Deep learning for automatic microscopy image analysis

Shenghua He
Washington University in St. Louis

Follow this and additional works at: https://openscholarship.wustl.edu/eng_etds

Part of the Engineering Commons

Recommended Citation
He, Shenghua, "Deep learning for automatic microscopy image analysis" (2021). McKelvey School of Engineering Theses & Dissertations. 725.
https://openscholarship.wustl.edu/eng_etds/725

This Dissertation is brought to you for free and open access by the McKelvey School of Engineering at Washington University Open Scholarship. It has been accepted for inclusion in McKelvey School of Engineering Theses & Dissertations by an authorized administrator of Washington University Open Scholarship. For more information, please contact digital@wumail.wustl.edu.
Deep Learning for Automatic Microscopy Image Analysis
by
Shenghua He

A dissertation presented to
The Graduate School
of Washington University in
partial fulfillment of the
requirements for the degree
of Doctor of Philosophy

December 2021
St. Louis, Missouri
# Table of Contents

List of Figures ............................................................................................................................... v
List of Tables ................................................................................................................................. ix
List of Abbreviations ...................................................................................................................... x
Acknowledgments ........................................................................................................................ xii
Abstract ........................................................................................................................................ xv

## Chapter 1: Introduction ............................................................................................................. 1

1.1 Microscopy Image Analysis ........................................................................................................ 4
1.2 Thesis outline and contributions ............................................................................................... 6

## Chapter 2: Deeply-Supervised Density Regression for Cell Counting ............................ 12

2.1 Introduction ............................................................................................................................... 12
2.2 Density Regression-based Cell Counting .................................................................................. 17
2.3 The Proposed Deeply-Supervised Density Regression .............................................................. 18
  2.3.1 Concatenated FCRN-based cell counting method ............................................................. 18
  2.3.2 Deeply-supervised C-FCRN training with auxiliary CNNs ............................................ 20
2.4 Datasets and method implementation ..................................................................................... 23
  2.4.1 Ground truth density map generation ............................................................................... 25
  2.4.2 C-FCRN and AuxCNN network parameter settings ....................................................... 27
  2.4.3 C-FCRN+Aux training and testing ................................................................................ 28
  2.4.4 Other methods for comparison ...................................................................................... 29
  2.4.5 Performance evaluation metrics ................................................................................... 32
2.5 Experimental results ................................................................................................................. 33
  2.5.1 Comparison between FCRN, C-FCRN, and C-FCRN+Aux ........................................... 34
  2.5.2 Comparison between C-FCRN+Aux detection-based methods ................................... 36
2.5.3 Density regression, count regression, and cell detection ........................................ 39
2.5.4 Benefits of using AuxCNNs to support C-FCRN training ................................................ 42
2.5.5 Computation efficiency comparison ...................................................................... 44
2.6 Discussion .................................................................................................................. 45
2.7 Conclusion ................................................................................................................. 47

Chapter 3: EfficientNet-based U-Net with Spatial Attention for Multi-Class
Nuclei Segmentation ............................................................................................................. 48
3.1 Introduction ................................................................................................................ 48
3.2 The Proposed AE-U-Net Based Method .................................................................... 52
  3.2.1 The proposed AE-U-Net ................................................................................. 52
  3.2.2 Transfer Learning ..................................................................................... 58
3.3 Experiments ............................................................................................................... 60
  3.3.1 Datasets ......................................................................................................... 60
  3.3.2 Network training .......................................................................................... 63
  3.3.3 Performance metrics .................................................................................... 63
  3.3.4 The overall performance of AE-U-Net .......................................................... 64
  3.3.5 Ablation Studies ........................................................................................... 65
  3.3.6 U-Net vs. AE-U-Net with different network sizes ........................................ 68
3.4 Discussion .................................................................................................................. 69
3.5 Conclusion ................................................................................................................. 72

Chapter 4: Multi-channel EfficientNet-based U-Net for Label-free 3D Phase-
to-fluorescent image translation ......................................................................................... 73
4.1 Introduction ................................................................................................................ 73
4.2 E-U-Net based phase-to-fluorescent Translation ......................................................... 77
  4.2.1 Overview of E-U-Net .................................................................................... 77
  4.2.2 U-Net structure ......................................................................................... 78
  4.2.3 EfficientNet ................................................................................................. 79
4.3 Experiments ............................................................................................................... 80
  4.3.1 Datasets ......................................................................................................... 80
  4.3.2 E-U-Net training .......................................................................................... 83
  4.3.3 Performance Metrics .................................................................................... 88
List of Figures

| Figure 1.1: | A comparison between conventional feature engineering-based MIA and deep learning-based MIA [26]. | 3 |
| Figure 2.1: | Framework of the proposed C-FCRN based automatic cell counting method. Different from the original FCRN, three shortcut connections (red lines) are established to connect the first three intermediate blocks to the fifth to seventh blocks, respectively. | 18 |
| Figure 2.2: | Framework of the AuxCNN-supported C-FCRN training process. The blue dash-line region indicates the C-FCRN. The orange dash-line region indicates the three AuxCNNs. EST and GT represents the estimated and ground truth density maps with varied resolutions, respectively. | 20 |
| Figure 2.3: | Example of constructing ground truth low-resolution density maps from an original ground truth of $128 \times 128$ pixels by summing up every local regions with size $2 \times 2$, $4 \times 4$ and $8 \times 8$ pixels, respectively. | 22 |
| Figure 2.4: | Example images of the four datasets used in this study. From left to right: Synthetic bacterial cells, Bone marrow cells, Colorectal cancer cells, and Human embryonic stem cells. | 24 |
| Figure 2.5: | Example of generating density map from a given cell centroid set. | 26 |
| Figure 2.6: | The MAE performances related to FCRN, C-FCRN and C-FCRN+Aux. | 34 |
| Figure 2.7: | Robustness related to FCRN, C-FCRN and C-FCRN+Aux for validation hESC images. | 35 |
| Figure 2.8: | The MRE performance evaluated on four different datasets. “C-FCRN+Aux” represents the proposed method in this study. No MRE results were reported for Arteta’s method. | 36 |
Figure 2.9: Estimated density or count maps from a sample image in each of the
four datasets. The panels from left to right on each row show the
cell images and the density/count maps estimated by the FCRN, the
Count-Ception, and the proposed method (C-FCRN+Aux), and the
associated ground truth density maps, respectively. ........................ 37

Figure 2.10: MAE performances related to detection-based methods. ................. 38

Figure 2.11: Robustness related to detection-based methods for bacterial cells
dataset .................................................................................. 38

Figure 2.12: Example results of three deep-learning detection-based cell counting
methods (Mask R-CNN, U-Net, and StructRegNet). Panels (a) and (b)
show the prediction results on the bacterial and bone marrow datasets,
respectively. The green cycles and red dots in each image represent the
ground truth annotations and the detected cell centroids, respectively. 40

Figure 2.13: Example prediction results based on the proposed C-FCRN+Aux
method and the detection-based method (StructRegNet). Here, “image”,”C-Aux”
and “RegNet” represent the processed image and the estimated density map using the “C-FCRN+Aux” method and the
computed proximity map using the “StructRegNet” methods. The
red dots represent the detected cell centroids based on the computed
proximity map, respectively ....................................................... 41

Figure 2.14: MAE performance related to density regression-based, count regression-
based, and cell detection-based methods ....................................... 41

Figure 2.15: Robustness related to the three groups of methods for bacterial cells
dataset .................................................................................. 42

Figure 2.16: Validation losses as the functions of epochs are plotted for the DRM
training on the four datasets. C-Aux and C-FCRN are abbreviations
of C-FCRN+Aux and C-FCRN-Only methods, respectively .......... 43

Figure 3.1: The overview of the AE-U-Net network for multi-class nuclei segmen-
tation. The major differences between an AE-U-Net and the standard
one are the EfficientNet used in encoding path and the spatial attention
modules applied on each skip connection. ................................. 54

Figure 3.2: Efficiency of EfficientNets compared to other methods [88]. .... 55

Figure 3.3: The network architecture of an EfficientNet-B0 and the MBConvX
module .................................................................
Figure 3.4: The schematic overview of an attention module [67]. The high-level features here are those from the decoding path; the low-level features are those from the encoding path. ............................................. 58

Figure 3.5: Transfer learning strategy used in the AE-U-Net training. ............... 59

Figure 3.6: Image examples from two datasets used in this study. The sizes of all the displayed images are $832 \times 832$ pixels..................................... 61

Figure 3.7: Phase image acquisition and ground-truth semantic segmentation map generation. (a) shows the phase imaging with computational specificity (PICS) imaging system. (b) shows the acquired phase image; (c) and (d) correspond to the fluorescent images from two biologically labeled channels; (e) shows ground-truth semantic segmentation maps generated from (c) and (d). ...................................................... 62

Figure 3.8: Pixel-wise F1 scores over each class............................................. 65

Figure 3.9: Ablation studies................................................................. 66

Figure 3.10: Visualized predictions for testing examples from HeLa and CHO datasets. The columns from left to right show the testing phase images, ground truth semantic maps, prediction results related to U-Net, those related to E-U-Net, and those related to AE-U-Net, respectively. The E-U-Net and AE-U-Net were trained with pre-trained EfficientNet-B0. 67

Figure 3.11: Network size under different settings. ........................................ 68

Figure 3.12: Average nuclei F1 scores related to different network sizes............. 68

Figure 4.1: Comparison between label-free imaging and Fluorescent imaging techniques. ............................................................. 74

Figure 4.2: Challenges related to 3D phase-to-fluorescence image translation....... 76

Figure 4.3: Multichannel E-U-Net for 3D phase to fluorescent image translation. 78

Figure 4.4: Network architectures of EfficientNet-B0 and EfficientNet-B7........ 80

Figure 4.5: Neuron example of size $300 \times 1774 \times 1774$ pixels...................... 81

Figure 4.6: Examples in spheroid cell and bead datasets. ............................ 82

Figure 4.7: 3D cell imaging pipeline [14]................................................... 83
Figure 4.8: Stopping rule metric for the E-U-Net training on neuron dataset. The training metric was defined as the Pearson’s $r$ between the predicted fluorescent image patches and their corresponding ground truth values; the validation metric was defined as the predicted image slices and their corresponding ground truths.

Figure 4.9: Stopping rule metric for the E-U-Net training on neuron dataset. The training metric was defined as the Pearson’s $r$ between the predicted fluorescent image patches and their corresponding ground truth values; the validation metric was defined as the predicted image slices and their corresponding ground truths.

Figure 4.10: Stopping rule metric for the E-U-Net training on bead dataset. The training and validation metrics were defined as the Pearson’s $r$ between the predicted image slices and their corresponding ground truths.

Figure 4.11: Validation results on the neuron dataset.

Figure 4.12: Prediction for unseen neuron stack.

Figure 4.13: Validation results on the spheroid cell dataset.

Figure 4.14: Prediction for unseen neuron stack.

Figure 4.15: Validation results on bead dataset.

Figure 4.16: Prediction for unseen neuron stack.

Figure 4.17: BA plots of E-U-Net with 1-plane input and that with 3-plane input on neuron and spheroids, respectively. The red dash lines indicate the mean of the differences of validation PCCs.

Figure 4.18: Fluorescence prediction with 1-plane input and that with 3-planes input for an example spheroid cell image slice.

Figure 4.19: BA plots of the E-U-Net and U-Net on validation neuron and spheroid cell image stacks, respectively. The red dash lines indicate the mean of the PCC differences.

Figure 4.20: Fluorescence prediction with 1-plane input and that with 3-planes input for an example spheroid cell image slice.
List of Tables

Table 2.1: Four datasets employed in the cell counting study. .......................... 23
Table 2.2: MAE±STD cell counting performance. ....................................... 33
Table 2.3: Computational efficiency comparison between different cell counting methods. .......................................................... 44
Table 4.1: PCC and PSNR performances on unseen neuron stacks................. 90
Table 4.2: PCC and PSNR performances on unseen spheroids cell stacks ........ 93
Table 4.3: PCC and PSNR performances on unseen bead stacks .................. 93
List of Abbreviations

**MIA** Microscopy Image Analysis

**CNN** Convolutional Neural Network

**FCRN** Fully Convolutional Regression Network

**FCNN** Fully Convolutional Neural Network

**DRM** Density Regression Model

**MSE** Mean Square Error

**MAE** Mean Absolute count Error

**MRE** Mean Relative count Error

**C-FCRN** Concatenated Fully Convolutional Regression Network

**C-FCRN+Aux** Concatenated Fully Convolutional Regression Network+Auxiliary CNNs

**E-U-Net** EfficientNet-based U-Net

**AE-U-Net** EfficientNet-based U-Net with Attention module

**CHO** Chinese Hamster Ovary

**QPI** Quantitative Phase Imaging

**PICS** Phase Imaging with Computational Specificity

**DIC** Differential Interference Contrast

**CV** Cross Validation
PCC Pearson Correlation Coefficient

PSNR Peak Signal-to-Noise Ratio
Acknowledgments

First and foremost, I would like to express my sincere gratitude to my advisors, Dr. Mark A. Anastasio and Dr. Hua Li, for their invaluable guidance and supports during the course of my PhD degree. I feel very fortunate to work with and learn from them.

I am also deeply grateful to the rest of my thesis committee: Dr. Tao Ju, Dr. Chengyang Lu, Dr. Alvitta Ottley, and Dr. Lilianna Solnica-Krezel, for their insightful comments and suggestions on thesis. Specially, I would like to thank Dr. Tao Ju for his insightful suggestions on my doctoral student seminar (DSS) talk and for chairing my thesis committee.

I would like to offer my special thanks to our collaborating groups: Dr. Solnica-Krezel’s group and Dr. Popescu’s group, who provided me with the opportunities to work on the interesting research problems in my thesis. Particularly, I would like to thank Dr. Kyaw Thu Minn from Dr. Solnica-Krezel’s group and Dr. Chenfei Hu, Mr. YuChen He, Dr. Xi Chen, Dr. Mikhail E. Kandel, and Mr. Michael John from Dr. Popescu’s group for their excellent work during our collaborations.

I would like to extend my thanks to all of my current and previous labmates in the Computational Imaging Science Laboratory and Medical Imaging and Bioinformatics Lab for the helpful discussions during these years. Particularly, I would like to thank Dr. Siddharth Muralidaran, Dr. Jian Wu, Mr. Zong Fan, Dr. Weizhou Zhou, Mr. Souyra Sengupta, and
Mr. Fu Li for their excellent work during the collaborations. I also would like to thank Dr. Yujia Chen, Dr. Thomas Matthews, Dr. Seonyeong Park, Dr. Frank J. Brooks, and Dr. Joemini Poudel for their insightful discussions.

Finally, I would like to thank my parents Heping He and Jinrong Wu for their unconditional love since I was born. Their love and supports have been always encouraging me to bravely and sturdily chase my dream. I would also like to thank my lovely girlfriend Tianyu Yan for her company over the past two years, which brought lots of joy and funs to my life.

Shenghua He

Washington University in Saint Louis

December 2021
Dedicated to my parents.
ABSTRACT OF THE DISSERTATION
Deep Learning for Automatic Microscopy Image Analysis

by
Shenghua He

Doctor of Philosophy in Computer Science
Washington University in St. Louis, 2021

Professor Tao Ju, Chair
Professor Mark A. Anastasio, Co-Chair

Microscopy imaging techniques allow for the creation of detailed images of cells (or nuclei) and have been widely employed for cell studies in biological research and disease diagnosis in clinic practices. Microscopy image analysis (MIA), with tasks of cell detection, cell classification, and cell counting, etc., can assist with the quantitative analysis of cells and provide useful information for a cellular-level understanding of biological activities and pathology. Manual MIA is tedious, time-consuming, prone to subject errors, and are not feasible for the high-throughput cell analysis process. Thus, automatic MIA methods can facilitate all kinds of biological studies and clinical tasks. Conventional feature engineering-based methods use handcrafted features to address MIA problems, but their performances are generally limited since the handcrafted features can lack feature diversity as well as relevancy to specific tasks. In recent years, deep learning, especially convolutional neuronal networks (CNNs), have shown promising performances on MIA tasks, due to their strong ability to automatically learn task-specific features directly from images in an end-to-end learning manner. However, there still remains a large gap between deep learning algorithms shown to be successful on retrospective datasets and those translated to clinical and biological practice. The major challenges for the application of deep learning into practical MIA problems include: (1) MIA tasks themselves
are challenging due to limited image quality, the ambiguous appearance of inter-class nuclei, occluded cells, low cell specificity, and imaging artifacts; (2) training a learning algorithm is very challenging due to the potential gradient vanishing issue and the limited availability of annotated images. In this thesis, we investigate and propose deep learning methods for three challenging MIA tasks: cell counting, multi-class nuclei segmentation, and 3D phase-to-fluorescent image translation. We demonstrate the effectiveness of the proposed methods by intensively evaluating them on practical MIA problems. The proposed methods show superior performances compared to competitive state-of-the-art methods. Experimental results demonstrated that the proposed methods hold great promise to be applied in practical biomedical applications.
Chapter 1

Introduction

The metabolism and activities of cells have been proved to play crucial roles in the pathophysiology of human diseases and the evolution of human life [9]. The analysis of cell activities and metabolism is thus critical for the cellular-level understanding of human diseases and human life, which can help with new drug discovery, clinical decision support, and further unveil the mystery of human life. Microscopy imaging techniques have been widely employed in cell analysis in clinical applications and biological research, since they permit the creation of high-resolution images that capture cell details, such as structures, functionality, and morphology, etc. The analysis of microscopy images to provide quantitative information regarding the observed cells is an important step for many biomedical applications [99]. Conventionally, this image analysis process was conducted by pathological or biological experts manually, which is time-consuming, tedious, and prone to large inter-observer variations. On the other hand, current microscopy imaging methods allow to produce vast amounts of data. Thus, automated methods for microscopy image analysis (MIA) are in great demand and have been actively studied over the past few decades.
Previously, the design of automatic MIA methods was mainly focused on building up automated pipelines that integrate separate steps including image processing, image feature extraction, model learning for specific tasks, and post-processing. This design strategy has been applied to various MIA problems, such as cell (nuclei) segmentation, cell (nuclei) classification, cell (nuclei) detection in 2D microscopy images. Though these methods have shown promising performance on these tasks, they have several major limitations. First of all, this pipeline heavily relies on biological or clinical experts to generate sophisticated handcrafted image features, such as curves, edges of cells, and intensity gradient, etc [4, 58]. The feature engineering process requires strong domain knowledge in order to generate reliable and representative features for the tasks of interest. This makes the designed method hard to adapt to different tasks when a new dataset comes. Furthermore, the handcrafted features are extracted with feature extraction algorithms that usually generate dedicated features that lack variety. Also, the features are usually generated without the awareness of the specific tasks of interest, so the features can contain irrelevant information that are not useful for the task at hand. The lack of variety and relevancy in the handcrafted features can largely degrade the performances of the learned models.

In recent years, deep learning techniques that use neural networks to learn patterns directly from raw image data have achieved remarkable success in computer vision tasks including image classification [22, 29, 30, 75, 76, 77, 88], image segmentation [39, 62, 73], and object detection [28, 89]. Deep learning techniques, especially convolutional neuronal networks (CNNs), have consequently been actively investigated in MIA tasks, which also rely heavily on the interpretation of images as computer vision tasks. In deep learning, image features are typically learned together with deep neural networks in an end-to-end manner based on specific tasks, as shown in Figure 1.1, and the extracted features are more relevant to the tasks of interest. As a result, deep learning models are generally able to achieve better performance
Figure 1.1: A comparison between conventional feature engineering-based MIA and deep learning-based MIA [26].

than conventional feature engineering-based methods. However, there still remains a large gap between the number of deep learning algorithms that have been shown to be successful on retrospective datasets and the number translated to clinical and biological practice.

The major challenges for the application of deep learning into practical MIA problems include: (1) MIA tasks themselves are challenging due to limited image quality, the ambiguous appearance of inter-class nuclei, occluded cells, low cell specificity, and imaging artifacts; (2) training a learning algorithm is very challenging due to the potential gradient vanishing issue and the limited availability of annotated images. In this thesis, we will explore advanced deep learning methods for a variety of MIA problems by tackling these challenges.
1.1 Microscopy Image Analysis

Microscopy image analysis (MIA) has been actively investigated by researchers who attempt
to automate the process of understanding the content of microscopy images and gain insights
for supporting cell studies or clinical decision-making. The common MIA problems include
cell (nuclei) detection, cell (nuclei) segmentation, cell classification, and image modality
translation, etc. In this thesis, we do not attempt to cover all of these topics. Instead,
we focus on only three of them: cell counting, multi-class nuclei segmentation, and image
modality translation. The three tasks are briefly described below.

The cell counting problem is the estimation of the number of cells (or nuclei) in microscopy
images. In the biomedical field, the number of cells is a very useful information for quantitative
cell analysis in cell studies or clinical diagnosis. For example, the number of bacterial cells
in an infected tissue can more precisely indicate the stage of diseases, thus helping doctors
make a personalized treatment strategy for patients. Automatic cell counting in microscopy
images can be challenging, due to the low signal-to-noise ratio (SNR), low cell contrast, and
complex irrelevant background in the images. In some practical scenarios, the cells in the
images can even be occluded with each other, which makes the task more challenging. In this
thesis, we explore advanced deep learning-based solutions to this problem. We will investigate
the problem in a variety of microscopy image modality including fluorescent images and
histopathological images.

The multi-class nuclei segmentation is a process of classifying each pixel in microscopy
images into different nuclei classes and background class. Addressing this MIA task can help
researchers identify and analyze the morphological and count changes of nuclei from different
classes to investigate how the cells react to discrete cures [43]. Automating this segmentation
process will alleviate the burden of doctors and biologists on tedious and time-consuming
manual labor, and allow them to focus more on solutions or remedies and improve the throughput of research and insights. Nevertheless, automatic multi-class nuclei segmentation is very challenging due to the presence of digital noise, low nuclei contrast, and complex and irregular background in microscopy images. Though CNN-based methods, such as the U-Net [73], have achieved impressive performance on classifying background and nuclei pixels, it is practically difficult to train a powerful CNN model that learns more discriminative features to distinguish pixels from different nuclei classes due to the ambiguous appearance between inter-class nuclei and the limited amount of available annotated images. In this thesis, we explore advanced deep learning-based approaches to address these challenges.

The *microscopy image translation* problem is the translation from one microscopy image modality to another. This task is greatly demanded when the target image modality is of great interest to researchers or clinical experts but there is a variety of limitations when it comes to the direct acquisition, such as long acquisition time, the introduction of side effects to cells being investigated, etc. The alternative is the translation from a different image modality (source image modality) that is relatively easier to acquire in practice. Image translation problems can include two types of translations. The first one is the translation of image style so that the translated image visually looks similar to the target image modality. The second one requires not only image styles but also pixel-level intensity is as close to the target image modality as possible. In this thesis, we focus on the second type of translation. Specifically, we will investigate the translation from 3D phase images to the corresponding 3D fluorescent images. Besides the common issues related to microscopy images, there are another two challenges in this task. First of all, phase images contain less cell specificity than fluorescent images, thus it relies on the translation process to reconstruct that information from the phase images. Second, the acquired 3D phase images contain serious artifacts due to the limitation of an imaging system. Therefore, the method must be able to clear these
artifacts to obtain artifact-free fluorescent images. Third, there exist serious noise in the ground truth fluorescent images, which can result in large uncertainty in the measured loss function and the difficulty of network training. In this study, we explore advanced deep learning-based methods to address this problem. Phase images with different cell structures will be investigated to validate our methods.

1.2 Thesis outline and contributions

The outline of the remaining chapters and our contributions in each of the topics are described as follows.

In Chapter 2, we present our proposed deeply-supervised density regression method for the cell counting problem. Our contributions to this topic are summarized below.

Contributions to the computer science field: Previously, the number of cells in microscopy images was commonly counted by cell detection and segmentation methods, which usually deliver poor results when the cells are seriously occluded in the images. Over the past decade, density regression-based methods have been proposed to count the number of cells by first estimating a density map of cell counts and then integrating the estimated density map. These methods allow for cell counting without cell detection and can potentially avoid cell detection challenges caused by cell occlusions, but their counting accuracy relies on the accuracy of density estimation. Recently, a fully convolutional regression network (FCRN) has been proposed to perform end-to-end density regression from images. There are two potential pitfalls related to the FCRN. First, the FCRN only uses high-level features for the estimation of densities. The lack of local information in these features results in a less accurate estimation of local density values. Besides, training a deep neural network generally
might encounter gradient vanishing issues due to the long path of back-propagation, which can result in the insufficient training of the early layers. To address these challenges, in Chapter 2, we present a deeply supervised-based density estimation method for automatic cell counting. Our contributions to the computer science field in this work are three fold. First of all, motivated by a U-Net [73], we redesigned the architecture of the FCRN by adding to it skip concatenations, which is referred to C-FCRN. This skip concatenation design allows the C-FCRN to fuse multi-level image features thus providing fine granular features for more accurate density estimation. Secondly, motivated by deeply-supervised nets [56], we proposed a deeply-supervised learning framework for the C-FCRN training. Specifically, several extra auxiliary CNNs (AuxCNNs) were set up to perform low-level density map estimation on the features of the intermediate layers. These AuxCNNs were jointly trained with the C-FCRN, which provides extra and direct supervision to the earlier layers in the C-FCRN to mitigate the gradient vanishing issue. This multi-task learning strategy also allows to sufficiently use the annotation information to regularize the network training, and potentially mitigate the over-fitting issue. To the best of our knowledge, this is the first work that utilized the deeply-supervised learning framework to train a DCNN for density map estimation and object counting, as of our conference paper [31] was published. Thirdly, we systematically validate our method on four challenging microscopy image datasets of different modalities and demonstrated the effectiveness of our proposed method. Lastly, we compared the proposed method to other state-of-the-art methods. Our method achieved superior cell counting performance than the compared methods. The works related to these contributions have been published in papers [31, 32, 33].

**Contributions to the biomedical field:** In this study, we provided a useful deep learning-based method for cell counting in practical biomedical applications. Specifically, we evaluated our method on three biomedical applications: counting bone marrow cells, counting colorectal
cancer cells, and counting hESC cells. All of them are clinically or biologically significant in practice. Specifically, the number of bone marrow cells can indicate the health status of a patient; the number of colorectal cancer cells can tell a doctor the level of cancer a patient has; the number of hESC cells can help analyze the stages of hESC differentiation in hESC studies. The experimental results related to the three cell counting tasks show that the proposed approach can predict the cell counts that are close to the ground truth counts. This demonstrates that our method can hold a great promise to be applied in practical biomedical applications. The work related to the application of our method on an hESC study has been published in a paper [64].

In Chapter 3, we will present our proposed EfficientNet-based U-Net with spatial attention module (AE-U-Net) for multi-class nuclei segmentation. The contributions to this topic are summarized as follows.

Contributions to the computer science field: The contributions to this field are three fold. First, we proposed a novel CNN-based approach for multi-class nuclei segmentation tasks based on our proposed compact but powerful AE-U-Net architecture and a transfer learning strategy. The AE-U-Net uses a U-Net as the basic architecture but uses a compact but powerful CNN model, EfficientNet, in the encoding path for feature extraction. This allows the AE-U-Net to have a strong capacity to learn more discriminative features to distinguish nuclei from different classes but maintain a relatively small network size. Besides, the AE-U-Net uses attention modules on skip connections to select salient local features and improve the quality of fused features in the decoding path. This powerful but compact AE-U-Net can mitigate the challenge of training a powerful segmentation model with a limited number of training images. In the transfer learning strategy, the EfficientNet [88] used in the AE-U-Net was pre-trained on an ImageNet dataset that contains millions of natural images for an image classification task. This transfer learning strategy greatly reduces the number
of training images required in network training and further address the training issue caused by data scarcity. Secondly, we conducted ablation studies on two challenging datasets to investigate the effectiveness of the EfficientNet, the spatial attention module, and the transfer learning strategy. Thirdly, we compared our proposed method with other state-of-the-art nuclei segmentation methods, and the results show that our proposed method consistently outperformed the compared methods on the two investigated datasets. The works related to these contributions have been published in papers [37, 40, 42].

**Contributions to the biomedical field:** Phase cell imaging techniques have been widely employed for live-cell studies since they are non-invasive to live cells. Multi-class nuclei segmentation in quantitative phase images (QPIs) is one of the fundamental steps in these applications. For example, in a drug testing task, the nuclei segmentation of cancer cells can provide information on nuclei morphology, cell count, and dry mass related to cancer cells with different states, which can help to analyze how cells react to the medicine and drug. Label-free cell viability assay is important for assessing the impact of drug, physical, or chemical stimulants, and other potential factors on cell dynamics. In this study, we evaluated our method in two label-free cell viability assay tasks, where the goal is to classify pixels in the phase images into “background” class and different viability states including “live”, “injured” and “dead”, which is essentially a multi-class nuclei segmentation task. The corresponding two datasets are a set of phase images of HeLa cells and that of phase images of Chinese hamster cells (CHOs), respectively. These two cells are biologically significant in biomedical applications. The experimental results on the two datasets demonstrated that our method can provide reasonable segmentation results, compared to the ground truths. This indicates that our method holds a great promise to be applied in label-free cell viability assay applications.
In Chapter 4, we will present our proposed Multi-channel E-U-Net-based approach for 3D phase to fluorescent image translation. Our contributions to this topic are described below.

**Contributions to the computer science field:** The contributions to this field are three fold. First of all, we investigated a 3D phase-to-fluorescent image translation problem and proposed an effective approach based on a multichannel EfficientNet-based U-Net (E-U-Net). In the proposed approach, a multichannel E-U-Net performs an end-to-end mapping from 3 neighboring phase image slices to the corresponding central fluorescent slice. This design allows an E-U-Net to exploit information from phase images acquired at multiple neighboring imaging planes to better predict the fluorescent image. In addition, compared to a standard U-Net, the E-U-Net uses a compact but powerful convolutional neural network (CNN) model, EfficientNet, as the encoder to extract representative features from phase images. This compact design allows mitigating the difficulty in training a powerful model when only a limited number of training data are available. Secondly, we evaluated our proposed E-U-Net-based approach on three 3D phase image datasets that are related to microspheres, hippocampal neurons, and liver cancer spheroids, respectively. To the best of our knowledge, this is the first work that uses deep neural networks to solve the 3D phase-to-fluorescent translation problem. The works related to this topic have been published in papers [13, 14].

**Contributions to the biomedical field:** Label-free imaging techniques have been widely employed for live cell studies, because they are cheap and quick, and they are non-invasive to live cells so that cells can be maintained in their natural state during cell imaging. This allows for the investigation of live cells over a long time. However, due to the lack of biological labeling, the cell specificity in the acquired images is very low, which limits its application to subcellular cell studies. In contrast, fluorescence imaging allows for acquiring images that have high specificity, but image acquisition can be time-consuming and expensive. On top
of that, the phototoxicity in the imaging process can cause the death of live cells, which makes it less suitable for long-term investigation of live cells. These motivate researchers to translate label-free images to fluorescence images for long-term live-cell studies. Recently, 3D cell imaging has gained increasing interest in the biomedical field, because it allows to better capture the interaction of cells and their biological surroundings. Also, it permits obtaining more cell details, like volume, surfaces, etc, which a 2D imaging system cannot provide. Therefore, 3D label-free translation is getting more and more important in live-cell studies. In this study, we investigated and evaluated our proposed method on the problem of 3D phase-to-fluorescent translation for neurons, spheroid cells, and microbeads. Experimental results show that the predicted 3D fluorescent images show great agreement with their corresponding ground-truth fluorescent images. The results demonstrated that our method can be potentially applied in biomedical applications that require 3D phase-to-fluorescent image translation processes.

Finally, we summarize the contributions in this thesis in Chapter 6.
Chapter 2

Deeply-Supervised Density Regression for Cell Counting

2.1 Introduction

Cell counting is the task of counting the number of cells in microscopy images. It is a basic but important step in biological studies and clinical applications. The number of cells in a microscopy image can indicate the presence of diseases [92], help differentiate tumor types [16], assist in understanding cellular and molecular genetic mechanisms [85, 103], and provide useful information to many other applications [55, 90]. Manually counting cells in microscopy images is tedious, time-consuming, prone to subjective errors, and not feasible for high-throughput processes in real-world biomedical applications. During the past decades, many automatic cell counting methods have been proposed [4, 6, 8, 15, 34, 63, 98]. However, designing efficient automatic methods with sufficient counting accuracy still remains a challenging task due to various image acquisition techniques, low image contrast, complex
tissue background, large variations in cell sizes, shapes, and counts, and significant inter-cell occlusions in two-dimensional (2D) microscopy images.

The reported automatic cell counting methods can be categorized into detection-based and regression-based methods. Generally, detection-based methods first determine the cell centroid locations and subsequently count them to estimate the number of cells [4, 15, 59, 98]. Therefore, the performance of these methods highly relies on the accuracy of cell centroid detection results. Traditional detection-based methods have been designed based on feature extraction [87], morphological processing [84], H-minima/maxima transform [84], Laplacian of Gaussian filtering [53], maximally stable extremal region detection [4], radial symmetry-based voting [71], or conventional supervised learning strategies [100]. Recently, deep learning strategies have shown superior ability of extracting informative image features and generating inferences in all kinds of medical image analysis tasks [15, 35, 36]. Several deep learning-based detection methods have been proposed [10, 12, 21, 54, 59, 91, 97, 99, 108]. For example, Falk et al. [21] trained a fully convolutional neural network (U-Net) to compute a probability map of cell existing in a given image. The number of cells can then be determined by searching for the local maxima on the probability map with a non-maxima suppression method. Xie et al. [97] applied the non-maxima suppression process to a dense proximity map for cell detection. The proximity map was produced by a fully residual convolutional network-based structural regression model (StructRegNet) and exhibits higher responses at locations near cell centroids to benefit local maximum searching. Tofighi et al. [91] used a prior-guided deep neural network for cell nuclei detection. In their method, nuclei shape a prior is employed as a regularizer in a model learning process to improve the cell detection accuracy. Liu et al. [59] trained a CNN model to determine the final cell detection result from the results generated by several traditional cell counting methods. The selection process was formulated as a maximum-weight independent set (MWIS) problem, a combinatorial
optimization problem that has been studied in many applications of clustering, segmentation, and tracking. Paulauskaite et al. [69] recently performed an experimental investigation of the Mask R-CNN method, which was proposed by He et al. [28], to detect overlapping cells with a two-stage procedure of determining potential cell regions and jointly classifying and predicting cell masks. The method was validated on fluorescence and histology images and showed promising results in detecting overlapping cells. However, it still remains difficult to detect cells that are highly occluded, densely concentrated, and surrounded by histopathological structures.

Compared to detection-based methods, regression-based cell counting methods have received more and more attention due to their superior performance on counting occluded cells [2, 5, 17, 23, 51, 58, 66, 93, 96, 101] Some regression-based methods learn a cell counter through a regression process directly without requiring cell detection. In these methods, the number of cells is the direct and only output, and no cell location information can be provided. For example, Khan et al. [51] and Xue et al. [101] learned a convolutional neural network-based cell counter from small image patches which can increase the number of training samples. The total number of cells across the whole image can then be obtained by summing those on image patches. These methods might suffer from redundant estimation issues across the patch boundaries, and might not be efficient since they have to infer for each image patch separately before cell counting. Differently, Cohen et al. [17] learned a cell counter with a fully convolutional neural network (FCNN). They utilized the “sliding window” mechanism associated with the convolutional layers of the FCNN to address the redundant counting issues across the overlapped regions among image patches. Their method counts the number of cells by directly inferring a count map for the whole image. The method performance might be affected by the sizes of sliding windows.
Other regression-based methods learn a spatial cell density regression model (DRM) across a full-size image instead of learning direct cell counters [23, 58, 61, 96]. In these methods, the number of cells can be obtained by integrating the regressed density map, and the local maxima in the density map can be considered as cell centroid locations. Therefore, both the number and the centroid locations of cells can be obtained. Conventional density regression-based methods learn DRMs from extracted handcrafted image features, in which the feature extraction is independent of the DRM learning. For example, Lempitsky et al. [58] used local features (e.g. scale-invariant feature transform (SIFT) features) to learn a linear DRM by use of a regularized risk regression-based learning framework. Differently, Fiaschi et al. [23] learned a nonlinear DRM based on regression random forest methods. In their method, image features computed by ordinary filter banks were employed as the model input. The performance of these methods relies on the effectiveness of feature extraction methods, that of the DRM learning algorithms, and the match between them.

Instead of using handcrafted image features to learn a DRM, some methods were proposed to integrate feature learning into end-to-end nonlinear DRM learning by use of deep convolutional neural networks. The learned end-to-end DRMs use images as their direct inputs to compute the corresponding density maps [61, 80, 96, 106]. As one of the pioneering works using this strategy, Xie et al. [96] proposed a fully convolutional regression network (FCRN) to learn such a DRM integrating image feature extraction and density map estimation for arbitrary-sized input images. By use of CNNs in feature extraction and model learning, their method demonstrated superior cell counting performance than conventional density regression-based methods, especially on microscopy images containing severely overlapped cell regions. Following Xie et al.’s work, Zheng et al. [106] trained an FCRN by incorporating a manifold regularization based on the graph Laplacian of the estimated density maps to
reduce the risk of overfitting. Liu et al. [60] employed a post-processing CNN to further regress the estimated density map to improve the accuracy of cell counting.

However, in the original FCRN work, the network layers of an FCRN are structured hierarchically and the output of each layer relies merely on the output of its direct adjacent layer. This restricts the FCRN to produce a more authentic density map for cell counting. In addition, the training of the original FCRN is based on a single loss that is measured at the final output layer, and all its intermediate layers are optimized based on the gradients back-propagated from this single loss only. The decreased gradients potentially trap the optimization of intermediate layers into unsatisfying local minima and jeopardize the overall network performance.

Recently, CNNs that concatenate multi-scale features by shortcut connections of non-adjacent layers have been reported and demonstrated promising performance than conventional hierarchical networks for many applications [19, 73]. In these concatenated network architectures, the multi-scale image features extracted by all the layers along the down-sampling path can be integrated into the input of the layers along the up-sampling path to further improve the model performance. Also, deeply-supervised (or deep supervision) learning strategies, aiming at enhancing the training of intermediate layers of designed neural networks by providing direct supervisions for them, have been proposed and have yielded promising performance for several computer vision tasks including image classification [56] and segmentation [20, 102]. To the best of our knowledge, deeply-supervised learning has not been employed in learning a density regression model for cell counting tasks except our preliminary work [31].

In this study, a novel density regression-based method for automatically counting cells in microscopy images is proposed. It addresses the two shortcomings that exist in the original FCRN by integrating the concatenation design and deeply-supervised learning strategy into
the FCRN. Specifically, the density regression model (DRM) is designed as a concatenated FCRN (C-FCRN) to employ multi-scale image features for the estimation of cell density maps from given images. The C-FCRN can fuse multi-scale features and improve the granularity of the extracted features to benefit the density map regression. It also facilitates the learning of intermediate layers in the down-sampling path by back-propagating the gradients conveyed via the shortcut connections. In addition, auxiliary convolutional neural networks (AuxCNNs) were employed to assist in training the C-FCRN by providing direct and deep supervision on learning its intermediate layers to improve the cell counting performance.

### 2.2 Density Regression-based Cell Counting

The salient mathematical aspects of the density regression-based counting process can be described as below. For a given two-dimensional microscopy image \(X \in \mathbb{R}^{M \times N}\) that includes \(N_c\) cells, the density map corresponding to \(X\) can be represented as \(Y \in \mathbb{R}^{M \times N}\). Each value in \(Y\) represents the number of cells at the corresponding pixel of \(X\). Let \(\phi(X)\) be a feature map extracted from \(X\), a density regression function \(F_\phi(\phi(X), \Theta)\) can be defined as a mapping function from \(X\) to \(Y\):

\[
Y = F_\phi(\phi(X); \Theta),
\]

where the vector \(\Theta\) parameterizes \(F_\phi\). The number of cells in \(X\) can be subsequently computed by:

\[
N_c = \sum_{i=1}^{M} \sum_{j=1}^{N} Y_{i,j} = \sum_{i=1}^{M} \sum_{j=1}^{N} [F_\phi(\phi(X); \Theta)]_{i,j},
\]

where \([F_\phi(\phi(X); \Theta)]_{i,j}\) is the computed density associated with the pixel \(X_{i,j}\). The key component of density regression-based methods is to learn \(F_\phi(\phi(X), \Theta)\) from \(\phi(X)\) and the corresponding \(\Theta\) [23, 58]. In the fully convolutional regression network (FCRN) [96], \(F_\phi(\phi(X), \Theta)\) can be simplified to \(F(X, \Theta)\) because it can be learned directly from \(X\).
2.3 The Proposed Deeply-Supervised Density Regression

2.3.1 Concatenated FCRN-based cell counting method

The proposed concatenated FCRN (C-FCRN) is shown in Figure 2.1, which integrates a concatenated neural network design and deeply-supervised learning strategy into the original FCRN. The C-FCRN network includes 8 blocks. Three concatenation layers (red lines in Figure 2.1) are established to connect the intermediate outputs along the down-sampling path to the input of the fifth to seventh blocks along the up-sampling path, respectively. This C-FCRN design integrates multi-scale features from non-adjacent layers to improve the granularity of the extracted features for density map regression, and subsequently improve the model performance on cell counting. The first three blocks in the C-FCRN are employed to extract low-dimension feature maps. Each of them includes a convolutional (CONV) layer, a ReLU layer, and a max-pooling (Pool) layer. The fourth block, including a CONV layer and
a ReLU layer, is used to further extract highly-representative features. The fifth to seventh blocks are employed to gradually restore the resolutions of feature maps while refining the extracted feature maps. Each of these blocks includes an up-sampling (UP) layer, a CONV layer, and a ReLU layer. The last block, including a chain of a CONV layer and a ReLU layer, is employed to estimate the final density map.

In C-FCRN, the CONV layer in each block is associated with a set of learnable kernels and is employed to extract local features from the output of its previous layer. The ReLU layer in each block is employed to increase the nonlinear properties of the network without affecting the receptive fields of the CONV layer by setting negative responses from its previous layer to zero while keeping the positive ones unchanged. Each Pool layer in the first three blocks performs a down-sampling operation on an input feature map by outputting only the maximum value in every down-sampled region in the feature map. Therefore, multi-scale informative features are extracted progressively along with the decrease of the spatial size of an input feature map. In contrast, each Up layer in the fifth to seventh block performs an up-sampling operation to gradually restore the resolution of the final estimated density map. This network design permits integration of feature extraction into the density regression process. Therefore, no additional feature extraction methods are required.

Given a to-be-tested image $X \in \mathbb{R}^{M \times N}$ and the trained density regression function $F(X; \Theta)$, the density map corresponding to $X$ can be estimated as $\hat{Y} = F(X; \Theta)$. Therefore, the number of cells in $X$ can be conveniently estimated based on the equation below:

$$\hat{N}_c = \sum_{i=1}^{M} \sum_{j=1}^{N} \hat{Y}_{i,j} = \sum_{i=1}^{M} \sum_{j=1}^{N} [F(X; \Theta)]_{i,j}, \quad (2.3)$$

where $[F(X; \Theta)]_{i,j}$ represents the estimated density of pixel $(i, j)$ in the $X$. 

19
2.3.2 Deeply-supervised C-FCRN training with auxiliary CNNs

Figure 2.2: Framework of the AuxCNN-supported C-FCRN training process. The blue dash-line region indicates the C-FCRN. The orange dash-line region indicates the three AuxCNNs. EST and GT represents the estimated and ground truth density maps with varied resolutions, respectively.

The task of training the C-FCRN corresponds to learning a nonlinear density regression function $F(X, \Theta)$ with parameters $\Theta$. However, training such a hierarchical and concatenated deep neural network by solving the corresponding highly non-convex optimization problem is a challenging task. Motivated by the deeply-supervised learning strategies [20, 56, 102], we employed three auxiliary convolutional neural networks (AuxCNNs) to provide direct supervision for learning the intermediate layers of the C-FCRN. The AuxCNN-supported C-FCRN training process is shown in Figure 2.2. Each AuxCNN contains two CONV-ReLU blocks, which estimate a low-resolution density map from each input feature map, respectively. The difference between the estimated density maps and the related ground truth are employed to support the C-FCRN training.
The Θ in the density regression function \( F(X, \Theta) \) can be re-defined as \( \Theta = (\Theta_1, \Theta_2, \Theta_3, \Theta_4) \), in which \( \Theta_1 \) represents the trainable parameters in the first four blocks, \( \Theta_2 \) represents the parameters in the 5-th block, \( \Theta_3 \) represents the parameters in the 6-th block, and \( \Theta_4 \) represents the parameters in the last 7-th and 8-th blocks, respectively. The outputs of the 4-th, 5-th, and 6-th blocks can then be denoted as \( \phi_1(X; \Theta_1) \), \( \phi_2(X; \Theta_1, \Theta_2) \), and \( \phi_3(X; \Theta_1, \Theta_2, \Theta_3) \). They are also the inputs of the 1-st, 2-nd, and 3-rd AuxCNNs, respectively. Given each input \( \phi_k (k = 1, 2, 3) \), the output of each AuxCNN is a low-resolution density map \( A_k(\phi_k; \theta_k) \), where \( \theta_k \) represents the parameter vector of the \( k \)-th AuxCNN.

\( F(X; \Theta) \) and \( A_k(\phi_k; \theta_k) \) are jointly trained through the minimization of a combined loss function [56],

\[
L_{crn}(\Theta, \theta_1, \theta_2, \theta_3) = L(\Theta) + \sum_{k=1}^{3} \alpha_k L_k(\Theta_1, \ldots, \Theta_k, \theta_k) + \lambda(\|\Theta\|^2 + \sum_{k=1}^{3} \|\theta_k\|^2), \quad k = 1, 2, 3,
\]

where \( L(\Theta) \) represents a loss function that measures the average mean square errors (MSE) between the estimated density map from the C-FCRN and the corresponding ground truth density map. \( L_k(\Theta_1, \ldots, \Theta_k, \theta_k) \) represents a loss function that measures the average MSE between a low-resolution density map estimated by the \( k \)-th AuxCNN and the corresponding low-resolution ground-truth (LRGT) density map. The parameter \( \alpha_k \in [0, 1] \) controls the supervision strength under the \( k \)-th AuxCNN. The parameter \( \lambda \) controls the strength of \( l_2 \) penalty to reduce overfitting and \( L_k(\Theta_1, \ldots, \Theta_k, \theta_k)(k = 1, 2, 3) \) and \( L(\Theta) \) are defined as:

\[
\begin{align*}
L_k(\Theta_1, \ldots, \Theta_k, \theta_k) &= \frac{1}{B} \sum_{b=1}^{B} \| A_k(\phi_k(X_b; \Theta_1, \ldots, \Theta_k); \theta_k) - Y_b^k \|^2, \\
L(\Theta) &= \frac{1}{B} \sum_{b=1}^{B} \| F(X_b, \Theta) - Y_b \|^2, \quad b = 1, \ldots B,
\end{align*}
\]
where $Y_b$ represents the full-size ground truth density map of the $b$-th training data $X_b$ of $B$ training images. Here, $Y^k_b$ represents the low-resolution ground-truth (LRGT) density map, which is generated from $Y_b$ by summing local regions in the original ground truth density map. An example of the summing process is shown in Figure 2.3.

Figure 2.3: Example of constructing ground truth low-resolution density maps from an original ground truth of $128 \times 128$ pixels by summing up every local regions with size $2 \times 2$, $4 \times 4$ and $8 \times 8$ pixels, respectively.

The loss $L_{cmb}$ can be numerically minimized via momentum stochastic gradient descent (SGD) methods [11] based on the Eqn. (2.6) shown below:

$$\begin{align}
\Delta \Theta_k^{(t+1)} &= \beta \Delta \Theta_k^{(t)} - (1 - \beta) \left( \eta \frac{\partial L_{cmb}^{(t)}}{\partial \Theta_k^{(t)}} \right), \\
\Theta_k^{(t+1)} &= \Theta_k^{(t)} - \Delta \Theta_k^{(t+1)},
\end{align}$$

where $\Theta_k^{(t)}$ is the updated parameters $\Theta_k$ at the $t$-th iteration; $\beta$ is a momentum parameter that controls the contribution of the result from the previous iteration; and $\eta$ is a learning
rate that determines the parameter updating speed. Since \( L_k(\Theta_1, \ldots, \Theta_k, \theta_k) \) only relates to \( \theta_k \) and \( \Theta_m \) \((m = 1, 2, \ldots, k)\), the gradient w.r.t the model parameters \( \Theta_k \) can be computed by:

\[
\frac{\partial L_{cmb}^{(t)}}{\partial \Theta_k^{(t)}} = \frac{\partial L^{(t)}}{\partial \Theta_k^{(t)}} + \sum_{m=k}^{3} \alpha_m \frac{\partial L_m^{(t)}}{\partial \Theta_k^{(t)}} + 2\lambda \Theta_k^{(t)},
\]

with the back-propagation algorithm [74]. The learned \( F(X; \Theta) \), represented by the trained C-FRCN model, can be used to estimate density maps for arbitrary-sized images because fully convolutional layers are employed in the C-FCRN.

In the rest of this paper, the proposed C-FCRN deeply-supervised by auxiliary CNNs during the training process is denoted as **C-FCRN+Aux**.

### 2.4 Datasets and method implementation

#### Datasets

Four microscopy image datasets were considered in this study, which are synthetic images of bacterial cells, experimental images of bone marrow cells, colorectal cancer cells, and human embryonic stem cells (hESCs), respectively. Table 2.1 illustrates the data details. Sample images from the four datasets are shown in Figure 2.4.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Bacterial cells</th>
<th>Bone marrow cells</th>
<th>Colorectal cancer cells</th>
<th>hESCs</th>
</tr>
</thead>
<tbody>
<tr>
<td># of images</td>
<td>200</td>
<td>40</td>
<td>100</td>
<td>49</td>
</tr>
<tr>
<td>Image size</td>
<td>(256 \times 256)</td>
<td>(600 \times 600)</td>
<td>(500 \times 500)</td>
<td>(512 \times 512)</td>
</tr>
<tr>
<td>Count statistics</td>
<td>174 ± 64</td>
<td>126 ± 33</td>
<td>310 ± 216</td>
<td>518 ± 316</td>
</tr>
</tbody>
</table>

Image size is represented by pixel, and count statistics is represented by mean and standard variations of cell numbers in all the images in each dataset.
Figure 2.4: Example images of the four datasets used in this study. From left to right: Synthetic bacterial cells, Bone marrow cells, Colorectal cancer cells, and Human embryonic stem cells.

**Synthetic bacterial cells:** This is a public synthetic dataset generated by Lempitsky et al. [58] by use of the method proposed by Lehmussola et al. [57]. This dataset contains 200 RGB synthetic fluorescent microscopy images of bacterial cells. The size of each image is $256 \times 256 \times 3$ pixels. The cells in these images are designed to be clustered and occluded with each other. This dataset is appropriate for testing the performance of the proposed method.

**Bone marrow cells:** This dataset includes 40 Hematoxylin-Eosin (H&E) stained bright-field RGB microscopy images, which were created from 10 images acquired from the human bone marrow tissues of 8 different patients [49]. The original image size of each H&E image is $1200 \times 1200 \times 3$ pixels. Each of the 10 original image was split into 4 images with the size of $600 \times 600$ pixels, following the process in Ception-Count [17]. The images in this dataset have inhomogeneous tissue background, and large cell shape variance.

**Colorectal cancer cells:** This dataset includes 100 H&E stained histology RGB images of colorectal adenocarcinomas acquired from 9 patients [83]. Knowing the number of colorectal adenocarcinomas can help with better understanding of colorectal cancer tumor for exploring various treatment strategies. Images in this dataset yield high inhomogeneous tissue region, noisy background, and large variance in the numbers of cells. This dataset is suitable to test the robustness and accuracy of given cell counting methods.
Human embryonic stem cells: This dataset contains 49 immunofluorescent images of human embryonic stem cells (hESC) that are differentiated into varied cell types [64]. The differentiation efficiency of the hESC population can be potentially observed based on the counted number of cells from each differentiation type in the images. The images in this dataset yield low image contrast and severe cell occlusion and clusters. In addition, high background noise exists in images.

2.4.1 Ground truth density map generation

Both the full-size and low-resolution ground truth (LRGT) density maps of the training images need to be constructed in order to train the C-FCRN and three AuxCNNs simultaneously. The full-size ground truth density map $Y$ of an image $X$ in the four data sets (described in Section 2.4) is defined as the superposition of a set of normalized 2D discrete Gaussian kernels [96]. The number of Gaussian kernels in $Y$ is identical to the number of cells in $X$, and each kernel is centered at a cell centroid in $X$ (as shown in Figure 2.5). Intuitively, the density map design can be interpreted in the perspective of microscopy imaging. Due to the limitation of imaging system and the point spread function (PSF), the intensity of each single pixel in image $X$ is affected by the PSF, and can be considered as a combination of the PSF-affected intensities of the pixel itself and its surrounding pixels. Accordingly, the density map is generated by simulating the imaging system and setting PSF as a Gaussian function. Integrating the density over $Y$ gives an estimate of the counts of cells in image $X$. This definition is also the same as the definition described in Lempitsky et al. [58], one of the compared methods in this study. This process would allow density regression-based methods to solve the problem of counting the overlapping cells. In the synthetic bacterial cell dataset, the ground truth cell centroids and numbers were automatically annotated during the image generation [58], while they are manually annotated on images in the other three experimental
datasets. The manual annotations for bone marrow cell images and colorectal cell images were provided by [49] and [83], respectively. The hESC annotation was performed by a graduate student under the supervision of and validation of a biologist expert [64].

Let \( S = \{(s_{xk}, s_{yk}) \in \mathbb{N}^2\} \) represent \( N_c \) cell centroid positions in \( X \), where \( k = 1, 2, ..., N_c \). Each \( Y_{i,j} \) in \( Y \) can be expressed as:

\[
Y_{i,j} = \sum_{k=1}^{N_c} G_\sigma(i - s_{xk}, j - s_{yk}),
\]

\[
G_\sigma(n_x, n_y) = C \cdot e^{-\frac{n_x^2 + n_y^2}{2\sigma^2}}, n_x, n_y \in \{-K_G, ..., 0, ..., K_G\}, \tag{2.8}
\]

where \( G_\sigma(n_x, n_y) \in \mathbb{R}^{(2K_G+1) \times (2K_G+1)} \) is a normalized 2D Gaussian kernel, and \( \sigma^2 \) is the isotropic covariance, \( K_G \) is an integer that determines the kernel size \( (2K_G + 1) \times (2K_G + 1) \) pixels, and \( C \) is a normalization constant. In light of the different sizes of cells in these four different datasets, the parameter \( \sigma \) was set to 5 pixels for bone marrow images and 3 pixels for images in the other three datasets, respectively. The parameter \( K_G \) was set to 10 pixels for all four image datasets.

Figure 2.5: Example of generating density map from a given cell centroid set.
Corresponding to the bi-linear interpolation performed by the Up layers in C-FCRN, the three low-resolution ground truth (LRGT) density maps \( Y^k \in \mathbb{R}^{M_k \times N_k} (k = 1, 2, 3) \) were generated from the original full-size ground-truth density map \( Y \in \mathbb{R}^{M \times N} \) by summing local regions with size of \( 8 \times 8, 4 \times 4, \) and \( 2 \times 2 \) pixels, respectively. Examples of ground truth of the images from the marrow bone cell dataset are shown in Figure 2.5, and the corresponding LRGT density map construction process is shown in Figure 2.3.

All images employed in the study were preprocessed by normalizing pixel values to a uniform range \([0, 1]\) in order to accelerate and stabilize the model training process [47]. The normalized images were subsequently employed as the inputs of the networks for both training and testing purpose. Random rotation with an arbitrary angle within \([0, 40^\circ]\) and random flipping on the training images was performed as a data augmentation operation to mitigate overfitting. During the training process, the ground truth density maps were amplified by 100 in order to force the C-FCRN and AuxCNNs to fit cell area rather than the background [96]. Correspondingly, the estimated density maps estimated from the testing image were scaled back by a factor of 0.01 before counting cell numbers.

### 2.4.2 C-FCRN and AuxCNN network parameter settings

The convolution kernel size in the first 7 blocks of C-FCRN was set to \(3 \times 3\), while that in the last block was set to \(1 \times 1\). The numbers of kernels in the first to 8-th CONV layers were set to 32, 64, 128, 512, 128, 64, 32, and 1, respectively. The pooling size in each pool layer was set to \(2 \times 2\), and the Up layers performed bi-linear interpolation. The size of the C-FCRN input image was set to \(128 \times 128\) pixels, so did the output density map. Three AuxCNNs yield the similar network structures, in which the kernel size of the first block in AuxCNN was set to \(3 \times 3\) and the number of kernels was set to 32, while that in the second block were set to \(1 \times 1\) and 1, respectively.
2.4.3 C-FCRN+Aux training and testing

Six thousand epochs were employed for model training, and that can permit the convergence of the training process in this study. In each training epoch, 100 image patches of 128 × 128 pixels were randomly cropped from each image for training. All the cropped image patches and their corresponding density maps were employed for training DRMs in the following epoch. The weight vector in the combined loss function $L_{cmb}(\Theta, \theta_1, \theta_2, \theta_3)$ in Eqn. 2.4 was set to $(\alpha_1, \alpha_2, \alpha_3) = (\frac{1}{64}, \frac{1}{16}, \frac{1}{4})$, considering that the task of a higher-resolution density estimation is more correlated to the task of original density estimation task. A momentum SGD method was used to minimize the combined loss function for jointly training the FCRN and AuxCNNs. The learning rates for training the C-FCRN+Aux were determined by operating a line search in a set of values $\{0.05, 0.01, 0.005, 0.0001, 0.0005, 0.001\}$ and selecting the one that results in the lowest validation loss. Other hyper-parameters were set to the fixed values of $\beta = 0.99$, $\lambda = 0.01$, and batch size = 100 considering the variations of these hyper-parameter values did not significantly improve the training performance based our trials. All the network parameters in the C-FCRN+Aux were orthogonally initialized [79].

The model performance was investigated by use of 5-fold cross validation on all four image datasets. When conducting cross validation on one of the four image datasets, the image dataset was randomly split into 5 folds of images for model training and validation. Specifically, every time, 4 of them were employed as the training dataset and the rest one as the validation dataset. Repeat the process for 5 times until each fold of data was used as validation dataset once. The average validation performance over the five times were measured as the evaluation result.

The proposed C-FCRN+Aux was implemented by use of python programming language with libraries including Python 3.5, NumPy 1.14, Keras 2.0, and Tensorflow 1.3.1. Model training
and validation were performed on a Nvidia Titan X GPU with 12 GB of VRAM and several Intel(R) Xeon(R) CPUs with E5-2620 v4 @ 2.10GHz.

2.4.4 Other methods for comparison

The proposed method (denoted as C-FCRN+Aux) was compared to other eight state-of-the-art methods, which include four regression-based counting methods [17, 58, 96], and four detection-based counting methods [4, 21, 28, 97].

Those four to-be-compared regression-based counting methods include the original FCRN method [96], the C-FCRN without AuxCNNs-supporting training (denoted as C-FCRN-only), the Count-Ception [17] method, and the Lempitsky’s method [58]. The original FCRN and the C-FCRN-only methods are nonlinear density regression-based methods. The Count-Ception [17] method is a nonlinear counter regression-based method, which employs a fully convolutional neural network (FCNN) to perform redundant cell counting in each overlapped local region and average out the estimated results to obtain the final cell count. The Lempitsky’s method is a linear density regression-based method, which learns the DRM by use of a regularized risk linear regression. Its hyper-parameter settings can be found in [58].

The loss functions for training the FCRN and C-FCRN were defined as the MSE between the ground truth density maps and the estimated density maps measured in a batch of training data. Differently, the loss function in the Count-Ception method was specified as the mean absolute error between the ground truth and the estimated count maps. The ground truth count map was generated according to its definition in the literature [17]. A momentum SGD method was used to minimize the loss functions in all these three methods. The learning rates and other hyper-parameters for model training in these methods were determined in the
same way as those were described in Section 2.4.3. All the network parameters in FCRN and C-FCRN-only were orthogonally initialized [79]; while those in the Count-Ception model were initialized by Glorot weight initialization [17]. The local region size in the Count-Ception was set to $32 \times 32$ as suggested in the literature [17].

The four referred detection-based counting methods include three deep-learning methods, StructRegNet [97], U-Net [21] and Mask R-CNN [28], and the Arteta’s method [4]. In the detection-based cell counting methods, the number of cells is determined by the number of detected cell centroids or cell regions. The StructRegNet used a fully residual convolutional network to regress a dense proximity map that exhibits higher responses at locations near cell centroids. Then the thresholding and non-maximum post-processes were employed to count the number of cell centroids. Differently, the U-Net method employed a U-Net to predict a cell probability map, and count cell centroids from it. The mask R-CNN method detects the cells by first detecting possible cell regions and then jointly predicting and segmenting these regions to get cells. The thresholds for the post-processes were tuned by visually checking detection results for two random validation images. The to-be-compared Arteta’s method [4] aims to segment an image into non-overlapped cell regions by use of a conventional machine learning technique. The results related to Arteta’s method on the bacterial dataset was referred to the literature [4].

The experiment settings related to the three deep learning detection-based counting methods are described as below. The StructRegNet model was built up based on the instructions presented by Xie et al. [97]. The ground truth proximity map was generated by a exponential function defined as:

$$
M(u, v) = \begin{cases} 
\frac{e^{\alpha(1 - \frac{D(i,j)}{d}) - 1}}{e^\alpha - 1}, & D(i,j) \leq d, \\
0, & D(i,j) > d,
\end{cases}
$$

(2.9)
where $D(i, j)$ is the Euclidean distance from a pixel $(i, j)$ to its closest annotated cell centroid; $d$ is a distance threshold and $\alpha$ is the decay ration, and both of them are used to control the shape of this exponential function. As suggested in literature [97], $\alpha = 3$, $d = 15$ was set in this study; the loss function for model training was a weighted MSE between the ground truth and estimated proximity map measured in a training batch. In this loss function, pixels closer to cell centroids were assigned to higher weights than those far-away pixels, and obtained more attention in the model training.

Although the task in this study is to annotate cell centroids, considering that the original U-Net method [73] requires fully annotation of complete cell masks, we reformulated the cell counting task as a segmentation problem in order to adapt the U-Net model to infer a segmentation map containing a small 2D disk at each cell centroid for each image, as suggested by Falk et al. [21]. When generating the ground truth segmentation maps, the radii of the 2D disks were set to 4 pixels, 8 pixels, 5 pixels and 3 pixels for the bacterial cell, bone marrow cell, colorectal cancer cell and hESC datasets, respectively, based on the average cell size of each dataset. The U-Net was trained by minimizing a binary cross-entropy loss with a momentum SGD method. The learning rates were determined by operating a line search in a set of values $\{0.05, 0.01, 0.005, 0.0001, 0.0005, 0.001\}$ and selecting the one that results in the lowest validation loss. Other hyper-parameters were set to the fixed values of $\beta = 0.99$, $\lambda = 0.01$, and batch size $= 100$. All the network parameters in the U-Net were orthogonally initialized. The same adaptation was performed for the Mask R-CNN method, except that a separate segmentation map was generated for each cell. For example, a set of $N_c$ separate segmentation maps were prepared as the ground truth for an image containing $N_c$ cells. ResNet-101 was chosen as feature extraction network in the Mask R-CNN model, since it yields better performance than the ResNet-50. The image scaling factor parameter was set to 2. The model was trained with image patches of $512 \times 512 \times 3$ pixels that were
randomly cropped from the scaled images in the training mode, and then tested on the whole scaled images. The sizes of anchors related to the region proposal networks for the bacterial cell dataset and the bone marrow cell dataset were set to \{8, 16, 32, 64\} and \{8, 16, 32, 64, 128\}, respectively. The Mask R-CNN model was trained by jointly minimizing the bounding box loss, classification loss, and segmentation loss. A stochastic gradient descent method was employed to minimize the losses. The batch size and learning rate were set to 4 and 0.001, respectively. The other parameter settings can be found in the repository [1].

The implementations of the six to-be-compared deep learning-based methods, including the FCRN, C-FCRN-only, Count-Ception, U-Net, Mask R-CNN, and StructRegNet, were based on the same Python, Tensorflow and Keras libraries as described in Section 2.4.3. In addition, the buildup of Mask R-CNN model was based on an open-sourced repository [1]. A Matlab implementation of Lempitsky’s method provided by Lempitsky et al. [58] was used to evaluate the Lempitsky’s method. The results related to Arteta’s method on the bacterial dataset was directly referred to the literature [4].

2.4.5 Performance evaluation metrics

Mean absolute count error (MAE), mean relative count error (MRE), and their related standard deviations (denoted by STDa and STDr) were employed as the evaluation metrics:

\[
\text{MAE} = \frac{1}{T} \sum_{t=1}^{T} |N_{ct} - \hat{N}_{ct}|, \\
\text{STDa} = \sqrt{\frac{1}{T-1} \sum_{t=1}^{T} (|N_{ct} - \hat{N}_{ct} - \text{MAE}|^2)}, \\
\text{MRE} = \frac{1}{T} \sum_{t=1}^{T} \frac{|N_{ct} - \hat{N}_{ct}|}{N_{ct}}, \\
\text{STDr} = \sqrt{\frac{1}{T-1} \sum_{t=1}^{T} \left(\frac{|N_{ct} - \hat{N}_{ct}|}{N_{ct}} - \text{MRE}\right)^2}. 
\] (2.10)
Table 2.2: MAE±STD cell counting performance.

<table>
<thead>
<tr>
<th>MAE ± STD</th>
<th>Bacterial cells</th>
<th>Bone marrow cells</th>
<th>Colorectal cancer cells</th>
<th>hESC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lempitsky’s method</td>
<td>3.52 ± 2.99</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Altera’s method</td>
<td>5.06*</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Mask R-CNN</td>
<td>36.92 ± 19.73</td>
<td>44.4 ± 14.17</td>
<td>45.97 ± 47.97</td>
<td>189.14 ± 231.86</td>
</tr>
<tr>
<td>U-Net</td>
<td>27.77 ± 25.48</td>
<td>48.00 ± 18.98</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>StructRegNet</td>
<td>9.80 ± 8.68</td>
<td>12.75 ± 8.62</td>
<td>45.97 ± 47.97</td>
<td>189.14 ± 231.86</td>
</tr>
<tr>
<td>FCRN</td>
<td>2.75 ± 2.47</td>
<td>8.46 ± 7.63</td>
<td>42.58 ± 33.51</td>
<td>44.90 ± 35.39</td>
</tr>
<tr>
<td>C-FCRN-Only</td>
<td>2.58 ± 2.28</td>
<td>8.68 ± 7.37</td>
<td>39.55 ± 35.80</td>
<td>42.17 ± 30.97</td>
</tr>
<tr>
<td>Count-Ception</td>
<td>2.79 ± 2.68</td>
<td>7.89 ± 6.83</td>
<td>34.14 ± 29.04</td>
<td>35.87 ± 35.77</td>
</tr>
<tr>
<td>C-FCRN+Aux</td>
<td><strong>2.37 ± 2.27</strong></td>
<td><strong>6.55 ± 5.26</strong></td>
<td><strong>29.34 ± 25.4</strong></td>
<td><strong>32.89 ± 26.35</strong></td>
</tr>
</tbody>
</table>

* indicates the result reported in the literature [4], in which the method was tested on a set of 100 testing bacterial cell images. Differently, the results from other methods related to this dataset were evaluated on a complete set of 200 bacterial cell images in this study, since the cross validation-based evaluation allows each image to be considered as a testing image for once. In addition, the Lempitsky’s method was only validated on the bacterial cell dataset because this dataset provides handcrafted image features for validation purpose. The results from the U-Net and Mask R-CNN were not reported on colorectal cancer cell and hESC datasets, due to their failure in providing reasonable detection results on the two datasets.

where \( T \) is the number of validation images, \( N_{ct} \) and \( \hat{N}_{ct} \) are the ground truth cell count and the estimated cell count in the \( t \)-th image respectively. MAE measures the mean of the absolute errors between the estimated cell counts and their ground truths for all the validation images. Considering the large variance in the numbers of cells in colorectal images and hESC images, MRE was also considered for method evaluation because they measure the relative errors between the ground-truth counts and the estimated counts. STDa and STDr indicate the stability of the cell counting process. A lower MAE or MRE indicates a better cell counting accuracy, and a lower STDa or STDr means a more stable counting performance.

### 2.5 Experimental results

Cell counting performance of the proposed “C-FCRN+Aux” method and the other eight methods on the four datasets are reported in the Figure 2.8 and Table 2.2. The proposed
method demonstrates superior cell counting performance to the other eight methods in terms of MAE and MRE. Compared to the regression-based methods, all four detection-based methods achieve worse counting performance in terms of MAE and MRE. Also, all three non-linear density regression-based methods (the proposed method, FCRN, C-FCRN-only) demonstrate superior counting performance compared to Lempitsky’s method, one of the conventional linear methods.

2.5.1 Comparison between FCRN, C-FCRN, and C-FCRN+Aux

First of all, we compared our method with original FCRN and the C-FCRN to investigate the effectiveness of the concatenation design and deeply-supervised learning strategy. Figure 2.6 shows the corresponding mean absolute errors tested on the four datasets. It can be observed in the figure that the C-FCRN+Aux consistently yields the lowest mean absolute errors on the four datasets. In addition, the C-FCRN generally has lower mean absolute errors than the original FCRN. These results suggest that both the concatenation design and deeply-supervised learning strategy can help to improve the cell counting performance in terms of MAE.

![Figure 2.6: The MAE performances related to FCRN, C-FCRN and C-FCRN+Aux.](image)

in the figure that the C-FCRN+Aux consistently yields the lowest mean absolute errors on the four datasets. In addition, the C-FCRN generally has lower mean absolute errors than the original FCRN. These results suggest that both the concatenation design and deeply-supervised learning strategy can help to improve the cell counting performance in terms of MAE.
The predicted cell counts for validation hESC images are plotted in Figure 2.7. In the

Figure 2.7: Robustness related to FCRN, C-FCRN and C-FCRN+Aux for validation hESC images.

plot, the horizontal and vertical axes represent the ground truth count and predicted count, respectively. Each dot represents a cell count prediction for a validation image. The black dash line indicates where the 100% accurate prediction should be located. From the figure we can see, generally, our results are closer to the target dash line than the compared methods. This indicate that the proposed method is more robust than the compared methods.

Besides, a paired t-test was performed on the absolute counting errors related to the proposed method (C-FCRN+Aux) and its closest counterpart C-FCRN-only. In this test, the null hypothesis $H_0$ was defined as the population means of absolute errors related to the C-FCRN+Aux is higher than that of C-FCRN, and vise versa for hypothesis $H_1$. The $p$-values for the tests on the synthetic cell, bone marrow cell, colorectal cancer cell, and hESC datasets are $6.19 \times 10^{-4}$, 0.042, $5 \times 10^{-7}$ and $2.8 \times 10^{-3}$, respectively. A similar paired t-test was performed on the absolute counting errors related to C-FCRN+Aux and original FCRN, and the corresponding $p$-values related to the four datasets are 0.024, 0.012, $7.35 \times 10^{-5}$ and 0.017,
respectively. The paired t-test results show that the MAEs related to the proposed method were lower than its two counterparts: C-FCRN and FCRN-only with statistical significance.

![Figure 2.8](image.png)

Figure 2.8: The MRE performance evaluated on four different datasets. “C-FCRN+Aux” represents the proposed method in this study. No MRE results were reported for Arteta’s method.

Figure 2.9 shows the estimated density/count map of a testing example in each of the four datasets. The density maps estimated by the C-FCRN+Aux appear visually closer to the ground truth density maps compared to the FCRN method. It is noted that the Count-Ception method predicts a count map directly without providing cell centroid locations, which is different from the other density regression-based methods.

### 2.5.2 Comparison between C-FCRN+Aux detection-based methods

To further investigate our method, we compared our method with three state-of-the-art cell detection-based methods, including U-Net, StructRegNet, and Mask R-CNN. The corresponding results in terms of MAE are shown in Figure 2.10. From the figure, we can see that our method yields the lowest MAE among these methods.
Figure 2.9: Estimated density or count maps from a sample image in each of the four datasets. The panels from left to right on each row show the cell images and the density/count maps estimated by the FCRN, the Count-Ception, and the proposed method (C-FCRN+Aux), and the associated ground truth density maps, respectively.
The predicted cell counts for validation hESC images are plotted in Figure 2.11. It can be observed in the figure that the proposed method has much more robust prediction than all the compared methods, especially when the number of cells in the validation image is high. These results suggest that our method are more accurate in terms of the MAE and more robust than the detection-based methods for cell counting tasks.

Figure 2.12 shows the result of a testing example in each of the bacterial and bone marrow cell datasets by use of three detection-based methods (Mask R-CNN, U-Net and StructRegNet). The StructRegNet achieves more accurate results than the other two. One of the possible
reasons is that the StructRegNet model is trained to regress a dense proximity map, in which the pixels closer to cell centroids can get more attention than those far-away pixels; this is different from the U-Net and Mask R-CNN model. This can benefit more for local maximum searching in the non-maximum post-process and yield better cell detection performance. It was also observed that the three detection-based methods commonly failed in detecting clustered and occluded cells in the bacterial image example. Also, they either under-detect or over-detect cells in the bone marrow image example. These images contain strongly heterogeneous backgrounds and the shapes of cells vary largely. The inaccuracy of cell detection with these detection-based methods confirms their lower cell counting accuracy shown in the Table 2.2 and Figure 2.8.

Figure 2.13 shows the result on an example in each of the colorectal and hESC datasets by use of the proposed method and the StructRegNet method, which are the best-performing regression-based method and detection-based method tested in this study, respectively. The cells are commonly concentrated in colorectal cell images and seriously clustered and occluded in the hESC images. Cell detection in these two scenarios is extremely challenging. The StructRegNet method shows much worse counting performance compared to the proposed method.

2.5.3 Density regression, count regression, and cell detection

In this study, we further compared the three groups of cell counting methods: density regression, count regression, and cell detection. Count-Ception is a count regression-based method. StructRegNet has the best performance among all the compared detection-based methods, so we use it for comparison. The corresponding results are shown in Figure 2.14. It can be observed in the figure that the proposed method shows better performance than the compared methods. This indicates that our method shows better performance than the
Figure 2.12: Example results of three deep-learning detection-based cell counting methods (Mask R-CNN, U-Net, and StructRegNet). Panels (a) and (b) show the prediction results on the bacterial and bone marrow datasets, respectively. The green cycles and red dots in each image represent the ground truth annotations and the detected cell centroids, respectively.
Figure 2.13: Example prediction results based on the proposed C-FCRN+Aux method and the detection-based method (StructRegNet). Here, “image”, “C-Aux” and “RegNet” represent the processed image and the estimated density map using the “C-FCRN+Aux” method and the computed proximity map using the “StructRegNet” methods. The red dots represent the detected cell centroids based on the computed proximity map, respectively.

Figure 2.14: MAE performance related to density regression-based, count regression-based, and cell detection-based methods.
count-regression and detection-based methods in terms of the MAE.

The predicted cell counts for validation hESC images are plotted in Figure 2.15. The right

![Figure 2.15: Robustness related to the three groups of methods for bacterial cells dataset.](image)

figure shows that our results are generally much closer than the target dash line compared to the other two. This suggest that our method is more robust than the other two.

### 2.5.4 Benefits of using AuxCNNs to support C-FCRN training

The accuracy of the estimated density map along the training process was investigated to demonstrate that AuxCNNs supports the overall model training process. Figure 2.16 shows the curves of validation losses vs. the number of epochs for the proposed method and the other two nonlinear density regression methods (C-FCRN-only and FCRN) on four datasets. One of the five validation curves generated during the 5-fold cross validation procedure is presented for each method as an example. The curves generated for the first 500 epochs are shown because the validation losses maintain stable after the 500-th epoch. As shown in Figure 2.16, the curves from all three methods converge when the number of epochs
increases, which reflects the stability of training process. In addition, the curves of the proposed C-FCRN+Aux method are significantly lower compared to the other two for all four datasets, which demonstrate that the proposed method allows to train a model that yields better model-fitting with the deep supervisions from the AuxCNNs. This analysis of validation loss over the training process is consistent with the results shown in Tables 2.2 and Figure 2.8, and reflects the better model fitting and generalization of our DRM to the validation data.

Figure 2.16: Validation losses as the functions of epochs are plotted for the DRM training on the four datasets. C-Aux and C-FCRN are abbreviations of C-FCRN+Aux and C-FCRN-Only methods, respectively.
2.5.5 Computation efficiency comparison

The computation efficiencies of the seven deep convolutional neural network-based methods, including the proposed method, FCRN, C-FCRN-Only, Count-Ception, StructRegNet, U-Net and Mask R-CNN, were compared. The average processing time on testing images with the same GPU settings was employed as the comparison metric. Table 2.3 shows that the proposed method costs comparable counting time compared to the FCRN and the C-FCRN-Only methods. The Count-Ception method is the more time-consuming one in comparison to the other three regression-based methods. In the Count-Ception method, max-pooling layers are not employed in the network, and filters with large spatial size (5 × 5 pixels) are employed for extracting multi-scale features from images. These two reasons induce a large amount of convolution operations between high-dimension feature maps and large-sized filters, therefore, leading to the high computation workload in the Count-Ception method.

Density regression-based methods are less time-consuming than the three detection-based methods (StructRegNet, U-Net, and Mask R-CNN). The main reason is that the non-maximum suppression post-processing for cell detection costs a considerable amount of time.

<table>
<thead>
<tr>
<th>Seconds/image</th>
<th>Bacterial cells</th>
<th>Bone marrow cells</th>
<th>Colorectal cancer cells</th>
<th>hESC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mask R-CNN</td>
<td>0.55279</td>
<td>0.89527</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>U-Net</td>
<td>0.07646</td>
<td>0.16125</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>StructRegNet</td>
<td>0.06648</td>
<td>0.08035</td>
<td>0.18690</td>
<td>0.36167</td>
</tr>
<tr>
<td>FCRN</td>
<td>0.00468</td>
<td>0.02568</td>
<td>0.017</td>
<td>0.01901</td>
</tr>
<tr>
<td>C-FCRN-Only</td>
<td>0.00511</td>
<td>0.02846</td>
<td>0.01925</td>
<td>0.02134</td>
</tr>
<tr>
<td>Count-Ception</td>
<td>0.25111</td>
<td>0.18185</td>
<td>0.16308</td>
<td>0.19208</td>
</tr>
<tr>
<td>C-FCRN+Aux</td>
<td>0.00554</td>
<td>0.03113</td>
<td>0.02233</td>
<td>0.02275</td>
</tr>
</tbody>
</table>

Seconds/image represents the processing time for one image.
Mask R-CNN takes particularly longer time because of its superior larger network size and its aim at predicting separate masks for each cell, which is a much more complex task compared to the cell counting task.

2.6 Discussion

The method proposed in this study combines the advantage of FCRN design with concatenation layers and a deeply-supervised learning strategy. It solves the two shortcomings that exist in the original FCRN. The concatenated layers integrate multi-scale features extracted from non-adjacent layers to improve the granularity of the extracted features and further support the density map estimation. The deeply-supervised learning strategy permits a direct supervision from AuxCNNs on learning its intermediate layers to mitigate the potential vanishing gradient issue and improve the cell counting performance. The results on four image datasets show superior cell counting performance of the proposed method compared to the other eight state-of-the-art methods. In addition, compared to the original FCRN, the proposed method improve the counting performance on four datasets ranging from 13% to 31% in terms of MAE. The computational efficiency of the proposed method is comparable to other density regression-based methods. The proposed method is capable of processing arbitrary-size images and estimating their density maps by use of fully convolutional layers in the C-FCRN. The proposed method could also be applied to heterogeneous cell assemblies, if cell types of interest are annotated in the training images. This deeply supervised learning framework will encourage the trained DRM to focus on the cell types of interest but consider cells of other types as background.

The proposed method, the four regression-based, and the four detection-based methods were investigated on four challenging datasets. In general, the density regression-based
methods yielded better performance and had three advantages over the detection-based methods. First, the regression-based methods count cells without cell detection, which can avoid challenging cell detection issues that commonly exist in microscopy images. Second, density regression-based methods are convenient for deployment, since they do not require trivial post-processings such as thresholding and non-maximum suppression. Thirdly, density regression-based methods can count cells more efficiently, i.e. the counting for an image of 512 × 512 pixels takes about 20ms. The three advantages enable the density-regression based methods to be potentially applied to real-time clinical applications. In addition, it should be noted that even though the detection-based methods yield lower performance on this cell counting task, they are more suitable for the segmentation of cells of other types for other applications [48, 105]. Generally, for those cell types of interest, the cells in the acquired microscopy images are less overlapped and the cell masks can be fully annotated. In addition, the kernel sizes shown in Eq. 2.8 is determined by $K_G$, which is chosen according to the sizes of cells in the processed image to guarantee that the touching areas between occluded cells have been appropriately represented on the related density map. In this study, the radii of cells in the four datasets are less than 8 pixels. We then set the kernel size $(2K_G + 1) \times (2K_G + 1)$ to 21 × 21 pixels.

In the current study, all images were pre-processed by simply normalizing the intensities to the range of [0, 1] to increase the stability of the model training process. In the future, we will investigate other image denoising and/or image enhancement methods to more accurately count cells for images that exhibit highly inhomogeneous tissue backgrounds and noises, or yield low image contrast. Also, the cell centroids used for generating ground truth density maps in the three experimental datasets were manually annotated by human experts, which may be subject to subjective errors. This might be one of the reasons that the MREs of these three experimental datasets (shown in Figure 2.8) were higher than that of the synthetic
bacterial dataset. More accurate annotation strategies will be investigated to reduce the uncertainty in generating ground truth density maps. In this study, a uniform network architecture of C-FCRN+Aux was applied to learn DRMs separately on each of the four distinct datasets. We will adapt some other variants of FCRNs in the future that aim at crowd counting tasks [82, 94, 104] for varied datasets. Last but not the least, the sizes of all the four datasets used in this study are small, thus, the generalization of the proposed method to a large-size dataset is one of the potential future directions.

### 2.7 Conclusion

A deeply-supervised density regression model was proposed in this study for accurately and robustly counting the number of cells in microscopy images. The proposed method was capable of processing varied-size images containing dense cell clusters, large variations of cell morphology and inhomogeneous background noise. Extensive experiments based on four datasets representing different image modalities and image acquisition techniques demonstrated the efficiency, robustness, and generality of the proposed method. The proposed method can be potentially to be applied to real-time clinical applications. It also holds the promise to be applied to a number of different problems, such as object counting (other than cells) in crowded scenes.
Chapter 3

EfficientNet-based U-Net with Spatial Attention for Multi-Class Nuclei Segmentation

3.1 Introduction

The nucleus is the brain of the cell consisting of the nuclear envelope, nucleolus, chromatin, and pore, among others, carrying the genetic information, DNA, which controls each cell [27, 70, 86]. Microscopy imaging techniques have been widely employed to generate high-quality cell images for the studies of nuclei [43]. Multi-class nuclei segmentation is the classification of pixels in microscopy images into different nuclei classes and background class, which helps researchers identify and analyze the morphological and count changes of nuclei from different classes to investigate how the cells react to discrete cures [43]. Automating this segmentation process will alleviate the burden of doctors and biologists on tedious and time-consuming
manual labor work, and allow them to focus more on solutions or remedies and improve the
throughput of research and insights. Nevertheless, automatic multi-class nuclei segmentation
is very challenging due to the presence of digital noise, low nuclei contrast, and complex and
irregular background in microscopy images.

In recent years, deep learning techniques, especially convolutional neural networks (CNNs),
have been commonly employed for generic cell or nuclei segmentation tasks due to their
strong ability in learning discriminative features from images to distinguish background and
nuclei pixels. To date, the U-Net [73] is the most widely used CNN architecture for nuclei
segmentation tasks, since it allows to effectively fuse localization information from low-level
features and semantic information from high-level features to deliver more accurate pixel-level
classification; it also permits an efficient prediction of semantic maps in an end-to-end manner.
A standard U-net contains an encoding path and a decoding path that are connected via skip
connections. In the encoding path, a sequence of down-sampled CNNs are employed to learn
multi-level feature maps from input images; in the decoding path, symmetrically, a sequence of
up-sampled CNNs gradually restore the resolution of feature maps and fuse low-level features
from the encoding path and high-level features from decoding path to generate more fine
granular features that deliver better pixel-level classification. Based on a standard U-Net, a
variety of U-Net-like architectures have been proposed to improve the U-Net performance for
generic biomedical image segmentation. For instance, Zhou et. al [107] proposed a U-Net++
arquitectura that adds dense shortcut connections in a standard U-Net to reduce the semantic
gap between the low-level and high-level features in the feature fusion; Oktay et. al [67]
placed attention modules on each skip connection to rule out irrelevant low-level features
and improve the qualities of fused features. These U-Net variants have achieved remarkable
segmentation performance in distinguishing nuclei and background pixels [65, 78]. However,
when it comes to multi-class nuclei segmentation tasks, their segmentation performances are
still inadequate. The reasons are described as follows. Due to the ambiguous appearance of nuclei, the multi-class task requires a powerful U-Net model to learn more discriminative image features for accurately distinguishing pixels from different nuclei classes. However, traditionally, such a powerful U-Net comes with a large network size, which is very difficult to train when the availability of annotated microscopy images is limited due to the expensive and time-consuming annotation process.

Recently, the efficiency of CNN models has been actively investigated to seek compact but powerful CNN models. In this trend, Tian et al. [88] has recently proposed an EfficientNet for natural image classification. Their studies on the ImageNet dataset demonstrated that, with a much more compact size, EfficientNets can achieve comparable or even stronger abilities in learning representative features than other state-of-the-art CNN models, such as ResNet, VGG nets, etc. Thus, EfficientNets can be potentially applied in a scenario where more discriminative features need to be learned from images but the availability of annotated images is limited.

In recent years, transfer learning has been widely employed to reduce the need for annotated images for training deep learning models by pre-training a deep learning model on a previous task and then fine-tuning them on a relatively small dataset of the current task [3, 25, 30, 37, 44, 45]. It has been proven to be an efficient method when there is a lack of target data, which is commonly found in medical imaging due to the difficulty of collecting medical image datasets. The ILSVRC-2012 competition of ImageNet [18] is the most well-known pretraining dataset and has been extensively utilized to improve the performance of image processing tasks such as segmentation [25, 37], detection [45], and classification [44].

In this study, motivated by the above mentioned works, we propose a novel approach to address the challenge of training a powerful CNN model for multi-class nuclei segmentation
with a limited amount of annotated microscopy images. The proposed approach is based on a
compact but powerful AE-U-Net architecture and a transfer learning strategy. The proposed
AE-U-Net uses a U-Net as the basic architecture but uses a compact but powerful CNN
model, EfficientNet, in the encoding path for feature extraction. This allows the AE-U-Net
to have a strong capacity to learn more discriminative features to distinguish nuclei from
different classes but maintain a relatively small network size. Besides, the AE-U-Net uses
attention modules on skip connections to select salient local features and improve the quality
of fused features in the decoding path. This powerful but compact AE-U-Net can mitigate the
challenge of training a powerful segmentation model with a limited number of training images.
In the transfer learning strategy, the EfficientNet [88] used in the AE-U-Net was pre-trained
on an ImageNet dataset that contains millions of natural images for an image classification
task. This transfer learning strategy greatly reduces the number of training images required in
network training and further address the training issue caused by data scarcity. Secondly, we
conducted ablation studies on two challenging datasets to investigate the effectiveness of the
EfficientNet, the spatial attention module, and the transfer learning strategy. Experimental
results show that the proposed approach can greatly boost the segmentation performance in
classifying pixels from different nuclei classes, compared to a standard U-Net and other state-
of-the-art methods. Thirdly, we compared our proposed method with other state-of-the-art
nuclei segmentation methods, and the results show that our proposed method consistently
outperformed the compared methods on the two investigated datasets.

The remainder of this section is organized as follows. In Section 3.2, we introduce our
proposed method. Section 3.3 contains experiments results related to the method evaluation.
Finally, in Section 3.4 and Section 3.5, we discuss and conclude this work.
3.2 The Proposed AE-U-Net Based Method

The goal of a multi-class nuclei segmentation task is to predict the corresponding semantic map for a given image, in which each pixel is assigned to a categorical value that represent the class of the corresponding pixel in the image. In this study, we proposed a compact but powerful AE-U-Net to achieve this task by mitigating the challenge of training a powerful segmentation model with limited number of training images. Furthermore, a transfer learning strategy was employed to train the proposed AE-U-Net for better network training with limited amount of annotated images. The AE-U-Net and the transfer learning strategy will be introduced in the following subsections.

3.2.1 The proposed AE-U-Net

Overview of the proposed AE-U-Net

The overview of the proposed AE-U-Net network is shown in Fig. 3.1. The input and output of the network are a microscopy image and the corresponding predicted probability map, respectively. Each pixel in the probability map corresponds to a vector of probabilities related to different nuclei classes and background class. The length of each probability vector is the number of nuclei classes plus 1. The predicted semantic map is finally generated from the probability map by use of a softmax decision rule.

The AE-U-Net architecture was designed based on three key components: a standard U-Net, an EfficientNet, and a spatial attention module. It maintains a basic U-Net-like structure of an encoding and a decoding path that are connected via skip connections. The encoding and decoding paths symmetrically perform a sequence of down-sampled convolution operations and up-sampled convolution operations to learn multi-level feature maps and fuse these
feature maps, respectively. The skip connections deliver the multi-level features from the encoding path to the decoding path for the feature fusion. This classic design allows our AE-U-Net to learn and fuse multi-level features from images to generate more fine granular features for a more accurate pixel-level classification as a standard U-Net does.

Nevertheless, our AE-U-Net differs from a standard U-Net in two aspects. First, instead of using conventional CNN models with large sizes in the encoding path to obtain strong feature learning capacity, we employed an EfficientNet model that is compact but powerful to achieve that. This makes the AE-U-Net have an overall compact network size and allows for better training a powerful CNN-based segmentation model when the availability of annotated images is limited. More details about EfficientNet will be introduced in Section 3.2.1. Here, we would like to mention, our work was not the first one to use the EfficientNet in a generic image segmentation problem. Before our work, Baheti et al. [7] employed EfficientNet to extract features for semantic segmentation in road traffic images. The challenges related to that work are different from the challenges related to the problems in this study. This was the first work that employed an EfficientNet for multi-class nuclei segmentation tasks as of when our pre-print paper [43] was released online.

Furthermore, motivated by an Attention U-Net [67], our AE-U-Net uses spatial attention modules (SAMs) on each skip connection, as shown in Fig. 3.1. The attention modules are employed to generate an attention map to enforce attention on salient features in the low-level feature maps. Through the attention module, the features in the salient area of the low-level feature maps will be selected and fused with the high-level feature map in the decoding path, so that the quality of fused features can be potentially improved. This attention-aware feature fusion process allows the AE-U-Net to rule out the irrelevant contextual information in the low-level feature maps, and deliver an efficient feature fusion in the decoding path. More details about the spatial attention module will be introduced in Section 3.2.1. The two
aspects mentioned above will be introduced remaining of this subsection. To the best of our knowledge, the AE-U-Net is the first U-Net-like architecture that integrates both EfficientNet and spatial attention module and is used in multi-class nuclei segmentation.

Figure 3.1: The overview of the AE-U-Net network for multi-class nuclei segmentation. The major differences between an AE-U-Net and the standard one are the EfficientNet used in encoding path and the spatial attention modules applied on each skip connection.

**EfficientNet**

EfficientNets [88] refer to a family of compact CNN models proposed by Tian et. al [88] for image classification with natural images. They deliver comparable capacity in learning image features compared to other popular CNN models, such as VGG [81], ResNet [29], etc., but have much smaller network sizes. The efficiency of EfficientNets compared to other state-of-the-art models is shown in Fig. 3.2. They have later been commonly used in other computer vision tasks, due to their excellent feature learning ability with compactness. This compact design can potentially benefit a CNN model training for multi-class nuclei segmentation problems. In this problem, there is a great need for learning more discriminative
features from microscopy images in order to more accurately distinguish nuclei pixels from different classes, but the availability of annotated images is quite limited compared to other computer vision problems since annotation for natural images are much cheaper. As such, an EfficientNet was employed in our network design for image feature learning.

There are 8 CNN models in the EfficientNet family, including EfficientNet-B0, EfficientNet-B1, ..., and EfficientNet-B7. Here, a larger index number indicates that the corresponding model has a larger network size. An EfficientNet typically consists of a stack of CNN blocks that each is a sequence of MBConvX modules, as shown in Fig 3.3 that shows the architecture of EfficientNet-B0. The structure inside an MBConvX module is shown in Fig. 3.3. It reformulates a standard convolutional operation into a sequence of operations including expansion, depthwise convolution, and residual connection layers, which allows an

---

**Figure 3.2:** Efficiency of EfficientNets compared to other methods [88].
MBConvX module to use much fewer network parameters to express comparable feature learning capacity compared to a standard convolutional layer. In an EfficientNet, there are two types of MBConvX: MBConv1 and MBConv6, which indicate the use of ReLu and ReLu6 activation functions in the corresponding MBConvX module, respectively. The other 7 EfficientNet models (from B1 to B7) have similar structures to the EfficientNet-B0, except for a different number of MBConvX modules used in the CNN blocks. The EfficientNet with a larger index uses more MBConvX modules, so the EfficientNet-B0 and EfficientNet-B7 are the smallest and largest among the 8 EfficientNet models. In this study, the index of EfficientNet is a tunable parameter that controls the complexity of an EfficientNet model.

![EfficientNet-B0 network architecture](image)

Figure 3.3: The network architecture of an EfficientNet-B0 and the MBConvX module.

When applied to an input image, an EfficientNet CNN model will generate intermediate outputs at each CNN block. These intermediate outputs are feature maps where each pixel represents a feature vector that corresponds to a receptive field of pixels in the input image. The deeper CNN blocks have larger receptive fields, thus their produced feature maps contain features with higher levels. In addition, due to the down-sampled operations, these feature maps also have smaller dimensions. In this way, the EfficientNet model in an AE-U-Net can learn multi-level and multi-scale features from input microscopy images for the multi-class nuclei segmentation.
Spatial Attention Module

In recent years, attention modules have been actively investigated in a variety of computer vision tasks, such as image classification [95], image segmentation [24], etc. Recently, attention modules have also been applied to the medical image segmentation task, where a spatial attention module was placed on the skip connection of U-Net to enforce attention to select a salient area of features in the feature maps [67]. This allows to rule out the irrelevant low-level features before their fusion with high-level features in the decoding path, and greatly improves the feature fusion efficiency, compared to a conventional U-Net. Motivated by this attention module-based work, a spatial attention module was employed in our proposed AE-U-Net model to improve its feature fusion efficiency.

The architecture of a spatial attention module is shown in Fig. 3.4. As shown in the figure, an attention module takes as inputs both a high-level feature map from the decoding path and a low-level feature map from the encoding path and generates an attention map where each pixel has a real-valued number in the range of $[0, 1]$. The generated attention map has the same size as the low-level feature map input, and the value of each pixel in the attention map indicates the importance level of a feature at the corresponding pixel in the feature map. A higher pixel value in the attention map means the corresponding feature is more salient, vice versa. This attention map is finally applied on the low-level feature map to select the salient features in the feature map by a pixel-wise multiplication of the attention map and the low-level feature map. The selected salient feature map is the final output of the attention module.
Figure 3.4: The schematic overview of an attention module [67]. The high-level features here are those from the decoding path; the low-level features are those from the encoding path.

### 3.2.2 Transfer Learning

Training a deep neural network with a limited amount of training images is generally challenging, so does the proposed AE-U-Net. To mitigate this challenge, a transfer learning strategy was employed in the AE-U-Net training. The training strategy is shown in Fig. 3.5. Specifically, the weights of the EfficientNet in the encoding path of an AE-U-Net were initialized with the weights of a pre-trained EfficientNet, which was trained by [88] on an ImageNet [25] dataset for an image classification task. The ImageNet is a public image set that contains millions of annotated natural images. Subsequently, the whole AE-U-Net was fine-tuned with the limited amount of annotated microscopy images in the target dataset.

In this study, a weighted combination of dice loss and focal loss is employed for the AE-U-Net fine-tuning, since both these two loss can help tackle class imbalance issue and better train the networks. This combined loss function is defined as:
Figure 3.5: Transfer learning strategy used in the AE-U-Net training.

\[ L_{\text{cmb}} = \lambda L_{\text{focal}} + (1 - \lambda) L_{\text{dice}} \]
\[ L_{\text{dice}} = 1 - \frac{1}{C} \sum_{c=1}^{C} \sum_{i=1}^{M\times N} 2y_{c,i}^T p_{c,i} \]
\[ L_{\text{focal}} = \frac{1}{C} \sum_{c=1}^{C} (1 - y_{c,i} p_{c,i})^\gamma y_{c,i}^T \log p_{c,i} \]

where the \( y_{c,i} \) and \( p_{c,i} \) are the predicted probability of belonging to \( c \)-th class and the ground truth class label for pixel \( i \) in each image, respectively. Both of the dice loss and focal loss can help to re-balance the nuclei and background pixel class and better train the networks. Here, \( \lambda \) was empirically set to 0.5 to make the focal and dice losses contribute equally in the combined loss.
3.3 Experiments

3.3.1 Datasets

In this study, two image datasets were employed to evaluate the proposed AE-U-Net-based method. The two datasets contain phase images of HeLa cancer cells (HeLa dataset) and those of CHO cells (CHO dataset), respectively. Both the two datasets were acquired by use of a quantitative phase imaging (QPI) system for label-free cell viability assay [42], which are important for assessing the impact of drug, physical, or chemical stimulants, and other potential factors on cell dynamics. Label-free cell viability assay is to classify pixels in the phase images into “background” class and different viability states including “live”, “injured” and “dead”, which is essentially a multi-class nuclei segmentation task. The details related to the two image datasets are described below.

The HeLa dataset contains images of HeLa cancer cells with all three viability states: “live”, “injured”, and “dead”. It consists of 1197 images of size $832 \times 832$ pixels and their corresponding ground-truth semantic maps. For a given image, the related semantic map is a map of categorical values that represent the class each pixel in the image belongs to, which includes background class, live nuclei class, injured nuclei class, and dead nuclei class. The segmentation task related to this dataset is accurately classifying each pixel in HeLa cell images into four classes: background, live, injured, or dead. Two image examples from the HeLa dataset are shown in Fig. 3.6(a). The whole HeLa dataset was randomly split into training, validation, and testing sets that contain 899, 199, and 99 images, respectively. Among the three image sets, the training set was used to train the networks; the validation set was used to tune the hyper-parameters and select the weights of the network parameters in the
training process; the testing set was held out as the unseen data for testing the performance of trained models once the model was selected based on the validation set.

The CHO dataset contains images of CHO cells with only two viability states: live and dead. Accordingly, the segmentation task is to classify pixels in the phase images into three classes: background, live, and dead. It includes 2112 images of size 832 × 832 pixels and their corresponding ground-truth semantic maps. The dataset was split into training, validation, and testing sets that contain 1536, 288, and 288 images, respectively. Two image examples from the CHO dataset are shown in Fig. 3.6(b).

The phase images and their ground truth semantic maps used in this study were generated by Prof. Gabriel Popescu’s group with their advanced QPI system, called phase imaging with computing specificity (PICS) [38, 41, 42, 43], as shown in Fig. 3.7. The PICS system integrates...
Figure 3.7: Phase image acquisition and ground-truth semantic segmentation map generation. (a) shows the phase imaging with computational specificity (PICS) imaging system. (b) shows the acquired phase image; (c) and (d) correspond to the fluorescent images from two biologically labeled channels; (e) shows ground-truth semantic segmentation maps generated from (c) and (d).

QPI and fluorescent imaging into the cell imaging process and enables the simultaneous acquisition and co-registration of the two image modalities. The ground-truth semantic maps were generated by thresholding the fluorescent images. The thresholds were selected based on the histogram of intensities of fluorescent signals from cells with each viability state. More details related to the acquisition and ground truth preparation can be found in the literature [42].

Multi-class nuclei segmentation on the two datasets is very challenging. As we can see in Fig. 3.6(a) and Fig. 3.6(b), cell nuclei in phase images have extremely low contrast, and the nuclei with different viability classes are visually distinguishable. Furthermore, the number of images in both two datasets is far less than natural image datasets in generic computer problems.
3.3.2 Network training

In this study, all the networks were trained by minimizing a loss function that measures the difference between the predicted semantic maps and the corresponding ground-truth values in the training set. The loss function is a weighted combination of focal loss and dice loss as defined in Section 3.2.2. The loss function was minimized with a batch of randomly cropped training patches of size 512 × 512 pixels using an Adam algorithm [52]. In the Adam algorithm, the exponential decay rates for 1st and 2nd-moment estimates were set to 0.9 and 0.999, respectively; a small constant was set to 10^{-7} for the numerical stability. The batch size was set to 6. The learning rate was initially set to 5 × 10^{-4}. At the end of each epoch, the loss of the being-trained networks was computed on the whole validation set. When the validation loss did not decrease for 10 training epochs, the learning rate was multiplied by a factor of 0.8. This validation loss-aware learning rate decaying strategy benefits for mitigating the overfitting issue that commonly occurs in deep neural network training. Furthermore, data augmentation techniques, such as random cropping, flipping, shifting, and random noise and brightness adding, etc., were employed to augment training samples on the fly for further reducing the overfitting risk. The E-U-Net was trained for 100 epochs. The parameter weights that yield the lowest validation loss were selected and subsequently used for model testing.

The networks were implemented using the Python programming language with libraries including Python 3.6 and Tensorflow 1.14. The model training, validation, and testing were performed on an NVIDIA GTX GPU of 12 GB VRAM and a CPU.

3.3.3 Performance metrics

In this study, pixel-wise recall, precision, and F1 scores were employed to evaluate the segmentation performances of the trained networks. Given the predicted semantic maps and
their corresponding ground truth values for all the images in a testing set, the three scores over a viability class $c$ are computed as follow.

\[
\text{Recall}_c = \frac{TP_c}{TP_c + FP_c}, \quad (3.1) \\
\text{Precision}_c = \frac{TP_c}{TP_c + FN_c}, \quad (3.2) \\
F1_c = \frac{\text{Precision}_c \times \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}, \quad (3.3)
\]

where $TP_c$ is the number of pixels that belong to class $c$ and are correctly classified into class $c$ among all the pixels in these testing images; $FN_c$ is the number of pixels that classified into class $c$ but belong to other classes among all the pixels in these testing images; $FP_c$ is the number of pixels belong to class $c$ but are incorrectly classified into other classes among all the pixels in these testing images.

### 3.3.4 The overall performance of AE-U-Net

First of all, we compared our method with the standard U-Net to investigate the overall effectiveness of our method. To make a fair comparison, we set the EfficientNet as EfficientNet-B0 to make sure that AE-U-Net and U-Net have a comparable network size. Their corresponding pixel-wise F1 score performances over each class on the HeLa cell dataset and CHO dataset are shown in Figure 3.8.

From the two figures, we can see that both the U-Net and AE-U-Net have good performances in classifying the background pixels, which means that both of them have a very good performance for binary segmentation performance. However, the U-Net has poor classification accuracy for the live and dead classes for the HeLa dataset and live class for the CHO cell dataset. In contrast, our method has much higher accuracy than the U-Net on each nuclei
class. The results suggest that, by use of our proposed AE-U-Net and transfer learning strategy, the nuclei classification accuracy is greatly improved compared to a standard U-Net.

### 3.3.5 Ablation Studies

In this section, we conducted ablation studies on the HeLa and CHO datasets to investigate the effectiveness of EfficientNet, spatial attention module, and transfer learning strategy that was used in our proposed approach. For simplicity, we denote our method as “AE-U-Net+TL”. Besides the standard U-Net and our method, we also evaluated a U-Net with attention modules, a U-Net with EfficientNet, and an AE-U-Net trained from scratch for this study. All these methods have comparable network sizes for a fair comparison. The corresponding pixel-wise F1 scores are shown in the two figures in Figure 3.9. Here, we only show the performances on each nuclei class, since all of these methods have good performances on classifying background pixels.
From the two figures we can no matter we added EfficientNet and Attention modules to separately or together, the network performance on nuclei classes get better compared to the standard U-Net. Also, an AE-U-Net with transfer learning show much better performance than an AE-U-Net trained from scratch. These results demonstrated that both AE-U-Net and transfer learning in our proposed method can effectively improve the performance of the U-Net.

Besides the analysis of F1 score performances, the prediction results related to E-U-Net and AE-U-Net for two testing HeLa cell images and those for two testing CHO images are visualized as shown in Fig. 3.10(a) and Fig. 3.10(b), respectively. The two networks were both trained with pre-trained EfficientNets. The prediction results related to U-Net are also displayed in the two figures as a baseline comparison.

It can be seen from the highlighted region in Fig. 3.10(a) and Fig. 3.10(b) that the AE-U-Net can deliver more visually accurate segmentation results compared to the E-U-Net. However, both the AE-U-Net and E-U-Net can predict much better semantic maps compared to the standard U-Net.
Figure 3.10: Visualized predictions for testing examples from HeLa and CHO datasets. The columns from left to right show the testing phase images, ground truth semantic maps, prediction results related to U-Net, those related to E-U-Net, and those related to AE-U-Net, respectively. The E-U-Net and AE-U-Net were trained with pre-trained EfficientNet-B0.
The quantitative results in terms of F1 scores and visualized prediction results suggest that the spatial attention modules in AE-U-Net allow for a more effective feature fusion to improve the discriminability of the fused features for distinguishing pixels from different nuclei classes.

### 3.3.6 U-Net vs. AE-U-Net with different network sizes

To further investigate our method, we compared our method with the U-Net by configuring them with different network sizes. The network sizes increase from the setting net-1 to net-3, as shown in Figure 3.12. The network training processes were the same as described in Section 3.3.2. The average F1 scores over nuclei classes (excluding background class) on the two datasets are shown in the two figures in Figure 3.12.

<table>
<thead>
<tr>
<th></th>
<th>Net-1</th>
<th>Net-2</th>
<th>Net-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>U-Net</td>
<td>U1</td>
<td>U2</td>
<td>U3</td>
</tr>
<tr>
<td>AE-U-Net</td>
<td>B0</td>
<td>B1</td>
<td>B2</td>
</tr>
<tr>
<td># of parameters (M)</td>
<td>~10</td>
<td>~12</td>
<td>~14</td>
</tr>
</tbody>
</table>

Figure 3.11: Network size under different settings.

![Graphs showing average nuclei F1 scores](image)

Figure 3.12: Average nuclei F1 scores related to different network sizes.
From the two figures, we can see that under the three different network settings, the proposed AE-U-Net (with or without transfer learning) can consistently improve performance on nuclei classes in terms of average nuclei F1 score. In addition, we can see that with smaller network size, the proposed method still can achieve much better performance than the U-Net. This further demonstrates the effectiveness of our proposed method.

3.4 Discussion

Multi-class nuclei segmentation is one of the fundamental tasks in biomedical cell analysis. Though CNN-based methods, such as the U-Net, have achieved impressive performance on classifying background and nuclei pixels, it is practically difficult to train a powerful CNN model that learns more discriminative features to distinguish pixels from different nuclei classes due to the ambiguous appearance of inter-class nuclei and a limited amount of available annotated images. In this study, to address these challenges, we proposed a novel CNN-based approach that involves our proposed compact but powerful CNN architecture AE-U-Net and a transfer learning strategy in the network training. The AE-U-Net was designed based on a standard U-Net, an EfficientNet, and a spatial attention module. Specifically, it replaces the encoding path of a U-Net with a compact but powerful CNN model EfficientNet, which allows our AE-U-Net to have a powerful capacity of image feature learning but retain a compact size. In addition, it contains a spatial attention module on each skip connection to generate attention maps to select salient features on low-level feature maps and boost the quality of the fused features in the decoding path. This powerful but compact AE-U-Net can mitigate the challenge of training a powerful segmentation model with a limited number of training images. The EfficientNet used in the AE-U-Net was pre-trained by Tan et. al [88] on an ImageNet dataset containing millions of natural images for an image classification task. This
transfer learning strategy greatly reduces the number of training images needed in network training and further address the training issue caused by data scarcity.

The proposed method was evaluated on two challenging datasets that contain 1197 phase images of HeLa cells and 2112 phase images of CHO cells, respectively. The results obtained on the two datasets show that the proposed approach can greatly boost segmentation performance in terms of average pixel-wise F1 scores, compared to a standard U-Net with comparable network size. As shown in Figure 3.8, when all the available training images were used, it improved the average F1 score from around 0.6 to 0.73 and from 0.78 to 0.85 on the HeLa dataset and CHO dataset, respectively. The improvement percentages are 21.7% and 8.6%, respectively. In addition, the proposed method greatly improved the F1 score for classifying pixels from different nuclei classes, though they both delivered impressive performance in classifying background and nuclei pixels. Figure 3.8(a) shows that on the HeLa dataset, the proposed approach improve the F1 scores for live, injured, dead classes from 0.34, 0.74, and 0.32 to 0.44, 0.79, and 0.73, respectively. The improvement percentages are 26.7%, 7.7%, and 114.4%, respectively. We can see that there is an extremely large improvement for the dead nuclei class. Figure 3.8(a) shows that the proposed approach improved the F1 scores for live and dead classes from around 0.60 and 0.76 to 0.74 and 0.82, respectively. The corresponding improvement percentages are 23.78% and 7.8%, respectively. The nuclei segmentation task on the HeLa dataset is to classify the pixels into four classes, while the task on the CHO dataset is to classify pixels into three classes. Intuitively, the task on the HeLa dataset is more difficult than that on the CHO dataset. Thus, the larger performance improvement related to the HeLa dataset indicates that the proposed approach can potentially demonstrate a greater performance improvement, compared to a standard U-Net, for a more challenging multi-class task.
The ablation studies conducted in this study showed that both the EfficientNet, spatial attention module, and transfer learning strategy are effective in improving the segmentation performance. As shown in Figure 3.9, no matter we added EfficientNet and Attention modules to separately or together, the network performance on nuclei classes get better compared to the standard U-Net for both the two datasets. Also, an AE-U-Net with transfer learning show much better performance than an AE-U-Net trained from scratch.

Despite the impressive segmentation performance of our proposed approach, there are some limitations related to our work. First of all, the proposed approach was evaluated on two QPI datasets, though they are acquired for different cells and have different characteristics. The proposed method might need further investigation when applied to other microscopy image modalities. However, the challenge related to the two investigated datasets, that is training difficulty caused by the limited number of available training images, also commonly exists in multi-class nuclei segmentation tasks on other image modalities. Therefore, the proposed approach still holds great promise to be applied to the other microscopy image modalities.

Secondly, though the EfficientNet showed great promise on the multi-class nuclei segmentation task, the EfficientNet family has customized CNN models originally proposed for a natural image classification task on ImageNet. The choice of EfficientNet is quite limited, except for the 8 EfficientNet models. This could limit its adaptation to some problems that need more tuning of the network size. However, the studies related to the AE-U-Net can still provide a meaningful insight that a compact but powerful CNN model is very important for multi-class nuclei segmentation when there are not many training images available. This could provide guidance for network choice in the practical scenario.

Thirdly, in this study, we did not conduct intensive comparisons between our proposed approach with other CNN networks that were dedicated to some generic computer vision
problems, where there are usually a large amount of annotated images for the network training. However, the proposed approach still holds a great promise to address the generic segmentation problems. In the future, we will further validate our proposed method on medical images or even natural images.

Last but not the least, the sizes of the two datasets used in this study are small. In future, the generalization of the proposed method will be investigated on larger-size datasets.

### 3.5 Conclusion

In this study, we proposed an AE-U-Net-based approach for multi-class nuclei segmentation tasks. The proposed approach involves a compact but powerful CNN model, called AE-U-Net, and a transfer learning strategy in the network training. The AE-U-Net uses an EfficientNet in the encoding path, which allows the AE-U-Net to have a strong capability of image feature learning but have a compact size. In addition, the attention module enforces attention on the low-level feature maps to improve the quality of fused features in the decoding path. The compact but powerful AE-U-Net greatly mitigates the challenge of training a powerful network with a limited amount of available annotated microscopy images. The transfer learning strategy further addresses the challenge caused by data limitation. The experimental results on two challenging phase image datasets show that the proposed approach holds great promise to be applied to the practical scenario when only a limited amount of annotated images are available.
Chapter 4

Multi-channel EfficientNet-based U-Net for Label-free 3D Phase-to-fluorescent image translation

4.1 Introduction

Imaging of live cells have been increasingly performed by cell biologists to provide clues into the fundamental nature of cellular and tissue structure and function. A number of technical challenges must be overcome to perform successful live-cell imaging experiments, including the ability to maintain cells in a healthy state on the microscope stage for extended periods of time. Label-free imaging techniques have been widely used in live-cell biological experiments since they are quick, cheap, and non-invasive to live cells so that cells can be investigated over long periods. However, due to the lack of biological labeling, the acquired images suffer from low cell specificity, which limits its applications to sub-cellular studies. In contrast, as
one of the most widely used cell imaging techniques with biological labeling, fluorescence imaging can provide high cell specificity at the cost of more time and high expense as well as the phototoxicity to cells, which limits its application to a long-term investigation of live cells. A comparison between a label-free imaging technique and fluorescent imaging technique is shown in Fig. 4.1. These motivate researchers to translate label-free images to fluorescent images for long-term live-cell studies, which raise the label-free image modality translation problem.

Label-free image modality translation is a very complicated problem due to the fundamentally biophysical difference between label-free images and fluorescence images. Thanks to the advance of deep learning techniques, especially convolutional neuronal networks (CNN), this problem becomes solvable in recent years due to their strong capacity in learning informative

<table>
<thead>
<tr>
<th>Applications</th>
<th>Label-free</th>
<th>Fluorescence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Study cellular dynamics &amp; cell morphology</td>
<td>Study cellular processes and functions. Visualize specific structures and proteins</td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td>Quick</td>
<td>Can be very time consuming</td>
</tr>
<tr>
<td>Cost</td>
<td>Cheap</td>
<td>Can be very expensive</td>
</tr>
<tr>
<td>Invasiveness</td>
<td>Non-invasive. Cells unperturbed and in their natural state</td>
<td>Invasive labelling-techniques and phototoxicity</td>
</tr>
<tr>
<td>Measurement periods</td>
<td>Can analyze cells over long periods of time</td>
<td>Shorter periods of analysis due to phototoxicity.</td>
</tr>
</tbody>
</table>

Figure 4.1: Comparison between label-free imaging and Fluorescent imaging techniques.
image features for the modality translation. To date, a few works have investigated the translation problems. Most of them are focused on two-dimensional (2D) image translation. For example, Kandel et. al [50] and Jiao et. al [46] have trained U-Nets to perform an end-to-end mapping from 2-dimensional phase images to 2-dimensional fluorescent images. Lee et. al employed two U-Net models to translate the unstained brightfield images to fluorescent images. The first one is used for the translation and the second is used to refine the translated images. Rivenson et. al [72] used pairs of phase images and brightfield images to train a generative adversarial network (GAN) to transform quantitative phase images into stained brightfield images. Overall, the works discussed above were focused on the 2D label-free image translation problem.

Recently, three-dimension cell imaging has gained increasing interest in the biomedical field because it allows to better capture the interaction of cells and their biological surroundings and obtain more cell details, such as volume, surfaces, etc., which a two-dimension system can not provide. Therefore, 3D label-free image modality translation is getting more and more important in live-cell studies. Ounkomol et. al [68] demonstrated that U-Net can be used to predict 3D fluorescent images with bright-field images as inputs. However, the investigation of this topic is still insufficient so far. Quantitative phase imaging (QPI) is one of the most promising label-free imaging techniques that have been actively investigated to generate 3D cell images since they allow for high spatial resolution. As of now, the translation from 3D quantitative phase images to 3D fluorescent images has not been investigated yet. 3D phase-to-fluorescent translation is very challenging due to a couple of reasons as illustrated in Fig. 4.2. First of all, the phase images lack specificity, but fluorescent images contain cell specificity, as shown in the spatial plane in the 3D phase image in Fig. 4.2(a). The translation task then requires to infer those missing sub-cellar information from low-specificity images. Second, due to limitations of 3D phase microscopy, the acquired 3D phase images have low
axial resolutions and there exists scattering artifacts in the vertical plane in the 3D phase image. However, the target 3D fluorescence images have much higher axial resolution and no such imaging artifacts. Thus, the 3D translation process also needs to reconstruct artifact-free fluorescent images from artifact-contained phase images. Lastly, the target fluorescence images have low SNR so the ground truth in this translation task contains a large uncertainty. As shown in Fig. 4.2(b), the intensities of the cell signals are quite noisy in the profile.

Figure 4.2: Challenges related to 3D phase-to-fluorescence image translation.

In this study, we investigate this challenging 3D phase-to-fluorescent image translation problem and proposed an effective approach for it. In the proposed approach, a multichannel E-U-Net is employed to perform an end-to-end mapping from three neighboring phase image slices to the corresponding central fluorescent slice. This multichannel design allows an E-U-Net to exploit information from phase images acquired at multiple neighboring imaging planes to better predict the fluorescent image. In addition, different from a standard U-Net, an E-U-Net uses a compact but powerful CNN model, EfficientNet, as the encoder to extract
representative features from phase images. This compact design allows for mitigating the challenge in training a powerful model when only a limited amount of training data are available. We evaluated our proposed E-U-Net-based approach on three 3D phase image datasets that are related to microspheres, hippocampal neurons, and liver cancer spheroids, respectively. Experimental results show that the predicted 3D fluorescent images show great agreement with their corresponding ground-truth fluorescent images. To the best of our knowledge, this is the first work that uses deep neural networks to solve the 3D phase-to-fluorescent translation problem.

4.2 E-U-Net based phase-to-fluorescent Translation

4.2.1 Overview of E-U-Net

The overall architecture of a multichannel E-U-Net is shown in Fig. 4.3. The inputs of an E-U-Net are three neighboring phase slices, and the output is the corresponding central fluorescent slice. This network design allows an E-U-Net to exploit information from phase images acquired at multiple neighboring imaging planes to better predict the fluorescent image.

The E-U-Net uses a U-Net [73] as a basic architecture: an encoding and decoding paths that are connected via skip connections. Different from a standard U-Net, a compact but powerful CNN model, EfficientNet [88], is used in the encoding path of the AE-U-Net. The EfficientNet generally has a powerful capacity of feature extraction but is relatively small in network size. This compact design allows for training a powerful CNN model with relatively small size when the number of image samples is limited.
4.2.2 U-Net structure

U-Nets [73] were first proposed for biomedical image segmentation tasks, where the goal is to predict the corresponding segmentation map for a given biomedical image. In the recent few years, U-Nets have already been widely employed in all kinds of image processing tasks, such as medical image segmentation, image reconstruction, image translation.

As described in Chapter 3, a standard U-Net includes an encoding path and a decoding path that are connected via skip connections. In the encoding path, a sequence of down-sampled convolution layers is used to extract multi-level and multi-scale features from the input images. The layers close to input have smaller receptive fields thus the corresponding extracted features are local features. As the layers go deeper, the receptive fields of convolutional layers get larger and the extracted features are more and more global. These multi-level and multi-scale
features carry both localization and semantic information at each pixel and both are useful for recovering local details and global semantic patterns in the predicted fluorescent images.

In the decoding path, a sequence of up-sampled convolution layers is used to gradually restore the resolution of feature maps while fusing the local information from the encoding path and global information from the decoding path to generate fine granular features for a more accurate fluorescent image prediction. Furthermore, the skip connection allows providing a shortcut path for back-propagating the gradient to the layers of the encoder during the supervised learning of the U-Net.

Due to these advantages, our proposed approach also employed a standard U-Net as the basic architecture.

### 4.2.3 EfficientNet

EfficientNets refer to a set of 8 compact convolutional neural network (CNN) models that were proposed by Tan et al. [88] for an image classification task on the ImageNet [18] dataset containing millions of natural images. They deliver comparable capacity in learning image features compared to other popular CNN models, such as VGG [81], ResNet [29], etc., but have much smaller network sizes. The efficiency of EfficientNets compared to other state-of-the-art models is shown in Fig. 3.2.

As two EfficientNet examples, the architectures of EfficientNet-B0 and EfficientNet-B7 are shown in Fig. 4.4. More details related to EfficientNet can be found in Section 3.2.1. When applied to an input image, an EfficientNet CNN model will generate intermediate outputs at each CNN block. These intermediate outputs are feature maps where each pixel represents a feature vector that corresponds to a receptive field of pixels in the input image. The deeper CNN blocks have larger receptive fields, thus their produced feature maps contain features
with higher levels. In addition, due to the down-sampled operations, these feature maps also have smaller dimensions.

In the proposed E-U-Net-based approach, this compact but powerful CNN model EfficientNet is used to extract multi-level and multi-scale features. The strong capacity of an EfficientNet can guarantee the E-U-Net to learn informative features related to fluorescence prediction from the input phase images. In addition, the compact design allows for mitigating the challenges of network training with a limited amount of training data.

### 4.3 Experiments

#### 4.3.1 Datasets

In this study, a neuron dataset, a spheroid cell dataset, and a bead dataset were used to train, validate and test the proposed AE-U-Net-based method. The neuron dataset contained 22 3D...
image stacks that each contained 300 neuron phase images of size $1744 \times 1744$ pixels and their related two-channel fluorescent images, which correspond to fluorescent signals from Tau and MAP2 proteins, respectively. One example of neuron stack is shown in Fig. 4.5. The spheroid cell dataset contained 21 3D image stacks that each contained 100 spheroid cell phase images of size $1744 \times 1744$ pixels and the related two-channel fluorescent images, which correspond to fluorescent signals from DNA and RNA, respectively. One example of spheroid cell stack is shown in Fig. 4.6(a). The bead dataset contained 18 3D image stacks that each contained 250 bead phase images of size $128 \times 128$ pixels and the associated fluorescent images. One example of bead stack is shown in Fig. 4.6(b).

All the three datasets were acquired by Prof. Gabriel Popescu’s group with their advanced QPI system, called phase imaging with computing specificity (PICS) [38, 41, 42, 43]. The pipeline for the 3D cell imaging is shown in Fig. 4.7. The image acquisition was performed by use of a multichannel imaging system that consists of confocal microscopy (LSM 900,
Zeiss) and laser-scanning interference microscopy. The laser-scanning interference microscopy upgrades a differential interference contrast (DIC) microscopy with a phase-shifting assembly (Fig. 4.7(a)). The laser-scanning interference microscopy shares the same two-laser lines (488nm, 561nm) inside of the confocal microscopy. The laser source from the confocal microscopy goes up through the DIC prism (63, 40), the objective (63, 40), and then is scattered by the sample. After the sample, the light is collected by the condenser of the DIC microscopy. The DIC prism (DIC III) inside of the condenser does not have a polarizer. Next, the light travels through the phase-shifting assembly, which consists of a liquid crystal variable retarder (LCVR, Thorlabs) and a linear polarizer. The stabilization time of the LCVR is about 70 ms. Four intensity frames are recorded by the photomultiplier tube (PMT, Zeiss) corresponding to each phase shift, as shown in Fig. 4.7(b). The acquisition time of each frame takes about the same time as the acquisition time of a confocal fluorescence image, which depends on the dwell time and pixel numbers in the image. The dwell time for all
the images was chosen to be 1.21 μs. For an image with 1744 × 1744 pixels, the acquisition time is about 4s with a 1.21 μs dwell time. The quantitative phase images are generated in real-time by the phase-retrieval reconstruction algorithm and Hilbert-inspired transform algorithm. The system registers pairs of confocal fluorescence and quantitative phase z-stack images, which serve as ground truth and input images for machine learning (Fig. 1(b)). The z sampling was chosen to be 0.2, 0.2, 1 for microbeads, neurons, and spheroids, respectively. The XY sampling was 0.092 for all the data used in this study.

![Diagram](image.png)

Figure 4.7: 3D cell imaging pipeline [14].

### 4.3.2 E-U-Net training

The multichannel E-U-Nets were trained with paired phase and fluorescent images. The input channels of an E-U-Net are three neighboring phase slices, and the output is the
corresponding central fluorescent slice. Training an E-U-Net from scratch can be challenging when the number of paired phase and fluorescent images is limited. To mitigate this challenge, a transfer learning strategy was used in the E-U-Net training. Specifically, the weights of EfficientNet were initialized with weights pre-trained on an ImageNet dataset for an image classification task. The ImageNet is a benchmark image set that contains millions of labeled natural images.

To facilitate network training, the pixel values in each fluorescent image stack were scaled to a range of $[0, 255.0]$. This was accomplished as: $x_0 = 255.0 \times \frac{x_i - x_{0.01\%}}{x_{99.9\%} - x_{0.01\%}}$, where $x_{0.01\%}$ and $x_{99.99\%}$ represent the $0.01\%$-th and $99.99\%$-th values among all the pixel values in the image stack after they were sorted in a non-decreasing order; $x_i$ and $x_o$ represent the original value and the scaled value of a pixel, respectively. The estimated fluorescent image stack was subsequently rescaled to its original range using $x_i = \frac{x_0}{255.0} (x_{99.99\%} - x_{0.01\%}) + x_{0.01\%}$. For those image stacks without ground truth values, the $x_{0.01\%}$ and $x_{99.9\%}$ can be estimated as the average of $x_{0.01\%}$ and $x_{99.9\%}$ related to ground truths in the training set, respectively.

Considering the small number of image stacks in the three datasets described above, 3-fold cross-validation (CV) approach was employed to train and validate the E-U-Nets after a few testing image stacks were held out for the E-U-Net testing. For a given dataset in which the testing stacks have been held out, the 3-fold CV approach involves randomly dividing all the stacks in the dataset into three folds of approximately equal size. The first two folds and the remaining one fold were treated as a training set and a validation set to train and validate E-U-Nets, respectively. The procedure was repeated three times; each time, a different fold was treated as the validation set. The three procedures resulted in the validation of the E-U-Nets on each image stack. The trained E-U-Nets were finally tested on the held-out unseen testing samples. Details related to the cross-validation of E-U-Nets on the neuron, spheroids cell, and bead datasets are described below.
For the neuron dataset, two separate E-U-Nets were trained: one to translate phase images into each of the two-channel fluorescent images. The EfficientNet-B7 network was employed in the two E-U-Nets. The network architecture of the EfficientNet-B7 is shown in Fig. 4.4. Two neuron image stacks were held out as unseen testing stacks; the remaining 20 stacks were employed in the 3-fold CV process described above. In the 3-fold CV process, the 20 image stacks were randomly divided into three folds that contained 6, 7, and 7 image stacks, respectively. For each data split, the E-U-Nets were trained by minimizing a mean square error (MSE) loss function that measures the difference between the predicted fluorescent images and their corresponding ground truth values. The loss function was minimized by the use of an ADAM optimizer [52] with a learning rate of $5 \times 10^{-4}$, which was empirically determined. In each training iteration, a batch of paired 3 neighboring phase images and the corresponding central fluorescent image was sampled from the training image stacks, and then randomly cropped into patches of $512 \times 512$ pixels as training samples to train the networks. The batch size was set to 4. A decaying strategy was applied to the learning rate to mitigate the overfitting by multiplying the learning rate by 0.8 when the validation MSE loss did not decrease for consecutive 10 epochs. An epoch is a sequence of iterations that walk through all the image slices in the training set. The validation MSE loss was computed between the predicted fluorescent images and their ground truth values for validation images. In the network training, an early-stopping strategy was employed to determine the end of the network training. Specifically, at the end of each epoch, the being-trained E-U-Net model was evaluated by computing the average of Pearson correlation coefficients (PCCs) between the predicted fluorescent images and the related ground truth values. The network training stopped if the average validation PCC did not increase for 20 epochs as shown in Fig. 4.8(a) and Fig. 4.8(b). The two figures show the average training and validation PCCs for training the two E-U-Nets respectively in one of the three training procedures of the 3-fold CV process. After the E-U-Nets were trained, the performances of the trained networks were evaluated on
the validation set by computing the peak signal-to-noise ratio (PSNR) and PCC between the predicted fluorescent stacks and the related ground truth values. The 3-fold CV process resulted in validation results for each of the 20 stacks. In addition, the E-U-Nets trained in the CV process were tested in the two unseen neuron stacks.

Figure 4.8: Stopping rule metric for the E-U-Net training on neuron dataset. The training metric was defined as the Pearson’s $r$ between the predicted fluorescent image patches and their corresponding ground truth values; the validation metric was defined as the predicted image slices and their corresponding ground truths.

For the spheroids cell dataset, two separate E-U-Nets were trained: one to translate phase images into each of the two channel fluorescent images. The EfficientNet-B7 network was employed as the encoder in the two trained E-U-Nets. Two spheroid cell image stacks were held out for E-U-Net testing; the remaining 19 stacks were randomly split into three folds that contain 6, 6, and 7 stacks, respectively, in the 3-fold CV process. The other training settings were the same as those described above for network training on the neuron dataset. The training and validation PCCs over epochs correspond to training the two E-U-Nets in one of the three training procedures of the 3-fold CV process are displayed in Fig. 4.9(a) and Fig. 4.9(b), respectively.
Figure 4.9: Stopping rule metric for the E-U-Net training on neuron dataset. The training metric was defined as the Pearson’s $r$ between the predicted fluorescent image patches and their corresponding ground truth values; the validation metric was defined as the predicted image slices and their corresponding ground truths.

For the bead dataset, a single E-U-Net was built for the phase-to-fluorescent image translation. The EfficientNet-B0 was employed as the encoder in the E-U-Net. The architecture of the EfficientNet-B0 network is shown in Fig. 4.4. One of the bead image stacks was held out as unseen testing stack for the E-U-Net testing; the remaining 17 bead stacks were randomly divided into three folds that each contains 5, 6, and 6 image stacks, respectively, for the 3-fold CV process. Paired images of size 128×128 pixels were employed for the E-U-Net training. The batch size was 32. The other training settings were the same as those for the network training on neuron and spheroid cell datasets, as described above. The training and validation PCCs over epochs for one of the three training procedures of the 3-fold CV process are displayed in Fig. 4.10.

The E-U-Nets were implemented by use of the Python programming language with libraries including Python 3.6 and Tensorflow 1.14. The model training, validation and testing were performed on a NVIDIA Tesla V100-GPU with 32 GB VRAM. The E-U-Net training on the neuron dataset and spheroid dataset took approximately 24 hours. The E-U-Net training on
the bead dataset took approximately 2 hours. The inference time for a fluorescent image slice of $1744 \times 1744$ pixels was about approximately 400 ms.

### 4.3.3 Performance Metrics

In this study, the prediction performances on validation and unseen testing stacks were evaluated with two metrics: pearson correlation coefficient (PCC) and peak-signal-noise-rate (PSNR), respectively. The PCC for a predicted image stack is defined as:

$$PCC = \frac{\sum (\hat{I}_i - \bar{\hat{I}})(I_i - \bar{I})}{\sqrt{\sum (\hat{I}_i - \bar{\hat{I}})^2 \sum (I_i - \bar{I})^2}}$$

(4.1)

where $\hat{I}_i$ and $I_i$ are the intensities at pixel $i$ in the predicted fluorescent image stack and its corresponding ground truth value; the $\bar{\hat{I}}$ and $\bar{I}$ are the mean of intensity values in the predicted stacks and their corresponding stacks, respectively.
The PSNR for a predicted image stack is defined as:

\[
PSNR = 10 \log \frac{I_{max}^2}{MSE},
\]

where \(I_{max}\) is the maximum intensity in the ground truth stack, and \(MSE\) is the mean square error computed between all the intensities in the predicted fluorescent image stack and their corresponding ground truth values.

### 4.3.4 Experimental Results

The PCC and PSNR related to each fluorescent channel were computed for each validation stack in the three datasets: neuron dataset, spheroids cell dataset, and bead dataset. The validation results for each dataset were combined separately, and the are displayed in Fig. 4.11, Fig. 4.13 and Fig. 4.15, respectively. In addition, the PCC and PSNR computed for the unseen testing stacks in the three datasets are presented in Table 4.1, Table 4.2, and Table 4.3, respectively. Furthermore, the predicted fluorescent images and the related ground truths for these unseen stacks in the three datasets are displayed in Fig. 4.12, Fig. 4.14 and Fig. 4.16, respectively.

Fig. 4.11 shows that the average PCCs related to Tau and MAP2 fluorescent channels over the 20 neuron image stacks are around 0.74 and 0.88, respectively; the average PSNRs related to the two fluorescent channels are 28 and 30, respectively. The CV results demonstrate that when trained on neuron datasets, the trained E-U-Nets can accurately predict 3D fluorescent images. In addition, Table 4.1 shows that the PCCs and PSNR performances on the unseen neuron stacks are consistent with the validation results. Besides, Fig. 4.12 shows that, visually, the predicted fluorescent images are consistent with their ground truth for both the two...
fluorescent channels. This indicates that the trained E-U-Net models generalize well to unseen testing neuron stacks.

Figure 4.11: Validation results on the neuron dataset

Table 4.1: PCC and PSNR performances on unseen neuron stacks

<table>
<thead>
<tr>
<th>Unseen samples</th>
<th>Channels</th>
<th>PCC</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Model1</td>
<td>Model2</td>
</tr>
<tr>
<td>neuron stack1</td>
<td>FL1</td>
<td>0.7923</td>
<td>0.8023</td>
</tr>
<tr>
<td></td>
<td>FL2</td>
<td>0.8978</td>
<td>0.9187</td>
</tr>
<tr>
<td>neuron stack2</td>
<td>FL1</td>
<td>0.7286</td>
<td>0.7733</td>
</tr>
<tr>
<td></td>
<td>FL2</td>
<td>0.8942</td>
<td>0.9127</td>
</tr>
</tbody>
</table>

Fig. 4.13 shows that the average PCCs related to DNA and RNA fluorescent channels over the 19 spheroids cell stacks are around 0.87 and 0.75, respectively; the average PSNRs related to the two fluorescent channels are both around 27. Again, the CV results demonstrate that when trained on the spheroid cell dataset, the trained E-U-Nets can also accurately predict 3D fluorescent images for the two fluorescent channels. In addition, Table 4.2 shows that the PCCs and PSNR performances on the unseen spheroid cell stacks are consistent
Figure 4.12: Prediction for unseen neuron stack.

Figure 4.13: Validation results on the spheroid cell dataset
Figure 4.14: Prediction for unseen neuron stack.
with the validation results. Besides, it can be observed from Fig. 4.14 that, visually, the predicted fluorescent images for both the two fluorescent channels look consistent with their corresponding ground truth. It indicates that the trained E-U-Net models generalize well to unseen testing spheroids cell stacks. These results demonstrate that the E-U-Net-based approach can also deliver an accurate predictions on the spheroids cell dataset.

Table 4.2: PCC and PSNR performances on unseen spheroids cell stacks

<table>
<thead>
<tr>
<th>Unseen samples</th>
<th>Channels</th>
<th>PCC</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Model1</td>
<td>Model2</td>
</tr>
<tr>
<td>spheroids cell stack1</td>
<td>FL1</td>
<td>0.8360</td>
<td>0.8340</td>
</tr>
<tr>
<td></td>
<td>FL2</td>
<td>0.6890</td>
<td>0.6957</td>
</tr>
</tbody>
</table>

Fig. 4.15 shows that the median of PCC over the 17 bead stacks is around 0.979; the median PSNR is 46. Table 4.3 shows that the PCC and PSNR performances on the unseen bead stack are consistent with the validation results. Besides, Fig. 4.16 shows that the predicted fluorescent images are visually consistent with ground truth. This indicates that the trained E-U-Net models can generalize well to the unseen bead image stack. These results demonstrate that the E-U-Net-based approach can deliver accurate predictions on the bead dataset as well. In addition, PCCs and PSNRs obtained on the bead dataset are higher than those obtained on the neuron and spheroids cell datasets. That is because bead has a much simpler structure than neuron and spheroids cells and the corresponding phase-to-fluorescence translation task is much simpler than the other two.

Table 4.3: PCC and PSNR performances on unseen bead stacks

<table>
<thead>
<tr>
<th>Unseen samples</th>
<th>Channels</th>
<th>PCC</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Model1</td>
<td>Model2</td>
</tr>
<tr>
<td>bead stack1</td>
<td>FL1</td>
<td>0.97790</td>
<td>0.9799</td>
</tr>
</tbody>
</table>

93
Figure 4.15: Validation results on bead dataset

Figure 4.16: Prediction for unseen neuron stack.
4.3.5 Comparison to one-plane input

To investigate the effectiveness of multi-channel design, we compared a E-U-Net with 1-plane input and that with 3-planes input. The only difference between the two E-U-Nets were the number of planes in their inputs. For simplicity, let the two networks be denoted as E-U-Net(3p) and E-U-Net(1p), respectively. Without loss of generality, EfficientNet-B0 was used in the encoding paths of both the two E-U-Net. The two E-U-Nets were evaluated on both the neuron dataset and spheroids dataset with the 3-fold cross-validation process described in Section 4.3.2. On the neuron dataset, the method evaluation was performed for Tau fluorescent channel; on the spheroids cell dataset, the evaluation was performed for DNA fluorescent channel, considering the ground truth fluorescent images have better image qualities in these two channels. In the CV process, for each given image stack, both the two E-U-Nets will generate a validation result. The corresponding validation PCCs at the image stack can then be computed based on the validation results. Bland-Altman (BA) plots were used to plot the PCC differences between the two networks at each validation stack in each of two datasets. In the BA plot, the vertical axis represents the subtraction of the PCC related to E-U-Net(3p) and that related to E-U-Net(1p) at any image stack; horizontal axis represents the average of the two PCCs at the stack. A image stack with positive PCC difference indicates that the E-U-Net(3p) yields higher validation PCC than E-U-Net(3p) does at the stack. The BA plot results on the two datasets are displayed in Fig. 4.17(a) and Fig. 4.17(b), respectively.

It can be seen from Fig. 4.17(a) that E-U-Net(3p) yielded higher validation PCC than E-U-Net(1p) did at almost all the neuron image stacks and all the spheroids cell image stacks.

As an example, the prediction results related to the two E-U-Nets for an example spheroid cell image slice are shown in Fig. 4.18, respectively. From the highlighted regions both the
two figures, we can see that the predictions related to the E-U-Net(3p) are visually much closer to the ground truth than that related to E-U-Net(1p). Also, the error maps related to the E-U-Net(3p) have less errors than those related to the E-U-Net(1p). These visualized results further support that multichannel inputs can improve the performances of fluorescent image prediction.

4.3.6 Comparison to a standard U-Net

In this study, we also compared the proposed approach with a standard U-Net, considering that it was most widely employed in microscopy image translation tasks. For a fair comparison, both the two networks used three planes as input. EfficientNet-B0 was used in this E-U-Net to make sure the two networks have comparable network sizes. The evaluation of the two methods was performed in both the Tau fluorescent channel on the neuron dataset and the DNA channel on the spheroids dataset. Both of the methods were evaluated with 3-fold cross-validation from scratch. Bland-Altman (BA) plots were used to plot the PCC differences related to the two trained networks that were computed on each image stack in the two
datasets. The corresponding results related to neuron and spheroid cell datasets are shown in Fig. 4.19(a) and Fig. 4.19(b), respectively.

Figure 4.19: BA plots of the E-U-Net and U-Net on validation neuron and spheroid cell image stacks, respectively. The red dash lines indicate the mean of the PCC differences.
It can be observed in the Fig. 4.19(a) and Fig. 4.19(b) that the proposed E-U-Net based approach yielded higher validation PCC ($PCC_{E-U-Net} - PCC_{U-Net} > 0$) at the majority of the validation neuron stacks and all the validation spheroids stacks. These results demonstrated the superior performance of the proposed method compared to the standard U-Net.

In addition, the fluorescent predictions for one spheroid cell phase image slice are displayed in Fig. 4.20. From the highlighted regions, we can see that the prediction related to the E-U-Net is visually much closer to the ground truth than that related to the U-Net. Also, the error map related to the E-U-Net has much less errors than that related to the U-Net. These visualized results further support that E-U-Net show superior prediction performance to the standard U-Net.

Figure 4.20: Fluorescence prediction with 1-plane input and that with 3-planes input for an example spheroid cell image slice.
4.4 Discussion

3D cell imaging has shown increasing interest in the biomedical field because it allows to better capture the interaction of cells and their biological surroundings and obtain more cell details. 3D phase-to-fluorescence translation is getting more and more important in live-cell studies, but it is very challenging due to the lack of cell specificity related to phase images, scattering phase imaging artifacts, and low SNR in the target fluorescence images.

In this study, we have explored this 3D phase-to-fluorescent image translation problem. Our contributions related to this task are three fold. First, we proposed a multichannel E-U-Net-based approach for the 3D phase-to-fluorescent image translation problem. To the best of our knowledge, this is the first work related to this problem. Second, we have systematically evaluated the proposed approach on three challenging yet biologically-significant datasets that related to hippocampal neurons, liver cancer spheroids, and microspheres, respectively. For the neuron dataset and live cancer spheroids dataset, the prediction of two fluorescent channels has been investigated. The experimental results demonstrate the proposed approach has impressive performance on the three datasets. In addition, we have compared the proposed approach with existing methods for similar problems. The results suggest that the proposed approach show superior performance compared to the compared method.

Nevertheless, the limitations related to this work are summarized below. First of all, due to difficulty in the acquisition of a large number of image stacks, our current study for this problem was based on three datasets that each have a small number of image stacks. Although a cross-validation approach was employed to evaluate the proposed approach, it remains greatly necessary to validate the method on a large-size dataset. Thus, validation of the proposed method on a large-size dataset is one of the future directions to be investigated if a larger dataset is available in the future.
Secondly, despite the impressive overall prediction performance on the three datasets, the method has relatively worse prediction performance for the fluorescent channel related to TAU protein on the neuron dataset and for the fluorescent channel related to DNA on spheroids datasets. We believed that was caused by the much lower signal-noise-ratio (SNR) in the ground truth fluorescent images related to those two channels. The lower image quality of ground truths results in the higher uncertainty in the MSE loss used in network training and then degraded the prediction performance of the trained models. In addition, the noise in low SNR images will more easily degrade the computed overall performance, compared to those in the images with higher SNR. Thus, another future direction is to investigate how to improve the prediction performance for those two fluorescent channels by including de-noising techniques in the ground truth preparation.

Thirdly, the interpretability of impressive prediction delivered by the proposed E-U-Net methods is quite limited, as all the other deep learning-based approaches may have for other image processing tasks. However, the interpretability of deep neural networks is a challenging problem, and still remains an open question to which lots of researchers are striving to find answers.

4.5 Conclusion

In this study, we investigated a challenging 3D phase-to-fluorescent translation problem and proposed a multichannel E-U-Net-based translation approach. The proposed approach has been evaluated on three 3D phase image datasets that are related to microspheres, hippocampal neurons, and liver cancer spheroids, respectively, and shown promising prediction performances qualitatively and quantitatively. We have also compared our method with other
competing microscopy image translation methods. The proposed approach show superior performance than the compared methods in terms of average PCC and average PSNR.
Chapter 5

Summary

In this dissertation, we have investigated deep learning methods for the automatic microscopy image analysis (MIA) tasks of cell counting, multi-class nuclei segmentation, and 3D phase-to-fluorescent image modality translation.

In Chapter 3, a deeply-supervised density regression model was proposed for accurately and robustly counting the number of cells in microscopy images. The proposed method was capable of processing varied-size images containing dense cell clusters, large variations of cell morphology and inhomogeneous background noise. Extensive experiments based on four datasets representing different image modalities and image acquisition techniques demonstrated the efficiency, robustness, and generality of the proposed method. The proposed method can be potentially be applied to real-time clinical applications. It also holds the promise to be applied to a number of different problems, such as object counting (other than cells) in crowded scenes.

In Chapter 4, we proposed an AE-U-Net-based approach for multi-class nuclei segmentation tasks. The proposed approach involves a compact but powerful CNN model, called AE-U-Net,
and a transfer learning strategy in the network training. The AE-U-Net uses an EfficientNet in the encoding path, which allows the AE-U-Net to have a strong capability of image feature learning but have a compact size. In addition, the attention modules enforce attention on the low-level feature maps to improve the quality of fused features in the decoding path. The compact but powerful AE-U-Net greatly mitigates the challenge of training a powerful network with a limited amount of available annotated microscopy images. The transfer learning strategy further addresses the challenge caused by data limitation. The experimental results on two challenging phase image datasets show that the proposed approach holds great promise to be applied to a practical scenario when an only limited amount of annotated images are available.

In Chapter 5, we investigated a challenging 3D phase-to-fluorescent translation problem and proposed a multichannel E-U-Net-based translation approach. The proposed approach has been evaluated on three 3D phase image datasets that are related to microspheres, hippocampal neurons, and liver cancer spheroids, respectively, and shown promising prediction performances qualitatively and quantitatively. We have also compared our method with other competing microscopy image translation methods. The proposed approach show superior performance than the compared methods in terms of average PCC and average PSNR.

Many topics remain for future investigation. First of all, in biomedical applications, though annotated images are expensive to obtain, unannotated images are easy to obtain. Semi-supervised learning methods that used a large number of unannotated images and a small amount of annotated images hold great promise to improve the performance of deep learning methods on MIA tasks.

Secondly, the accuracy of image annotation is important for supervised deep learning. In the study related to cell counting, the annotation of images in three experimental datasets was
annotated with one-time annotation, which could cause uncertainty in the prepared ground truth. This eventually causes uncertainty in deep learning training and method evaluation. Thus, in the future, it might need to use datasets with more accurate annotation for this study.

Thirdly, in the studies related to the multi-nuclei segmentation task, quantitative phase images were the only image modality that was employed in method evaluation, but the proposed AE-U-Net-based approach can be potentially employed in other image modality or other biomedical image segmentation tasks. Thus, in the future, the proposed AE-U-Net will be evaluated with image datasets with different modalities and for generic biomedical image segmentation tasks.

Fourthly, in the studies related to the 3D phase-to-fluorescent image translation, only 2D networks were investigated due to the challenge in training a 3D network from scratch. In the future, 3D CNN models will be investigated by overcoming those training challenges.

Last but not the least, the sizes of datasets used in the three topics are small, especially for the cell counting task and image translation task. In future, the generalization of our methods to larger-size datasets will be explored.
References


Shenghua He, Kyaw Thu Minn, Lilianna Solnica-Krezel, Mark A Anastasio, and Hua Li. “Deeply-supervised density regression for automatic cell counting in microscopy images.” In: Medical Image Analysis 68 (2021), p. 101892.


[40] Chenfei Hu, Shenghua He, Young Jae Lee, Yuchen He, Mark Anastasio, and Gabriel Popescu. “Label-free cell viability assay using phase imaging with computational specificity (PICS).” In: Quantitative Phase Imaging VII. Vol. 11653. International Society for Optics and Photonics. 2021, p. 116531D.


Kyaw Thu Minn, Yuheng C Fu, Shenghua He, Steven C George, Mark A Anastasio, Samantha A Morris, and Lilianna Solnica-Krezel. “High-resolution transcriptional and morphogenetic profiling of cells from micropatterned human embryonic stem cell gastruloid cultures.” In: *DEVELOPMENTAL-CELL-D-20-00079* (2020).


