Deep-learning based automatic segmentation of vesicles in cryo-electron tomograms

This manuscript ([permalink](#)) was automatically generated from [elatella/deep-prepyto-paper@efaef97](#) on July 19, 2022.

Authors

- **Amin Khosrozadeh**[*]
  ⬤ [XXXX-XXXX-XXXX-XXXX](#) · ◯ [ameen-khosrowzadeh](#)
  Institute of Anatomy, University of Bern, Bern, Switzerland; Graduate School for Cellular and Biomedical Sciences, University of Bern · Funded by Grant XXXXXXXX

- **Raphaela Seeger**[*]
  ⬤ [XXXX-XXXX-XXXX-XXXX](#) · ◯ [elatella](#)
  Institute of Anatomy, University of Bern, Bern, Switzerland; Graduate School for Cellular and Biomedical Sciences, University of Bern

- **Julika Radecke**
  ⬤ [0000-0002-5815-5537](#) · ◯ [julikaradecke](#)
  Institute of Anatomy, University of Bern, Bern, Switzerland; Department of Neuroscience, Faculty of Health and Medical Science , 2200 Copenhagen N, University of Copenhagen, Copenhagen, Denmark; Diamond Light Source Ltd, Didcot, Oxfordshire, United Kingdom

- **Guillaume Witz**
  ⬤ [0000-0003-1562-4265](#) · ◯ [guiwitz](#)
  Science IT Service, University of Bern, Bern, Switzerland; Microscopy Imaging Center, University of Bern, Bern, Switzerland

- **Jakob B. Sørensen**
  ⬤ [0000-0001-5465-3769](#) · ◯ [JBSorensen](#)
  Department of Neuroscience, University of Copenhagen, Blegdamsvej 3B, 2200 Copenhagen N, Denmark · Funded by Novo Nordisk Fonden, NNF17OC0028516.; Carlsbergfondet, CF17-0875; Independent Research Fond Denmark, 8020-00228A; Lundbeckfonden, R277-2018-802

- **Benoît Zuber**[✉]
  ⬤ [0000-0001-7725-5579](#) · ◯ [aseedb](#)
  Institute of Anatomy, University of Bern, Bern, Switzerland · Funded by Swiss National Science Foundation, 179520; ERA-NET NEURON, NEURON-119

✉ Address correspondence to benoit.zuber@ana.unibe.ch.

* These authors contributed equally.

Abstract

Cryo-electron Tomography (Cryo-ET) has the potential to reveal cell structure down to atomic resolution. Nevertheless, cellular cryo-ET data is often highly complex and visualization, as well as quantification, of subcellular structures require image segmentation. Due to a relatively high level of noise and to anisotropic resolution in cryo-ET data, automatic segmentation based on classical computer vision approaches usually does not perform satisfyingly. For this reason, cryo-ET researchers have mostly performed manual segmentation.

Communication between neurons rely on neurotransmitter-filled synaptic vesicle (SV) exocytosis. Recruitment of SVs to the plasma membrane is an important means of regulating exocytosis and is influenced by interactions between SVs. Cryo-ET study of the spatial organization of SVs and of their interconnections allows a better understanding of the mechanisms of exocytosis regulation. To obtain a faithful representation of SV connectivity state, an absolutely vital prerequisite is an extremely accurate SV segmentation. Hundreds to thousands of SVs are present in a typical synapse, and their manual segmentation is a burden. Typically accurately segmenting all SVs in one synapse takes between 3 to 8 days. This segmentation process has been widely recognized as a bottleneck by the community.

Several attempts to automate vesicle segmentation by classical computer vision or machine learning algorithms have not yielded very robust results. We addressed this problem by designing a workflow consisting of a U-Net convolutional network followed by post-processing steps. This combination yields highly accurate results. Furthermore, we provide an interactive tool for accurately segmenting spherical vesicles in a fraction of the time required by available manual segmentation methods. This tool can be used to segment vesicles that were missed by the fully automatic procedure or to quickly segment a handful of vesicles, while bypassing the fully automatic procedure. Our pipeline can in principle be used to segment any spherical vesicle in any cell type as well as extracellular vesicles.

Introduction

The fine architecture of cells can be investigated by cryo-electron tomography (cryo-ET) [1]. Cellular structures are preserved down to the atomic scale through vitrification and observation of the samples in a fully hydrated state. When a macromolecule is present in a sufficient number of copies in the cells imaged by cryo-ET, it is possible to obtain its atomic structure in situ using subtomogram averaging [2,3]. Cellular cryo-ET datasets are usually extremely complex, making them difficult to analyze. This is aggravated by the sensitivity of biological samples to electron radiation, which limits the signal-to-noise ratio in cryo-ET datasets [4]. Tomographic reconstructions are generated from a series of images of the sample acquired at different viewing angles. The geometry of the samples prevents acquisition at certain angles, resulting in anisotropic spatial coverage. The resolution in the directions close to the axis of the electron beam incident on the untilted sample is strongly reduced. This effect, commonly referred to as the missing-wedge artifact, further complicates data analysis. In particular, organelles fully bounded by a membrane appear to have holes at their top and bottom (relative to the electron beam axis) [4].

The synapse is the functional cellular contact at which information is transmitted from a neuron to another. The former neuron is called presynaptic and the latter is postsynaptic. In most cases, the signal is transmitted by the release of neurotransmitters into the intercellular space. Neurotransmitters are stored in SVs and are released following the fusion of a vesicle with the presynaptic plasma membrane. A synapse contains hundreds of SVs and their mobility and recruitability for neurotransmitter release depends on inter-vesicle interactions through so-called connector structures [5]. The characterization of these interactions can be performed automatically

with the pyto software, which implements a hierarchical connectivity approach to detect and annotate connectors [6]. For accurate connector segmentation, an exceptionally precise segmentation of SVs is prerequisite. To date, this SV segmentation has been achieved manually, but given the massive number of SVs per dataset, it is an extremely time-consuming process. Typically, one person spends 3 to 8 working days to segment a single dataset. Attempts to perform this task automatically based on classical computer vision algorithms have not yielded sufficiently accurate performance [7]. To alleviate this situation, we considered applying deep learning methods.

Convolutional neural networks (CNN) have been successfully employed to segment cryo-ET data [8]. Although entirely satisfying for visualization purposes, this approach has not met the requirements of pyto. A recent publication described accurate SV segmentation of transmission electron microscopy images using CNN, but it is limited to 2-dimensional images of resin-embedded synapses [9]. For our use-case, a common issue is that the input data consists of 2-dimensional images. In the first study, cryo-ET data are decomposed in individual 2-dimensional slices, which are handed as seperate input to the CNN. The independent output prediction images are re-assembled in a 3-dimensional stack.[8] As discussed above, membranes oriented approximately parallel to the plane of the 2-dimensional tomographic images are not resolved. In the absence of contextual knowledge of the other 2-dimensional images, the CNN fails to segment these regions of the vesicles. Hence, spherical vesicles appear open, whereas we expect closed spherical objects. To overcome this limitation, we used a U-Net CNN that takes 3-dimensional images as input. U-Net architecture has been introduced in 2015 by Ronenberger et al. [10]. It consists of a contracting path, typical of CNN, and a symmetric expanding path. At each expansion step, the correspondingly cropped feature map of the contracting path is concatenated. The contracting path captures context, while the expanding path coupled with concatenation enables precise localization. The U-Net architecture was developed to achieve a fast and accurate segmentation of biomedical two-dimensional images, with the requirement of only a small fraction of training data in comparison to previous CNNs. *might be redundant* Arranging the sequence of convolution layers in the contraction path along with skip connection and concatenating into the expansion path bring significant privilege for medical and semantic segmentation. Concurrent with the appearance of 3D convolutional neural networks (CNN) three-dimensional form of U-Net was also developed for volumetric image analysis.

It was then extended to segment 3-dimensional biomedical images (3D U-Net) [11]. Weigert et al. [12] implemented a U-Net for content-aware restoration (CARE) of 3-dimensional fluorescence microscopy datasets. They showed that it can restore information from anisotropic and very noisy datasets.

We implemented a 3D U-Net based on CARE building blocks and trained it with manually segmented datasets. This method provided good accuracy and was only slightly affected by the missing wedge artifact. Nevertheless, it was not quite sufficient for our downstream pyto analysis. Hence, we developed a post-processing method, which transforms the segmented objects into spheres and refines their radius and center location. This leads to a substantial accuracy improvement, which are reflected in better pyto performance. Additionally, we designed a multivariate ranking procedure, highlighting possibly wrongly segmented SVs. We also introduce a semi-automatic method to very quickly fix wrongly segmented and missed SVs.

Although our set of procedures was developed with the use case of SV segmentation in mind, it can be used to segment any other types of biological spherical vesicles, such as transport vesicles, secretory vesicles, endocytic vesicles, and extracellular vesicles.

Results

Cellular cryo-electron tomography is an upcoming field with a multitude of possible applications. The manual segmentation of the cellular features is a major bottleneck of this method. When segmenting

cryo-electron tomograms from presynaptic terminals, the manual segmentation of synaptic vesicles is one of the most time-intensive steps. Synaptic vesicles constitute a large, homogeneous group, constituting a large training set for deep learning applications. Therefore, we decided to initially develop the automatic segmentation for synaptic vesicles. The used tomograms were previously manually segmented with IMOD, these manual segmentations were further treated as the ground truth [13]. `\_maybe add this somewhere else: In a next step, filaments connecting the synaptic vesicles with each other (connectors) and to the active zone (AZ) were automatically segmented with the algorithm application Pyto [@doi:10.1016/j.jsb.2016.10.004].*`

The U-Net neural network was used to train its mask prediction on a training set of 9 tomograms containing untreated synaptosomes. The learning progress was tracked by calculating the Dice coefficient and the loss value after each training epoch (Figure 1). The dice value for the training dataset started at a value of ~0.25 and rose to a value of over 0.9 after the initial 50 epoches. The loss value of the training dataset had an initial value of over 0.55 and declined to values below 0.05 after the initial 50 epochs and further striving towards 0 in the following depicted epochs. `\_Vali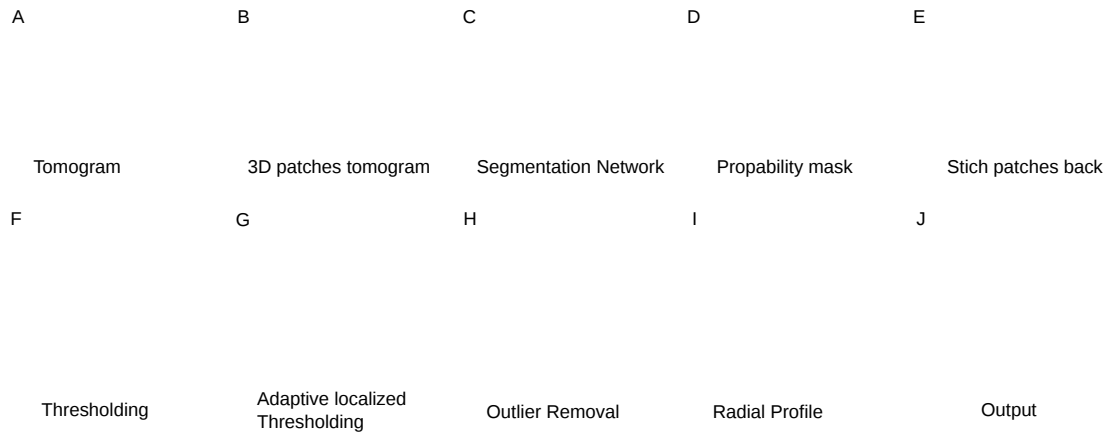dation Dice and loss from treated synaptosome dataset?*` The validation dataset showed much more fluctuations during both validation and loss progression. The dice value for the validation dataset started at a value of ~0.27 and rose to an average value of over 0.75 after the initial 50 epoches. The loss value of the validation dataset had an initial value 1 and declined to values below 0.3 after the initial 50 epochs.
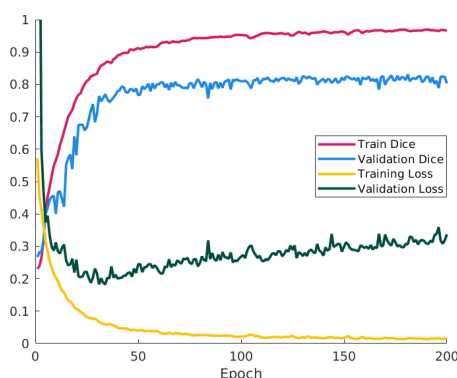
Figure 1: Dice coefficient and loss value for training and validation set. `\_legend in figure should say "Training Dice"*`

`\_---->add figures of local measurements such as diameter or center error as an extra figure, why else listed in M&M??"*`

After the neural network is trained to recognize the synaptic vesicles with a sufficient probability, the trained U-Net was implemented into a pipeline. The pipeline for automatic segmentation of vesicles consists of two major parts: the neural network consisting of a U-Net neural network, and the post-processing steps refining the labels generated by the U-Net (Figure 2). The three batches of tomograms (synaptosome control, synaptosome treatment and neuron) were each handed to the pipeline.

A	B	C	D	E
Tomogram	3D patches tomogram	Segmentation Network	Propability mask	Stich patches back

| F	G	H	I	J
Thresholding	Adaptive localized Thresholding	Outlier Removal	Radial Profile	Output

Figure 2: Pipeline of automatic segmentation. a) tomograms b) patchify the tomograms into 3D patches c) Segmentation Network/ trained U-Net d) probability masks e) stitching patches back together f) thresholding g) adaptive localized thresholding h) outlier removal i) radial profile

Each tomogram is split into patches of 32x32x32 `\_unit?*`. These patches are the fed into the trained U-Net, which outputs a probability mask for those patches. To obtain a complete probability mask, the patches are stitched back together. The probability mask is further refined by applying global and adaptive localized thresholding steps (Figure 2, Figure 3).
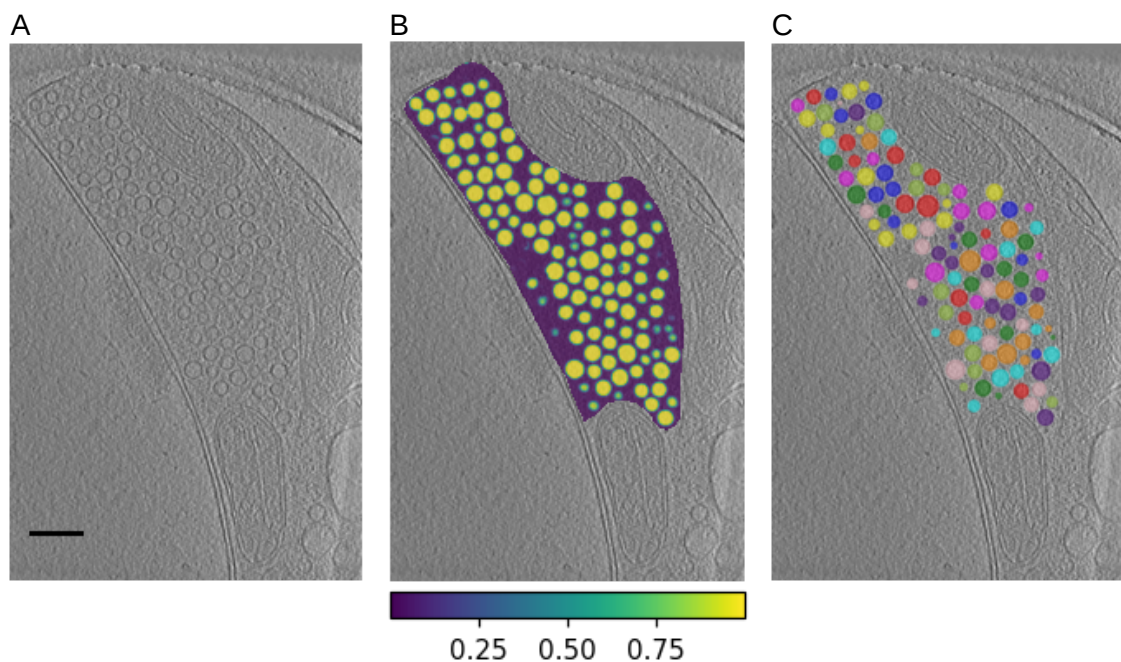
Figure 3: 2D Slices A) a section from z axis of a tomogram's presynaptic terminal of a neuron B) instance mask of the vesicles after post processing; purple corresponds to a low probability of SVs and yellow corresponds to a high probability of synaptic vesicles C) predicted probability mask by the segmentation network

`\_more detail about global and adaptive localized threshold*` For further optimization of the mask, outliers were removed. Removed outliers mostly consisted of vesicles which were only partially segmented, and vesicles which maks were adjacent due to proximity. The removed masks, which only partially traced the vesicles, were reevaluated by reducing or expanding their radius (Figure 4). `\_was the center of the vesicles also reevaluated?*`
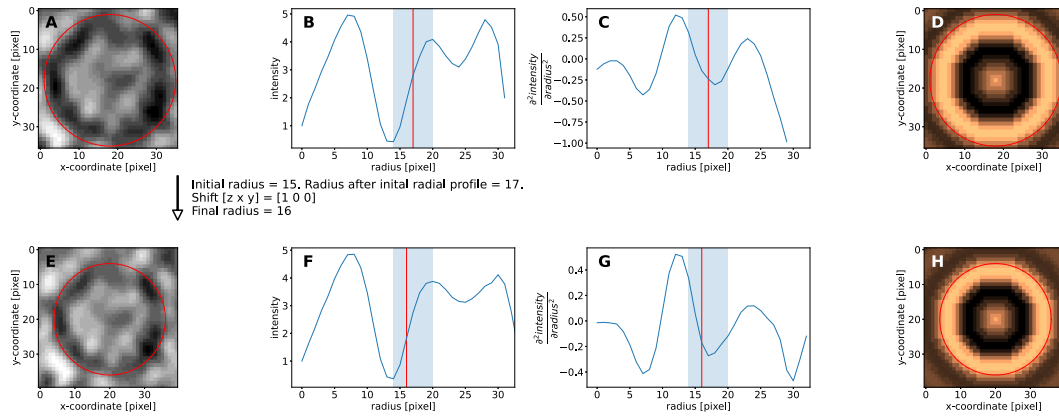
Figure 4: Vesicle radius and position through radial profile and cross-correlation Radial Profile Refinement A) couple of vesicles are not centered B) Radial Profile. Blue range is from membrane center to outer white halo center, this is the search range for the optimal radius. (smoothed by gaussian filtering) C) second derivative of radial profile E, F, H, G) Same as above columns after refinement

The adjacent vesicle masks were seperated (Figure `\_missing*`). `\_how?*`

`\_**missing Figure- Splitting adjacent vesicles. A) Examples of tomogram, no labels; B) raw label with connected vesicle-labels; C) modified label with seperated vesicles ---> for software: IMOD**`

The Dice coefficient was used to track the global congruence between the manually segmented mask and the predicted mask within the different tomograms (Figure 5).
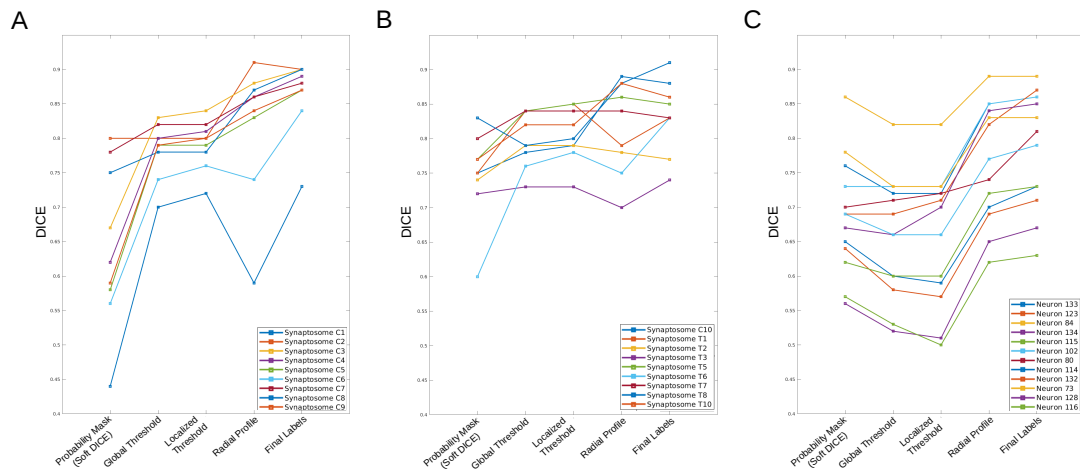
Figure 5: Dice developement during post-processing Dice developement at different post processing steps of initial predicted mask (different colors correspond to different tomograms): A) synaptosomal training datasets B) synaptosomal test datasets c) neuron test datasets

Final Eval Tables

Table 1- Evaluation of the segmentation: Mask Dice: Mask Dice coefficient for the predicted mask; Final Label Dice: Dice coefficient after post-processing; δ d: diameter error on correctly detected vesicle; Δ c: average error center (nm); # of Vesicles: number of expected vesicles; TP: True Positive; FN: False Negative; FP: False Positive

Train Dataset

Dataset	Mask DICE	Final Label DICE	δ d	Δ c (nm)	# of Vesicles	TP	FN	FP
Synaptosome C1	0.44	0.73	0.07	2.55±1.56	223	198	26	49
Synaptosome C2	0.8	0.9	0.05	2.12±1.06	105	103	2	1
Synaptosome C3	0.67	0.9	0.05	1.86±1.24	128	127	1	6
Synaptosome C4	0.62	0.89	0.03	1.78±0.92	144	141	3	4
Synaptosome C5	0.58	0.87	0.04	1.86±1.00	214	209	5	13
Synaptosome C6	0.56	0.84	0.04	1.92±1.05	104	102	2	16
Synaptosome C7	0.78	0.88	0.06	1.86±0.90	184	184	0	16
Synaptosome C8	0.75	0.9	0.05	1.70±0.93	132	126	6	1
Synaptosome C9	0.59	0.87	0.05	1.87±0.91	135	132	3	14
Average	**0.64±0.11**	**0.86±0.05**	**0.05**	**1.95±1.08**	**152.22**	**97.00%**	**3.00%**	**7.30%**

Test Dataset (Same preparation and microscope with training set)

Dataset	Mask DICE	Final Label DICE	δ d	Δ c (nm)	# of Vesicles	TP	FN	FP
Synaptosome C10	0.75	0.88	0.07	1.86±1.18	129	123	6	5
Synaptosome T1	0.75	0.83	0.11	2.66±1.52	699	687	12	33
Synaptosome T2	0.74	0.77	0.11	2.27±1.84	122	117	5	2
Synaptosome T3	0.72	0.74	0.11	3.64±2.22	434	397	37	57
Synaptosome T5	0.77	0.85	0.08	2.20±1.26	535	526	9	25
Synaptosome T6	0.6	0.83	0.07	2.02±1.12	373	353	20	42
Synaptosome T7	0.8	0.83	0.06	2.22±1.14	110	107	3	9
Synaptosome T8	0.83	0.91	0.04	2.09±1.04	100	99	1	2
Synaptosome T10	0.77	0.86	0.05	1.96±1.04	77	74	3	6
Average	**0.75±0.06**	**0.83±0.05**	**0.08**	**2.32±1.43**	**286.56**	**96.30%**	**3.70%**	**6.10%**

Test Dataset 3 (Neuron Dataset)

Dataset	Mask DICE	Final Label DICE	δ d	Δ c (nm)	# of Vesicles	TP	FN	FP
Neuron 133	0.76	0.86	0.05	2.16±1.32	523	467	56	8
Neuron 123	0.64	0.71	0.05	2.05±1.18	66	58	8	2
Neuron 84	0.86	0.89	0.06	1.44±0.75	498	484	14	1
Neuron 134	0.56	0.67	0.09	2.87±2.50	638	384	254	63
Neuron 115	0.57	0.63	0.08	3.56±3.23	170	123	47	32
Neuron 102	0.73	0.86	0.05	1.47±0.79	103	86	17	1
Neuron 80	0.7	0.81	0.07	2.67±2.00	111	102	9	3
Neuron 114	0.65	0.73	0.07	2.68±1.79	131	93	38	9
Neuron 132	0.69	0.87	0.03	1.65±1.26	135	129	6	32
Neuron 73	0.78	0.83	0.06	2.93±2.00	526	483	43	2

Dataset	Mask DICE	Final Label DICE	δ d	Δ c (nm)	# of Vesicles	TP	FN	FP
Neuron 128	0.67	0.85	0.04	2.33±1.70	252	232	20	19
Neuron 116	0.62	0.73	0.07	2.38±1.82	296	207	89	35
Average	**0.69±0.09**	**0.79±0.09**	**0.06**	**2.35±1.83**	**287.42**	**83.60%**	**16.40%**	**7.90%**

Our method transfers well across datasets even without fine-tuning which show robustness and generalization.

Comparison of manual segmentation with automatic deep-learning based segmentation
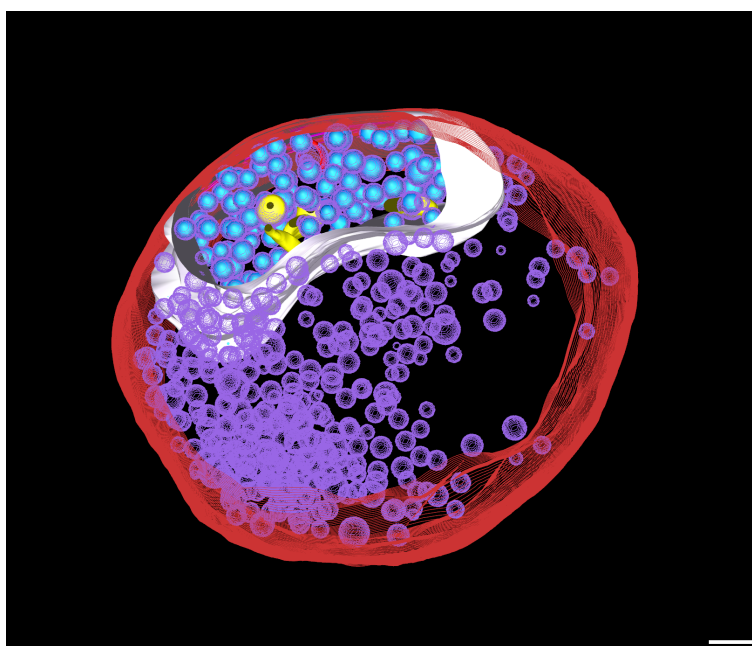
Figure 6: 3D model of manual segmented and automatically segmented synaptosome.

Discussion

While the Dice coefficient is a good global measure to assess the predictions in comparison to the ground truth, it is difficult to asses local segmentation accurracy. For example, a single generated vesicle label containing several close connected vesicles would not be practical for further analysis for the researcher although it could have almost the same dice value. What is important for actual usage of the software would be the number and percentage of true-detected vesicles, false-positive and false-negative rates.

Center error: If we measure each axis error it will reveal that human bias in segmentation is more affected on the Z-axis. [we didn't show it in number but its checked the hypothesis] `\_we cannot claim something without showing it. ---> this would belong into results*`

3d unet good for 3D processing recent Nature methods paper by Ben Engel, DeepFinder -> Relion for STA creates mask to find more using dl -what are they doing, maybe compare that in the text, different aims; we might compare results we achieve (keep as bonus, revision)

Outlook

implement automatic cell-outline and active zone segmentation as deep learning workflow using UNet
implement automatic connector and tether segmentation as a deep leaning workflow using UNet

Materials and methods

Cryo-electron Tomography Datasets

Two datasets of different origin were used as input and test subjects for the automatic segmentation pipeline, rat synaptosomes as well as astrocytic and neural cell cultures derived from mice. Both datasets have been previously analyzed [14].

Rat synaptosomes

The preparation of the rat synaptosomes were previously described [15]. After the purification, the synaptosomes were incubated for 30 min at room temperature. 1.3mM $CaCl_2$ and 10 nm gold fiducials were added (gold fiducials, #s10110/8. AURION Immuno Gold Reagents & Accessories. Wageningen, The Neatherlands). The synaptosome solution was applied to a 200-mesh lacey finder carbon film grid (#AGS166-H2. Agar Scientific. Elektron Technology UK Ltd. Stansted, UK). Manual blotting was used to remove excess liquid with a filter paper. Thereafter, the grid was immediately plunge frozen in liquid ethane using a home built plunge freezer controlled with a LabView script (National Instruments Corporation. Mopac Expwy Austin, TX, USA). The grids coated with rat synaptosomes were mounted in a cryo-holder (Gatan, Pleasonton, CA, USA) and transferred to a Tecnai F20 (FEI, Eindhoven, The Netherlands) which was set to low dose conditions, operated at 200 kV, and equipped with a field emission gun. Images were recorded with a 2k x 2k CCD camera (Gatan) mounted after a GIF Tridiem post-column filter (Gatan) operated in zero-loss mode. The sample was kept at about -180 °C. Tilt series were acquired using SerialEM [16] for automated acquisition recorded typically from -50° to 50° with a 2° angular increment and an unbinned pixel size of 0.75 or 1.2 nm. Due to sample thickness (400-700 nm), tomograms were usually not recorded with higher tilt angles. Defocus was set between -8 to -12 μm and the total electron dose used was about 80-100 e$^-$/$Å^2$. Some tomograms were acquired at a Titan Krios equipped with a K2 direct electron detector (Gatan) without energy filter. The K2 camera was operated in super-resolution counting mode and between 8-40 frames per tilt angle were taken. Tilt series were acquired using the Latitude software (Gatan) for automated acquisition recorded typically from -60° to 60° with a 2° angular increment and an unbinned pixel size of 0.6 nm. Defocus was set between -8 to -12 μm and the total electron dose used was about 80-100 e$^-$/$Å^2$. Before image processing the frames at each tilt angle, frames were aligned and averaged in 2dx MC_Automator [17] with motioncor [18]. 3D reconstruction was done in IMOD [13]. The alignments were done using the automated fiducial tracking function and the 3D reconstructions were done using the weighted back projection followed by a nonlinear anisotropic diffusion (NAD) filtering.

Astrocytic and neuronal mouse culture

The preparation of astrocytic and neuronal culture has been published before [19]. After 12 to 14 days of incubation, grids with mouse neurons were plunge frozen with a Vitrobot (Thermofisher Scientific, Mark IV) with a blot time of 3 s and a blot force of -10. Wait time and drain time were not used. Humidity was set to 100% at 4 °C. 4 μl undiluted 10 nm BSA gold tracer (Aurion) was added directly onto the grid prior to plunge freezing. Cultured mouse neurons tilt series were acquired at a Titan Krios, equipped with a Falcon 3 direct electron detector (Thermofisher Scientific) without energy filter. The Falcon camera was operated in linear mode. Tilt series were acquired using the TEM Tomography software (TFS) for automated acquisition, recorded typically from -60° to 60° with a 2° angular increment and an unbinned pixel size of 0.37 nm. Defocus was set between -6 to -10 μm and the total

electron dose used was about 80-100 e$^-$/Å^2. Tomogram reconstruction was done in the same way as for the synaptosome datasets.

Manual segmentation procedures

Manual segmentation of SVs, mitochondria, and the active zone PM was done in IMOD (Figure S4A&B). The boundary marked the region to be analyzed by Pyto [6]. The analysis by Pyto was essentially the same as described previously [5,6].

In short, the segmented area is divided in 1 voxel thick layers parallel to the active zone for distance calculations. A hierarchical connectivity segmentation detects densities interconnecting vesicles (so-called connectors) and densities connecting vesicles to the active zone PM (so-called tethers) (Figure `\_add figure number*`). Distance calculations are done with the center of the vesicle. Mainly default settings were used. The segmentation procedure is conservative and tends to miss some tethers and connectors because of noise. Consequently, the numbers of tethers and connectors should not be considered as absolute values, but rather to compare experimental groups. As it was done before, an upper limit was set between 2100 and 3200 nm^3 on segment volume. The tomograms that were used for this analysis were binned by a factor of 2 to 3, resulting in voxel sizes between 2.1 and 2.4 nm.

Pre-processing of manual segmentation outputs from IMOD for further use (jupyter notebook pre-pyto)

Description of Machine Learning

This experiment was conducted on a workstation with an AMD Ryzen Threadripper 3970X CPU with an NVIDIA GeForce RTX 2080 Ti GPU. All the framework has been implemented in Python using the Keras library (2.4.3) and TensorFlow (2.4.1). The workflow includes a GUI based on a multidimensional image viewer, Napari (0.4.15), enabling the user to add and remove vesicles. `\_any other packages required?*`

The used datasets included a total of 30 tomograms with heterogeneous pixel sizes, defocus and resolution.

1. All 9 tomograms Dataset: partially segmented and used for training (Synaptosome)
2. A single tomogram with the exactly same setup and sample preparation like the train dataset
3. 8 Synaptosome tomograms Dataset: with ground truth (with an exceeding treatment on the samples)
4. 12 Neuron fully segmented tomograms Dataset: with completely different sample preparation and microscope setup

`\_this is not good place to talk about evalution *`

Deep Model Training with a 3D U-Net

Create an Input Voxel Patch

The training set was prepared by splitting the 3D tomographic volume into 32x32x32 `\_voxels??*` sub volumes and keeping only volumes occupied by a sufficient amount (> 1000 voxels) of binarized vesicle labels. The obtained sub volumes were randomly divided into ten subsets of the training set, this method is termed k-fold cross-validation in the field of machine learning. `\_this is a wrong`

statment we dont do cross validation just split(fold) data * All of these subsets or "folds" were used as training sets, as an entirely separate set of tomographic subvolumes was used for validation, to avoid overfitting.

_are the folds overlapping? are the tomograms normalized any further than the NAD from previous segmentation before feeding it to the deeplearning model? add image from odt??, the 2D slices of the subset are supposed to be 32x32x32, but seem to have a different size... padding?? ---> results*

3D U-Net architecture

The previously prepared subsets are fed into the 3D U-Net in batches of 50. _cite Ronneberger, what was changed compared to the original U-Net, is there a better publication to cite?* These were passed through the U-Net in a total of 200 epochs. Batch normalization was applied before Rectified Linear Unit (ReLU) activation _was it? cite Ioffe & Szegedy, maybe? [@doi:10.48550/arXiv.1502.03167]* .

_we do not use deconv layers, instead we upsample (by repeating the data) and then apply a convolution.* _as far as I know, nobody uses analysis/expantion path, encoder/decoder is the common and correct term.* The 3D U-Net architecture is composed of a contracting or analysis path!! (convolutional layers), and an expanding or synthesis path (deconvolutional!! layers) (Figure 7). _did we wrote our own U-Net or do we need to quote someones github for the original framework* _we used CSBDeep for Convoultion blocks Benoit mentioned it in the introdction we can bring it here but github licence is BSD 3-Clause License whihc it means we dont need to reference *
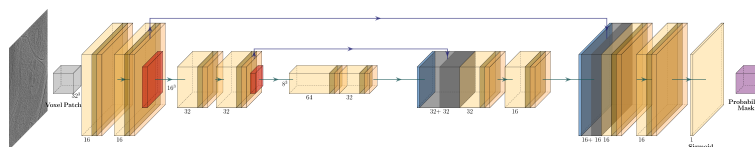
Figure 7: Segmentation Network: U-Net. Input Size is 32^3 _voxels??* , in each resolution we have two convolution layers followed by batch normalization layer and rectified Linear Units (ReLU) activation function. Intermediate sizes are written on top of arrows, number of convolution filters is written bottom of boxes. Skip connections show concatenation of the features from contracting path (left side of the network) and expansive path (right side of the network). _explain different colors, do they correspond with the different steps? for example: red- max pooling 2x2x2? add figure legend?*

_what do we use as initial encoder weights and backbone* _code checked: keras- The Glorot uniform initializer, also called Xavier uniform initializer.* During each layer of the analysis path, the convolution filter consisting of a 3x3x3 kernel, with randomly initialized weights, was moved over all the voxels in each subset twice _what do you mean by "twice"? * , each time taking their dot product. This kernel extracts and enhances the features in different parts of the image. This is followed by a ReLU function, which can be described as

$$f(x) = max(0, x)$$

which will output the positive input directly, while setting the negative input to 0.

Between each layer, the voxels were condensed, or down sampled, via a max-pooling step of 2x2x2, in which the maximum value of the 2x2x2 voxel cube is put forth. With every layer of the U-Net, the

input size thereby halves, while the number of channels doubles.

While the analysis path focuses on the identification of the areas of interest, the synthesis part focuses on their localization. The synthesis path of the U-Net uses the same basic architecture as the analysis path, with a slight variation of using up convolution operation and implementing skip connections, where feature maps of the analysis part are concatenated to the output of the transposed convolutions of the same level.

`\_code checked:final = Conv3D(n_channel_out, (1,) * n_dim, activation='sigmoid')(unet) *` The sigmoid/Softmax `\_in the image you write Sigmoid but in the text Softmax, which activation function was used?*` activation function is the last block of the U-Net (Figure 7), it triggers the loss function, which converts the output from the decoder path into a mask, where each voxel is assigned as either vesicle or not-vesicle. As the segmentation of vesicles can be considered a binary classification task, binary cross-entropy was implemented as a loss function.

$$Loss = -\frac{1}{outputsize} \sum_{i=1}^{outputsize} y_i * log\hat{y}_i + (1 - y_1) * log(1 - \hat{y}_1)$$

with the *output size* being the number of scalar values in the model output, $\hat{y}\_1\_$ being the *i*-th scalar value in the model output and $y\_i\_$ being the corresponding target value. `\_publication to cite? *`

To optimize the weights and reduce the loss function during training, the Adam optimizer was used [20]. 200 epochs were run until the loss function reached a global minimum. `\_it's a local minima, we are optimizing a non-convex function with thousands of parameters*`

Mask prediction

The tomograms are large, 3D volumes. They therefore were split into 32x32x32 patches with a step size of 24 (stride) before the U-Net. The small prediction patches, which were the output of the U-Net, were then stitched together to get the final probability mask.

Transfer Learning / Neuron Dataset

Postprocessing

Estimating Global Threshold

In electron microscopic images, dense areas of the specimen appear darker, as fewer electrons are able to pass through that area of the sample. Therefore, the cellular and vesicular membrane, among other features, appear darker than the background. Binary cross-entropy was utilized to binarize the obtained probability mask. To determine the required threshold, several threshold values (between 0.8 and 1.0) were tested and the one that minimized the average of luminance of all marginal voxels was selected.(srounnd of vesicles) `\_add image from odt??*`

Adaptive Localized Threshold

The global threshold identifies the majority synaptic vesicles. Some vesicles, however, are more difficult to detect. This might be, due to high intensity variations within the tomogram, or because the

predicted binarized labels are do not match the spherical shape of vesicles. `\_should go to results?*`

In terms of binarized lables not matching the spherical shape of vesicles, two main classes were identified: vesicles that were in close proximity to another and got connected, and vesicles, which were only partially captured. These two classes were then separated from the rightfully identified vesicles by comparing their label sizes and their extent value (Ratio of pixels in the region to pixels in the total bounding box computed as area / (rows * cols)). `\_write this into a clean formula*` In the next step, probability masks of the partially detected vesicles were expanded in an effort to find the correct mask. The probability masks of the connected vesicles, however, were minimized by searching for a more precise threshold, with the goal of separating the vesicles.

Radial Profile

`\_@Benoit?*`

Outlier Removal

The feature space of predicted vesicle labels was defined, containing thickness, membrane density, and estimated radius of a vesicle. `\_@Benoit: after radial profile, we can add the definition of thinness and membrane as well*` To detect outliers in this multivariate space, Mahalanobis Distance (MD) was applied to calculate L2 norm distance on normalized variables using the covariance matrix of observation. As an additional way to detect outliers, the p-value of MD was calculated, bringing this evaluation setup in iterative form. If the MD p-value of a specific vesicle was not in a specific margin range (0-10), their radial profile was recalculated, and the label entirely removed if they again failed to pass the margin of the p-value.

Radius Estimation (Cross Correlation through Radial Profile)

`\_@Benoit?*`

Analysis of Results

The evaluation framework was designed to assess the capabilities of the proposed toolbox for automatic synaptic vesicle segmentation. The framework was not only designed to evaluate quantitatively performance of the neural network, but rather assay the segmentation of vesicles in practice `\_unsure what the last part means*`. `\_I tried to say we develop the software rather than an algorithm paper with ablation study kinda more trasnfer learning but however for tranfer learning we might need add finetunning the network or say this sentence in other way*`. The pipeline also generates a specific output format, which is necessary to further analyze the presynaptic tomograms via another pre-developed toolbox (Pyto), which segments small molecular filaments associated to the synaptic vesicles, titled tethers and connectors.

DICE

The `\_general form??*` DICE coefficient for probabilistic subvolume maps was calculated after each epoch as a performance quantification while `\_during?*` training. The probabilistic mask subvolumes were stitched back together, creating a probabilistic map of the whole tomogram. The DICE for the whole tomogram was calculated to evaluate the similarity of the predicted probability

mask with ground truth. The binarization from the same formulation converges to this interpretation that we measure the overlap between two samples. `\_this sounds unnecessary difficult*`

$$1 - \frac{2 \sum_{pixels} y_{true} y_{pred}}{\sum_{pixels} y_{true}^2 + \sum_{pixels} y_{pred}^2}$$

`\_shouln't it be voxels instead of pixels??*` `\_yes voxel is right*`

THE DICE was also employed to monitor all stages of post-processing on the eventual label file, to observe the effect of each post-processing step.

Diameter Error

The diameter of a vesicle is one of its relevant characterizations, and it is predefined (see Outlier Removal). `\_which diameters are we using in this evaluation as input, pre- or post-outlier removal?*` `\_after! I meant from that perentesis that radius or dimater we assume as one charectiristic of vesicle we might can write it better *` The error of diameter estimation of true-detected vesicle is defined as 1 minus the proportion of diameters

$$\delta d = 1 - \frac{min(dSi, dGTi)}{max(dSi, dGTi)} \tag{1}$$

where dGTi is the diameter of each true manual segmented vesicle, and dSi is the diameter of its estimation.

Center error

The center error is an euclidean distance of ground truth and corresponds to true predicted vesicles `\_true pos or true neg*`. A vesicle was defined as a true-detected vesicle if the predicted center was located inside the hand-segmented vesicle and the other way around the center of prediction was located inside the predicted vesicle. `\_isn't this a bit too general, shouldn't this be a tighter evaluation?*` `\_we assume this as hard condition to be true postive we could define like some % liek 50% intersection but this condition is generally harder` This means the volume of intersection of the estimated vesicle with the distance of d to a ground truth vesicle with radius R is:

$$V = \frac{1}{12}\pi(4R + d)(2R - d)$$

Manuscript preparation

The manuscript was written with the open and collaborative scientific writing package Manubot [21]. The source code and data for this manuscript are available at https://github.com/aseedb/synaptic_tomo_ms.

References

1. **Towards Visual Proteomics at High Resolution**
    Felix JB Bäuerlein, Wolfgang Baumeister
    Journal of Molecular Biology (2021-10) https://doi.org/gn9t3v
    DOI: 10.1016/j.jmb.2021.167187 · PMID: 34384780

2. **Exploring high-resolution cryo-ET and subtomogram averaging capabilities of contemporary DEDs**
    Martin Obr, Wim JH Hagen, Robert A Dick, Lingbo Yu, Abhay Kotecha, Florian KM Schur
    Molecular Biology (2022-01-10) https://doi.org/gn92pd
    DOI: 10.1101/2022.01.10.475481

3. **High-resolution in situ structure determination by cryo-electron tomography and subtomogram averaging using emClarity**
    Tao Ni, Thomas Frosio, Luiza Mendonça, Yuewen Sheng, Daniel Clare, Benjamin A Himes, Peijun Zhang
    Nature Protocols (2022-02) https://doi.org/gn92pc
    DOI: 10.1038/s41596-021-00648-5 · PMID: 35022621 · PMCID: PMC9251519

4. **STRUCTURAL STUDIES BY ELECTRON TOMOGRAPHY: From Cells to Molecules**
    Vladan Lučić, Friedrich Förster, Wolfgang Baumeister
    Annual Review of Biochemistry (2005-06-01) https://doi.org/cfd27f
    DOI: 10.1146/annurev.biochem.73.011303.074112 · PMID: 15952904

5. **Quantitative analysis of the native presynaptic cytomatrix by cryoelectron tomography**
    Rubén Fernández-Busnadiego, Benoît Zuber, Ulrike Elisabeth Maurer, Marek Cyrklaff, Wolfgang Baumeister, Vladan Lučić
    Journal of Cell Biology (2010-01-11) https://doi.org/b9c26b
    DOI: 10.1083/jcb.200908082 · PMID: 20065095 · PMCID: PMC2812849

6. **Hierarchical detection and analysis of macromolecular complexes in cryo-electron tomograms using Pyto software**
    Vladan Lučić, Rubén Fernández-Busnadiego, Ulrike Laugks, Wolfgang Baumeister
    Journal of Structural Biology (2016-12) https://doi.org/f9d5t2
    DOI: 10.1016/j.jsb.2016.10.004 · PMID: 27742578

7. **Robust membrane detection based on tensor voting for electron tomography**
    Antonio Martinez-Sanchez, Inmaculada Garcia, Shoh Asano, Vladan Lucic, Jose-Jesus Fernandez
    Journal of Structural Biology (2014-04) https://doi.org/f5zkj8
    DOI: 10.1016/j.jsb.2014.02.015 · PMID: 24625523

8. **Convolutional neural networks for automated annotation of cellular cryo-electron tomograms**
    Muyuan Chen, Wei Dai, Stella Y Sun, Darius Jonasch, Cynthia Y He, Michael F Schmid, Wah Chiu, Steven J Ludtke
    Nature Methods (2017-10) https://doi.org/gkpj62
    DOI: 10.1038/nmeth.4405 · PMID: 28846087 · PMCID: PMC5623144

9. **Automated Detection and Localization of Synaptic Vesicles in Electron Microscopy Images**
    Barbara Imbrosci, Dietmar Schmitz, Marta Orlando
    eneuro (2022-01) https://doi.org/gn9n8k
    DOI: 10.1523/eneuro.0400-20.2021 · PMID: 34983830 · PMCID: PMC8805189

10. **U-Net: Convolutional Networks for Biomedical Image Segmentation**
    Olaf Ronneberger, Philipp Fischer, Thomas Brox
    arXiv (2015-05-19) https://arxiv.org/abs/1505.04597

11. **3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation**
    Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, Olaf Ronneberger
    arXiv (2016-06-22) https://arxiv.org/abs/1606.06650

12. **Content-aware image restoration: pushing the limits of fluorescence microscopy**
    Martin Weigert, Uwe Schmidt, Tobias Boothe, Andreas Müller, Alexandr Dibrov, Akanksha Jain,
    Benjamin Wilhelm, Deborah Schmidt, Coleman Broaddus, Siân Culley, … Eugene W Myers
    Nature Methods (2018-12) https://doi.org/gfkkfd
    DOI: 10.1038/s41592-018-0216-7 · PMID: 30478326

13. **Computer Visualization of Three-Dimensional Image Data Using IMOD**
    James R Kremer, David N Mastronarde, JRichard McIntosh
    Journal of Structural Biology (1996-01) https://doi.org/d9nfzw
    DOI: 10.1006/jsbi.1996.0013 · PMID: 8742726

14. **Morphofunctional changes at the active zone during synaptic vesicle exocytosis**
    Julika Radecke, Raphaela Seeger, Anna Kádková, Ulrike Laugks, Amin Khosrozadeh, Kenneth N
    Goldie, Vladan Lučić, Jakob B Sørensen, Benoît Zuber
    Neuroscience (2022-03-07) https://doi.org/gpm26v
    DOI: 10.1101/2022.03.07.483217

15. **A rapid Percoll gradient procedure for preparation of synaptosomes**
    Peter R Dunkley, Paula E Jarvie, Phillip J Robinson
    Nature Protocols (2008-11) https://doi.org/b7zwh8
    DOI: 10.1038/nprot.2008.171 · PMID: 18927557

16. **Automated electron microscope tomography using robust prediction of specimen movements**
    David N Mastronarde
    Journal of Structural Biology (2005-10) https://doi.org/ff7gzx
    DOI: 10.1016/j.jsb.2005.07.007 · PMID: 16182563

17. **2dx_automator: Implementation of a semiautomatic high-throughput high-resolution cryo-electron crystallography pipeline**
    Sebastian Scherer, Julia Kowal, Mohamed Chami, Venkata Dandey, Marcel Arheit, Philippe
    Ringler, Henning Stahlberg
    Journal of Structural Biology (2014-05) https://doi.org/f522h3
    DOI: 10.1016/j.jsb.2014.03.016 · PMID: 24680783

18. **Electron counting and beam-induced motion correction enable near-atomic-resolution single-particle cryo-EM**
    Xueming Li, Paul Mooney, Shawn Zheng, Christopher R Booth, Michael B Braunfeld, Sander
    Gubbens, David A Agard, Yifan Cheng
    Nature Methods (2013-06) https://doi.org/f4zpjf
    DOI: 10.1038/nmeth.2472 · PMID: 23644547 · PMCID: PMC3684049

19. **Preparation of Primary Neurons for Visualizing Neurites in a Frozen-hydrated State Using Cryo-Electron Tomography**
    Sarah H Shahmoradian, Mauricio R Galiano, Chengbiao Wu, Shurui Chen, Matthew N Rasband,
    William C Mobley, Wah Chiu
    Journal of Visualized Experiments (2014-02-12) https://doi.org/gmh9w3

DOI: [10.3791/50783](10.3791/50783) · PMID: [24561719](24561719) · PMCID: [PMC4089403](PMC4089403)

20. **Adam: A Method for Stochastic Optimization**
Diederik P Kingma, Jimmy Ba
arXiv (2014) https://doi.org/hnkr
DOI: 10.48550/arxiv.1412.6980

21. **Open collaborative writing with Manubot**
Daniel S Himmelstein, Vincent Rubinetti, David R Slochower, Dongbo Hu, Venkat S Malladi,
Casey S Greene, Anthony Gitter
PLOS Computational Biology (2019-06-24) https://doi.org/c7np
DOI: 10.1371/journal.pcbi.1007128 · PMID: [31233491](31233491) · PMCID: [PMC6611653](PMC6611653)