JOURNAL OF
EMERGING INVESTIGATORS

Article

# The role of a mask - understanding the performance of deep neural networks to detect, segment, and extract cellular nuclei from microscopy images

**Soham Dasgupta[1], Stephen Turban[2]**
[1]Mallya Aditi International School, Bangalore, India
[2]Harvard University, Cambridge, Massachusetts, United States

## SUMMARY

**Cell segmentation is the task of identifying cell nuclei instances in fluorescence microscopy images. It plays a key role in biomedical analysis tasks like cell characterization, cancer cell identification, and gene expression measurement. There has been a recent proliferation of deep neural network-based object detection techniques for cell segmentation. However, previous literature does not suitably address the proper understanding of the different methods and the merits of the various neural architectures proposed. One key design point relates to the learning of segmentation masks and how neural methods go about detecting, segmenting, and extracting nuclei from unseen images. The goal of this paper is to benchmark the performance of representative deep learning techniques for cell nuclei segmentation using standard datasets and common evaluation criteria. We investigate whether for cell nuclei segmentation, learning the nuclei masks in parallel with boundary detection features of the image provides superior performance and significantly cleaner separation of nuclei from images. We further explore the trade-off between increased accuracy, achieved through more complex deep learning models, and the heavy requirements imposed on both computational resources and training times. We believe this paper establishes an important baseline for cell nuclei segmentation, enabling researchers to continually refine and deploy neural models for real-world clinical applications.**

## INTRODUCTION

Fluorescence microscopy is an optical microscopy technique that uses fluorescence to detect and image biological specimens as well as three-dimensional subcellular structures (1, 2). It is an indispensable tool in pre-clinical research, enabling the systematic dissection of life's molecular machines (3). Accurate three-dimensional segmentation of biological specimens is important to characterize and quantify cells, nuclei, and other microscopic structures. For example, it plays a key role in identifying cancerous cells which have differently shaped nuclei and measuring expression levels of specific genes (4). Accurate cell nuclei segmentation is also important for clinical research and medical analyses including the release of open-source evaluation frameworks (5), study of the observable characteristics of cells (6), segmentation of blood cells (7), and cell type classification (8). Segmenting the image and isolating nuclei from the rest of the cytoplasm and associated elements (like background tissue) remains arguably the most vital research challenge for practitioners (9). Therefore, automating this task with precise algorithms can potentially save several hours of manual effort per sample and eliminate human subjectivity. In recent years, the advent of deep neural networks and associated machine learning techniques have had transformational impact in the broader field of image classification and object detection.

Object detection has its roots in computer vision. The early machine learning algorithms for object detection typically define features on the image and subsequently use statistical techniques like support vector machines to classify the presence or absence of the object (13). The new age deep learning techniques, in contrast, typically attempt to do end-to-end object detection without specifically defining features. These deep learning methods are largely based on convolutional neural networks (CNNs). For general-purpose object detection in images, a breakthrough paper from 2012 introduced Alex Net (10) – an eight-layer CNN trained with millions of parameters on the ImageNet dataset (11), using over one million training images. Since then, even larger, and deeper networks have been trained for image classification and object detection.

Image segmentation extends the notion of object detection to mark the presence of an object through *pixel-wise masks* generated for each object in the image. The goal is to determine the shape of each object and extract it from the background. Note that, in segmentation, each pixel needs to be assigned a label corresponding to the class of interest (e.g., nucleus or not). Broadly, there are two types of approaches – *semantic segmentation*, where each pixel is evaluated for the same class label (e.g., nucleus) in the output mask, and *instance segmentation*, where each pixel has a unique identity/class label (e.g., nucleus 1, nucleus 2 etc.) in the output mask. **Figure 1** demonstrates the difference in output

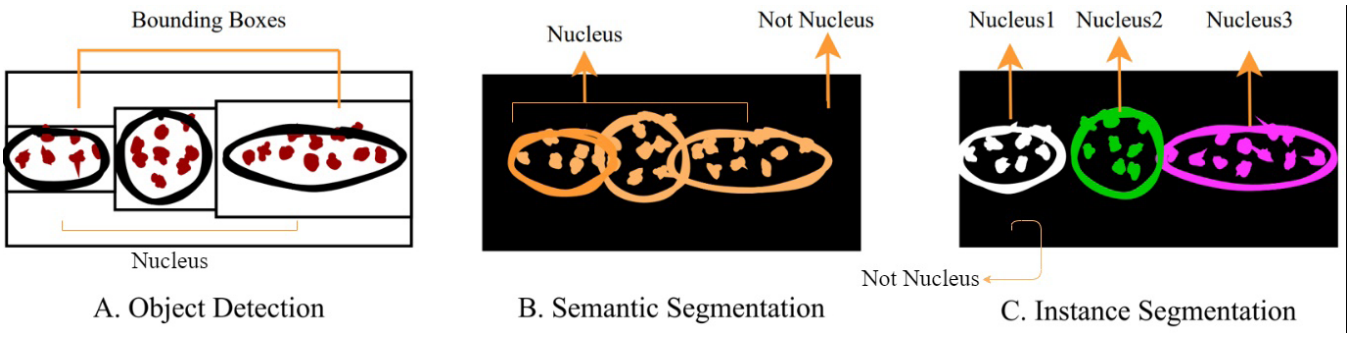


**Figure 1. Object Detection, Semantic Segmentation, and Instance Segmentation of cell nuclei.** This image compares the three different methods of cell nuclei segmentation. In object detection, each image pixel is classified whether it belongs to a particular class (nucleus) or not by grouping pixels together to form bounding boxes therefore reducing the problem to deciding if the bounding box is a tight fit around the object. Semantic segmentation is the task of clustering parts of images together which belong to the same object class: nucleus or not nucleus. Finally, instance segmentation involves assigning a unique label to each instance/object of a class, thereby being able to differentiate between multiple nuclei.

of object detection, semantic segmentation, and instance segmentation on a sample microscopy image. The U-Net encoder-decoder architecture, proposed by Ronneberger *et al*. (12), is arguably the most popular semantic segmentation approach giving superior results on biomedical images (13). Using substantially deeper networks and a residual learning framework, ResNet (14) is another semantic segmentation technique with significant improvements in accuracy, reported on the MS COCO (15) dataset for common object detection. Fast and Faster Region based-CNNs (R-CNNs) (16), on the contrary, are instance segmentation techniques that use fully convolutional networks to simultaneously predict object bounds and class labels at each position. Mask R-CNN extends Faster R-CNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition (17). Both methods have won several MS COCO competitions and represent state-of-the-art for object detection (15-17).

Masks play a key role in the task of segmentation. Specifically, a mask comprises the most important aspect of each input image for prediction of the original image, without (ideally) affecting the prediction accuracy. These segmentation neural models are used in combination with an input set of *training masks*. Training masks are manually annotated by biologists, and the segmentation task requires learning from the input masks and predicting a *target mask* for a new (previously unseen) image. The research challenge, therefore, is to determine masks for seen datasets, thereby evaluating their ability to generalize and predict segmentations on unseen images.

Different neural network-based techniques approach the segmentation process in different ways, which results in performance trade-offs in terms of prediction accuracies. At a high level, the process of generating a mask essentially entails labeling pixels in an image as belonging to the class of interest (e.g., a car, a human, or a cell nucleus) or not. Approaches by the authors of Faster R-CNN learn the candidate bounding boxes first and then from within the candidate boxes performs classification and separation of the object instances (16). In doing so, they learn the masks in two stages. In contrast, authors of the Mask R-CNN algorithm show that for the general problem of object detection, learning an object mask in parallel with the boundary detection features outperforms other approaches (17). The importance of identifying the object mask and the class label in parallel with detecting the class boundaries implies one single loss function to be optimized simultaneously across bounding box generation, classification, and mask generation.

We hypothesize that for the specific task of cell nuclei segmentation, learning the nuclei masks in parallel with the boundary detection features provides superior performance in terms of clean separation of nuclei from the images. While an increased accuracy in the task can be obtained by stacking multiple neural networks, it creates heavy requirements on computational resources. Therefore, we will also evaluate the trade-offs between performance and computing resources. We present a comprehensive study of four state-of-the-art representative neural networks on the Kaggle 2018 Data Science Bowl dataset (18). We believe this is a first-of-a-kind investigation in an emerging field of research and the experimental evidence presented herein can expedite the adoption and improvement of cell segmentation techniques.

## RESULTS

To evaluate the performance of different neural architectures on the nuclei segmentation task, we selected the *Intersection over Union (IoU)* measure as the common metric of measurement (24). IoU is known to be a reliable metric for measuring the overlap between two bounding boxes or masks and hence chosen for evaluating the efficacy of nuclei segmentation. If the prediction is completely correct, IoU would be equal to 100%. The lower the IoU, the worse the prediction (segmentation) results. The performance of the different models in terms of train and test accuracy of predicted masks, mean IoU, and training times are presented in **Table 1**.

**Table 1. Comparison of results on Kaggle dataset.**

| Method | Training Accuracy % | Test Accuracy % | Mean IoU % | Training Time (minutes) |
|---|---|---|---|---|
| Simple CNN | 93.10 | 93.70 | 62 | 135 |
| ResNet | 96.30 | 96.10 | 77 | 632 |
| U-Net | 99.75 | 98.00 | 83 | 238 |
| Mask R-CNN | 99.68 | 99.20 | 88 | 312 |

The training accuracy of a model signifies how well the model learns on "seen" data (images). Test accuracy signifies how well the model generalizes on "unseen data." Training times represent the time taken to create the model before it can infer. Lower training times are efficient, especially when computing resources are limited. They also enable faster retraining and redeployment in real world scenarios. As seen in **Table 1**, there is a wide variation in the mean (average) IoUs, ranging from 62% to 88%. The simple CNN has the least training time of about two hours, but also the minimum test accuracy of 93.70% and mean IoU of 62%. In contrast, the ResNet has a higher test accuracy than simple CNN of 96.30% and mean IoU of 77%, but also has the longest training time of nearly 11 hours (**Table 1**).

While CNNs work well with general-purpose object classification, they are demonstrably not suitable for cell nuclei segmentation. To address this problem, the next set of experiments focused on the U-Net encoder-decoder architecture for specialized semantic segmentation. U-Net delivers superior performance, in terms of test accuracy and IoU, when compared to CNNs (**Table 1**). For our experimental settings, it takes nearly four hours to train the U-Net network. The mean IoU score was 83% with a 99.75% training accuracy and a 98% test accuracy. The contraction and expansion layers serve as convolution and up sampling layers, respectively. Consequently, the image is recreated with segmented masks similar to the input size.

As highlighted in **Table 1**, it takes approximately five hours to train the Mask R-CNN network, thereby producing the highest mean IoU score of 88% and a test accuracy of 99.20%. The masks generated by Mask R-CNN were observed to be distinctly clearer and the nuclei cleanly separated from each other, compared to previous methods. **Figure 2** shows a probability density map of the segmented nuclei predictions produced by the Mask R-CNN model - with a value between 0 to 1 next to each nucleus, indicating the confidence of the model of a nucleus being present. The image is a 2D light microscopy image of stained nuclei from stomach lining tissue obtained under fluorescence microscopy conditions. The nuclei of the tissue have been stained primarily with DAPI or Hoechst. The aggregated very bright spots in the top left section represent multiple overlapping nuclei.

## DISCUSSION

The goal of our research is to understand various deep learning architectures and compare their efficiencies for cell nuclei segmentation. We found that the choice of the neural net architecture has a significant impact on the accuracy of the task. Hence, design choices are crucial for downstream diagnostics. We also found that instance segmentation techniques like Mask R-CNN outperform convolutional models - in line with our hypothesis that simultaneous learning of boundary detection features and masks works better for cell segmentation. Lastly, we also discovered a distinct trade-off between training times and prediction accuracies that is important for practitioners to consider.

Our first observation is that convolutional networks – both simple CNNs and deeper variations like ResNet, when trained end-to-end and on individual pixels – hold promise for the task of nuclei segmentation. Additionally, between the two classes of CNNs, ResNet has a higher accuracy because of its ability to learn deeper representations. ResNets take substantially longer time to train.

The second observation is that for fluorescent microscopy images with multiple nuclei, increasing the depth of neural networks (via multiple stacked layers in ResNet) vastly improve the accuracies of the predicted masks. Thus,

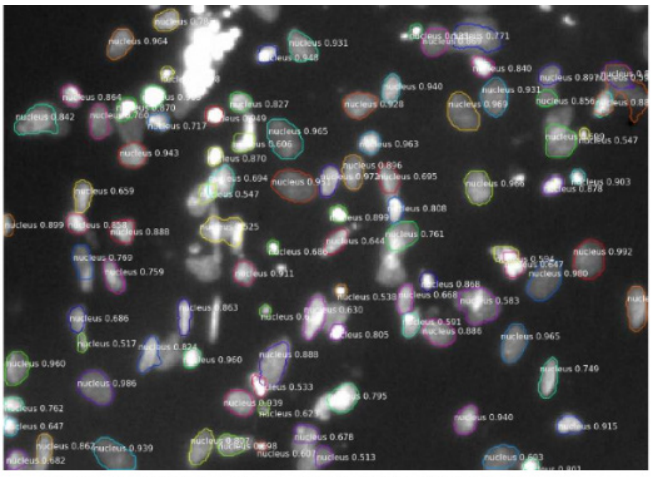

**Figure 2. Sample Mask R-CNN output for cell nuclei segmentation.** The image is a 2D light microscopy image of stained nuclei from stomach lining tissue obtained under fluorescence microscopy conditions. The nuclei of the tissue have been stained primarily with DAPI or Hoechst. The aggregated very bright spots in the top left section represent multiple overlapping nuclei. This is an example of instance segmentation where the image shows a probability density map of the segmented nuclei predictions produced by the Mask R-CNN model - with a value between 0 to 1 next to each nucleus, indicating the confidence of the model of a nucleus being present.

whenever longer training times are acceptable (or additional compute resources are available), the ResNet family of networks could serve as a viable backbone framework for convolutional networks in segmentation tasks.

The third observation highlights the fact that semantic segmentation frameworks like U-Net perform better and faster than CNNs on nuclei segmentation task. However, U-Net predicts a single mask for the entire image and needs complex post processing to separate a mask for each instance of the nucleus in the cell.

Next, we observe that the much-cited problem of vanishing gradients occurs with using the sigmoid activation function in Residual Networks, taking forever to train the network (10). This is more enhanced when the depth of the network increases.

Our final observation and arguably the most important one relates to the Mask R-CNN technique, which produces a pixel-level mask in parallel with the class label. Since Mask R-CNN decouples the problems of mask and class prediction, a binary mask is generated for each class with minimal competition (overlap) among classes; this is in stark contrast to convolutional techniques that attempt per-pixel multiclass categorization, coupling segmentation and classification. Our results suggest that networks like Mask R-CNN that learn the instance nuclei masks along with boundary features perform best for nuclei segmentation tasks as depicted in **Table 1**, with the network outperforming all other networks across various networks.

Further, as models become more complex, training times increase with better accuracies. While deep learning models demonstrate superior performance on their own, they can often be used in combination with one another. For instance, Hollandi *et al.* proposes combination of U-Net and Mask R-CNN, to predict masks (25). The algorithm outperformed the other 739 submissions to the 2018 Data Science Bowl - however, it was computationally exhaustive. Leveraging cloud-based computational services in such scenarios is therefore an important direction to pursue (26). Solving for the cell nuclei segmentation task under resource constraints creates new requirements on lean neural models that work in federated/distributed settings and address critical applications in biological cell research (27, 28).

## METHODS

Our work used the Kaggle dataset to serve as the benchmark for comparing deep learning models on the cell nuclei segmentation task, using IoU as the common metric of measurement (18). **Table 2** lists the models used in our study and the selected hyperparameters that produce optimal results on the Kaggle dataset. In our experimental setup Simple CNN, U-Net and Mask R-CNN models were implemented from scratch in Python with hyperparameter optimizations. The ResNet implementation was based on prior work (14). All models were trained using Keras deep learning API in TensorFlow - on a Google Colab instance with an NVIDIA Tesla T4 GPU and up to 12GB RAM/128GB disk space.

The 2018 Data Science Bowl Kaggle dataset contains a dataset of 670 training images and 65 test images. The dataset represents 22 different cell types including liver and stomach lining tissue. The images in the Kaggle dataset are two-dimensional light microscopy images of stained nuclei. Most of the dataset comprises fluorescent images with cells of different sizes and various types, primarily stained with DAPI or Hoechst. It also contains tissue samples stained with hematoxylin and eosin, displaying structures from a diversity of organs and animal models. These images were acquired under a variety of conditions and vary in the cell type, magnification, and imaging modality (brightfield vs. fluorescence). The dataset provides a training set of images containing the nuclei along with the corresponding masks and a test set of images for which one must generate the target masks. The nuclei segmentation methods were challenged to generalize on unseen data without any additional annotation or training.

We selected four representative neural network architectures, i.e., Simple CNN, ResNet, U-Net and Mask R-CNN and perform hyperparameter optimizations for each of the corresponding methods. These networks have been widely cited for their superior performance on general-purpose object detection and have their own unique approach to generating segmentation masks.

Using the Kaggle dataset, we evaluated the four neural network architectures of Simple CNN, ResNet, U-Net and Mask R-CNN. **Table 2** lists the selected hyperparameters that produced the optimal results for each of them. We next describe how each of these architectures were implemented for the cell nuclei segmentation task.

**Table 2. Hyperparameters for different deep learning methods.**

| Parameter | Simple CNN | ResNet | U-Net | Mask R-CNN |
|---|---|---|---|---|
| Epochs | 15 | 40 | 20 | 30 |
| Batch Size | 32 | 10 | 16 | 16 |
| Optimizer | Adam | Adadelta | Adam | SGD |
| Activation Unit | Sigmoid | ReLU | Sigmoid, ELU | ReLU |
| Learning rate | 0.1 | 0.1 | 0.01 | 0.01 |

### Simple convolutional neural networks

Simple convolutional neural networks are powerful tools that have demonstrated high performance on image classification tasks (19). For the image classification task, a CNN takes an input image, assigns importance (learnt weights and biases) to various aspects in the image and attempts to differentiate one from the other. It is composed of multiple building blocks - convolution layers, pooling layers, and fully connected layers, and is designed to adaptively learn spatial hierarchies of features through a backpropagation algorithm (20).

For the cell nuclei instance segmentation task, we leveraged the CNN to predict one of three possible class labels for each pixel – (i) Class 1: pixel belonging to the nuclei; (ii) Class 2: pixel bordering the nuclei; and (ii) Class 3: Background pixel, A simple CNN is the first deep learning model we evaluate using the following hyperparameters. We used one batch normalization layer to normalize the inputs to the next layer and reduce number of training epochs (overfitting) (29), followed by 6 fully connected dense layers and a 2-dimensional convolutional output layer, that gives us the predicted segmented mask for each image. The model is trained for 15 epochs with a batch size of 32. A stochastic gradient based 'Adam' optimizer (19) is used with a sigmoid activation function and a learning rate of 0.1. A kernel size of (3, 3) is used and a sigmoid activation function implemented for the output layer. The output is a high-resolution image with each pixel classified to a label (nucleus/not nucleus) thereby producing *one single mask* from the image.

### Residual Network (ResNet) (14)

One could in theory extend the eight-layer CNN to include more layers for better learning. However, the much-cited problem of vanishing gradients (10) occurs with the sigmoid function, taking forever to train the network. This is more enhanced when the depth of the network increases. The next variation of CNNs addresses this issue. The Residual Network (ResNet) is a variation of CNNs that uses a breakthrough architecture to train networks that are substantially deeper (150+ layers) than ones used previously (14). ResN*et al*low skip connections that mitigates the problem of vanishing gradients when using gradient descent and back-propagation. This model was the winner of ImageNet challenge (11) in 2015. Comprehensive empirical evidence demonstrate that these residual networks are easier to optimize and can gain accuracy from considerably increased depth (14). The depth of representations is extremely important for visual recognition tasks. We wish to briefly comment that the ResNet implementation uses transfer learning to leverage pre-trained weights (details in (14)) from the original model trained on the COCO common object dataset.

In this paper, we tune the ResNet for the cell nuclei segmentation task using the following hyperparameters. We implement the latest ResNet-which has 152 stacked neural nets referred to as ResNet-152. There are 20,000 steps involved in the training and checkpoints are created at the end of the 1,000$^{th}$ step. The model is trained for 40 epochs with a batch size of 10. A Rectified Linear Unit (ReLU) activation function is used with an Adadelta optimizer and a learning rate of 0.1 as mentioned in **Table 2**. Similar to the CNN, the output from the ResNet is a high-resolution image with each pixel classified to a label thereby producing a single image mask.

### U-Net (21)

Next, we consider the U-Net architecture. The U-Net encoder-decoder implementation was proposed by Ronneberger *et al*. developed along with TensorFlow (21). Initially developed for radio frequency interference mitigation, the network can be trained to perform image segmentation on arbitrary imaging data (22). The network starts with blocks of convolution layers and *max pooling* layers followed by deconvolution layers and recreating the image step by step. The output is similar to the input image size with segmented or labeled masks. U-Net is an end-to-end fully convolutional network but does not contain dense layers (unlike simple CNNs). We select U-Net as a candidate deep learning method primarily due to the state-of-art encoder decoder architecture. It follows a convolve-resample process where the convolution layers convolve through the image and get a detailed observation of the image. The model can be made more detailed by increasing the convolution layers. The deconvolution layers are generated by using the convolution transpose function in Tensorflow which performs the reverse of convolution (23). The output is generated by gradually recreating the image using the convolution layers used previously. Since the U-Net is a semantic image segmentation technique where each pixel is labeled to the corresponding class, the output is a high-resolution image, with labels, bounding box parameters and each pixel classified to a particular class. Note that, U-Net produces one single mask for the whole image which is the union of all masks. Post processing is required to split a single mask into a mask for each instance of the nuclei.

We implement the U-Net network using the following hyperparameters. We use 19 layers of convolution and 4 deconvolution layers. We also use 6 pooling layers to consider maximum values and reduce architecture size. The model is trained for 20 epochs with a batch size of 16. An 'Adam' optimizer is used with a sigmoid as well as Exponential Linear Unit (ELU) activation function and a learning rate of 0.01. Since the U-Net is a semantic image segmentation technique where each pixel is labeled to the corresponding class, the output is a high-resolution image, with labels, bounding box parameters and each pixel classified to a particular class.

### Mask R-CNN

The Mask R-CNN is an extension of the new genre of Fast and Faster R-CNN techniques, widely used for object detection tasks (16). For a given image, Faster R-CNNs simultaneously return the class label and bounding box coordinate for each

object in the image. The Mask R-CNN framework extends the Faster R-CNN, where a mask prediction branch is added in parallel to the class label and bounding box prediction branch. Thereby, in addition to the class label and bounding box coordinates for each object, Mask R-CNN also predicts the object mask. This only adds a small overhead to Faster R-CNN.

The Mask R-CNN is an instance segmentation network which can automatically predict both the bounding box and the pixel-wise mask of each object in an input image. For a given image, Mask R-CNN, in addition to the class label and bounding box coordinates for each object, also returned the object mask. In other words, each unique class instance (e.g., a cell nucleus) is simultaneously detected, separated, and extracted by this technique. The network generated 28x28 pixel masks for each detected region of interest. These masks were then passed through a threshold to get a complete binary mask for each nuclei instance. The parallel prediction of masks and class labels in Mask R-CNN underlines its flexibility and generality. This is also the reason why Mask R-CNN produces significantly superior performance compared to its peer methods.

We ran Mask R-CNN on the Kaggle dataset using the following hyperparameters. The network used six 2-D convolutional layers, six batch normalisation layers and six activation layers with a rectified linear unit activation function. A rectified linear unit (ReLU) activation function was used with a stochastic gradient descent optimizer as in **Table 2**. We train the model for 30 epochs with a batch size of 16 and learning rate of 0.01. The Mask R-CNN is an instance segmentation network which can automatically predict both the bounding box and the pixel-wise mask of each object in an input image.

## ACKNOWLEDGEMENTS

The authors would like to express their gratitude to Anish Raj Roy (Ph.D. scholar at Stanford University), for his invaluable mentorship and guidance, and the Lumiere Research Scholar Program, for giving us the opportunity to work on this challenging research problem.

**Received:** January 31, 2020
**Accepted:** May 15, 2021
**Published:** July 6, 2021

## REFERENCES

1. Vonesch, C., *et al*. "The Colored Revolution of Bioimaging." IEEE Signal Processing Magazine, vol. 23, no. 3, 2006, pp. 20–31., doi:10.1109/msp.2006.1628875.
2. Dunn, Kenneth W., *et al*. "Functional Studies of the Kidney of Living Animals Using Multicolor Two-Photon Microscopy." *American Journal of Physiology-Cell Physiology*, vol. 283, no. 3, 2002, doi:10.1152/ajpcell.00159.2002.
3. Lichtman, Jeff W, and José-Angel Conchello. "Fluorescence Microscopy." *Nature Methods*, vol. 2, no. 12, 2005, pp. 910–919., doi:10.1038/nmeth817.
4. Gharipour, Amin, and Alan Wee-Chung Liew. "Segmentation of Cell Nuclei in Fluorescence Microscopy Images: An Integrated Framework Using Level Set Segmentation and Touching-Cell Splitting." *Pattern Recognition*, vol. 58, 2016, pp. 1–11., doi:10.1016/j.patcog.2016.03.030.
5. Caicedo, Juan C., *et al*. "Evaluation of Deep Learning Strategies for Nucleus Segmentation in Fluorescence Images." Cytometry Part A, vol. 95, no. 9, 2019, pp. 952–965., doi:10.1002/cyto.a.23863.
6. Bougen-Zhukov, Nicola, *et al*. "Large-Scale Image-Based Screening and Profiling of Cellular Phenotypes." Cytometry Part A, vol. 91, no. 2, 2016, pp. 115–125., doi:10.1002/cyto.a.22909.
7. Tran, Thanh, *et al*. "Blood Cell Images Segmentation Using Deep Learning Semantic Segmentation." 2018 IEEE International Conference on Electronics and Communication Engineering (ICECE), 2018, doi:10.1109/icecome.2018.8644754.
8. Liu, Ying, and Feixiao Long. "Acute Lymphoblastic Leukemia Cells Image Analysis with Deep Bagging Ensemble Learning." 2019, doi:10.1101/580852.
9. Hiremath, P.S., and Y. Humnabad Iranna. "Automated Cell Nuclei Segmentation and Classification of Squamous Cell Carcinoma from Microscopic Images of Esophagus Tissue." 2006 International Conference on Advanced Computing and Communications, 2006, doi:10.1109/adcom.2006.4289885.
10. Krizhevsky, Alex, *et al*. "ImageNet Classification with Deep Convolutional Neural Networks." Communications of the ACM, vol. 60, no. 6, 2017, pp. 84–90., doi:10.1145/3065386.
11. Deng, Jia, *et al*. "ImageNet: A Large-Scale Hierarchical Image Database." 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, doi:10.1109/cvpr.2009.5206848.
12. Falk, Thorsten, *et al*. "U-Net: Deep Learning for Cell Counting, Detection, and Morphometry." *Nature Methods*, vol. 16, no. 1, 2018, pp. 67–70., doi:10.1038/s41592-018-0261-2.
13. Felzenszwalb, P F, *et al*. "Object Detection with Discriminatively Trained Part-Based Models." IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, 2010, pp. 1627–1645., doi:10.1109/tpami.2009.167.
14. He, Kaiming, *et al*. "Deep Residual Learning for Image Recognition." 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, doi:10.1109/cvpr.2016.90.
15. Lin, Tsung-Yi, *et al*. "Microsoft COCO: Common Objects in Context." Computer Vision – ECCV 2014, 2014, pp. 740–755., doi:10.1007/978-3-319-10602-1_48.
16. Ren, Shaoqing, *et al*. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks."

IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, 2017, pp. 1137–1149., doi:10.1109/tpami.2016.2577031.
17. He, Kaiming, *et al*. "Mask R-CNN." 2017 IEEE International Conference on Computer Vision (ICCV), 2017, doi:10.1109/iccv.2017.322.
18. Caicedo, Juan C., *et al*. "Nucleus Segmentation across Imaging Experiments: the 2018 Data Science Bowl." *Nature Methods*, vol. 16, no. 12, 2019, pp. 1247–1253., doi:10.1038/s41592-019-0612-7.
19. Guo, Tianmei, *et al*. "Simple Convolutional Neural Network on Image Classification." 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA), 2017, doi:10.1109/icbda.2017.8078730.
20. Huang, Gao, *et al*. "Densely Connected Convolutional Networks." 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, doi:10.1109/cvpr.2017.243.
21. Ronneberger, Olaf, *et al*. "U-Net: Convolutional Networks for Biomedical Image Segmentation." Lecture Notes in Computer Science, 2015, pp. 234–241., doi:10.1007/978-3-319-24574-4_28.
22. Akeret, J., *et al*. "Radio Frequency Interference Mitigation Using Deep Convolutional Neural Networks." *Astronomy and Computing*, vol. 18, 2017, pp. 35–39., doi:10.1016/j.ascom.2017.01.002.
23. Abadi, Martín. "TensorFlow: Learning Functions at Scale." Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming, 2016, doi:10.1145/2951913.2976746.
24. Rezatofighi, Hamid, *et al*. "Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression." 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, doi:10.1109/cvpr.2019.00075.
25. Hollandi, Reka, *et al*. "A Deep Learning Framework for Nucleus Segmentation Using Image Style Transfer." 2019, doi:10.1101/580605.
26. Shi, Weisong, *et al*. "Edge Computing: Vision and Challenges." *IEEE Internet of Things Journal*, vol. 3, no. 5, 2016, pp. 637–646., doi:10.1109/jiot.2016.2579198.
27. Verbraeken, Joost, *et al*. "A Survey on Distributed Machine Learning." *ACM Computing Surveys*, vol. 53, no. 2, 2020, pp. 1–33., doi:10.1145/3377454.
28. Chen, Mingqing, *et al*. "Federated Learning of N-Gram Language Models." Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL), 2019, doi:10.18653/v1/k19-1012.
29. Yuan, Xiaoyong, *et al*. "Generalized Batch Normalization: Towards Accelerating Deep Neural Networks." Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 1682–1689., doi:10.1609/aaai.v33i01.33011682.