Weakly Supervised Learning for Joint Image Denoising and Protein Localization in Cryo-EM

Qinwen Huang[1] , **Ye Zhou**[1] , **Hsuan-Fu Liu**[2] , **Alberto Bartesaghi**[1,2,3]
[1] Department of Computer Science, Duke University
[2] Department of Biochemistry, Duke University School of Medicine
[3] Department of Electrical and Computer Engineering, Duke University
{qinwen.huang, ye.zhou678, hsuanfu.liu, alberto.bartesaghi}@duke.edu

Abstract

Deep learning-based object detection methods have shown promising results in various fields ranging from autonomous driving to video surveillance where input images have relatively high signal-to-noise ratios (SNR). On low SNR images such as biological electron microscopy (EM) data, however, the performance of these algorithms is significantly lower. Moreover, these data typically lack standardized annotations further complicating the training of detection algorithms. Accurate identification of proteins from EM images is a critical task, as the detected positions serve as inputs for the downstream 3D structure determination process. To overcome the low SNR and lack of annotations, we propose a joint weakly-supervised learning framework that simultaneously performs image denoising while detecting objects of interest. Our framework denoises images without the need of clean images and is able to detect particles of interest even when less than 0.5% of the data are labeled. We validate our approach on very low SNR cryo-EM datasets and show that our strategy outperforms existing state-of-the-art methods used in the cryo-EM field by a significant margin.

1 Introduction

Deep learning based-algorithms for object detection have witnessed a dramatic improvement over the past few years. Given sufficient amount of data, a network can easily learn to perform object detection or tracking. Most of these applications, however, rely on the availability of images with relatively high signal-to-noise ratios (SNRs). Cryo-EM is a popular technique for structure determination that can produce 3D reconstructions of proteins by back-projecting a large number of 2D protein projections taken from different orientations. This requires the detection of individual molecular images from electron micrographs, a process commonly known as *particle picking*. The low SNR is caused by the limited electron doses used during acquisition to prevent radiation damage, and makes the detection problem very challenging. Recent efforts to tackle particle picking have focused on either improving SNR using pre-processing strategies, followed by automatic or semi-automatic detection algorithms. Under very low SNR conditions, however, the performance of these algorithms is sub-optimal, thus limiting the quality of the downstream 3D reconstructions.

In this paper, we propose a framework that performs image denoising and particle segmentation and identification simultaneously. By enabling information sharing, we are able to improve the performance of both tasks. Our strategy does not require any information on clean images and learns to segment particles using only a small fraction of labeled particles. We validate our approach on three challenging datasets: one from single particle cryo-EM and two from cryo-ET. We show that under increasingly challenging SNR conditions, our proposed method is able to outperform existing approaches by a significant margin. To our knowledge, this is the first example of a method that

Machine Learning for Structural Biology Workshop, NeurIPS 2021.

is able to perform both image denoising and particle segmentation and detection at the same time without the need of ground-truth clean images for denoising and per-pixel labeling for segmentation.

2 Related Work

Recent developments in deep learning have led to breakthrough performance in tasks such as image enhancement, object detection and segmentation. In this section, we introduce relevant recent work, including denoising without clean images, semi-supervised object detection and weakly supervised segmentation, and multi-task learning (MTL). Blind image denoising based on convolutional neural networks include deep image prior [35], noise2noise [20], and noise2void [17]. In addition, a number of related methods have been proposed in the literature [19, 13, 29, 28, 18]. In the cryo-EM field, implementations of noise2noise have been successfully applied in [2, 25]. The majority of existing semi-supervised object detection methods are either built upon one-stage detectors [31, 23] or two-stage detectors [10, 32]. Most of the semi-supervised learning frameworks incorporate the use of unlabeled data through the use of consistency regularization [34], along with supervised learning of labeled data. In cryo-EM, several deep learning-based object detection methods [3, 37, 1, 36] have been used successfully for particle detection. Topaz is a semi-supervised particle picking method based on positive-unlabeled learning [3]. crYOLO [36] is a fully supervised picking method built upon YOLO [31]. Unlike fully supervised segmentation methods which use per-pixel supervision, weakly supervised segmentation usually uses weak labels such as bounding box annotations [7, 30, 24], scribbles [21], or image-level labels [16, 8, 27, 38]. MTL which leverages information shared between related tasks to improve the performance of the original task has been widely applied in various image processing tasks [33, 14, 9, 22]. Specific applications of MTL in object detection include Mask-RCNN [12], joint detection of objects while estimating distance between them [5], learning of segmentation maps as attention to aid detection [26], denoising, segmentation and detection through a cascaded network in a supervised manner [11], and denoising and segmentation of florescence microscopy images [4].

3 Proposed Method

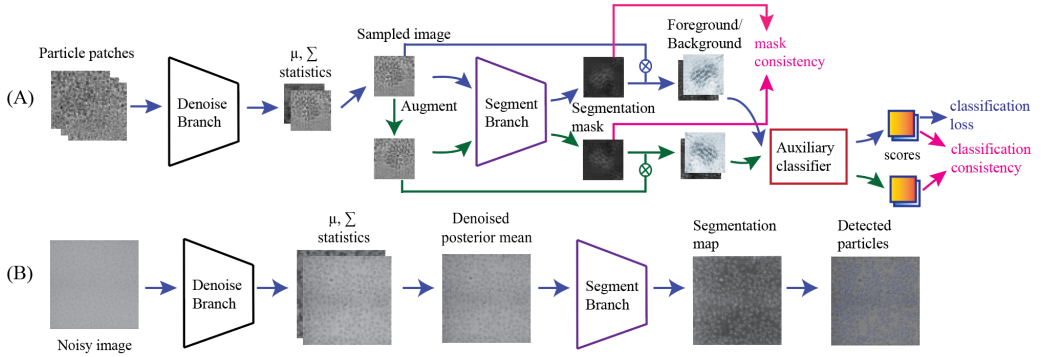

Figure 1: **Overall network architecture for joint image denoising and particle detection. A)** Network training: patches that contain particles are cropped from noisy images and serve as input to the network. **B)** Network evaluation: during inference, the entire image is fed into the network. The network outputs the denoised posterior mean and a segmentation mask of the entire image.

The overall architecture of our framework is shown in Figure 1. The denoising branch estimates the mean and covariance of the underlying noiseless data distribution from noisy inputs and feeds the approximated data statistics into the detection branch. The detection branch identifies particle locations by performing pixel-wise segmentation of the input and in return improves the denoising output as segmentation accuracy increases. Segmentation is performed in a weakly supervised fashion where only the center location of the particle is provided.

Joint denoising and detection Consider the prediction of the clean value x and its corresponding label l for a noisy pixel y. As pixels in an image are not independent, we assume that the clean value depends not only on the noisy measurement y, but also on the neighboring context Ω_y. We also assume that the label l of the pixel depends only on its clean value x, since the class of a pixel should not be affected by the noise. From this, performing denoising and detection jointly can be thought of as statistical inference on the probability distribution $p(x, l|y, \Omega_y)$ over the clean pixel value x and its label l, conditioned on the noisy input y and its context Ω_y. In cryo-EM applications, the noise is usually modeled as a Gaussian distribution. We therefore use this extra information to model $p(y|x)$ explicitly. We also model $p(x|\Omega_y)$ as a multivariate Gaussian $\mathcal{N}(\mu_x, \Sigma_x)$. Following this assumption, we train a network to map the context Ω_y to the mean μ_x and covariance Σ_x, and subsequently map the estimated statistics to the label l by maximizing the posterior likelihood under Equation (1).

$$\underbrace{p(x, l|y, \Omega_y)}_{\text{posterior}} \propto \underbrace{p(y|x)}_{\text{noise model}} \underbrace{p(x|\Omega_y)}_{\text{prior}} \underbrace{p(l|x)}_{\text{label model}} . \tag{1}$$

We therefore perform joint learning through maximization of this posterior distribution. Specifically, the denoising branch of our proposed framework corresponds to the first two terms in Equation (1) and the detection branch corresponds to the label model term. We describe details regarding the denoising term in the supplementary material.

Weakly-Supervised Detection Branch. Input to the detection branch is sampled from the prior Gaussian distribution $\mathcal{N}(\mu_x, \Sigma_x)$, where μ_x and Σ_x are the outputs from the denoising branch. In order to backpropagate the gradient, we adopt the re-parameterization trick proposed in [15]. The detection branch models $p(l|x)$. For a sampled input image I containing particles, the detection branch outputs a saliency map, M, which segments the image into two regions: particle (foreground) and background. To simulate binary hard thresholding while preserving differentiability, we add a modified sigmoid layer to the output saliency map: $\tilde{M} = \frac{1}{1+\exp[-C(M-t)]}$, where C is a constant and t is a threshold value. Segmentation is guided by an auxiliary classifier. If the image is segmented correctly, the classifier will be able to classify segmented images into their corresponding category. To do this, we multiply the sampled image I by $\tilde{M}$ and its complement to get two segmented images, F (foreground with particle) and B (background). Both are fed into the classifier g and the classifier outputs the probability of the input containing a particle. We adopt the hinge loss to train this classifier:

$$L_f = \min[0, -1 + g(F)], \quad L_b = \min[0, -1 - g(B)], \tag{2}$$

where L_f is the loss for the foreground and L_b is the loss for the background. We also incorporate consistency constraints to further regularize the detection branch. For each I, we randomly apply horizontal and vertical flipping to generate its augmented pair $A(I)$, where $A(\cdot)$ denotes the applied transformation. The augmented image is fed into the same network and the network outputs the segmentation map $M_{A(I)}$ and the classification probabilities $g(F_{A(I)}))$ and $g(B_{A(I)})$. Since the network is only able to access patch-level information, to impose further constraints on segmentation and classification, we assume equivariance in segmentation, i.e. $M_{A(I)} = A(M_I)$, and rotation invariance in classification, i.e. $g(F_{A(I)})) = g(F_I)$ and $g(B_{A(I)}) = g(B_I)$. Therefore, the consistency loss is defined as:

$$L_{cons} = \left\|M_{A(I)}, A(M_I)\right\|_2^2 + \left\|g(F_{A(I)})), g(F_I)\right\|_2^2 + \left\|g(B_{A(I)}), g(B_I)\right\|_2^2, \tag{3}$$

where $\|\cdot\|_2^2$ denotes the squared L2 norm error. While imposing equivariance is possible by modifying the convolution operation [6], we did not implement this approach in this work.

The final loss function is defined as: $L = \alpha L_{dn} + (1-\alpha)(L_f + L_b) + \lambda L_{cons}$, where α represents the assigned weights for each task and λ is the weight for the consistency regularization term. L_{dn} is used to denoise the input noisy images, L_f and L_b are used to guide the segmentation, and L_{cons} is used to further refine the segmentation result. Detailed implementation is discussed in the supplementary material.

4 Experiments

In this section, we evaluate our methods on real cryo-EM images of ribosomes, including one single-particle dataset and two cryo-ET datasets available from the Electron Microscopy Public Image

Archive (EMPIAR): EMPIAR-10304 and EMPIAR-10499. To simulate lower SNR conditions for the single-particle dataset, we use partial averages calculated from 10% of the total number of frames (6 frames). For two cryo-ET datasets, we only use the zero-degree tilt image. Tilt images have lower SNR than single-particle frame averages making the task of particle detection more challenging. A more detailed description of three datasets is included in the supplementary material.

We use PSNR values to evaluate denoising performance. We only calculate PSNR values for the single-particle ribosome dataset as we are able to treat full dose frame averages as ground truth. As full dosage micrographs can still be noisy, we apply a low-pass filter to remove potential high-frequency noise. The average PSNR values for the dataset are shown in Table 1. PSNR1 is calculated against the full dose micrographs and PSNR2 is calculated against low pass-filtered ones. Since low-pass filtering introduces image blurring, we expect the actual PSNR value to lie in between the two reported values. Our joint learning framework is able to achieve significantly higher PSNR values. For the cryo-ET datasets, we provide qualitative results in the supplementary material. Since Topaz requires pairs of noisy images, we compare the performance of our method against DivNoising [28], which is a self-supervised denoising method proven to perform well on fluorescence images.

Method	**PSNR1**	**PSNR2**
Topaz denoise	16.79 ± 2.12	19.49 ± 2.42
Ours (denoise only)	19.18 ± 3.14	19.33 ± 4.34
Ours (joint model)	$\mathbf{19.78 \pm 2.69}$	$\mathbf{21.62 \pm 3.12}$

Table 1: Denoising performance on single-particle cryo-EM images of ribosome complexes.

		Topaz Pick	*Topaz Denoise + Pick*	*crYOLO*	*Ours w/o Consistency*	*Ours w/ Consistency*
single-particle	Precision	0.615	0.656	0.501	0.681	**0.735**
	Recall	0.525	0.645	0.211	0.702	**0.825**
EMPIAR-10304	Precision	0.56	N/A	0.259	0.605	**0.618**
	Recall	0.448	N/A	0.146	0.443	**0.683**
EMPIAR-10499	Precision	0.289	N/A	0.102	0.216	**0.356**
	Recall	0.217	N/A	0.016	0.235	**0.326**

Table 2: Particle detection performance measured on single-particle cryo-EM and cryo-ET datasets.

Even though we perform particle detection through segmentation, our main focus is on the detection task, not on segmentation. We therefore calculate precision and recall values as evaluation criteria. True positive, false negative and false positive values are obtained by comparing against particles detected on full-dose images (single-particle ribosome dataset) and manually labeled particles (EMPIAR-10304 and EMPIAR-10499). To account for small variations in the detected particle centers, instead of looking at a single pixel, we also look at pixels located within a certain radius from the center. In Table 2 we compare the detection performance of our method against two of the most commonly used particle picking methods in cryo-EM: topaz [3] and crYOLO [36]. Qualitative visualization of our detection results is provided in the supplementary material.

5 Conclusion

In this paper, we present a novel joint training framework that performs image denoising and segmentation simultaneously without the need of noiseless images or per-pixel annotated datasets. We show that the complementary information shared between the two tasks allows us to improve the performance of both tasks, especially under extremely low SNR conditions. We validated our approach on real single-particle cryo-EM and cryo-ET datasets and showed that our model is able to outperform existing methods. Our future work will focus on handling more complex and diverse datasets, including datasets with particles of varying size corresponding to different species. We will also extend the applicability of our work to volumetric data to enable protein identification on 3D tomograms. We hope that our algorithm will facilitate structural analysis of challenging biomedical targets such as low molecular weight complexes imaged in their native environments using cryo-ET.

References

[1] A. Al-Azzawi, A. Ouadou, H. Max, Y. Duan, J. Tanner, and J. Cheng. Deepcryopicker: Fully automated deep neural network for single protein particle picking in cryo-em. *bioRxiv*, 2020.

[2] T. Bepler, K. Kelley, A. Noble, and B. Berger. Topaz-denoise: general deep denoising models for cryoem and cryoet. *Nature Communications*, 11, 2020.

[3] T. Bepler, A. Morin, A. Noble, J. Brasch, L. Shapiro, and B. Berger. Positive-unlabeled convolutional neural networks for particle picking in cryo-electron micrographs. *Nature Methods*, pages 1–8, 2019.

[4] T.-O. Buchholz, M. Prakash, A. Krull, and F. Jug. Denoiseg: Joint denoising and segmentation. In *ECCV Workshops*, 2020.

[5] Y. Chen, D. Zhao, L. Lv, and Q. Zhang. Multi-task learning for dangerous object detection in autonomous driving. *Inf. Sci.*, 432:559–571, 2018.

[6] T. Cohen and M. Welling. Group equivariant convolutional networks. In *ICML*, 2016.

[7] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1635–1643, 2015.

[8] T. Durand, T. Mordan, N. Thome, and M. Cord. Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5957–5966, 2017.

[9] T. Evgeniou and M. Pontil. Regularized multi–task learning. *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004.

[10] R. B. Girshick. Fast r-cnn. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015.

[11] I. Gubins and R. Veltkamp. Deeply cascaded u-net for multi-task image processing. *ArXiv*, abs/2005.00225, 2020.

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn, 2018.

[13] A. Hendriksen, D. M. Pelt, and K. J. Batenburg. Noise2inverse: Self-supervised deep convolutional denoising for linear inverse problems in imaging. *ArXiv*, abs/2001.11801, 2020.

[14] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018.

[15] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2014.

[16] A. Kolesnikov and C. H. Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. *ArXiv*, abs/1603.06098, 2016.

[17] A. Krull, T.-O. Buchholz, and F. Jug. Noise2void - learning denoising from single noisy images. pages 2124–2132, 06 2019.

[18] S. Laine, T. Karras, J. Lehtinen, and T. Aila. High-quality self-supervised deep image denoising. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[19] K. Lee and W.-K. Jeong. Noise2kernel: Adaptive self-supervised blind denoising using a dilated convolutional kernel architecture. 12 2020.

[20] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila. Noise2noise: Learning image restoration without clean data. 03 2018.

[21] D. Lin, J. Dai, J. Jia, K. He, and J. Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3159–3167, 2016.

[22] P. Liu, X. Qiu, and X. Huang. Adversarial multi-task learning for text classification. *ArXiv*, abs/1704.05742, 2017.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C.-Y. Fu, and A. Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016.

[24] Y. Liu, Q. lei Hui, Z. Peng, S. Gong, and D. Kong. Automatic ct segmentation from bounding box annotations using convolutional neural networks. *ArXiv*, abs/2105.14314, 2021.

[25] E. Palovcak, D. Asarnow, M. G. Campbell, Z. Yu, and Y. Cheng. Enhancing the signal-to-noise ratio and generating contrast for cryo-EM images with convolutional neural networks. *IUCrJ*, 7(6):1142–1150, Nov 2020.

[26] H. Perreault, G.-A. Bilodeau, N. Saunier, and M. H'eritier. Spotnet: Self-attention multi-task network for object detection. *2020 17th Conference on Computer and Robot Vision (CRV)*, pages 230–237, 2020.

[27] P. H. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1713–1721, 2015.

[28] M. Prakash, A. Krull, and F. Jug. Divnoising: Diversity denoising with fully convolutional variational autoencoders. *ArXiv*, abs/2006.06072, 2020.

[29] Y. Quan, M. Chen, T. Pang, and H. Ji. Self2self with dropout: Learning self-supervised denoising from single image. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1887–1895, 2020.

[30] M. Rajchl, M. J. Lee, O. Oktay, K. Kamnitsas, J. Passerat-Palmbach, W. Bai, B. Kainz, and D. Rueckert. Deepcut: Object segmentation from bounding box annotations using convolutional neural networks. *IEEE Transactions on Medical Imaging*, 36:674–683, 2017.

[31] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. pages 779–788, 06 2016.

[32] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015.

[33] S. Ruder. An overview of multi-task learning in deep neural networks. *ArXiv*, abs/1706.05098, 2017.

[34] K. Sohn, D. Berthelot, C. Li, Z. Zhang, N. Carlini, E. D. Cubuk, A. Kurakin, H. Zhang, and C. Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *ArXiv*, abs/2001.07685, 2020.

[35] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. *International Journal of Computer Vision*, 128, 07 2020.

[36] T. Wagner, F. Merino, M. Stabrin, T. Moriya, C. Antoni, A. Apelbaum, P. Hagel, O. Sitsel, T. Raisch, D. Prumbaum, D. Quentin, D. Roderer, S. Tacke, B. Siebolds, E. Schubert, T. Shaikh, P. Lill, C. Gatsogiannis, and S. Raunser. Sphire-cryolo is a fast and accurate fully automated particle picker for cryo-em. *Communications Biology*, 2, 2019.

[37] F. Wang, H. Gong, G. Liu, M. Li, C. Yan, T. Xia, X. Li, and J. Zeng. Deeppicker: A deep learning approach for fully automated particle picking in cryo-em. *Journal of Structural Biology*, 195(3):325–336, 2016.

[38] Y. Wang, J. Zhang, M. Kan, S. Shan, and X. Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12272–12281, 2020.

Supplementary Materials

Self-Supervised Bayesian Denoising

The denoising branch is based upon a blindspot convolutional neural network proposed in [1] that learns the underlying clean signal by maximizing the posterior likelihood in Equation (2), which includes maximization of the observed noise model $p(y|x)$ subject to prior belief $p(x|\Omega_y) \sim \mathcal{N}(\mu_x, \Sigma_x)$ and its label (we discuss segmentation in the following section). Assuming that y is corrupted by zero-mean Gaussian noise, we have $p(y|x) \sim \mathcal{N}(\mu_y, \Sigma_y)$, where $\mu_y = \mu_x$ and $\Sigma_y = \Sigma_x + \sigma^2\mathbf{I}$, with σ^2 being the noise variance. Since maximizing $p(y|x)$ is equivalent to minimizing its negative log-likelihood, we can write the denoise loss as:

$$L_{dn} = \frac{1}{2}[(y-\mu_y)^T\Sigma_y^{-1}(y-\mu_y)] + \frac{1}{2}\log|\Sigma_y| + C, \tag{1}$$

where C is a constant that can be discarded. Since σ^2 is unknown, an auxiliary network is used to estimate its value. The output of the denoising branch consists of μ_x and Σ_x. The posterior distribution for x is calculated by multiplying the noise model and the prior Gaussian distribution parameterized by μ_x and Σ_x, which also follows a Gaussian distribution with $E(p(x|y,\Omega_y) = (\Sigma_x^{-1} + \sigma^{-2}\mathbf{I})^{-1}(\Sigma_x^{-1}\mu_x + \sigma^{-2}y)$. The mean of the posterior distribution is the final denoised output.

Implementation details

Here we provide details about the architecture of the different components of our network and its training procedure. We adopt a U-Net based structure for the denoising branch that uses a shifted convolutional layer instead of a normal convolutional layer. The segmentation module of our detection branch is composed of four convolutional layers with two upsampling layers in between and a final max-pooling layer. The auxiliary classifier is composed of three residual blocks and a final convolutional layer. The estimation of the noise variance is also performed using a normal U-Net architecture. During training, only patches that contain particles of interest are fed into the framework. We use 64×64 for the patch size. The detection branch outputs a segmentation map. Foreground and background images are generated based on the segmentation map and the classifier outputs the probability of a foreground/background image containing a particle. For the modified sigmoid layer in Equation (4), we use $C = 7$ and $t = 0.5$. During the inference process, the entire image is fed into the framework. The auxiliary classifier is removed and the segmentation map of the entire image is used to identify particle locations. The center location for each particle is obtained by applying non-max suppression to the segmentation map. The model is trained on a single NVIDIA Tesla V100 GPU with 32G of RAM. We use a batch size of 32 and a cosine decay learning rate for the scheduler with initial learning rate of 0.001. We adopt the ADAM optimizer and use $\alpha = 0.75$ and $\lambda = 0.1$ for all experiments. Training with $240,000$ iterations takes less than an hour. Inference on a single image takes less than a second.

Training details

Here we provide additional training details regarding our network. All networks use a single input channel and 48 feature channels in all encode blocks of U-Net. With feature concatenation, all U-Net decode blocks have 96 feature channels. For the classifier, each residual block has 32, 64 and 128 feature channels respectively. Input to the modified sigmoid layer (soft segmentation map) is normalized to $[0, 1]$.

How many training iterations are needed? In general, our framework converges within 300,000 iterations. Due to limited number of training samples, we found out that the model can overfit if the training iterations are too large. Training time takes around 30 minutes to an hour based on the number of iterations used. Training of the model does not require very high computational power. The network takes around 7GB RAM on a NVIDIA TESLA V100 GPU. It can also work on other NVIDIA GPUs such as 1080Ti.

How to choose α and λ in loss function? We experimentally chose $\alpha = 0.75$ and $\lambda = 0.1$ for our loss function. However, the choice of α can be anywhere between 0.7 to 0.9. It slightly affects the convergence time and the overall performance. However, results usually vary within $1 - 2\%$. For λ,

we found out $\lambda = 0.1$ works the best experimentally. In general, our network is able to perform well without too much fine-tuning. Below we show the mean and standard deviation of precision/recall values calculated using $\alpha = 0.7, 0.75, 0.8, 0.85, 0.9$.

	single particle ribosome	**EMPAIR-10304**	**EMPIAR-10499**
Precision	0.72 ± 0.018	0.61 ± 0.014	0.34 ± 0.021
Recall	0.82 ± 0.013	0.67 ± 0.015	0.31 ± 0.023

Table 1: Detection mean and standard deviation measured on three different datasets with varying alpha.

How does the radius selection in precision-recall measurement affect the measurement results? When we use a radius of 3, we see an around 5% decrease in the precision-recall scores. This means that there is still room for improvements in the detection of center coordinates. This decrease mainly comes from the fact that we are inferring center coordinates from segmentation map. However, since manual labeling does not fully guarantee the precise localization of center coordinates, we believe a 3-6 pixels of deviation is acceptable.

Dataset Description

Single-particle cryo-EM ribosome dataset. This dataset contains 1000 movies, each cropped to size 4096×4096, with defocus values ranging from $0.8\,\mu\text{m}$ to $3.0\,\mu\text{m}$. The pixel size is 1.08 Å. Each movie contains 60 frames. Since the ribosome is a relatively large particle, frame averages of this dataset have relatively high SNR compared to most other proteins. We therefore treat the average of all 60 frames as the ground-truth images. Similarly, particles picked on these frame averages are treated as the ground-truth annotations. To simulate lower SNR conditions, such as those observed for lower molecular weight proteins and lower defocus datasets, we use partial averages calculated from 10% of the total number of frames (6 frames). Among these 1000 low SNR partial frame averages, 16 images are used as the training set and the remaining ones are used for testing. All images are further down-sampled by a factor of 8 to size 512×512. Training images are only partially labeled, with 15 to 25 particles identified on each. The entire training set is composed of 500 labeled particles from 16 images, which accounts for around 0.04% of the total number of particles in the entire dataset.

EMPIAR-10304. This dataset consists of 12 tilt-series from a sample of purified ribosomes. Each tilt series is composed of 41 projection images ranging from -60 degree to + 60 degree. A single tilt image has size of 4096×5760, with a pixel size of 2.1 Å. We evaluate our framework on the zero-degree tilt images of each tilt series. From the total of 12 zero-degree tilt images, 3 images are used for training and the remaining ones are the testing set. We also down-sampled each image by a factor of 8 to size 512×720. Training images are partially labeled as well, with 40 to 60 particles identified on each. The entire training set is composed of 200 labeled particles, which accounts for around 4% of all particles in the entire dataset. We use manually labeled particle locations on zero-tilt images as ground truth.

EMPIAR 10499. This is a cryo-ET dataset of ribosomes imaged within cells. This dataset is challenging because particles are observed within a crowded context that includes cell membranes and other sub-cellular components. We use a subset of the entire dataset which consists of 65 tilt series. Each tilt series is composed of 41 projection images ranging from -60 degree to +60 degree. A single tilt image has size of 3838×3710, with pixel size of 1.7 Å. We also evaluate our framework on the zero-degree tilt images. Of the 65 tilt images, 7 are used for training. Each image is down-sampled by a factor of 8 to 480×464. Partial-labeling is performed and a total of 90 particles are identified, which accounts for less than 1% of all particles in the dataset. We use manually labeled particle locations on zero-tilt images as ground truth.

References

[1] S. Laine, T. Karras, J. Lehtinen, and T. Aila. High-quality self-supervised deep image denoising. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Visualization of Denoising Results

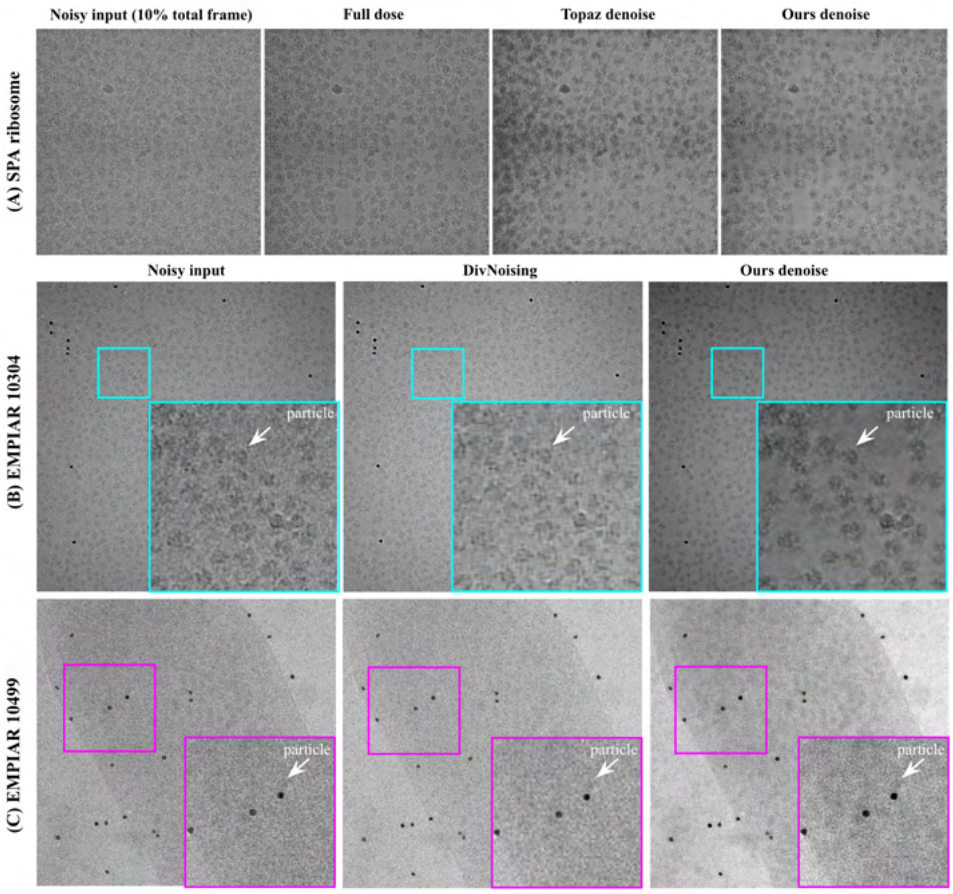

Figure 1: **Visualization of improvement in denoising performance on three real cryo-EM datasets compared to Topaz denoise and DivNoising.** (A). single particle ribosome image denoised using topaz and our method. (B). EMPIAR 10304 image denoised using DivNoising and our method. (C). EMPIAR 10499 denoised using DivNoising and our method. Note: Since EMPIAR-10304 and EMPIAR-10499 only have a single noisy image, Topaz denoise cannot be applied as the method is based on Noise2Noise and requires pairs of noisy images. We therefore use DivNoising for comparison. For the SPA ribosome dataset, we show a 10% image fraction and the full dose image for comparison. For EMPIAR-10304 and EMPIAR-10499, we provide zoomed in views of the noisy input and the denoised output for better visualization. Our method helps to visualize particles better, especially for zoomed in views of 10304 and 10499.

Visualization of Detection Results

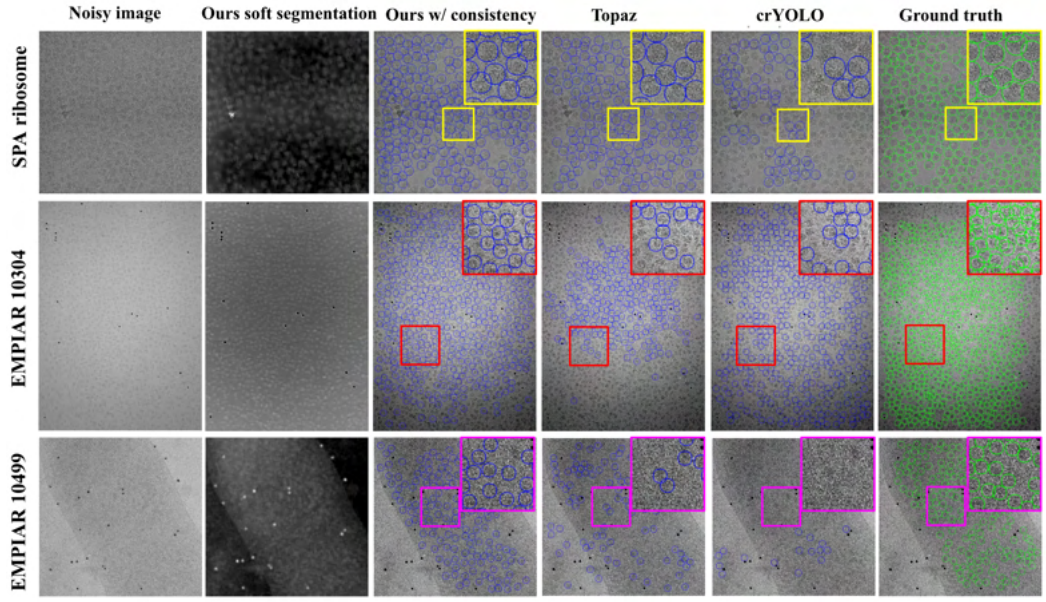

Figure 2: **Visualization of particle detection results on the three datasets.** Detected particles are circled in blue. The first row is an example of detection results for the single-particle dataset. The second row is the result for EMPIAR-10304 and the last row is for EMPIAR-10499. We show both, the soft segmentation map (2nd column) and the detection output based on this segmentation (3rd column), obtained using our proposed method. We also show results obtained using Topaz picking (without denoise), and crYOLO in the next two columns, respectively. The last column shows the ground truth. Note that for the single-particle dataset, ground truth was obtained using the Topaz particle picking method on the full dose micrograph. Ground truth for EMPIAR-10304 and 10499 were obtained using manual picking.

More Qualitative Visualizations

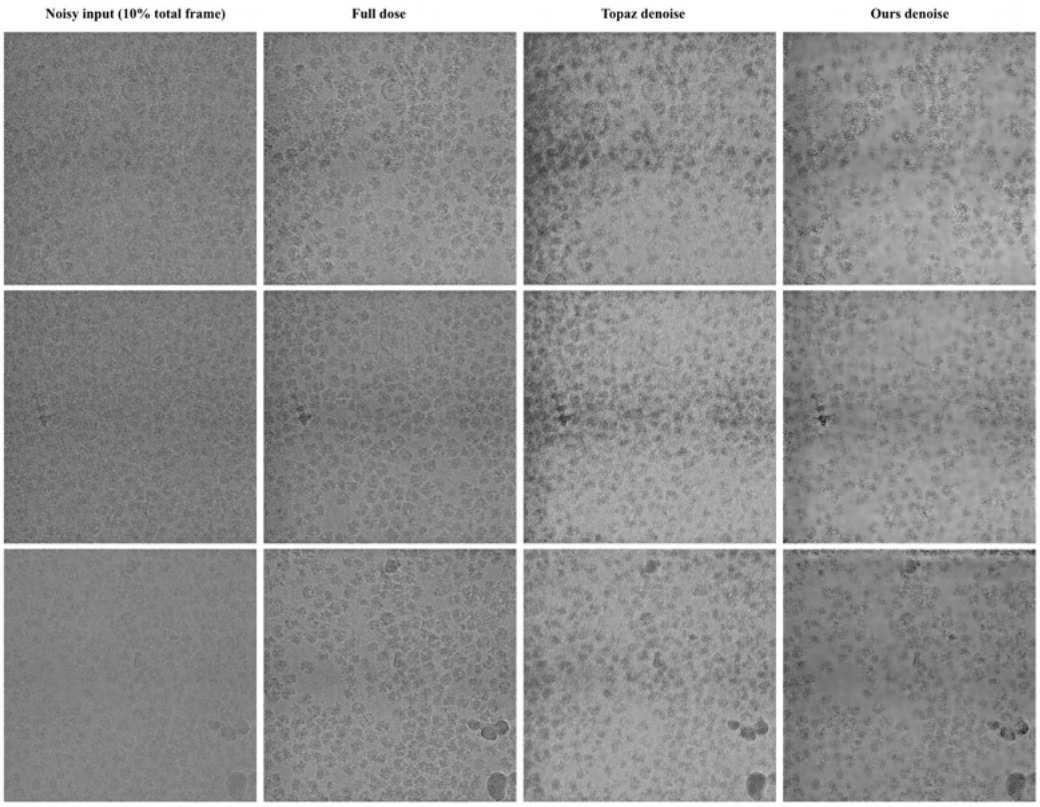

Figure 3: **Additional qualitative results for image denoising of single particle ribosome dataset.** Here we are showing the 10% dose fraction noisy input(1st column), full dose image (2nd column), Topaz denoise (3rd column) and ours denoise (4th column).

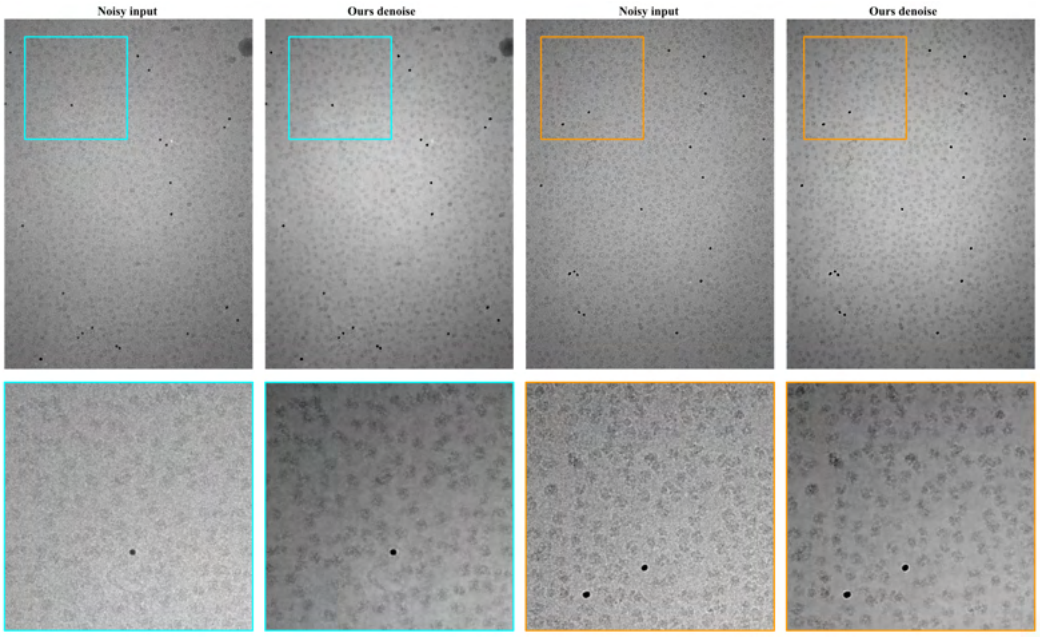

Figure 4: **Additional qualitative results for image denoising of EMPIAR 10304.** Here we are showing the full image. The bottom row is the zoomed-in view of the selected area. In zoomed in views, we can better visualize distinctions between background and particles.

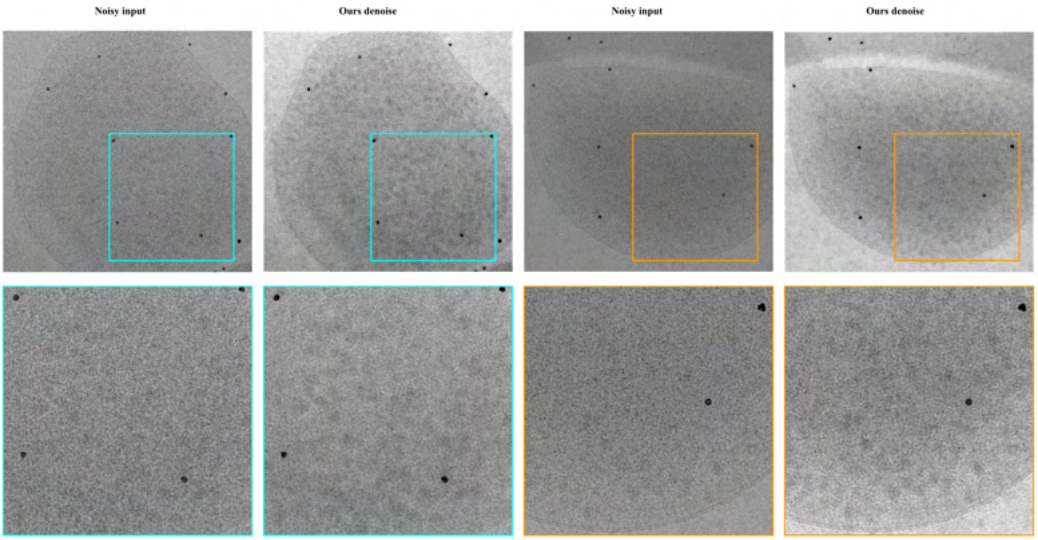

Figure 5: **Additional qualitative results for image denoising of EMPIAR 10499.** Here we are showing the full image. The bottom row is the zoomed-in view of the selected area. In zoomed in views, we can better visualize distinctions between background and particles.

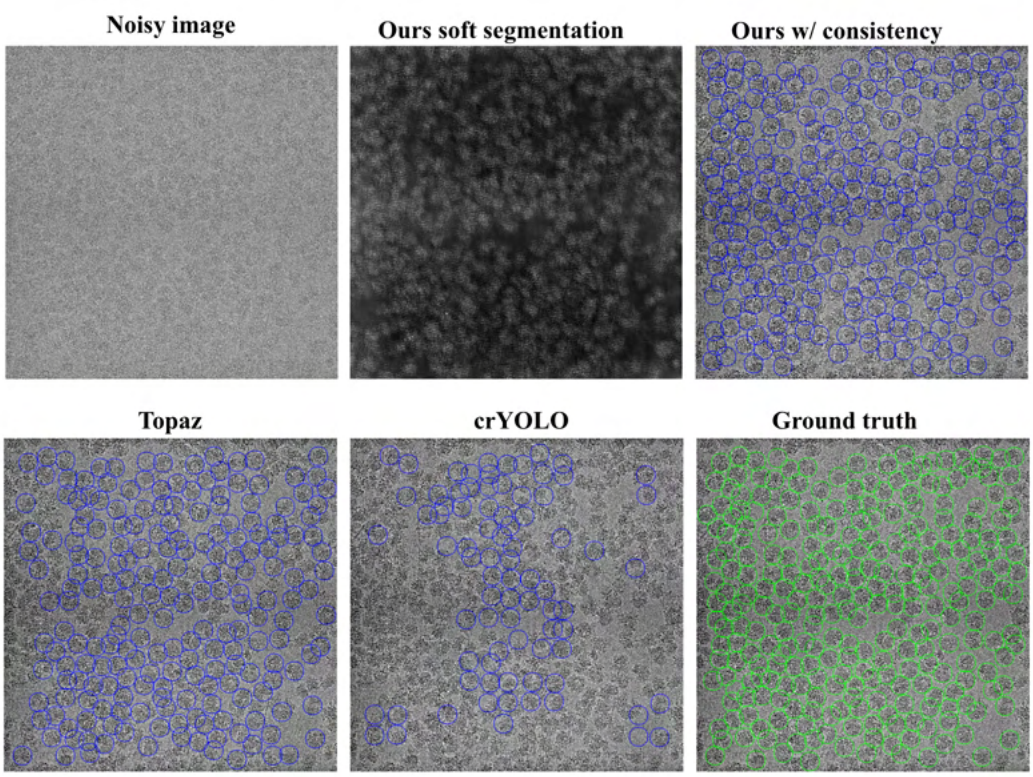

Figure 6: **Additional qualitative results for particle detection of single particle ribosome dataset.** Here we are showing the segmentation map and our detection output. We also show picking results using Topaz and CrYOLO. Ground truth is obtained by running Topaz and full-dose images.

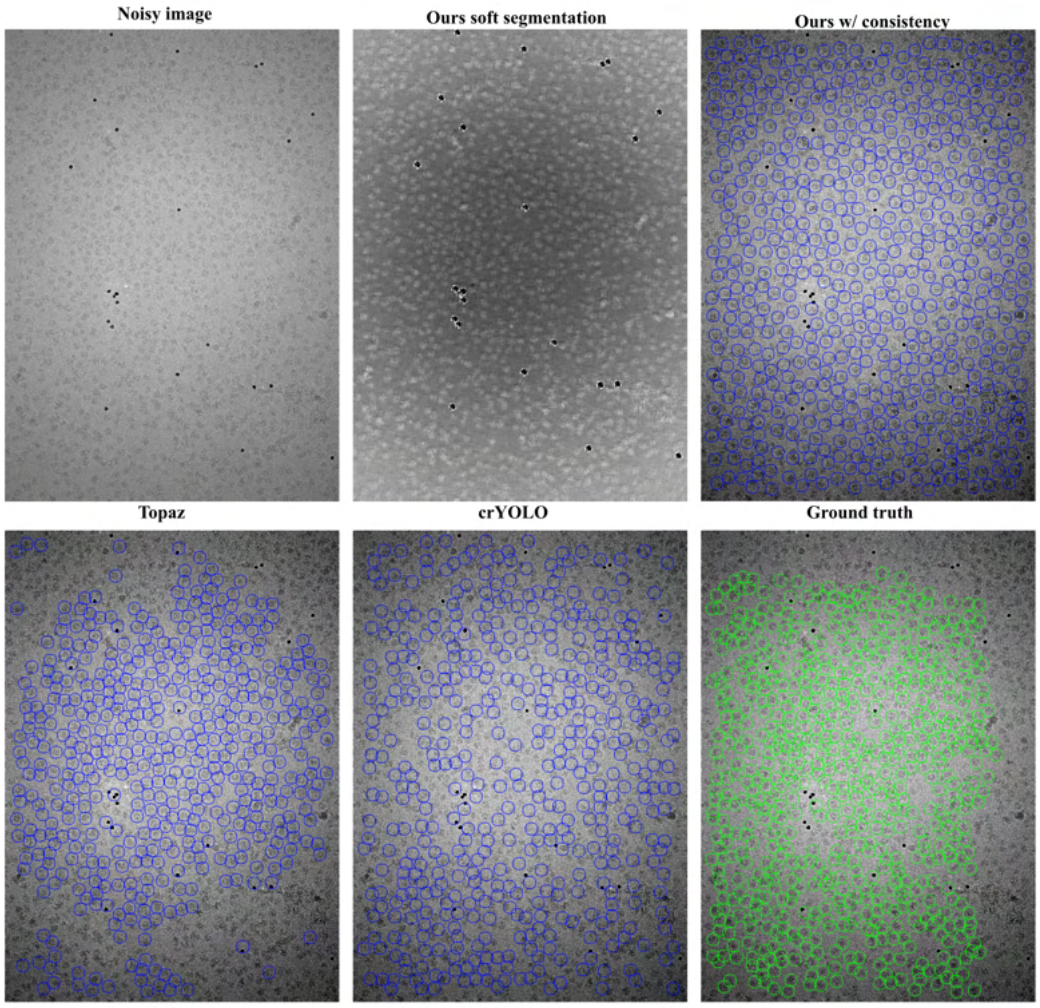

Figure 7: **Additional qualitative results for particle detection of EMPIAR 10304.** Here we are showing the full image with detection result. Note our method is able to avoid contamination areas and gold beads. Note: some of the particles near edges of the image are not getting labeled in manual picking. Our method is able to detect these particles near boundaries.

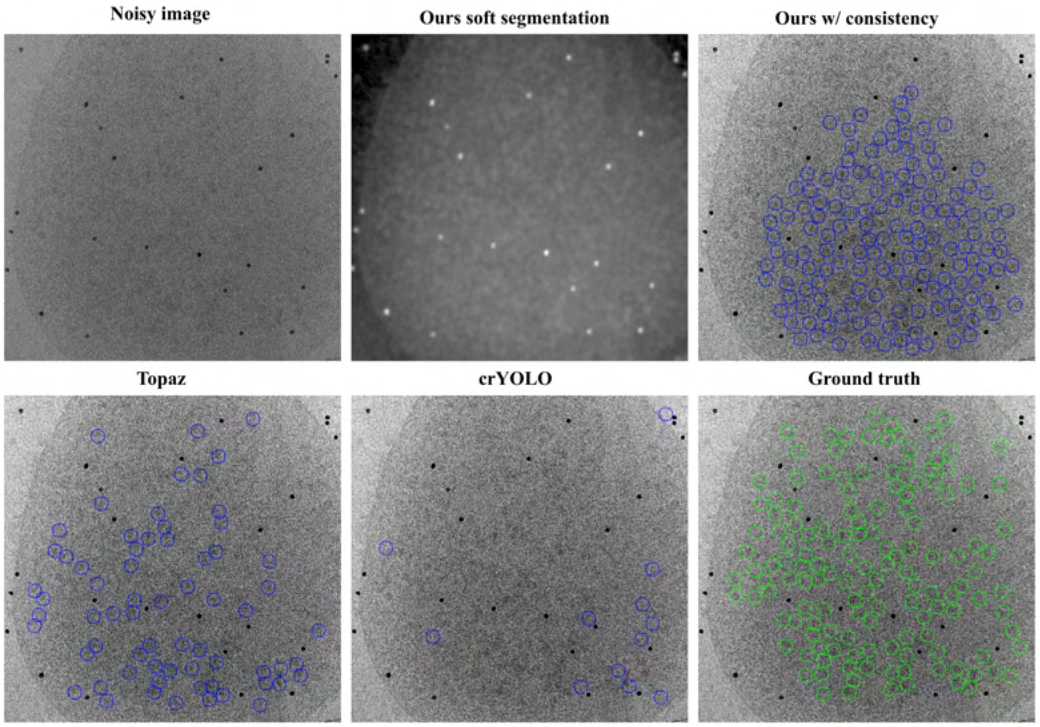

Figure 8: **Additional qualitative results for particle detection of EMPIAR 10499.** Here we are show the soft segmentation map and particle detection outputs using different methods. Our method is able to identify more particles compared to the other two methods.