS	0.066	0.209	0.194	0.395	0.197	0.223	0.139	0.273	0.232
DF cc3d ISS + Augmix	0.378	0.262	0.273	0.372	0.362	0.315	0.359	0.410	0.356
DF cc3d Consistency ISS + Augmix	0.384	0.210	0.236	0.466	0.292	0.291	0.366	0.437	0.346

Table 6 shows that consistency loss can work in complement with AugMix, PSS and ISS while showing improvements across various metrics with little trade-off. It also improved AugMix performance for $N = 3$ where it failed previously by a substantial margin. The inclusion of self-interpreted learning across all tested baseline methods demonstrates improvements across various metrics with little trade-off. In particular, the results show that self-interpreted learning improves either test-time robustness (T_9) or overall method performance (T_{8-9}) and improvement is particularly pronounced when combined with Augmix and SSL, suggesting that consistency mechanisms provide additional regularization that benefits model generalization.

7 Conclusion

In this work, we proposed the SaSi, a Self-augmented and Self-Interpreted framework for few-shot particle detection in cryo-ET data, incorporating methods like AugMix, self-supervised learning and self-interpreted learning to address the challenges posed by limited training data. We demonstrated that our approach could enhance particle localization while maintaining robustness across different few-shot setups by utilizing a combination of supervised and self-supervised learning strategies. The integration of cc3d post-processing further improves the reliability of the detection process by eliminating the need for hyperparameter tuning. We show that the core components of SaSi are compatible with the most popular model architectures like CNNs and ViT making the method more resilient. The results across various configurations confirm the effectiveness of SaSi, including real-world datasets, and establish the benchmark for few-shot particle picking of Cryo-ET particles.

Supporting information

S1 Fig. Subtomogram rotation and size adjustment. The image on the right is an image of subtomogram, obtained by applying rotation to subtomogram on left. When a rotation is applied, the orange region has to be filled with zero values. But, since the length of the diagonal is $\sqrt{2} \times x$, we can take a larger subtomogram of size $\sqrt{2} \times x$, rotate it and then crop it. This approach keeps maximal information without filling in zero values. In cases when the particle is located at the border and the size $\sqrt{2} \times x$ of a subtomogram is not possible, we choose the biggest subtomogram so that the amount of filling with zero values is decreased, to retain maximum information.

S2 Fig. Sample from Tomogram 0 of SHREC2021 dataset. Slices along the x-axis (left), y-axis (middle), z-axis (right)

S3 Fig. Sample from Tomogram from Real dataset. Slices along the x-axis (left), y-axis (middle), z-axis (right)

S4 Fig. In the Figure, Left Image is the Ground truth mask for the x-axis slice and the Right Image is the Ground truth mask on the input slice of the SHREC2021 tomogram

S5 Fig. In the Figure, Left Image is the Ground truth mask for the x-axis slice and the Right Image is the Ground truth mask on the input slice of the Real World tomogram

S6 Fig. Localization f1 scores on two SHREC tomograms, T_8 (red curve) and T_9 (cyan curve) for the baseline DF (cc3d) (left plot) and our improved approach DF (cc3d)+ISS (right plot) with the few-shot setting $N = 3$. For the baseline method (A), DF (cc3d), the curve goes up for tomogram 9 and down for tomogram 8. This fluctuation indicates that the model is not robust and does not generalize easily across different tomograms. In contrast (B), our approach of using DF (cc3d)+ISS is shown to work better on both the two tomograms, thus demonstrating its higher robustness and consistent effectiveness across diverse tomograms. Hence we state that T_9 is not enough to test model's robustness and report T_{8-9}

Acknowledgments

This work was also supported in part by U.S. NIH grants R01GM134020 and P41GM103712, NSF grants DBI-1949629, DBI-2238093, IIS-2007595, IIS-2211597, and MCB-2205148, Oracle for Research, and AMD HPC Fund.

References

1. Irobalieva RN, Martins B, Medalia O. Cellular structural biology as revealed by cryo-electron tomography. *Journal of cell science*. 2016;129(3):469–476.
2. Yahav T, Maimon T, Grossman E, Dahan I, Medalia O. Cryo-electron tomography: gaining insight into cellular processes by structural approaches. *Current opinion in structural biology*. 2011;21(5):670–677.
3. Liu C, Mendonça L, Yang Y, Gao Y, Shen C, Liu J, et al. The architecture of inactivated SARS-CoV-2 with postfusion spikes revealed by cryo-EM and cryo-ET. *Structure*. 2020;28(11):1218–1224.
4. Zeng X, Xu M. Gum-net: Unsupervised geometric matching for fast and accurate 3d subtomogram image alignment and averaging. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; 2020. p. 4073–4084.
5. Xu M, Singla J, Tocheva EI, Chang YW, Stevens RC, Jensen GJ, et al. De novo structural pattern mining in cellular electron cryotomograms. *Structure*. 2019;27(4):679–691.

6. Koning RI. Cryo-electron tomography of cellular microtubules. *Methods in cell biology*. 2010;97:455–473.
7. Briggs JA. Structural biology in situ—the potential of subtomogram averaging. *Current opinion in structural biology*. 2013;23(2):261–267.
8. Moebel E, Martinez-Sanchez A, Lamm L, Righetto RD, Wietrzynski W, Albert S, et al. Deep learning improves macromolecule identification in 3D cellular cryo-electron tomograms. *Nature methods*. 2021;18(11):1386–1394.
9. Gubins I, Chaillet ML, van Der Schot G, Veltkamp RC, Förster F, Hao Y, et al. SHREC 2020: Classification in cryo-electron tomograms. *Computers & Graphics*. 2020;91:279–289.
10. Liu G, Niu T, Qiu M, Zhu Y, Sun F, Yang G. DeepETPicker: Fast and accurate 3D particle picking for cryo-electron tomography using weakly supervised deep learning. *Nature Communications*. 2024;15(1):2090.
11. Hendrycks D, Mu N, Cubuk ED, Zoph B, Gilmer J, Lakshminarayanan B. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*. 2019;.
12. Pari J, Shafiullah NM, Arunachalam SP, Pinto L. The surprising effectiveness of representation learning for visual imitation. *arXiv preprint arXiv:2112.01511*. 2021;.
13. Doersch C, Gupta A, Efros AA. Unsupervised visual representation learning by context prediction. In: *Proceedings of the IEEE international conference on computer vision*; 2015. p. 1422–1430.
14. Noroozi M, Vinjimoor A, Favaro P, Pirsiavash H. Boosting self-supervised learning via knowledge transfer. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2018. p. 9359–9367.
15. Frangakis AS, Böhm J, Förster F, Nickell S, Nicastro D, Typke D, et al. Identification of macromolecular complexes in cryoelectron tomograms of phantom cells. *Proceedings of the National Academy of Sciences*. 2002;99(22):14153–14158.
16. Voss N, Yoshioka C, Radermacher M, Potter C, Carragher B. DoG Picker and TiltPicker: software tools to facilitate particle selection in single particle electron microscopy. *Journal of structural biology*. 2009;166(2):205–213.
17. Chen Y, Hrabe T, Pfeiffer S, Pauly O, Mateus D, Navab N, et al. Detection and identification of macromolecular complexes in cryo-electron tomograms using support vector machines. In: *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*. IEEE; 2012. p. 1373–1376.
18. Che C, Lin R, Zeng X, Elmaaroufi K, Galeotti J, Xu M. Improved deep learning-based macromolecules structure classification from electron cryo-tomograms. *Machine vision and applications*. 2018;29:1227–1236.
19. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. 2014;.
20. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2016. p. 770–778.

21. Luengo I, Darrow MC, Spink MC, Sun Y, Dai W, He CY, et al. SuRVoS: super-region volume segmentation workbench. *Journal of Structural Biology*. 2017;198(1):43–53.
22. Chen M, Dai W, Sun SY, Jonasch D, He CY, Schmid MF, et al. Convolutional neural networks for automated annotation of cellular cryo-electron tomograms. *Nature methods*. 2017;14(10):983–985.
23. Li R, Zeng X, Sigmund SE, Lin R, Zhou B, Liu C, et al. Automatic localization and identification of mitochondria in cellular electron cryo-tomography using faster-RCNN. *BMC bioinformatics*. 2019;20:75–85.
24. Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*. 2016;39(6):1137–1149.
25. Li R, Yu L, Zhou B, Zeng X, Wang Z, Yang X, et al. Few-shot learning for classification of novel macromolecular structures in cryo-electron tomograms. *PLoS computational biology*. 2020;16(11):e1008227.
26. Silversmith W. cc3d: Connected components on multilabel 3D & 2D images.; 2021.
27. Gupta K, Ajanthan T, Hengel Avd, Gould S. Understanding and improving the role of projection head in self-supervised learning. *arXiv preprint arXiv:2212.11491*. 2022;.
28. Gubins I, Chaillet ML, White T, Bunyak F, Papoulias G, Gerolymatos S, et al. SHREC 2021: Classification in Cryo-electron Tomograms. In: Biasotti S, Dyke RM, Lai Y, Rosin PL, Veltkamp RC, editors. *Eurographics Workshop on 3D Object Retrieval*. The Eurographics Association; 2021.
29. Guo Q, Lehmer C, Martínez-Sánchez A, Rudack T, Beck F, Hartmann H, et al. In situ structure of neuronal C9orf72 poly-GA aggregates reveals proteasome recruitment. *Cell*. 2018;172(4):696–705.
30. Tong Z, Song Y, Wang J, Wang L. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*. 2022;35:10078–10093.