NU-Net: a self-supervised smart filter for enhancing blobs in bioimages

Seongbin Lim
Laboratoire d'Optique et Biosciences, CNRS, INSERM, École Polytechnique, Institut Polytechnique de Paris
91128 PALAISEAU Cedex, France
sungbin246@gmail.com

Emmanuel Beaurepaire
emmanuel.beaurepaire@polytechnique.edu

Anatole Chessel
anatole.chessel@polytechnique.edu

Abstract

While supervised deep neural networks have become the dominant method for image analysis tasks in bioimages, truly versatile methods are not available yet because of the diversity of modalities and conditions and the cost of re-training. In practice, day-to-day biological image analysis still largely relies on ad hoc workflows often using classical linear filters. We propose NU-Net, a convolutional neural network filter selectively enhancing cells and nuclei, as a drop-in replacement of chains of classical linear filters in bioimage analysis pipelines. Using a style transfer architecture, a novel perceptual loss implicitly learns a soft separation of background and foreground. We used self-supervised training using 25 datasets covering diverse modalities of nuclear and cellular images. We show its ability to selectively improve contrast, remove background and enhance objects across a wide range of datasets and workflow while keeping image content. The pre-trained models are light and practical, and published as free and open-source software for the community. NU-Net is also available as a plugin for Napari.

1. Introduction

Machine learning solutions, especially those based on deep neural networks, have had huge impact on bioimage analysis. Very successful segmentation models, in particular, have been introduced, starting with U-Net [32]. Over the years, semantic segmentation task evolved into instance segmentation with its ability to address touching boundaries of nuclei and cells and identify instances, such as StarDist [34], nucleAIzer [17], Cellpose [37], CDNet [15], and Mesmer [14].

These segmentation models were essentially based on supervised learning, resulting in capable and performant models once sufficient annotated data were provided. However, in biological imaging, each experiment is different, with varied image modality or sample type, condition and labeling, making the design of a truly generic pre-trained model conplicated. In practice, it is common in analysis pipelines to annotate a few local data again, which is a time-consuming process needing expertise, and to retrain or fine-tune these models. Alternatively, users resort to less demanding machine learning software, compromising performance, *e.g.* ilastik [5] and LABKIT [3], to interactively acquire a model, usually built on random forest classifier.

One very common way to circumvent this issue, in particular, since machine learning expertise is not always available in experimental labs, is to use the same set of classical linear filters in increasingly complex pipelines. Use cases include a quick and light way to adjust contrast, a post-acquisition step to denoise, enhance images, or clean their

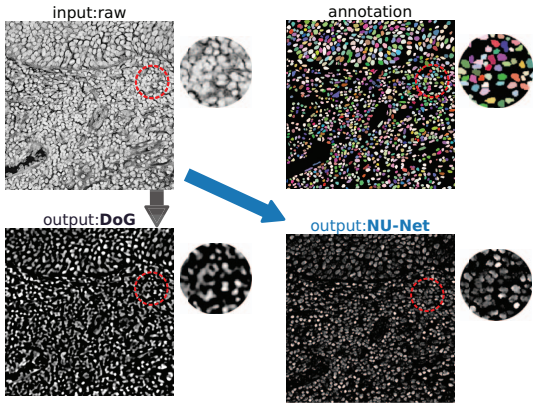

Figure 1. Proposed model, NU-Net, is a nuclear and cellular filter built on convolutional neural network and self-supervised learning scheme via perceptual losses, to be used in place of linear filters. This example compares NU-Net to difference of Gaussians (DoG). Sample image is a histopathology image containing dense cells and is sourced from [26]. (The sample image was converted into gray-scaled).

background or a pre-analysis step to detect/segment/count objects of interest. Linear filters are reliable and well proven over time, but most of them are parametric algorithms with strong assumption, meaning needs of parameter search and low versatility, respectively.

In this paper, we propose a novel filter, we call NU-Net (Not a U-Net and read new net), built on convolutional neural networks (CNNs) and self-supervised learning scheme. NU-Net enhances contrast of images based on morphological traits of nuclei and cells perceived by a large pre-trained vision model. NU-Net's behavior is positioned between a binary semantic segmentation model that classifies foreground and background pixels and an autoencoder that reconstructs input images as accurately as possible. During training, it learns morphological perception as well as a generative loss and does not require paired target labels as in supervised learning. This self-supervision advantage can address highly diverse nature of biological images across many modalities because one can include as many images as suited regardless of their target label availability. Also, it makes NU-Net highly adaptable to local images and practical in real application.

To leverage its training benefit, we gathered 25 nuclear and cellular datasets with or without target labels and trained NU-Net with this meta dataset to promote it as a generalized filter that can cover a large range of different modalities of nuclear and cellular images. We kept the architecture of NU-Net rather small, so it can be used on a local CPU without heavy computing power or reliance of graphical processing units (GPUs). In addition, training NU-Net from scratch can be done within an hour with a proper setup, if needed.

Evaluation of NU-Net is complicated by the fact that it is not directly comparable to other recent work and thus cannot reuse existing metrics and benchmarks. Since the main objective of this work is a practical one, adding a new tool to the bioimage analysis toolbox, we focused on integrating NU-Net into actual object detection workflow, to show its usefulness in practice, as well as try it on a wide range of images to show its versatility.

Contributions of this work can be summarized as followed:

- introducing a novel filter built on machine learning and CNN, which can replace classical linear filters
- using perceptual losses for self-supervised learning to circumvent supervised learning and avoid costly labeling process in biological images
- gathering 25 different nuclear and cellular datasets and training NU-Net to cover large range of modalities

1.1. Related work

Perceptual losses were used effectively for neural style transfer application [22]. They take advantage of a pre-trained large vision model such as VGG [35] and its perception or feature vectors from certain neural layers. The losses reconstruct input images given features or details that each layer keeps. In neural style transfer, there are two losses in general [11, 10, 19]: content loss and style loss. The content loss is a generative loss to preserve content and the style loss is a modified generative loss using Gram's matrices that attempts to reconstruct a certain style.

To seek generalizability in machine learning models, foremost one needs diverse sources of data. Many works focused on segmentation tasks. In biological images, Cellpose [37] is an instance segmentation model that transforms target masks based on diffusion process, and for training they scraped nuclear and cellular images from multiple sources including internet to make their models more generic. NucleAIzer [17] is another instance segmentation model that defined a set of modalities and combined simulation and neural style transfer (pix2pix [20]) to augment training data. TissueNet [14] is a dataset that provides about 2.5M of annotations for tissue images across 9 organs and 6 different modalities to support general-purpose tissue segmentation models.

For machine learning applications beside segmentation tasks, image restoration and denoising models turned out to be practical and effective. CARE [40] is a super-resolution model that learns how to restore poorly resolved microscopy images from pairs of high and low quality images. Noise2Noise [27] demonstrated denoising MRI images as a supervised model. To avoid the necessity of having paired labels, Noise2Void [25] proposed self-learning model to denoise images. Self-learning approach made it easy to apply Noise2Void model to any images since it does not need paired images nor annotations.

As supervised learning approach remains difficult in biological images in practice, classical filters are still frequently used as off-the-shelf methods in many biomedical image analysis pipelines. For contrast enhancement, Gamma and logarithm adjustment methods are the simplest, yet remain versatile and available in most software tools. More elaborated approach is to remove background, that is to identify foreground and background pixels, often with morphological assumption. DoG (difference of Gaussians) and LoG (Laplacian of Gaussian) are such filters that consider blobs as objects of interest or foreground pixels given univariate Gaussian kernels. These linear filters are not dependent on data unlike machine learning approach and thus universally applicable. However, they are rigidly bound by their strong assumption and prone to visual artifacts and errors. Despite being thoroughly characterized mathematically, their use is still very much empirical and problem

specific. DoG and LoG are popular choices for object detection or segmentation task when it comes to bioimage analysis and widely adopted for their easy implementation to many applications and software, as in Fiji [33], ilastik [5], TrackMate [38], BigStitcher [18] and so on.

2. Method

We will describe losses, training scheme and architecture of NU-Net, a self-supervised nuclear and cellular filter we propose. In addition, we will introduce 25 datasets that we used and the concept of meta-dataset with which we trained NU-Net.

2.1. Losses

NU-Net uses perceptual losses and the work of Johnson *et al*. [22] as a starting point. They used a pre-trained VGG [35] for perception and demonstrated neural style transfer and super-resolution applications. Neural style transfer applications usually have two losses: content loss and style loss [11, 10, 19, 12, 4].

2.1.1 Content loss

The content loss is a generative loss that attempts to reproduce input data through a single latent feature vector from a certain layer of pre-trained deep neural network. The definition of content loss is shown as in equation 1. The content loss (L_c) optimizes $\hat{\boldsymbol{y}}$ to reproduce *content* $\boldsymbol{c}$ (which is identical to $\boldsymbol{x}$ in our setting) based on features at layer F^l of a pre-trained deep neural network. The loss is defined incorporating the mini-batching technique where N denotes a mini-batch size:

$$L_c(\boldsymbol{x}, \hat{\boldsymbol{y}}, l) = \frac{1}{N} \sum_{n=1}^{N} \sum_{i,j} (F_{ij}^l(\boldsymbol{x}) - F_l^l ij(\hat{\boldsymbol{y}}))^2 \quad (1)$$

The choice of layer l has great importance in particular for content loss, because what is considered as *content* is determined largely by convolutional layers and the size of their receptive fields at layer l. In general, the concept of *content* is recognized as a high-level feature, thus the required receptive field should not be too small or too large.

2.1.2 Morphological loss

The morphological loss we propose is an extension of the style loss from [22]. The plain style loss from literatures [11, 10, 22] uses Gram matrix to represent *styles*. Gram matrix being $G^l \in R^{N_l \times N_l}$, the vectorized feature map $F_l \in R^{N_l \times M_l}$ has N_l number of filters and has the size of M_l, where l indicates l-th block inside a CNN: $G_{ij}^l = \sum_k F_{ik}^l F_{jk}^l$.

In the context of style transfer applications [11, 10, 22], the morphological loss can be interpreted to generalize a certain style, which has binary values with a set of targets having a consistent morphology. In fact, there have been few attempts to model or generalize one particular style with multiple images, because supposedly a style usually meant one from a single painting instead of a group of the same style of paintings, possibly due to their high complexity. Our idea started from an assumption that generalizing a simple style from multiple images with the same style, as in our case, should be easier. In principle, the morphological loss could generalize any morphologies as long as the given morphologies are consistent. We focused on round shapes or blobs, which can represent nuclei or cells. We will call this morphological prior "blob-mask style".

Morphological loss is calculated in mini-batch fashion same as for the content loss in equation 1. The idea is similar to that of the memory bank [42], to sample mini-batches from a bank that contains a fixed number of predefined classes. Our bank represents only one style, to be specific blob-mask style. Morphological loss (L_m) is defined as below with N_s referring to a mini-batch size of morphological targets $\boldsymbol{s}$, $\hat{\boldsymbol{y}}$ to an image to be optimized, and $\boldsymbol{s_n}$ to n-th target within a mini-batch $\boldsymbol{s}$:

$$L_m(\boldsymbol{s}, \hat{\boldsymbol{y}}) = \frac{1}{N_s N_l} \sum_{n=1}^{N_s} \sum_{l} (G^l(\boldsymbol{s_n}) - G^l(\hat{\boldsymbol{y}}))^2 \quad (2)$$

2.1.3 Total loss

Combining equation 1 and 2, total loss (L_t) is defined as below, where we put linear weights on each loss with w_c and w_m, respectively.

$$\begin{aligned} L_t &= w_c L_c(\boldsymbol{x}, \hat{\boldsymbol{y}}, l) + w_m L_m(\boldsymbol{s}, \hat{\boldsymbol{y}}) \\ L_t &\propto L_c + \frac{w_m}{w_c} L_m = L_c + w L_m \text{ where } w = \frac{w_m}{w_c} \end{aligned} \quad (3)$$

The fraction (w) between two coefficients play an important role to balance two losses and can bring in practical functionality later in inference time as we will show in the result section.

2.2. Training scheme

Overall training scheme is displayed in figure 2. There are two separate groups of image sources. One is for the content loss and the other is for the morphological loss. We will call the former content images and the later style images for brevity. The content images are a set of raw images whose contents need to be preserved. The style images are also a set of binary images or mask labels that serves two purposes. The primary purpose of the style images is to define "contents" to keep, against what its name suggests, or

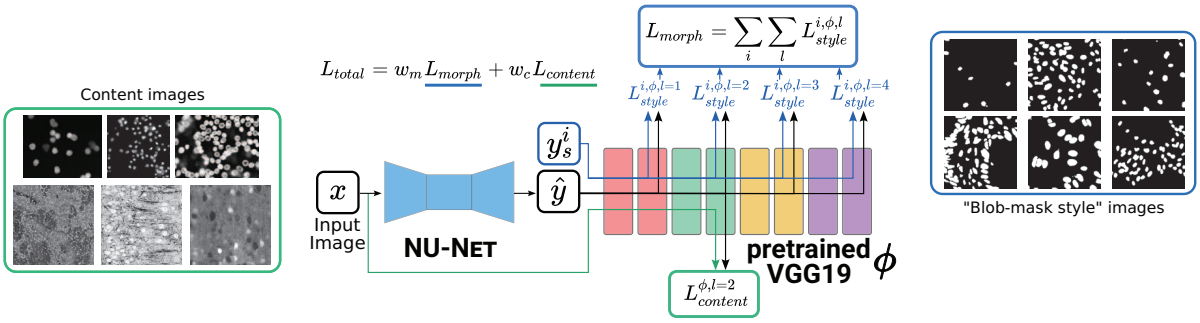

Figure 2. Diagram describing training losses. Total loss consists of two losses: content loss and morphological loss. Both are perceptual losses from a pre-trained large vision model, VGG19 [35] in this case. Note that the content images and the "blob-mask style" images are not paired, neither are the two losses. x denotes input image, $\hat{y}$ output image. y_s^i is a sample target from a mini-batch used for morphological loss. Find details in equation 1, 2 and 3. $l \in \{1, 2, 3, 4\}$ indicates four feature block in VGG19, described in [22].

to apply "blob-mask style" to the content images in the perspective of neural style transfer. The secondary purpose is to maximize contrast between foreground and background. Note that the content images and the style images are entirely separate and not paired unlike a typical supervised learning scheme.

We chose a pre-trained VGG19 [35] as our perception network and employed the same architecture from the work of Johnson *et al*. [22]. The architecture of NU-Net makes use of residual blocks [16] which are important for the purpose because they are effective to preserve input data. We followed the recipe from [22] to calculate perceptual losses. The number of parameter is about 1.7M, which seems small but capable enough to apply complex styles of paintings. NU-Net is light and fast benefiting from small architecture.

2.3. Data

2.3.1 Meta dataset

We wanted to make NU-Net as generic and applicable to many modalities of nuclear and cellular images as possible and promote it as a superior alternative to classical linear filters. Whether it is supervised learning or self-supervised learning, it is imperative to use various sources of data in order to make a generic machine learning model. Massive and well curated datasets such as ImageNet [23] and MS-COCO [29] are great examples that paved the road for computer vision foundation models [6]. It was not until TissueNet [14] that such large dataset appeared for nuclear or cellular images. TissueNet sourced images from 6 platforms and 9 organs, which were unprecedented diversity for a dataset curated for machine learning. However, it cannot cover all types of images due to highly diverse nature of biomedical image modalities, such as our local images of mouse cerebral cortex acquired via THG (third-harmonic generation) microscopy as TissueNet simply does not have the same platform nor the same sample.

To address the issue of lacking diversity of data source in nuclear and cellular datasets, we gathered 24 datasets having nuclei, cells or both, plus one additional local THG dataset. Thus, the total number of gathered datasets was 25 and the total number of images amounted to 12K. They are listed in table 1. Not all of them provide annotation, nor they provide the same types of annotation. However, they are all useful for training NU-Net as long as they contain nuclei or cells in images.

Dealing with the diversity of datasets, each with its format and preprocessing step, across many numerical experiments is a non-trivial MLOps task. To handle it, we used BioImageLoader [28].

2.3.2 Data pre-processing and augmentation

We carried out a set of data pre-processes to have consistent inputs format across different datasets. Due to the nature of microscopy and biological images, in particular of staining techniques, most datasets offer a single channel which tagged nuclei or cells. We selected those channels if we could and converted multichannel images to gray-scaled ones otherwise. Another process was to ensure consistent contrast of foreground and background pixels across datasets. While most datasets have object of interests in higher pixel values than background pixels, a few datasets have the opposite contrast. Therefore, we inverted contrast of these few datasets to match the others. This process turned out to be important since all the target images for the morphological loss had the same contrast.

Balancing volumes of each dataset was an inevitable step. Every dataset is different not only in the number of images, but in resolution of images, size of objects and density of objects. To account for all the datasets rather equally, we designed data augmentation protocols, which consist of random cropping, resizing and flipping for each dataset with appropriate parameters. Then we sampled im-

ID	Acronym	number	Annotation	Ref.	ID	Acronym	number	Annotation	Ref.
1	TissueNet	6,990	◯	[14]	14	S-BSST265	79	◯	[24]
2	BBBC041	1,328	△ (U)	[30]	15	FRUNet	72	◯	[13]
3	BBBC026	864	X (B,C)	[30]	16	BBBC016	72	X (B)	[30]
4	BBBC006	768	◯	[30]	17	BBBC018	56	◯	[30]
5	DSB2018	735	◯	[1]	18	TNBC	50	◯	[31]
6	BBBC021	240	X (B)	[30]	19	BBBC002	50	X (C)	[30]
7	BBBC039	200	◯	[30]	20	ComPath	30	◯	[26]
8	BBBC015	144	X (B)	[30]	21	UCSB	58	△	[9]
9	DigitPath	141	△	[21]	22	BBBC020	25	◯	[30]
10	MurphyLab	100	◯	[8]	23	BBBC007	16	◯	[30]
11	Cellpose	100	◯	[37]	24	BBBC008	12	◯	[30]
12	BBBC013	96	X (B)	[30]	25	LOCAL-THG	14	◯	-
13	BBBC014	96	X (B)	[30]	-	**Total**	**12,336**		

Table 1. List of nuclear and cellular datasets used to train NU-Net. Annotation availability and type are usually the most important factors when it comes to selecting datasets for machine learning, but not for NU-Net thanks to a self-supervised learning scheme. In total, 25 datasets were used, including a local dataset (LOCAL-THG), and the number of images amounted to 12,336. Symbols for annotation (◯: provide complete segmentation mask targets, △: partially annotated masks, X: do not provide mask targets; C: counts, B: biological labels, U: bounding boxes). Note that BBBC021 originally has massive 132,000 images and only a portion was used because data comes from the same platform and would have led huge imbalance.

ages to roughly match the relative sizes of nuclei and cells from one dataset to another. Last but not least, we chose the minimum number of samples among 25 datasets after considering data augmentation, and simply set it as a constant number of samples throughout all the datasets mainly to avoid big bias towards those having larger number of samples. Our rationale was that we did not need huge amount of one modality when our primary goal was to cover many modalities and the architecture was rather small. In the end, we randomly sampled 900 images as inputs for every epoch.

2.3.3 Blob-mask style images

As shown in figure 2, so-called blob-mask style images are merely binary segmentation masks. Therefore, in a sense, NU-Net is a neural style transfer network that mimics behavior of binary segmentation. We selected mask labels of DSB2018 [1], BBBC039 and BBBC006 datasets [30] and used them as targets for the morphological loss. DSB2018 dataset was introduced through a competition on online platform Kaggle and has been used in many computer vision models for biological images [34, 17, 7]. We selected DSB2018 because its wide inclusiveness of data sources. BBBC039 is one of sources of DSB2018 and represents typical cellular microscopy images. We included BBBC006, whose annotation was automated, to show that it is possible to use automated labels to train NU-Net.

3. Numerical experiments

We think of NU-Net as filling a new niche in biological image analysis and as such it does not belong to a clear existing class of deep learning models, like segmentation or image restoration models. Thus, classical metrics and comparison are hard to apply. To evaluate it, we will 1) use a simple contrast enhancement metric and compare with classical filters, 2) use a realistic biological image analysis quantitative workflow, to show how it can improve day-to-day bioimage informatics work 3) apply it to a range of openly available images, to qualitatively show its versatility.

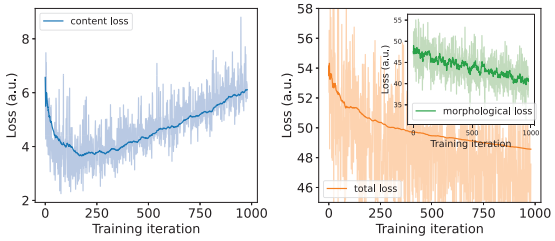

Figure 3. Ideal training loss curves. Total loss is a weighed sum of a content loss and a morphological loss. The desired trajectory would be to have a constantly decreasing morphological loss and a content loss to increase after a decent drop. Note that the curves were from a real case but smoothed out using exponentially weighted moving average (EWMA) to demonstrate the ideal forms. Transparent lines indicate raw data.

3.1. Training losses

Two main losses, the content loss and the morphological loss, are supposed to compete, because the content loss, being a generative loss, hinders outputs to become blob-mask style as per the morphological loss, and vice versa. Thus, balancing two losses was crucial, meaning their magnitudes should match one to the other. Hence, we implemented a dry-run before an actual training, where NU-Net goes through a few iterations of dry-run to estimate magnitudes of the losses and match them. During the training, a desired loss behavior is to have a constantly decreasing morphological loss and a content loss increasing in the middle of training. The rationale is that, in the beginning, we want NU-Net to learn images themselves through the content loss, and to learn how to enhance blobs according to the morphological loss afterwards. As a result, the ideal loss curves looks as in figure 3. In general, weight coefficient w (in equation 3) in range of 5 to 10 yielded good results. Training took about 1-2 hours on a workstation with a computational GPU for 20 epochs with batch sizes being 16 and 128 for the content loss and the morphological loss, respectively.

3.2. Comparison to classical filters

We compared results of NU-Net to existing filtering and contrast enhancement algorithms in terms of a contrast gain. Those were logarithmic lookup table (LUT) mapping, median filter (med), Gaussian filter (gaus), and difference of Gaussians (DoG) filter. Logarithmic lookup table mapping

	LUT	Gauss	Med	DoG	NU-Net
contrast gain avg.	12.9±24.3	4.9±6.7	8±8	90.6±100.4	99.5±121.5
change avg. (times)	x2.7±2.2	x2±1.1	x2.4±1.4	x14.6±11.7	x16±16.7

Table 2. Improvement of contrast ratio (CR) over raw images from several classical algorithms and NU-Net, as the before/after difference and the fold increase. Average over datasets of average over test images from each dataset. NU-Net shows slightly better contrast gains compared with DoG filter across the datasets. See Fig. 1 and 4 for the much more significant but harder to assess difference in image quality.

(LUT) is the simplest method among them, which makes use of logarithm to remap pixel values and to stretch value intervals. Median and Gaussian filters are both denoising filters using parameterized kernels. While denoising does not have a direct impact on contrast, it could enhance contrast by smoothing out noisy background signals, combined with value range stretching strategy based on percentile values (upper bound to 99.8% and lower bound to 2%).

We defined contrast ratio ($CR = \frac{\mu_{fg}}{\mu_{bg}}$) to quantify contrast change and to compare NU-Net to other filters, quantitatively. CR is a fraction of a mean value of foreground pixels over that of background pixels. Note that contrast gain depends a lot on the initial contrast of a given image. Note as well that this metric is blind to resulting image quality.

The result of contrast change can be found in table 2. In addition, a qualitative result on a sample from LOCAL-THG dataset can be found in figure 4. Contrast gain was significant once the algorithm had a notion of foreground and background. For examples, LUT mapping is a merely pixel-wise method, and kernel-based filtering methods like median filter (med) and Gaussian filter (gaus) made the distinction by blurring off the noise and the background as well as the foreground. They accentuated cells, but the contrast was barely improved because they did not make clear distinction between foreground pixels and background pixels.

Among the tested filters, the DoG filter showed the most comparable results to NU-Net. The DoG filter is a band pass filter, and assumes that the objects of interest have a consistent size. It enhances visibility of objects by subtracting two Gaussian filters with different kernel sizes. DoG is, in a way, aware of contents and could detect foreground from background.

Figure 1 compares details of NU-Net to those to DoG filter. While both DoG filter and NU-Net picked up objects, DoG suffered from blurring artifacts from its algorithmic nature, just as other rule-based algorithms suffer from their own pitfalls. Blurring merged objects together making it hard to recognize individual objects. Moreover, DoG accentuated other structures too in presence of noise. NU-Net, however, did not suffer from blurring and was more robust to noise. Additionally, NU-Net not just detected contents but also actively enhances contrast by pushing foreground pixels up towards 255 (maximum value in unsigned integer

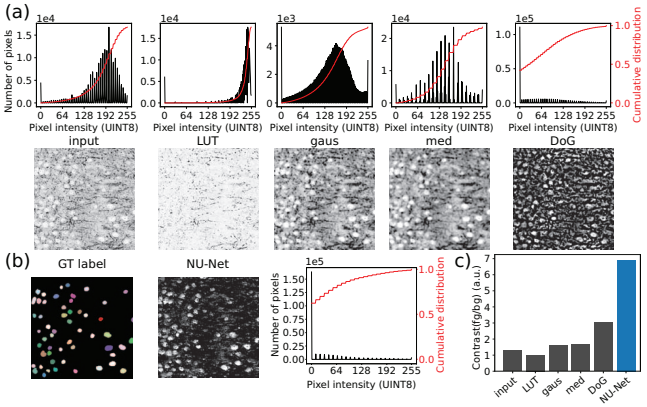

Figure 4. Comparing NU-Net to various filters. They were logarithmic lookup table (LUT) mapping, Gaussian filter (gaus), median filter (med) and difference of Gaussians (DoG) filter. (a) It shows the input image and the filtered images as well as their histograms. The red line shows the cumulative distribution of pixel values and gives a rough idea of image contrast. (b) It shows the ground truth (GT) mask and NU-Net's result. Though, its cumulative distribution looks similar to that of DoG, the qualitative result is much cleaner than DoG. (c) Quantified contrast ratio (CR) of foreground pixels over background pixels.

8-bit) and background pixels down towards 0 though the morphological loss. As shown in the histogram in figure 4, histogram of image after NU-Net had all over lower background pixel values than that after DoG or than those after any filters presented.

3.3. Evaluation on realistic workflows

Given the clear aim of having an impact on actual, daily use of image analysis across labs and facility, the clearest path to evaluate NU-Net is through studying its use within a classical, end-to-end workflow.

3.3.1 Evaluation on a training set dataset

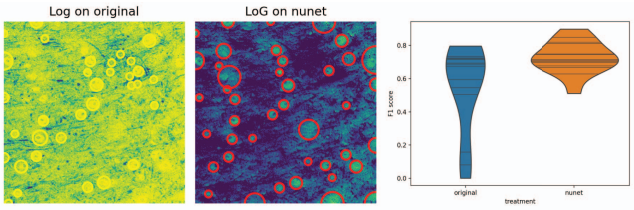

Figure 5. Use of NU-Net to improve a cell detection workflow of neuron in inverted contrast in a THG dataset. Left image: original image with detections from scikit-image blob-lob in red, right image: same with NU-Net filtered image. Right: distribution of F1 score across the dataset

First we study the effect of NU-Net in an object detection workflow on the LOCAL-THG dataset. To voluntarily use

a simple and straightforward workflow, similar to one that would be tried first in practice, we use the scikit-image[39] function `blob_log` as a point detection algorithm. It is actually not so simple as it scan through the Gaussian scale space of a Laplacian of Gaussian filter for stable detection, using a set threshold. The dataset comprises 14 2D images taken from a 3D Third Harmonic Generation (THG) microscopy volume of a slice of mouse brain. Due to the fact that THG signal is mainly from myelinated axons and not from cell bodies, the neuron are seen as noisy black spots in inverted contrast. A manual annotation of the positions of the cells was done, and the `blob_log` function was applied, with and without NU-Net. The results can be seen Fig. 5 and show a clear improvement of the detection, with the average F1 score across the dataset going from 0.54 to 0.72.

3.3.2 Evaluation on an unseen dataset

Obviously this assessment is biased since the LOCAL-THG dataset is part of the training set. As an unbiased assessment, we performed a similar numerical experiment on another dataset, unseen by the algorithm. We used a published dataset from [2], available on the Image Data Resource[41] https://idr.openmicroscopy.org under the accession idr0048, image ID 9846151, accessed through an OME-zarr endpoint (see code in section 5). It shows a section of a mouse cortex where the astrocytes are labeled with brainbow, and thus where the neuron are not labeled and again seen in inverted contrast. This problem is more challenging though since the background, consisting of astrocytes of various colors and intensities, is quite diverse. We randomly selected 15 crops out of the large volume and manually annotated them. This time, since the problem is more challenging and to be fair, the detection threshold was optimized independently for both conditions. The results are shown Fig. 6, with the mean F1 score rising from 0.49 to 0.61 from the use of NU-Net.

3.4. Controlling filtering magnitude

One big advantage of NU-Net as a filter is an ability to control the filtering magnitude. It can be achieved by adjusting the ratio between the morphological loss and the content loss. In practice, it is as simple as to manipulate the fraction of loss weights, that is w in equation 3, during training. Essentially, the larger the ratio w is, the more binarized the outputs become. An example of varying w is shown in figure 7.

We noticed that as w decreases, the content loss curves do not follow ideal curves anymore, which means that the resulted model would become closer to an antoencoder. Yet, due to presence of the morphological loss, resulted NU-Nets could still filter blobs. In the perspective of histogram of images, it also means that filtered output roughly keeps the original distribution as close as possible. In comparison, when w increases, outputs becomes more like those of binary segmentation.

3.5. Inference time and resources

NU-Net is light and fast. The size of a model is about as small as 6.5 MB. To run inference over a volume whose size is 130 x 400 x 400 pixels (depth, height, width), it consumed 1.3 GB of GPU memory, which can fit in laptop hardware. On a workstation, it took only 3 second (23 ms/plane), while it took around 20 seconds on CPU (150 ms/plane).

3.6. Assessing versatility on open images

To test pre-trained NU-Net and to assess its versatility, we explored IDR (Image Data Resource) [41], an online database where researchers share their biological image data, and selected a handful of image crops that contained nuclei or cells; they include various modality in various sample and conditions. A pre-trained NU-Net with $w = 8.0$ was applied, and the results are shown in figure 8. Overall, NU-Net could spot and enhance blobs, acting as denoising, textured or inhomogeneous background subtraction and selective object enhancement, depending on the image.

3.6.1 Graphical user interfaces

NU-Net is easily accessible and available via Napari viewer [36], a multidimensional image viewer. We made NU-Net available as a plugin of Napari. The plugin comes with simple graphical user interfaces (GUIs) and 5 pre-trained NU-Net with varying w value from 5.0 to 10.0. Users may adjust filtering magnitude via w value and run the process either on CPU or GPU.

4. Conclusion

We propose a nuclear and cellular filter, NU-Net, built on perceptual losses as a drop-in alternative to classical filters. Those are very useful in biological image analysis pipelines but limited by their simplicity and strong assumptions. We introduced a novel loss based on perceptual losses, which attempts to generalize a single style given multiple images. Training is self-supervised as it does not require paired labels contrary to supervised learning. Our model result in appreciable contrast gain against even complex and textured background while keeping details, proved useful in practical use cases and is versatile, giving good results across the diversity of biological images. By adjusting loss coefficients during training, NU-Net can exert different degrees of filtering power. We made NU-Net easy to use via a Napari plugin with GUIs, including 5 pre-trained models. Overall,

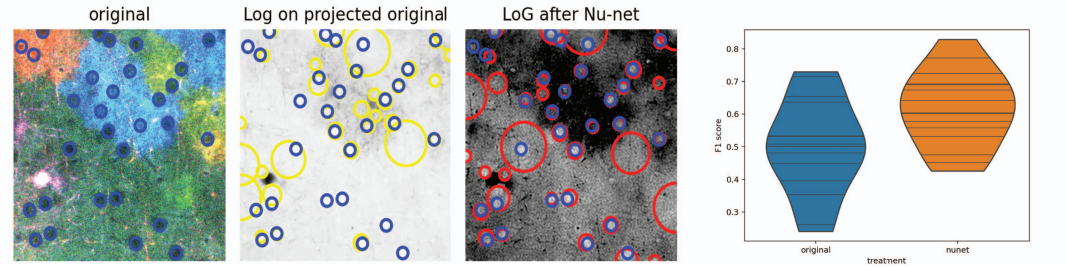

Figure 6. Use of NU-Net to improve a cell detection workflow of neuron in inverted contrast in a public brainbow astrocytes dataset. Left image: original image with annotation in blue, middle image: grayscale projection with detections from scikit-image blob-lob in red, right image: same with NU-Net filtered image. Right: distribution of F1 score across the dataset

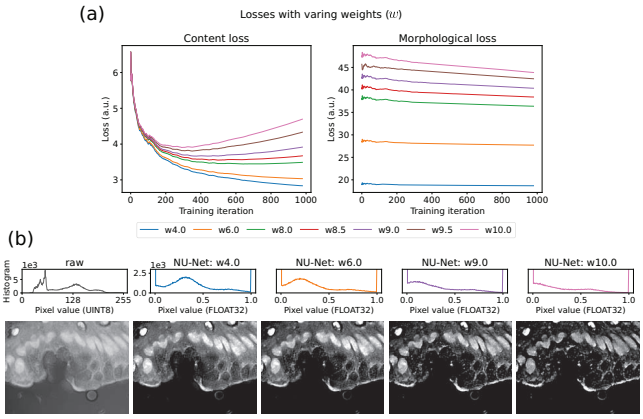

Figure 7. Controlling filtering magnitude by manipulating the loss coefficient w. (a) Loss curves with varying w values. Each morphological loss looks barely changing, but that is because w value is a multiplier of the morphological loss. (b) An example of applying NU-Nets trained with different magnitudes of weight (w). Note that w is a ratio of the morphological weight to the content weight in equation 3. Image source: DSB2018 dataset [1].

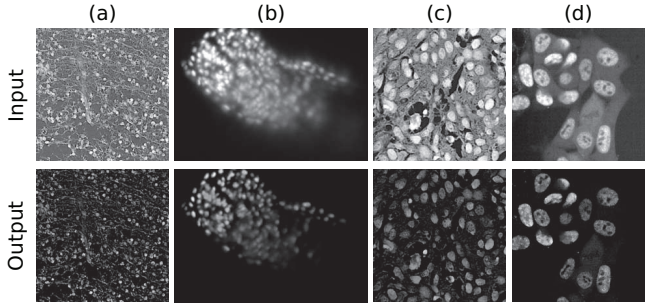

Figure 8. Application of a pre-trained NU-Net ($w = 8.0$) to nuclear and cellular images hosted on IDR (Image Data Resource) [41].

our aim is for NU-Net to be included into existing biological analysis workflow and allow for improved analysis of biological images across the board, to enable biological understanding.

5. Code and data availability

The source code of NU-Net resides at https://github.com/LaboratoryOpticsBiosciences/nunet and comes with codes to reproduce a large part of the figure of the paper. Additionally, the module is packaged and available through Python Package Index (PyPI) for easy installation and usage. A Napari plug-in is available at https://github.com/sbinnee/napari-nunet. Finally, the data used for training is used through https://github.com/LaboratoryOpticsBiosciences/bioimageloader, where information on how to obtain the various datasets is available.

6. Acknowledgement

We thank Tanguy Rolland for contributing to the Napari plugin. This work was supported by HOPE - European Research Council - Horizon 2020 programme (grant No 951330 HOPE), Morphoscope - Agence Nationale de la Recherche - ANR-EQPX-0029, and France BioImaging - Agence Nationale de la Recherche - ANR-10-INBS-04.

References

[1] 2018 Data Science Bowl. https://kaggle.com/competitions/data-science-bowl-2018. 5, 8

[2] Lamiae Abdeladim, Katherine S. Matho, Solène Clavreul, Pierre Mahou, Jean-Marc Sintes, Xavier Solinas, Ignacio Arganda-Carreras, Stephen G. Turney, Jeff W. Lichtman, Anatole Chessel, Alexis-Pierre Bemelmans, Karine Loulier, Willy Supatto, Jean Livet, and Emmanuel Beaurepaire. Multicolor multiscale brain imaging with chromatic multiphoton serial microscopy. *Nature Communications*, 10(1):1–14, Apr. 2019. 7

[3] Matthias Arzt, Joran Deschamps, Christopher Schmied, Tobias Pietzsch, Deborah Schmidt, Pavel Tomancak, Robert Haase, and Florian Jug. LABKIT: Labeling and Segmentation Toolkit for Big Image Data. *Frontiers in Computer Science*, 4, 2022. 1

[4] Mohammad Babaeizadeh and Golnaz Ghiasi. Adjustable Real-time Style Transfer. In *International Conference on Learning Representations*, Sept. 2019. 3

[5] Stuart Berg, Dominik Kutra, Thorben Kroeger, Christoph N. Straehle, Bernhard X. Kausler, Carsten Haubold, Martin Schiegg, Janez Ales, Thorsten Beier, Markus Rudy, Kemal Eren, Jaime I. Cervantes, Buote Xu, Fynn Beuttenmueller, Adrian Wolny, Chong Zhang, Ullrich Koethe, Fred A. Hamprecht, and Anna Kreshuk. Ilastik: Interactive machine learning for (bio)image analysis. *Nature Methods*, 16(12):1226–1232, Dec. 2019. 1, 3

[6] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the Opportunities and Risks of Foundation Models. *arXiv:2108.07258 [cs]*, Aug. 2021. 4

[7] Tim-Oliver Buchholz, Mangal Prakash, Alexander Krull, and Florian Jug. DenoiSeg: Joint Denoising and Segmentation. *arXiv:2005.02987 [cs]*, June 2020. 5

[8] L. P. Coelho, A. Shariff, and R. F. Murphy. Nuclear segmentation in microscope cell images: A hand-segmented dataset and comparison of algorithms. In *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 518–521, June 2009. 5

[9] Elisa Drelie Gelasca, Boguslaw Obara, Dmitry Fedorov, Kristian Kvilekval, and BS Manjunath. A biosegmentation benchmark for evaluation of bioimage analysis methods. *BMC Bioinformatics*, 10:368, Nov. 2009. 5

[10] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A Learned Representation For Artistic Style. *arXiv:1610.07629 [cs]*, Feb. 2017. 2, 3

[11] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image Style Transfer Using Convolutional Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016. 2, 3

[12] Golnaz Ghiasi, Honglak Lee, Manjunath Kudlur, Vincent Dumoulin, and Jonathon Shlens. Exploring the structure of a real-time, arbitrary neural artistic stylization network. *arXiv:1705.06830 [cs]*, Aug. 2017. 3

[13] Estibaliz Gómez-de-Mariscal, Martin Maška, Anna Kotrbová, Vendula Pospíchalová, Pavel Matula, and Arrate Muñoz-Barrutia. Deep-Learning-Based Segmentation of Small Extracellular Vesicles in Transmission Electron Microscopy Images. *Scientific Reports*, 9(1):13211, Sept. 2019. 5

[14] Noah F. Greenwald, Geneva Miller, Erick Moen, Alex Kong, Adam Kagel, Thomas Dougherty, Christine Camacho Fullaway, Brianna J. McIntosh, Ke Xuan Leow, Morgan Sarah Schwartz, Cole Pavelchek, Sunny Cui, Isabella Camplisson, Omer Bar-Tal, Jaiveer Singh, Mara Fong, Gautam Chaudhry, Zion Abraham, Jackson Moseley, Shiri Warshawsky, Erin Soon, Shirley Greenbaum, Tyler Risom, Travis Hollmann, Sean C. Bendall, Leeat Keren, William Graf, Michael Angelo, and David Van Valen. Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. *Nature Biotechnology*, 40(4):555–565, Apr. 2022. 1, 2, 4, 5

[15] Hongliang He, Zhongyi Huang, Yao Ding, Guoli Song, Lin Wang, Qian Ren, Pengxu Wei, Zhiqiang Gao, and Jie Chen. CDNet: Centripetal Direction Network for Nuclear Instance Segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4026–4035, 2021. 1

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*, Dec. 2015. 4

[17] Reka Hollandi, Abel Szkalisity, Timea Toth, Ervin Tasnadi, Csaba Molnar, Botond Mathe, Istvan Grexa, Jozsef Molnar, Arpad Balind, Mate Gorbe, Maria Kovacs, Ede Migh, Allen Goodman, Tamas Balassa, Krisztian Koos, Wenyu Wang, Juan Carlos Caicedo, Norbert Bara, Ferenc Kovacs, Lassi Paavolainen, Tivadar Danka, Andras Kriston, Anne Elizabeth Carpenter, Kevin Smith, and Peter Horvath. nucleAIzer: A Parameter-free Deep Learning Framework for Nucleus Segmentation Using Image Style Transfer. *Cell Systems*, 10(5):453–458.e6, May 2020. 1, 2, 5

[18] David Hörl, Fabio Rojas Rusak, Friedrich Preusser, Paul Tillberg, Nadine Randel, Raghav K. Chhetri, Albert Cardona, Philipp J. Keller, Hartmann Harz, Heinrich Leonhardt, Mathias Treier, and Stephan Preibisch. BigStitcher: Reconstructing high-resolution image datasets of cleared and expanded samples. *Nature Methods*, 16(9):870–874, Sept. 2019. 3

[19] Xun Huang and Serge Belongie. Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization. *arXiv:1703.06868 [cs]*, July 2017. 2, 3

[20] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-To-Image Translation With Conditional Adversarial Networks. In *Proceedings of the IEEE Conference*

on Computer Vision and Pattern Recognition, pages 1125–1134, 2017. 2

[21] Andrew Janowczyk and Anant Madabhushi. Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. *Journal of Pathology Informatics*, 7, July 2016. 5

[22] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. *arXiv:1603.08155 [cs]*, Mar. 2016. 2, 3, 4

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. 4

[24] Florian Kromp, Eva Bozsaky, Fikret Rifatbegovic, Lukas Fischer, Magdalena Ambros, Maria Berneder, Tamara Weiss, Daria Lazic, Wolfgang Dörr, Allan Hanbury, Klaus Beiske, Peter F. Ambros, Inge M. Ambros, and Sabine Taschner-Mandl. An annotated fluorescence image dataset for training nuclear segmentation methods. *Scientific Data*, 7(1):262, Aug. 2020. 5

[25] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2Void - Learning Denoising from Single Noisy Images. *arXiv:1811.10980 [cs]*, Apr. 2019. 2

[26] Neeraj Kumar, Ruchika Verma, Sanuj Sharma, Surabhi Bhargava, Abhishek Vahadane, and Amit Sethi. A Dataset and a Technique for Generalized Nuclear Segmentation for Computational Pathology. *IEEE Transactions on Medical Imaging*, 36(7):1550–1560, July 2017. 1, 5

[27] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2Noise: Learning Image Restoration without Clean Data. *arXiv:1803.04189 [cs, stat]*, Oct. 2018. 2

[28] Seongbin Lim, Xingjian Zhang, Emmanuel Beaurepaire, and Anatole Chessel. BioImageLoader: Easy Handling of Bioimage Datasets for Machine Learning. https://arxiv.org/abs/2303.02158v1, Mar. 2023. 4

[29] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context. *arXiv:1405.0312 [cs]*, Feb. 2015. 4

[30] Vebjorn Ljosa, Katherine L. Sokolnicki, and Anne E. Carpenter. Annotated high-throughput microscopy image sets for validation. *Nature Methods*, 9(7):637–637, July 2012. 5

[31] Peter Naylor, Marick Laé, Fabien Reyal, and Thomas Walter. Segmentation of Nuclei in Histopathology Images by Deep Regression of the Distance Map. *IEEE Transactions on Medical Imaging*, 38(2):448–459, Feb. 2019. 5

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv:1505.04597 [cs]*, May 2015. 1

[33] Johannes Schindelin, Ignacio Arganda-Carreras, Erwin Frise, Verena Kaynig, Mark Longair, Tobias Pietzsch, Stephan Preibisch, Curtis Rueden, Stephan Saalfeld, Benjamin Schmid, Jean-Yves Tinevez, Daniel James White, Volker Hartenstein, Kevin Eliceiri, Pavel Tomancak, and Albert Cardona. Fiji: An open-source platform for biological-image analysis. *Nature Methods*, 9(7):676–682, July 2012. 3

[34] Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. Cell Detection with Star-convex Polygons. *arXiv:1806.03535 [cs]*, 11071:265–273, 2018. 1, 5

[35] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556 [cs]*, Apr. 2015. 2, 3, 4

[36] Nicholas Sofroniew, Talley Lambert, Kira Evans, Juan Nunez-Iglesias, Grzegorz Bokota, Philip Winston, Gonzalo Peña-Castellanos, Kevin Yamauchi, Matthias Bussonnier, Draga Doncila Pop, Ahmet Can Solak, Ziyang Liu, Pam Wadhwa, Alister Burt, Genevieve Buckley, Andrew Sweet, Lukasz Migas, Volker Hilsenstein, Lorenzo Gaifas, Jordão Bragantini, Jaime Rodríguez-Guerra, Hector Muñoz, Jeremy Freeman, Peter Boone, Alan Lowe, Christoph Gohlke, Loic Royer, Andrea PIERRÉ, Hagai Har-Gil, and Abigail McGovern. Napari: A multi-dimensional image viewer for Python. Zenodo, May 2022. 7

[37] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: A generalist algorithm for cellular segmentation. *Nature Methods*, 18(1):100–106, Jan. 2021. 1, 2 , 5

[38] Jean-Yves Tinevez, Nick Perry, Johannes Schindelin, Genevieve M. Hoopes, Gregory D. Reynolds, Emmanuel Laplantine, Sebastian Y. Bednarek, Spencer L. Shorte, and Kevin W. Eliceiri. TrackMate: An open and extensible platform for single-particle tracking. *Methods*, 115:80–90, Feb. 2017. 3

[39] Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, and Tony Yu. Scikit-image: Image processing in Python. *PeerJ*, 2:e453, June 2014. 7

[40] Martin Weigert, Uwe Schmidt, Tobias Boothe, Andreas Müller, Alexandr Dibrov, Akanksha Jain, Benjamin Wilhelm, Deborah Schmidt, Coleman Broaddus, Siân Culley, Mauricio Rocha-Martins, Fabián Segovia-Miranda, Caren Norden, Ricardo Henriques, Marino Zerial, Michele Solimena, Jochen Rink, Pavel Tomancak, Loic Royer, Florian Jug, and Eugene W. Myers. Content-aware image restoration: Pushing the limits of fluorescence microscopy. *Nature Methods*, 15(12):1090–1097, Dec. 2018. 2

[41] Eleanor Williams, Josh Moore, Simon W. Li, Gabriella Rustici, Aleksandra Tarkowska, Anatole Chessel, Simone Leo, Bálint Antal, Richard K. Ferguson, Ugis Sarkans, Alvis Brazma, Rafael E. Carazo Salas, and Jason R. Swedlow. Image Data Resource: A bioimage data integration and publication platform. *Nature Methods*, 14(8):775–781, Aug. 2017. 7, 8

[42] Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised Feature Learning via Non-Parametric Instance-level Discrimination. *arXiv:1805.01978 [cs]*, May 2018. 3