



Developing Automated Cell Segmentation Models Intended for MERFISH Analysis of the Cardiac Tissue by Deploying Supervised Machine Learning Algorithms

A project carried out in the Kosuri Lab at Salk Institute for Biological Studies

Julia Rune

Degree project in Chemical Engineering (KE200X)
Second cycle, 30 credits

Department of Chemistry
KTH, Royal Institute of Technology
Stockholm, Sweden
2023-06-22

Abstract

The following study delves into the development of automated cell segmentation models, with the intention of identifying boundaries between cells in the cardiac tissue for analysing spatial transcriptomics data. Addressing the limitations of alternative techniques like single-cell RNA sequencing (ScRNA-seq) and single molecule fluorescence in situ hybridization (smFISH), the study underscores the innovative use of multiplexed error-robust fluorescence in situ hybridization (MERFISH) deployed by the Kosuri Lab at Salk Institute for Biological Studies. This advanced imaging-based technique allows for a single-cell transcriptome profiling of hundreds of different transcripts while retaining the spatial context of the tissue. The technique can accordingly reveal how the organization of cells within a healthy heart is altered during disease. However, the extraction of meaningful data from MERFISH poses a significant challenge - accurate cell segmentation. This thesis therefore presents the development of a robust model for cell boundary identification within cardiac tissue, leveraging some of the advanced supervised machine learning algorithms in the field, named Cellpose and Omnipose. Due to the dense and highly heterogeneous tissue- stemming from a wide distribution of cell types and shapes- two separate models had to be developed; one that covers the smaller cells and the cross-sectioned cardiomyocytes, and correspondingly one to cover the longitudinal cardiomyocytes. The cross-section model was successfully developed to achieve an accuracy of 91.2%, whereas the longitudinal model still needs further improvements before being implemented. The thesis acknowledges potential areas for improvement, emphasizing the need to further improve the segmentation of longitudinal cardiomyocytes, tackle the challenges with segmenting cells within fibrotic regions of the diseased heart, as well as achieving a precise 3D cell segmentation. Nonetheless, the generated models have paved the way towards enabling efficient downstream MERFISH analysis to ultimately understand the structural and functional dynamics of heart failure at a cellular level, aiding the development of more effective therapeutic strategies.

Keywords:

Cell Segmentation, Cellpose, Supervised Machine Learning, MERFISH, Heart Failure.

Sammanfattning

Följande studie behandlar utvecklandet av automatiserade cellsegmenteringsmodeller med avsikt att identifiera gränser mellan celler i hjärtvävnad. Syftet är att möjliggöra analys av data genererad från *multiplexed error-robust in situ hybridization* (MERFISH). MERFISH är en spatial transcriptomics-teknik som till skillnad från exempelvis *single-cell RNA sequencing* (ScRNA-seq) och *single molecule fluorescence in situ hybridization* (smFISH), möjliggör profilering av hundratals RNA-sekvenser hos enskilda celler utan att förlora dess rumsliga kontext. I Kosuri laboratoriet på Salk Institute of Biological Studies i San Diego tillämpas MERFISH på mushjärtan. Syftet är att få en djupare insikt i hur celler är organiserade i friska hjärtan, och hur denna struktur ändras i och med åldring och sjukdom. Att extrahera meningsfull information från MERFISH medför dock en betydande utmaning - en exakt cellsegmentering. Studien bidrar följaktligen till utvecklandet av segmenteringsmodeller för att kringgå de utmaningar som står i vägen för all efterföljande analys. Då klassiska segmenteringsalgoritmer är otillräckliga för att segmentera den komplexa vävnad som hjärtat utgörs av, tillämpades några av dagens mest avancerade och framstående maskininlärningsalgoritmer inom fältet, kallade Cellpose och Omnipose. Givet den täta och heterogena hjärtvävnaden, som härstammar från en bred distribution av celltyper och geometrier, utvecklades två separata modeller; en för att täcka både mindre celler och kardiomyocyter skurna på tvärsnittet; och en för att enbart segmentera kardiomyocyter skurna i longitudinell riktning. Den förstnämnda modellen utvecklades och tränades i Cellpose, och uppnådde en träffsäkerhet på 91.2%. Modellen för longitudinella kardiomyocyter utvecklades istället både i Cellpose och Omnipose för att utvärdera vilket nätverk som är bäst lämpat för ändamålet. Ingen av nätverken lyckades uppnå en tillräckligt hög träffsäkerhet för att vara applicerbar, och är därmed i behov av fortsatt träning. Modellen genererad i Omnipose bedöms dock vara mest lovande, givet dess mer heltäckande segmentering. Ytterligare utvecklingsområden för framtiden innefattar segmentering av celler i fibros-täta regioner, samt att utveckla en 3D-segmentering av hela hjärtat för att uppnå en mer komplett MERFISH-analys. Sammanfattningsvis har de genererade segmenteringsmodellerna banat väg för möjliggörandet av en rigorös MERFISH-analys av hjärtat. Genom att avslöja några av de strukturella och funktionella orsakerna till hjärtsvikt på en cellulär nivå, kan vi således på sikt bidra till utvecklingen av mer effektiva terapeutiska strategier.

Nyckelord:

Cellsegmentering, Cellpose, Övervakad Maskininläring, MERFISH, Hjärtsvikt.

Acknowledgements

I would like to express my deepest gratitude to several individuals who have played instrumental roles in the completion of this master thesis. First and foremost, I would like to thank my supervisor Asst. Prof. Pallav Kosuri for inviting me to his lab for this semester. His expertise, enthusiasm, and dedication have been incredibly inspiring and fostering for my intellectual growth. Additionally, I wish to express my gratitude to my mentor Dr. Yuening Liu, whose mentorship, insightful feedback, and continuous encouragement have significantly contributed to the development and refinement of this work. Her commitment to excellence and willingness to share her knowledge and expertise have been truly transformative, and I am grateful for the opportunities I have had to learn from her. I certainly also want to thank the rest of the team at the Kosuri Lab for all the support. In addition, I would like to express my appreciation to my examiner Prof. Afshin Ahmadian and my opponent Roj Satis for the insightful feedback and input. I am grateful for their time and expertise invested in reviewing this thesis. Finally, I would like to thank the Department of Chemistry at KTH and Salk Institute for Biological Studies for providing me the opportunity to write my thesis on this influential topic, and to the Karl Engvers Foundation for the generous funding to make this possible.

Author

Julia Rune
Department of Chemistry
KTH Royal Institute of Technology
Stockholm, Sweden

Place for Project

The Kosuri Lab
Salk Institute for Biological Studies
San Diego CA, USA

Supervisors

Pallav Kosuri & Yuening Liu
The Kosuri Lab
Salk Institute of Biological Studies
San Diego CA, USA

Examiner

Afshin Ahmadian
KTH Royal Institute of Technology
Stockholm, Sweden

Contents

1	Introduction	10
2	Theoretical Background	12
2.1	Cell Type Classification & Function of the Mammalian Heart	12
2.2	Spatial Transcriptomics & Fluorescence In Situ Hybridization	13
2.2.1	Multiplexed Error-Robust FISH	15
2.3	Cell Segmentation	16
2.3.1	Classical Segmentation Algorithms	17
2.3.2	Machine Learning Algorithms	18
3	Aim	25
3.1	Limitations & Delimitations	25
3.2	Outline of Thesis	25
4	Methodology	26
4.1	Cell Segmentation - Prerequisites, Requirements & Strategy	26
4.1.1	Quantifying Model Progression	28
4.2	Adapted Segmentation Algorithms	28
4.2.1	Cellpose 2.0	29
4.2.2	Omnipose	31
4.3	Model Development	32
4.3.1	Over & Underfitting	33
4.3.2	Cross-Section Model	33
4.3.3	Longitudinal Model	34
4.4	Supplementary Software for Image Processing	34
5	Results & Discussion	35
5.1	Cross-section Model	35
5.1.1	Benchmarking	35

5.1.2	Model Training & Performance	36
5.1.3	Threshold Optimization	39
5.2	Longitudinal Model	43
5.2.1	Stitching and Flat-field Correction	43
5.2.2	Cellpose Model	45
5.2.3	Omnipose Model	50
5.2.4	Cellpose VS Omnipose	52
5.3	Overall Performance	53
6	Conclusions & Final Remarks	55
6.1	Future Prospects	55
	References	61
	Appendices	62

List of Figures

1	MERFISH analysis pipeline	10
2	Cell Types of the Mammalian Heart	13
3	Barcode-readout Schematics of MERFISH	16
4	Watershed Transform	18
5	U-Net Architecture	20
6	Main Steps of Gradient Flow Tracking	21
7	Sink Attraction Basins	21
8	Cellpose Model Architecture	22
9	Microscopy Images of the Cardiac Tissue	26
10	Examples of Inaccurate Segmentation	27
11	Cellpose Graphical User Interface (GUI)	29
12	Cell Probability Function	30
13	Omnipose GUI	31
14	Default Model CPx Performance on Cross-section area of Cardiac Tissue	36
15	Ground-truth Masks For Training a Cross-section Model	37
16	Segmentation from Each Model Version of the Cross-section Models Version 1-3	37
17	Diameter Adjustments Cross-section Model Version 3	39
18	Flow Threshold Comparison For Cross-section Model Version 3	40
19	Cell Probability Threshold Comparison For Cross-section Model Version 3	41
20	Threshold Chart For Cross-section Model Version 3	42
21	FOV Stitching	44
22	FF Corrected and Stitched FOVs	44
23	Ground-truth Masks Longitudinal Model	45
24	Cross-section Model Version 3 Applied on a Longitudinal Region	46
25	Cellpose Benchmark Longitudinal Model	47
26	Segmentation From Longitudinal Cellpose Model Trained From Scratch	47
27	Training Progress Longitudinal Model Cellpose	48
28	Changing flow and cell probability threshold CPx based longitudinal model	49
29	Comparison Cell Probability Threshold Longitudinal Cellpose Model	49
30	Benchmark Omnipose Default Model	51
31	Close-up of a Default Segmentation in Omnipose Using the bact_phase_omni Model	51
32	Resulting Segmentation From Longitudinal Omnipose Model Trained From Scratch	52
33	Longitudinal Model Comparison Between Cellpose & Omnipose	52
34	Merged Segmentations	53
35	Close-ups of a Full-scale Cross-section Segmentation of the Cardiac Tissue	54
36	Microscopy Image of Diseased Heart	56
B1	Varying Diameter of Cross-section Cellpose Model Version 3	64

List of Tables

1	Benchmark For Cross-section Model	35
2	Training Specifics For Cross-section Model Development in Cellpose	36
3	Segmentation Results From Cross-section Models Version 1-3	38
4	Results From Flow Threshold Adjustments Cross-section Model Version 3	40
5	Results From Cell Probability Threshold Adjustments Cross-section Model Version 3	41
6	Number of Output Masks as a Function of Threshold Parameters	43
7	Training Data For Longitudinal Model	45

Acronyms

CM	Cardiomyocyte
CNN	Convolutional Neural Network
DAPI	4'6-diamidino-2-phenulindole
DNN	Deep Neural Network
FFC	Flat-field Correction
FISH	Fluorescence In Situ Hybridization
FOV	Field of View
GUI	Graphical User Interface
HD	Hamming Distance
HF	Heart Failure
HITL	Human In The Loop
MERFISH	Multiplexed Error-Robust Fluorescence In Situ Hybridization
ROI	Region of Interest
scRNA-seq	Single-cell RNA sequencing
smFISH	Single Molecule Fluorescence In Situ Hybridization

1 Introduction

Heart failure (HF) is considered a global pandemic and affects over 26 million people worldwide [1]. The syndrome is characterized by reduced cardiac pumping capacity and/or impaired blood filling ability of the heart. Its prevalence is increasing dramatically - especially with an ageing population [2]. Despite significant advancements in both therapy and prevention, the mortality and morbidity rates associated with HF remain high. To develop treatments which directly target the underlying causes of disease, thus promoting longer, healthier lives, a comprehensive understanding of the pathophysiology of HF is required [1]. Gaining insights at a single-cell resolution is herein crucial in understanding the progression of HF as it stems from cell-specific dysregulation of genes [3]. Furthermore, the intricate interplay between multiple signaling pathways and various cell types in the context of cardiac function and dysfunction remains incompletely understood [4].

Consequently, to expand our knowledge and unravel the complexity of HF initiation and progression, it is of paramount importance to not only establish the complete repository of cardiac cells and their corresponding gene expression profiles, but to reveal how these factors are affected by diverse cellular contexts [2]. To address this knowledge gap, a number of techniques have been developed with the purpose of disclosing how genes are spatially expressed in thin sections of fixed tissue. Studies using single-cell RNA sequencing (ScRNA-seq) have managed to reveal both the characterization of individual cells, the range of cell types and states, and the molecular functions in both healthy and diseased hearts [5, 6]. Albeit, the technique requires tissue dissociation, meaning that all essential information regarding cellular micro-environments and cell interactions in space (entailing details of how cardiac homeostasis is maintained, or how disease progression is promoted) are lost [7]. Single molecule fluorescence in situ hybridization (smFISH) is one method to combat this challenge. Herein, individual RNA molecules can be mapped in intact tissue at a single-cell resolution. What is however deficient in this approach is the fact that only a limited number of RNA species can be simultaneously imaged in individual cells, thus restricting the scope of the study [8].

With the goal of overcoming the above-mentioned limitations, the Kosuri Lab at Salk Institute for Biological Studies is applying multiplexed error-robust in situ hybridization (MERFISH) to study cardiac tissue [9]. This novel, images-based, method is enabling single-cell transcriptome profiling of hundreds of different transcripts, whilst sustaining the spatial context of the tissue [10]. By implementing this technique to pressure-overloaded mouse hearts, the cell types and states in local microenvironments are identified. Herein, functional maps can be generated to study both the molecular and cellular structure of the heart, which in turn will elucidate factors giving rise to cardiac hypertrophy and fibrosis. Ultimately, the goal is to unravel their influence on HF initiation and progression, and in the long-term being able to establish new therapeutic strategies for HF patients.

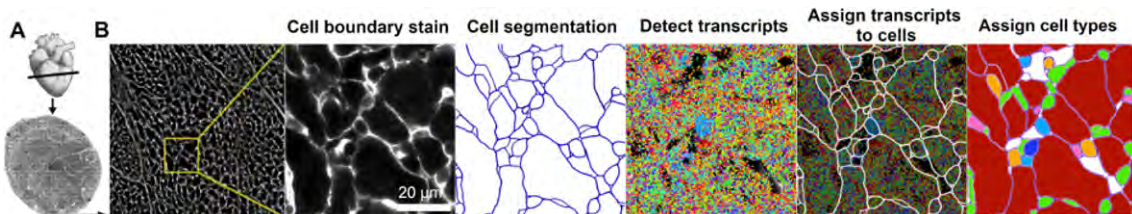


Figure 1: Delineation of the MERFISH analysis pipeline flow. Included are illustrations of cell membrane staining, cell segmentation, transcript detection and assignment, and lastly cell type characterization. Unpublished data, *Liu et al.*

Before revealing how the molecular and cellular structure of the heart give rise to function and dysfunction, there are further challenges hampering any downstream analysis of extracted MERFISH data. In regard to cardiac tissue in particular, the predominant challenge is to accurately distinguish boundaries between individual cells- a procedure known as cell segmentation. The main reason for why cardiac tissue is particularly challenging to segment, is the fact that (1) the cells are densely packed within the tissue, making it difficult to discriminate cells from one another; (2) the tissue is highly heterogeneous in terms of cell size and shape, thus aggravating an already difficult task; and lastly (3) it is troublesome to acquire an even and clear membrane staining, especially for larger cells such as cardiomyocytes. How cell segmentation is affecting the interpretation of the MERFISH data can be understood by inspecting Figure 1. This is unpublished data from the Kosuri Lab, delineating their MERFISH-analysis pipeline. Evidently, depending on how the boundaries are drawn, transcripts will be assigned to a particular cell accordingly. This will in turn determine the corresponding type and state of each segmented cell in the tissue. Consequently, an inaccurate cell segmentation will put the validity of any conclusions drawn from the MERFISH data at risk. This highlights the decisive need for an accurate cell segmentation, being the focal point of this study.

2 Theoretical Background

The following section is concerned with the theoretical underpinnings of this study. First, the cell type classification and function of the mammalian heart is enlightened. A background to spatial transcriptomics and fluorescent in situ hybridization, together with its further development, is then provided. Included herein is a more in depth description of multiplexed error robust fluorescent in situ hybridization, being the underlying spatial transcriptomics method of concern for this study. Finally, the principle of cell segmentation is examined as pertinent to this study. In this section, existing classical tools, together with more recently developed ones, are brought to attention to shed light on the current state of the field and what is yet unresolved. In particular, the supervised machine learning algorithm Cellpose, together with its successor Omnipose, are examined more carefully.

2.1 Cell Type Classification & Function of the Mammalian Heart

The heart, a muscular pump, is the first solid organ to become functional in the human embryo, being composed of four morphologically and functionally distinct chambers [2, 4]. The chambers are separated by interatrial and interventricular septa, separating the atrial chambers and contributing to the mechanical function of both ventricles (the two large chambers toward the bottom of the heart) respectively [11]. Its anatomical and functional complexity is subject to a precise architecture of heterogeneous cell populations, allowing continuous contraction and relaxation depending on different conditions in each chamber [2]. The cardiac tissue consists of a highly organized structure of a broad range of cell types, all being crucial in both physiological and pathophysiological conditions [12]. This pluricellularity administer the ability of multi-directional intercellular communication within the organ, providing successive optimization of its overall function. Importantly, this intricate communication has proved to play a key modulatory role not only during cardiac development, but in the normal and failing heart of adults [13].

There are several ways in which the cardiac cells communicate and modulate each others' functions. Either they communicate through direct cell-cell interaction, or they secrete e.g., paracrine or juxtacrine factors. The former refers to an indirect communication wherein signaling substances are released onto nearby target cells, whereas the latter specifically involves secretion of protein and receptor proteins between adjacent cells [13, 14]. Cell interactions between the same cell type, and between different types (i.e., homotypic and heterotypic respectively) are both contributing to the organized structure and overall function of the heart [12]. Some of the cell types in the mammalian heart includes cardiomyocytes (CMs), endothelial cells (ECs), fibroblasts (FBs), pericytes (PCs), vascular smooth muscle cells (SMCs), immune cells, mesothelial cells (stem cells) and neuronal cells [2]. A chart delineating the major cardiac cell types and their respective main function is presented in Figure 2. As evident, the cell types do not only differ in function but also in shape and structure.

CMs and ECs are two of the most abundant cardiac cell types, both being essential in cardiac remodeling and regeneration [16]. CMs are by far the most abundant cells in the heart in terms of volume. In short, CMs constitutes the muscular walls of the heart and are characterized by their relatively large and rod-shaped appearance. Their main function is to maintain blood circulation and is thus subject to their pump ability; CMs are the most central cardiac cell type, both in regard to normal and pathological conditions [16]. As evident from Figure 2, CMs are further divided into ventricular and atrial CMs, diverging in both the ultrastructure and gene expression patterns in respect to e.g., transcription factors, structural proteins and ion channels [16]. The ECs exhibit

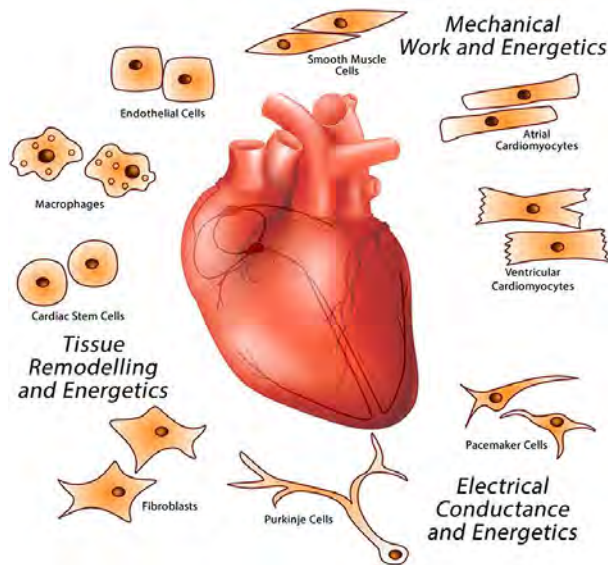


Figure 2: The major cell types and their overall function in the mammalian heart. Adapted from [15].

several fundamental functions, where the production of paracrine factors is one; paracrine factors are for instance regulating CM contractility, growth, and survival, thus crucial for maintaining normal myocardial function [13]. Among other things, ECs contribute with structural integrity in the interior surfaces of blood vessels and are also metabolically active [12]. SMCs, FBs and PCs are additional cell types crucial for maintaining normal heart function. Cardiac FBs mainly contribute to producing and secreting various extracellular matrix (ECM) components (including collagen), as well as to contribute to the cardiac remodeling during pathological conditions (e.g., in myocardial function and hypertension) [17]. Cardiac PCs are vital to preserve tissue homeostasis, i.e., the self-regulatory function, as they are crucial regulators of e.g., vascular development and remodeling [18]. Together with SMCs, PCs also regulate blood flow in the cardiac vasculature [12].

2.2 Spatial Transcriptomics & Fluorescence In Situ Hybridization

In multicellular organisms cells are organized into tissues, being the structures making up entire organs. Within the tissues, the cells proliferate, differentiate, and operate whereupon continuously communicating with its surrounding [10]. Consequently, as cells are highly influenced by its surrounding it is crucial to understand both the organization of cells and the cell-cell interactions in order to understand their origin, function and fate [10]. Compartmental distribution of RNA within a cell provides an efficient way of producing proteins in close proximity to the location of function and in response to local stimuli [19]. Notably, the cell-specific gene expression dictates cell types and states; cells belonging to the same cell type can express markedly different genes depending on their state or their local environment, such as in response to surrounding cells [20]. At tissue level, spatial organization of cells is highly associated with both the development and function of normal tissue, as well as to pathogenesis. Revealing the physical positions of different transcripts and cells would thus bridge the molecular view of the cell with e.g., morphology, electrophysiology and other cellular phenotypes. This would in turn shed further light on how these processes influence tissue organization during both development and disease [21]. Sustaining spatial context is thus essential for understanding key features of cell and developmental biology, e.g., by placing

cell-cell interactions within a broader tissue-context. Cell types and their specific organization are in fact highly associated to biological activity [22, 10]. The heterogeneous behavior of cells as well as the development, migration, polarity and maintenance of cell fate are for instance a product of intracellular organization of RNA [8]. This is yet another aspect corroborating the crucial need for a high-resolution, spatially resolved omics analysis of single-cell transcriptomic profiling [23, 8].

Single-cell RNA sequencing (ScRNA-seq) and epigenomic analyses have previously provided deep insights into both the evolution of cellular diversity and the molecular characteristics of different cell types. However, how the spatial relationships, as well as their interactions, vary across complex tissue (and species) remain herein unknown [7]. Indeed, the cell type composition and organization are to a large extent inconclusive for most biological tissue. As the number of cell types can be extensive, e.g., in the human brain and heart, there is a pronounced demand for a tool that can systematically and quantitatively reveal how genes are expressed across space in tissue [23]. Spatially resolved transcriptomics techniques, being elected as the method of the year in 2020 by Nature Methods, has been proposed as a solution [24]. These methods provide unbiased spatial profiling of tissue transcriptomes, paving the way towards obtaining extensive atlases of the cellular architecture of tissues and thus understanding morphological features. In fact, one of the main advantages is the ability of capturing mRNA in an objective manner and thus delineating both known and unknown morphological features - solely based on the molecular characterizations [10]. This has broken ground for numerous novel analyses, such as the establishment of which cellular processes are governing the morphogenesis in developing organs. In 2019 for instance, Asp et al. published a comprehensive study of the development of the human heart, whereupon spatial transcriptomics was utilized to create an organ-wide developmental transcriptional atlas. Ultimately their findings generated a publicly available web resource to feature the human developing heart, which in turn may contribute to future discoveries within the studies on human cardiogenesis [25].

Fluorescence In Situ Hybridization (FISH) is one such technique that has been developed for this purpose, first developed in the late 1960s [26]. FISH is a powerful technique to image cellular RNA transcripts, enabling in situ quantification of both the number of individual transcripts and their spatial distribution within single cells [27]. In this method cells are fixed and permeabilized, whereupon being introduced to fluorescently labeled oligonucleotide probes. The sequence of each probe is complementary to the sequence of the RNA molecule of interest [28]. Base pairing between probes and their target RNA is favoring specific bindings, which is utilized in this approach; by using multiple probes per target, the probes will become concentrated within a narrow volume of the cell for each probe-RNA assembly. This will in turn induce bright fluorescent spots that can be discriminated from e.g., auto-fluorescent spots or stray probes. By identifying each spot, the copy number of the specific RNA- together with its corresponding location within the individual cell or the tissue- can be established [28]. In contrast to single-cell transcriptomic profiling (e.g., scRNA-seq), whereby cells must dissociate from the tissue for sequencing analysis and thus lose their spatial frame of reference, FISH enables in-situ imaging.

Single-molecule FISH (smFISH), first developed in 1998, is considered to be the beginning of the hybridization based approaches of spatial transcriptomics [29, 24]. This method enables localization and quantification of RNAs (either in individual cells or in their native tissue context) with high precision; it detects nearly 100% of all mRNAs, albeit for a limited set of target genes [30]. By applying combinatorial color schemes to the profiling of single-cell gene expression, more than ten genes can be simultaneously measured [23]. smFISH is however limited by the number of RNA

species that can be simultaneously measured (and the number of distinct color channels), thus hampering the possibilities of genome-scale imaging [28]. Consequently, smFISH is not sufficient to answer many of the unresolved biological enigmas as they would require measurements of hundreds to thousands of RNAs – just within a single cell [8].

2.2.1 Multiplexed Error-Robust FISH

To combat the challenges stemming from smFISH, multiplexed error-robust FISH (MERFISH) was introduced by Moffitt et. al. in 2015 [28]. This method is described as a massively multiplexed smFISH imaging method that expands the technique towards a transcriptome-scale [28]. From correlation analyses of the copy number variations and the spatial distributions of genes one can identify groups of genes being co-regulated, as well as the groups of genes that share analogous spatial distribution patterns inside the cell [8]. In contrast to smFISH, MERFISH assigns so-called error-robust barcodes to each individual RNA species through a combinatorial FISH labeling approach. Herein, each barcode is uniquely associated with an individual RNA species. Non-fluorescent targeting probes bind to the mRNA, whereby fluorescent readout probes hybridize to the targeting probes [24]. The barcodes, distinct to each mRNA, are subsequently read through successive rounds of hybridization and smFISH imaging measurements [24, 28]. The readout probes used for the corresponding imaging round are successively washed out between each round. Herein, each gene is ascribed a N-bit binary barcode, where “1” or “0” indicates whether fluorescence is detected or not in a certain hybridization round. The assemblage of readout sequences bound to a specific RNA dictates which bits in the barcode reads “1” or “0” and accordingly the barcode identity [19]. Each readout sequence involved in the RNA transcript can in this way be determined, thus revealing the identity of each barcode [19].

A MERFISH scheme is depicted in Figure 3. In the first round of hybridization a subset of RNA species is stained with encoding probes and subsequently assigned either “1” or “0” in the first bit if fluorescence is detected or not. In the consecutive round of smFISH, a new set of probes is hybridized to the same sample and then imaged. Each RNA species is assigned either “1” or “0” anew, however now to the second bit. The procedure is repeated for N rounds of smFISH imaging, generating an N-bit barcode for each labeled RNA species [28]. The sequential rounds of hybridization, together with an error correction procedure, enable the capturing of multifold genes. 2^N genes can be distinguished using N rounds of hybridization, and correspondingly 2^{CN} genes when using C colors for each N rounds respectively [23]. For comparison, only 16 imaging rounds would generate over 65,000 (2^{16}) unique barcodes, which is enough to identify the majority of the expressed RNA species within single human cells [8]. MERFISH specifically have demonstrated an efficiency of 80% for more than 10,000-gene imaging, allowing detection and quantification of genes even with low expression levels [19].

Misidentification of barcodes, and thus misidentification of genes, is however a challenge with this approach, being inherent to smFISH. The source of these errors is twofold: either an RNA that is supposed to fluoresce in one round does not accumulate the probes sufficiently to generate a bright enough signal to be identified as an RNA; or there are stray probes or autofluorescent spots in the cell that is falsely identified as RNA [28]. By identifying a molecule with multiple signals, i.e., in this combinatorial barcoding approach, the finite per-signal (per-bit) error will accumulate with the length of the barcode which prompts a significant overall misidentification rate. Consequently, this type of error is growing exponentially with N, as compared to smFISH which only proceeds through one single measurement [24, 23]. To address this issue, MERFISH utilizes a coding scheme

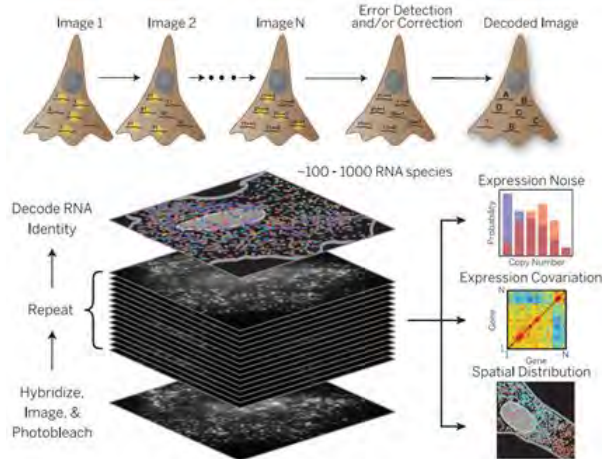


Figure 3: Schematic illustration of the process of reading out binary barcodes associated with each labeled RNA in a sample using MERFISH, adopted from [8].

in which error detection and correction is allowed, being yet another key concept of this method. The idea behind the error detection and/or correction is to utilize the so-called Hamming Distance (HD) approach. HD corresponds to the number of bits that must be interchanged in order to transform one barcode into another [28]. This implies that if e.g., HD is selected to 2 (i.e., HD2), then no single-bit readout error can convert one barcode into another, which accordingly reduces the misidentification rate. Hence, by further increasing the HD, the reduction in misidentification will improve [28]. An additional key concept of this error-robust encoding scheme is to leave a number of possible binary barcodes unassigned, rather than assigning each of them to a RNA species. Hence, if one of these unassigned barcodes would be identified, it implies that an error has occurred [28].

2.3 Cell Segmentation

The transcriptional data extracted from e.g., MERFISH unfold the coordinates of each detected molecule (i.e., genes or transcripts) within its field of view (FOV). Despite the method's ability to reveal intricate tissue organization, it is still challenging to tell individual cells apart when dealing with such data. This in turn is restricting the subsequent analysis of extracted data, such as the analysis of context-dependent cell expression states, interactions, spatial dependencies between cell types and formation of tissue architecture [21]. The effective spatial resolution is thus subject to discriminating features in the successive analysis; to accomplish cellular resolution, a precise cell segmentation is a prerequisite [21]. Cell segmentation serves as the primary challenge and bottleneck in image cytometry (the physical and chemical characterizations of cells) [31, 32].

The principle behind cell segmentation is to localize the cell object boundaries within an image, and to distinguish these from the background. This is referred to as a pixel-wise binary classification, whereupon a mask with the specific pixels belonging to the region of interest (ROI) is deduced [33]. Current studies utilize auxiliary nuclei, cytoplasmic, and/or membrane markers to discern specific cell features and thus facilitate the cell segmentation [34]. One nuclear staining method involves DAPI (4'-diamidino-2-phenylindole), a highly fluorescent stain which binds to adenine-thymine rich regions in nuclear DNA and fluoresces. This fluorescent tag has proven to be a convenient and efficient method of imaging cell centers, especially in confocal microscopy (using laser and

fluorescence to create 3D images of samples) [35]. A challenge, however, arises from the notion that nuclei placements do not indicate the actual shape and extent of the cell body [21]. One technique used to further extend the initial nuclei position is poly(A) staining, which is a cytoplasmic staining approach. This has been proven to be especially successful in tissues where cells are not densely packed [36]. Although DAPI and poly(A) staining both furnish feature-rich stains for segmentation in cell-sparse tissue, such as the brain, they are not sufficient in cell-dense tissue - as relevant to this study. For these tissues, membrane staining by e.g., immunofluorescence (IF) is a more successful approach [36]. Herein, fluorescently-labeled antibodies are used to specifically target and bind to a certain molecule in the membrane. After binding a primary antibody to the target molecule in the membrane, a secondary antibody- conjugated to a fluorescent dye- is added to the sample. A fluorescent signal is visualized upon binding the secondary antibody to the primary one, which can be imaged with a fluorescent microscope [37].

2.3.1 Classical Segmentation Algorithms

Despite the great advancements of both spatial transcriptomics, optical imaging and image processing, inadequate cell segmentation is restraining ensuing discoveries. Today, the state-of-the-art methods and software tools mainly rely on existing, basic, image processing algorithms formulated decades ago, most of them proposed back in the 1970s [32]. Among others, these include SMASH [38], ImageJ [39], Cellsegm [40] and CellProfiler [41]. These are all primitive and fully automated methods, developed to solve more general computer vision tasks. Generally, these are mainly successful when cells are sufficiently separated, for instance in a monodisperse in vitro cultures, and not in a tightly packed tissue [42]. Classical algorithms are thus hampered by their insufficient capabilities of solving more challenging and extensive cell segmentation problems at large [32]. For most classical approaches, the segmentation is requiring manual tuning and corrections, making the models incomplete [34]. Moreover, these classical methods are all based on the so-called Watershed algorithm, being an image processing transformation defined on a gray-scale image [43]. Herein, the gray-scale values of the image is used to create topological maps, where the brightness of each pixel is representing the "height" of that point. In this way, the algorithm defines a line that runs along the top of the formed ridges [44]. The principle is delineated in Figure 4, wherein two black blobs with fading edges are converted into its corresponding topological map. Here, the bright areas are considered "high" and the dark "low", creating two catchment basins surrounded by watershed lines [44]. In the Watershed algorithm, the basins represents the segmented regions of interest. As can be deduced from the concepts of this algorithm, the method is mainly successful when the object to be segmented has a smooth intensity decay from its center [42]. In this way the algorithm can form a clear, single basin. This is however rarely the case when performing cell segmentation; several cell types often form multiple intensity basins - due to e.g., inhomogeneous distribution of fluorescent markers along the cell border [42]. Moreover, a frequently occurring issue with this approach is over-segmentation. This issue is especially prone to occur in the case of high image complexity or poor image quality [32, 42]. The ever-increasing complexity, the considerable variety of cell types, low image contrast, deficient image quality and contrast, non-uniform pixel intensity and not to mention the connection or overlapping of neighboring cells are yet some of the challenges where these methods are insufficient [32].

One further aspect of cell segmentation is the available of thresholding algorithms. Thresholding refers to the process of discriminate ROIs from the background in an image [45]. Based on a predetermined value or range of values for pixel intensity, all pixels above or below this threshold are classified as foreground (i.e., a part of the cell) or background, respectively. This process is generating

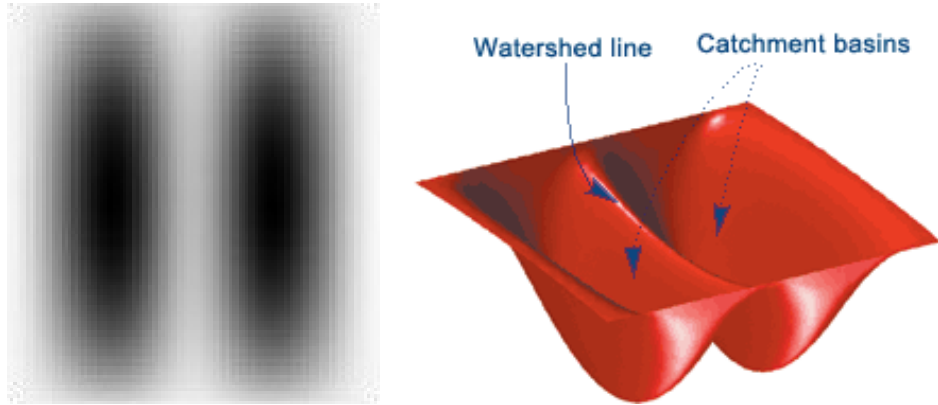


Figure 4: Schematic representation of a watershed transform. The pixel intensity decay of two dark blobs are translated into catchment basins and watershed lines depending on the corresponding "height" of each pixel. Adopted from [44].

the pixel-wise binary classification, as touched upon previously. As evident, thresholding is a crucial step in cell segmentation algorithms to allow for an accurate identification and quantification of cells in an image. Albeit, to select a proper threshold value is challenging and strongly depends on e.g., image quality, cell types, intensity of staining and the type of labeling used [45]. In the above-mentioned segmentation techniques, classical thresholding algorithms are used, wherein the threshold is found between two pixel classes with a rather large intensity interval [32]. This approach is in most cases not sufficient, in particular when (1) the intensity interval between background and darker cells is smaller than the one between darker and brighter cells; or (2) the image histogram (i.e., the graphical representation of the distribution of pixel intensity, with a frequency of occurrence of each pixel varying from e.g., 0 to 255 for an 8-bit gray-scale image [46]) is not properly equalized; or (3) there is a high degree of overlapping between different pixel classes in the histogram distribution [32]. Cell images do not only exhibit a lot of noise, but can often display great variations in intensities both between and within cells. This implies great distortions in the histograms, why the classical thresholding algorithms are not sufficient for more complex cell segmentation purposes [32]. Consequently, to facilitate an accurate cell segmentation, the thresholding algorithm must be more robust and flexible to adapt to varying types of cell images.

2.3.2 Machine Learning Algorithms

In an effort to overcome the aforementioned challenges of adopting classical cell segmentation algorithms to more complex cell images, new segmentation algorithms have emerged in the recent years. Deep neural networks (DNN) have been particularly successful for this purpose, utilizing parameters estimated from large training datasets [42]. In spite the fact that fully automated methods are advantageous in terms of both reducing the human effort, increasing the reproducibility and allowing for improved scalability to larger datasets, they are deficient in other aspects [42]. As the available image acquisitions become more diverse and complex, the challenge of automating the segmentation approaches is even more profound. Consequently, one must make a trade-off between flexibility and automation in the segmentation [42]. The main issue with employing fully automated algorithms is the fact that the segmentation style cannot be customized; the models often perform poorly upon being introduced to images markedly different from their training images [47].

Cellpose

The above-mentioned challenge is being addressed by Pachitariu’s and Stringer’s in 2021, introducing Cellpose. Cellpose is a generalist deep learning-based segmentation algorithm claimed to have better expressive power than alternative solutions [42]. The great expressive power and high capacity of Cellpose has been proven mainly from its ability to segment not only more complex cells such as elongated dendrites, but even non-cell objects such as jellyfish and rocks [42]. The difficulty with cells having multiple intensity basins is herein tackled by constructing an intermediate representation of the cells that forms just a single, smooth, topological basin. The main feature of this approach is to generate topological maps from a method called simulated diffusion, extracted from ground-truth masks. A ground-truth mask is, as compared to a predicted mask (which is the output from a deep learning model) a manual segmentation, in this case drawn by a human annotator [42]. Furthermore, a DNN is trained to predict three outputs: the horizontal and vertical gradients of these topological maps (forming vector fields), along with a so-called binary map to indicate if a given pixel is inside a ROI or not.

In Cellpose, the DNN that is predicting the spatial gradients and the binary map is based on a (slightly modified version of the) so-called U-Net architecture. This approach is proposed as an alternative to the typical use of convolutional neural networks (CNN), and is an extension to the so-called *fully convolutional neural network* [48]. Simply put, a CNN is a class of DNN, most commonly used for image recognition. An in-depth review of the specific construction of Cellpose’s DNN is beyond the scope of this report. However, at a high level, a CNN is used to train machines in image recognition and other computer vision tasks. Herein, the CNN takes an input image and process it through a sequence of so-called convolutional and pooling layers [49]. The convolutional layers extract features from the input image by successively converting all pixels in a specific region into a single value, hence decreasing the image size and simultaneously concentrating all information into a single pixel [50]. This is performed by screening the image with one or more filters (also known as a kernels), and for each pixel compute the dot product between the weight of the kernel and that specific pixel value. The weight in this case refers to how the specific kernel will respond to different patterns and features of the input image; it can be compared to a template or feature detector and is a decisive step during training of the network. The output from each kernel-weight dot product is a ”feature map” that contains a specific pattern or a feature that was found in the input image [49]. Subsequently, the pooling layers will downsample these feature maps, meaning that the dimensionality is reduced whilst sustaining the most integral information [51]. The resulting feature maps from several repetitions of the above-mentioned layers are then stored as a vector. This is subsequently fed into one or more connected layers, from which a final output classification is produced based on all extracted features [49]. The weights of each kernel and fully connected layers are changed when training a CNN, enabling the network to learn to recognize the visual patterns and characteristics of the input images [49].

A representation of the so-called U-net architecture is presented in Figure 5. Herein, in addition to the pooling layers that downsample the feature maps, upsampling operators are supplemented to increase the resolution of the output [52]. In contrast to downsampling, upsampling is a technique used to increase the spatial resolution of feature maps. This is essential in image segmentation since the output must have the same spatial resolution as the input [52]. High resolution features from the contracting path are accordingly integrated with the upsampled output, which enables localization (assigning a class label to each pixel). Based on this information, a consecutive convolutional

layer will be able to learn how to generate a more precise output. The U-shaped network thus emerges from the mirror-symmetric relationship between the contracting and expansive path of the architecture, whereupon both paths involve a large number of feature channels [42]. Moreover, as different image types are deemed to be processed in different ways, a global average pooling on the smallest convolutional maps is utilized in Cellpose to generate a classification of the image [42]. Together, the advantage with this U-net CNN approach is three-folded: the need for thousands of training images is greatly reduced, localization is enabled by assigning each pixel to a specific class and lastly, the segmentation is more precise [52].

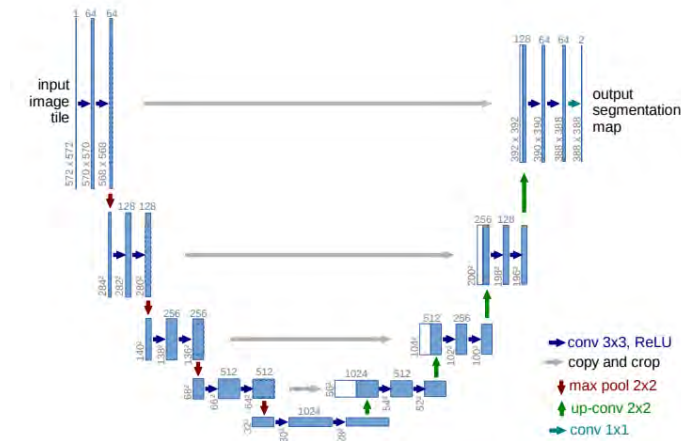


Figure 5: Example of a U-net architecture. The blue boxes represents the different feature maps between each layer of the network, whereas the white correspond to copied feature maps. The specific operations are denoted with differently coloured arrows, explained in the lower right corner. In this example, downsampling is performed to the lowest resolution of 32x32 pixels. Adopted from [52].

Upon introducing a test image to the CNN in Cellpose, it will accordingly predict horizontal and vertical gradients to generate vector fields. These fields are then followed through a procedure called gradient tracking, wherein each pixel belonging to a specific cell will be routed to its center [53]. Gradient flow tracking constitutes of three main steps, delineated in Figure 6. These include (1) generating a so-called diffused gradient flow vector field to overcome the challenges of generating trustworthy vectors in the case of having irregular cell shapes and intensities (to smooth out the otherwise noisy gradient vector field); (2) performing a gradient flow tracking procedure of this field to bring together points to the basin of a sink (i.e., the center of a specific cell); and lastly (3) dividing the FOV into smaller regions, each consisting of one cell/ nucleus and its adjacent background, and from there carry out local, flexible, thresholding for each region to withdraw the cell/nucleus from its background [53].

The gradient flow tracking will segment all points that are flowing to the same sink as a region called an *attraction basin of the sink*, being illustrated in Figure 7. Upon grouping together pixels which converge to the same sink, cells and their exact shape will emerge accordingly. Since all points are tracked independently, the attraction basin is not restricted to a specific shape, being an advantageous feature for more irregular cells [53]. Additional refinements are also performed on the model in Cellpose, wherein pixels predicted as outside of cells are removed [42]. Since the shape of the cell can be irregular, all gradients do however not necessarily point directly to the center, as

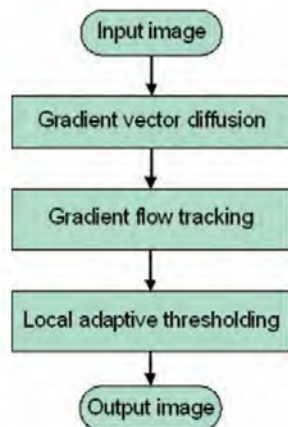


Figure 6: Flowchart of the main steps and their consecutive order in gradient flow tracking. Adopted from [53].

they might intersect with cell boundaries [42]. Albeit, locally the gradients will translate pixels into other pixels inside the cell; after a series of iterations, pixels will globally translate to fixed points of the gradient vector field. Finally, these fixed points are then assigned to be the cell centers.

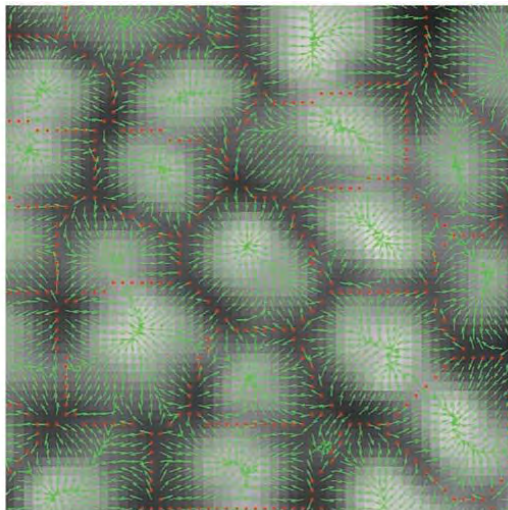


Figure 7: A representation of sink attraction basins generated from gradient flow tracking, where segmentation emerge from grouping together the pixels flowing towards the same sink. Gradients are generated from the derivatives of the image brightness. Adopted from [53].

One of the main features of Cellpose is the use of so-called an auxiliary vector flow representation. Auxiliary representation is widely used for machine learning models to help reducing the amount of data needed to train the model - while at the same time improving the overall performance [54]. The technique involves incorporating additional input data or features (which does not have to be directly related to the main prediction task) to the model in order to support the model to make better predictions. For the case of Cellpose, the auxiliary representation converts the cell masks into two images, also known as "maps", which are vector fields of the same size as the original image

[42]. These vector fields lead all pixels belonging to a certain cell toward its center. Ultimately, the neural network is trained to predict these gradients from raw images. These vector flow fields are generated from a process known as *heat diffusion simulation*. The idea behind this process is to introduce a "heat source" at the center pixel of the cell (being the pixel at the median values of horizontal and vertical pixels within the cell [42]). For several iterations, a pixel value of 1 is added to the heat source, whereupon all pixels within the cell are assigned a mean value of its nearby surrounding pixels. All pixels outside of the cell are assigned a pixel value of 0 in each iteration. After N iterations (proportional to the vertical and horizontal pixel range within each cell), the heat distribution is approaching equilibrium. This final distribution is generating an energy function, ultimately representing the two (horizontal and vertical) vector fields of the auxiliary vector flow representation [42].

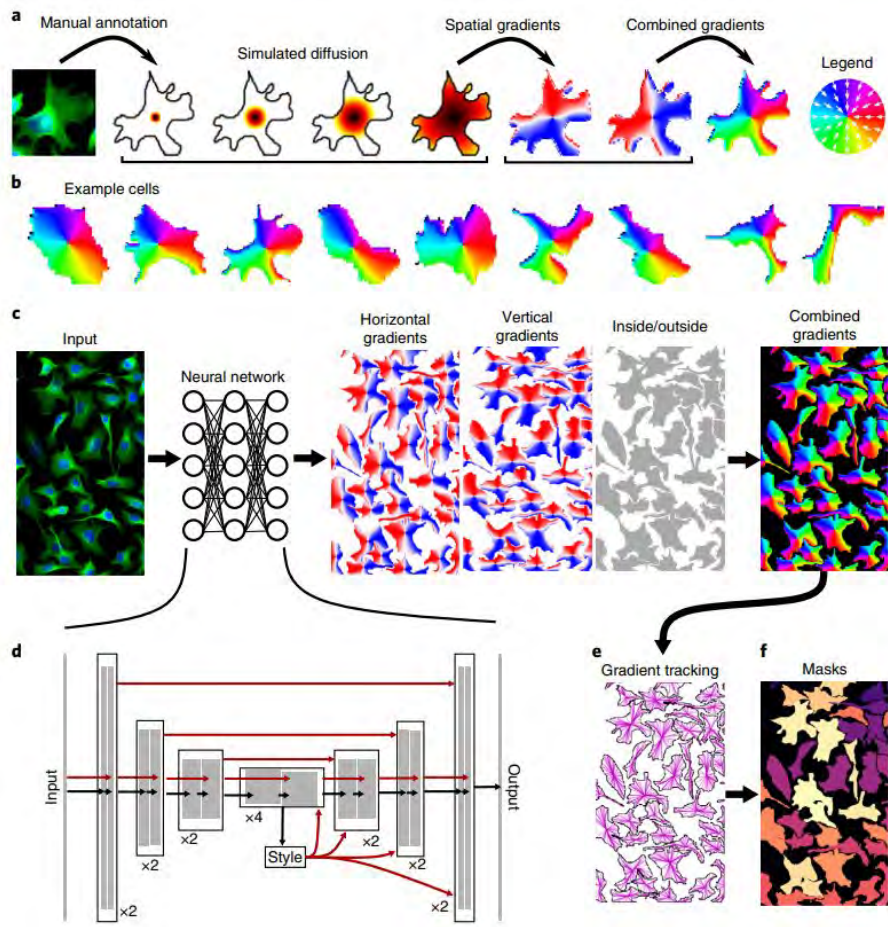


Figure 8: Schematics of the Cellpose model architecture. **a.** Manually annotated masks are transformed into vector flow representations by performing simulated diffusion. The generated spatial gradients (x and y) from the diffusion are transformed into combined gradients in just a single direction from 0° to 360° . **b.** Spatial gradients of example cells. **c.** Training of the neural network to predict three outputs: horizontal and vertical gradients as well as pixel probability. These outputs are combined into a gradient vector field. **d.** U-Net architecture of Cellpose. **e.** Predicted gradient vector fields are used in a dynamical system where every pixel will track these gradients toward their respective fixed points, being the basin of attraction from the gradient flow tracking. **f.** Pixels converging to the same basin are prescribed to the same mask. Adopted from [42].

The workflow of the Cellpose algorithm is delineated in Figure 8. As previously mentioned, the output from the neural network is three maps: horizontal gradients, vertical gradients and a binary pixel map - also known as the pixel probabilities [42]. In order to recover the masks from these outputs, a threshold of 0.5 for the pixel probability map is first set. Each pixel above this threshold is then considered in the next step, wherein a dynamical system is run for 200 iteration. This dynamical system starts from each of these pixel locations, and runs along the spatial derivatives of the horizontal and vertical gradient maps. For every iteration, a step is taken in the direction of the gradient. Pixels will in this way be clustered in accordance to the corresponding pixel they converge towards [42]. As the CNN can predict flow vectors that do not correspond to any real shapes, there is an element of uncertainty that can yield inconsistent flow outputs. In addition to the pixel probability threshold, Cellpose therefore involves a mask quality threshold. This threshold verifies that the obtained ROIs from the dynamics step are consistent with the real masks. This is implemented by recomputing the flow gradients based on the predicted labels from the dynamics and comparing them to the masks predicted by the network. This control is carried out pixel-by-pixel. Consequently, this threshold reflects the maximum allowed error of the flows, but averaged over the entire set of pixels within a specific mask [42]. The masks with a difference larger than the cell probability threshold are excluded from the final output [42].

Cellpose 2.0

As an extension to Cellpose, the new edition Cellpose 2.0 was introduced in 2022 [47]. This upgraded version also involves a new package which includes an ensemble of different pretrained models (called a model zoo) together with a human-in-the-loop (HITL) approach. The HITL approach involves human interaction, starting with a restricted amount of user-segmented data to train an initial, imperfect model. Upon being introduced to other images, the result from the model is subsequently corrected by the user [47]. During the development process, the model is thus continuously inspected, validated, and calibrated to allow for rapid prototyping of new custom models [55]. In this way, Cellpose 2.0 is considered an interactive, supervised, machine learning method. In Cellpose 2.0, they utilized two large-scale datasets on Cellpose, whereupon taking advantage of these to develop a model zoo that the user can select between as starting points, and then evaluate based on their specific data. These were subsequently used in the HITL annotation and in further retraining procedures [47]. Consequently, this has reduced the required user annotation in Cellpose 2.0, without compromising on high quality segmentation [47]. The algorithmic and software tools involved in Cellpose 2.0 manage to adapt CNN segmentation models to novel image categories, utilizing minimal training data, thus reducing the time consuming task of manual segmentation. Altogether, the main advances with this approach is threefold. Firstly, it is crucial since the style of annotations may differ substantially among annotators; there are several different styles of human annotator styles, all equally valid. Secondly, it is efficient as it thanks to HITL only requires fewer segmentation of ROIs to meet the same accuracy of alternative methods. Lastly, the approach is effective; models developed in this manner have comparable accuracy with human researchers in the field [47].

Omnipose

As an extension to Cellpose, the DNN image-segmentation algorithm Omnipose was developed - featuring one of the authors of Cellpose, Carsen Stringer [56]. The primary motive for developing this algorithm was the need for a tool that could segment cell types for which Cellpose and other state-of-the-art cell algorithms currently are deficient. These cell types mainly involve bacterial cultures and cells of elongated or branched morphology - which are highly different from the more uniform cell shapes that other segmentation algorithms have mostly been trained on. The algorithm is based on the same framework as Cellpose, including file handling, DNN architecture, training functions, network predictions and mask reconstruction [56]. This section will thus not go into details of how the Omnipose algorithm is constructed. The central differences, however, lies within the network outputs. These are a cell boundary probability map, a distance field and a flow field - being the gradient of the distance field. These discrepancies were implemented to circumvent issues Cellpose encounters when segmenting bacteria and other miscellaneous morphologies, namely over and under-segmentation. Notably, upon comparing Cellpose and other state-of-the-art algorithms (incl. StarDist, MAsk-R-CNN and MiSiC) against Omnipose's benchmarking data, Cellpose was the only algorithm that exhibited errors just for segmenting elongated cells [56]. Not only does this prove that Cellpose is superior to other alternatives, but it also demonstrates the need for an algorithm as Omnipose to resolve the challenge of segmenting bacteria and other anisotropic cells.

3 Aim

In the Kosuri Lab at Salk Institute for Biological Studies, MERFISH is deployed to visualize simultaneously the location of hundreds of different transcripts in sections of mouse hearts, with the goal of revealing novel underlying principles of cellular organization in healthy and diseased tissue. To accomplish this goal, it is essential to (1) identify cells and (2) quantify their spatial organization based on the MERFISH data. As previously stated, these tasks are major challenges in the field of spatial transcriptomics for which there currently are no adequate solutions. To resolve these challenges, one needs to (1) adapt and optimize computational methods to achieve an accurate cell segmentation of the tissue; and correspondingly (2) develop and apply statistical methods to analyze the spatial distributions of different subsets of cells.

3.1 Limitations & Delimitations

Upon exploring potential solutions to task (1), it became clear that this was more challenging than anticipated; there are no existing segmentation models that sufficiently can segment the cardiac tissue. As previously elaborated on, the main reason for this is the dense and heterogeneous nature of the cardiac tissue. As task (2) highly depends on an accurate cell segmentation, this bottleneck must first be resolved before proceeding with any downstream analysis. Consequently, to narrow down the project to a feasible extent, it is exclusively focusing on solving the first challenge. The aim of this study is thus *to develop and refine a cell segmentation algorithm that accurately determines the cell boundaries in dense, heterogeneous mouse heart tissue.*

Ultimately, the ambition is to successfully acquiring an accurate segmentation across a full-scale MERFISH dataset of the tissue. To achieve this, the project will be concerned with understanding and utilizing supervised machine learning algorithms including Cellpose & Omnipose, to accordingly generate and optimize customized model(s) applicable to diverse datasets of the mouse heart. Due to the great heterogeneity of cell types and shapes within the tissue, a major part of the project is to find a strategy to cover segmentation of all cell types and features. In summary, this project intends to support the development of tools capable of addressing one of the challenges associated with deciphering MERFISH data of the cardiac tissue.

3.2 Outline of Thesis

Section 1 introduces the background, purpose and significance of the thesis. This is followed by the theoretical framework relevant to the study in section 2, providing insights into the constituents of the heart, spatial transcriptomics, and cell segmentation. Section 3 delineates the aim of the project, whereas the experimental procedure is setted in section 4. Section 5 reports the result of the study, which is followed by a discussion where these results are elaborated and expounded upon. Possible shortcomings and improvements are suggested here as well. Lastly, conclusions and some final remarks are declared in section 6.

4 Methodology

The following section involves the methodological procedure of achieving an accurate segmentation model for the cardiac tissue. Included herein is a review of the prerequisites and requirements on the segmentation, along with a manifestation of the overall strategy to meet these. The chosen machine learning algorithms for addressing this challenge is presented as well. Additionally, a detailed description of the practical and theoretical considerations for developing the model(s) are introduced in this section. These encompasses the procedure of generating and training a model, along with the training and segmentation parameters that will be considered when optimizing each model.

4.1 Cell Segmentation - Prerequisites, Requirements & Strategy

The morphology of the cardiac tissue is highly diverse, having CMs differently oriented depending on where in the heart they are located. As the CMs are large (relative to remaining cardiac cells) and rod-like in shape, they will be featured in various forms when looking at the tissue in just one plane. This implies that depending on where the thin tissue sections are cut, and in what angle each CM are cut, the morphology will be different; some areas will contain large, round, cross-sections of the CMs, whereas others will display longitudinal CMs. Examples of epifluorescence microscopy images taken at different parts of the tissue (in the same z-plane) are presented in Figure 9, delineating the great heterogeneity in the morphology. Notably, there is a substantial difference in appearance of the CMs depending on how they are oriented in the tissue. What is also evident from Figure 9 is that the cells are densely packed in the tissue - being yet another noteworthy challenge as also mentioned in previous sections. Evidently, the great diversity in morphology makes the cell segmentation particularly challenging, especially for the longitudinal sections of the CMs.

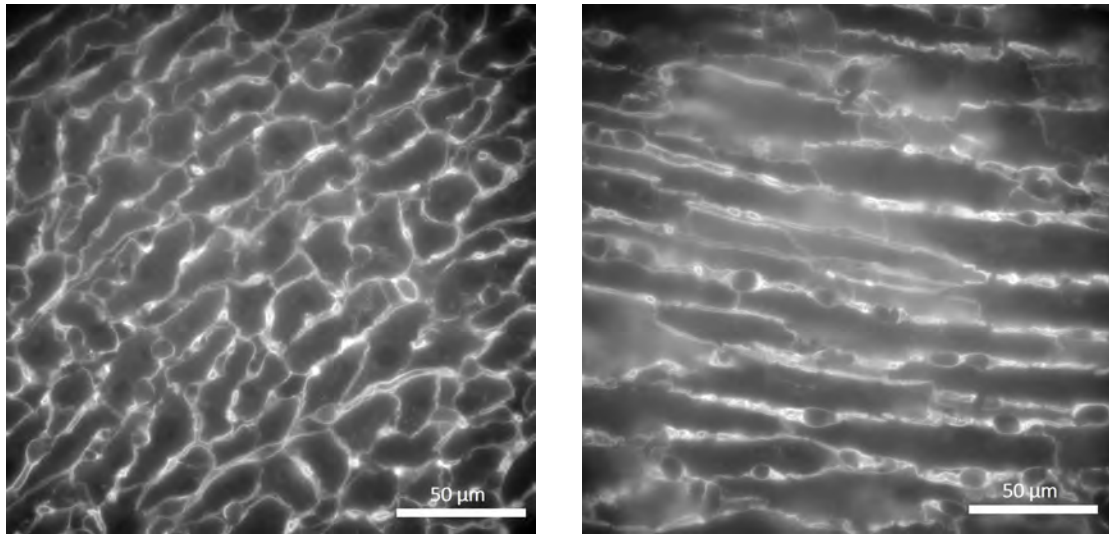


Figure 9: Epifluorescence microscopy images of the cardiac tissue using cell boundary staining, showcasing cross-section cardiomyocytes (left) and longitudinal cardiomyocytes (right) respectively. Each section of the heart tissue is sliced into $10\ \mu\text{m}$ sections.

As previously stated, an accurate cell segmentation is a prerequisite for drawing valid conclusions about the extracted MERFISH data. To benchmark different cell segmentation models, as well as following the improvement of these, one must set the standard for what is considered an *accurate*

and *inaccurate* segmentation respectively. Inaccurate segmentations may appear in various forms, and will in this report be categorized as the following;

1. Cell-splitting
2. Cell-merging
3. Oversegmentation
4. Incomplete segmentation
5. Absent segmentation

Examples of these inaccurate segmentations are pictured in Figure 10. Herein, cell-splitting refers to when one cell is segmented as several. In contrast, cell-merging implies that several cells are combined into one. As opposed to cell-merging, oversegmentation refers to when the boundaries around a cell also capture the interstitial space between the ROI and its surrounding cells. Incomplete segmentation is simply when the segmentation does not capture the entire cell body. Lastly, an absent segmentation refers to when a cell is not segmented in any way, and instead is deemed as a part of the background.

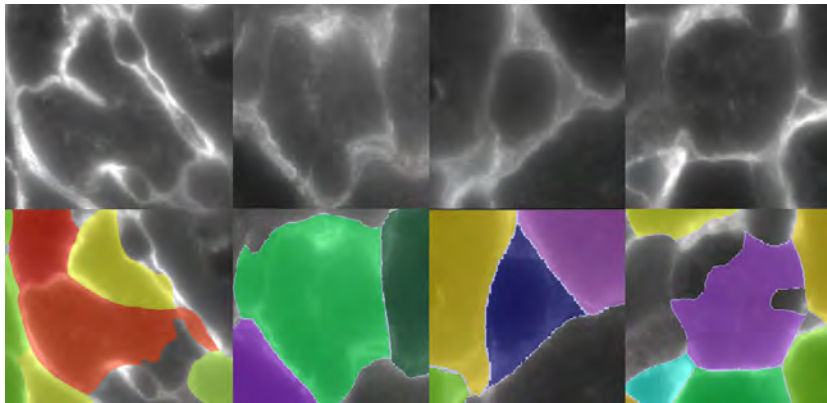


Figure 10: Examples of inaccurate cell segmentation from the respective category, extracted from a default model in Cellpose. Each color represent uniquely segmented cells. From left to right: cell-splitting; cell-merging; oversegmentation; incomplete segmentation.

The consequences of these five categories varies in severity. For instance, splitting cells would mean that some transcripts belonging to a certain cell will be assigned to another ROI. Consequently, this would imply that the cell type (and state) most probably would not be correctly determined, since certain marker-genes may instead be assigned to another cell. Cell-merging would naturally have similar consequences. These two inaccuracies are thus considered a more severe than e.g., incomplete segmentation, since no transcripts will falsely be assigned to another cell; as long as the marker genes are included within the ROI, the cell type and state will still be correctly assigned. Likewise, an oversegmentation is not as severe either, since the interstitial space ultimately should not contain any transcripts. To completely disregard cells in the so-called *absent segmentation* could in fact be detrimental, especially if the cells being neglected contain crucial information for understanding a certain micro-environment within the tissue. As highlighted in section 2.1, each cell types have their own respective function, all being significant in maintaining cardiac function. If a certain cell type rarely is segmented, crucial information thus risks being disregarded, compromising

further analysis. Consequently, as achieving a 100% segmentation accuracy is an elusive task, one must accordingly evaluate which inaccuracies are more alarming than others. For this project, cell-splitting and cell-merging will be the top-priorities in terms of inaccurate segmentations to minimize.

With the heterogeneity and variety of inaccurate segmentations in mind, the strategy for this project is to create two separate models; one for the smaller cells and cross-sectional CMs, and one for the longitudinal CMs. This approach will accordingly facilitate a more fine-tuning of the models, to each meet the requirements of the respective cell shape and morphology. With this approach, the idea is for the models to jointly achieve an accurate segmentation of a complete dataset - covering the entire morphology of the cardiac tissue. The workflow for this project can accordingly be depicted as per below:

1. Identify and adapt a segmentation algorithm suitable for the specific morphology of interest.
2. Benchmark default models.
3. Collecting training data in terms of cardiac tissue images.
4. Generate ground truth masks for these images.
5. Develop and train a model.
6. Optimize parameters, including thresholds, cell diameter and other segmentation parameters for each model.

4.1.1 Quantifying Model Progression

For benchmarking default models and recording the model progression for the cross-section models, the segmentations will be quantified in terms of *segmentation accuracy*. In this report, this is defined as the number of cells being *accurately* segmented by the model, divided by the number of cells that are "segmentable" within that FOV - i.e., the number of ground truth ROIs (see Eq.1):

$$A = \frac{\# \text{ Accurate output ROI}}{\# \text{ Ground truth ROI}} \quad (1)$$

The fraction of inaccurate masks will also be quantified for the cross-section model. As neither of the longitudinal models achieved a quantifiable accuracy, the assessment will herein solely be based on qualitative evaluations.

4.2 Adapted Segmentation Algorithms

Considering what is previously stated about available segmentation tools, Cellpose and its successor Omnipose are the algorithms prospected to be suitable for this project. Notably, the authors to Cellpose and Omnipose have both benchmarked their algorithm against similar alternative solutions (including Stardist, Mask R-CNN and U-Nets [57, 58, 52])- with the conclusion that they outperform their competitors ([42, 56]).

4.2.1 Cellpose 2.0

Visualization, manual segmentation and model training are all enabled directly in the graphical user interface (GUI) of Cellpose, being delineated in Figure 11. For visualizing the segmentation, the user can switch between cell outlines and filled masks. Additionally, the image brightness and contrast can be adjusted for visualization purposes. Parameters for adjusting the segmentation output can be altered in the GUI as well. Included herein are the flow and cell probability threshold respectively.

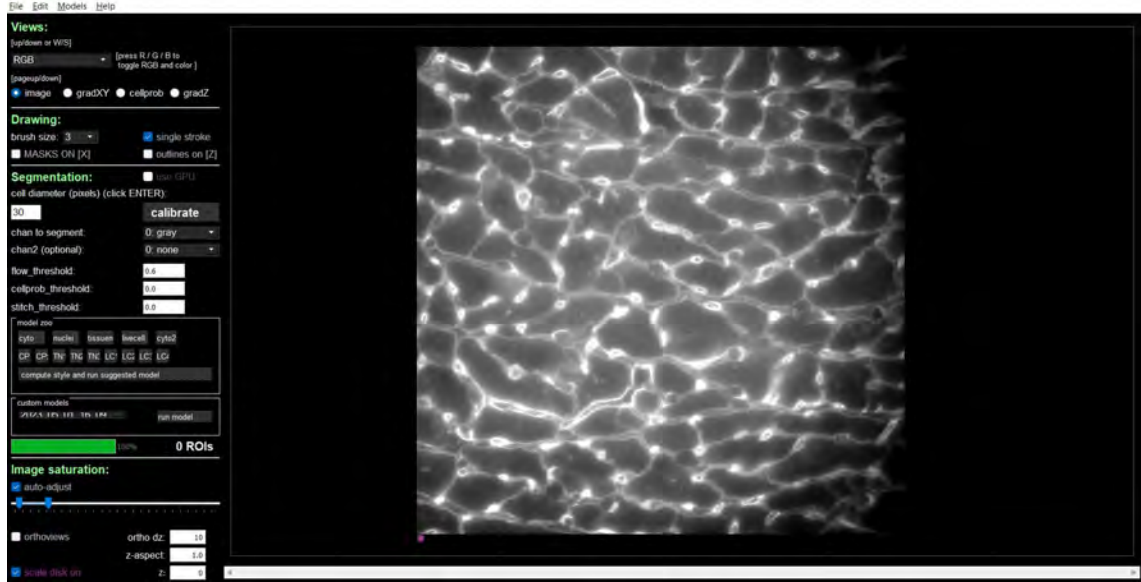


Figure 11: The graphical user interface (GUI) of Cellpose. Visualization, manual segmentation and training can be performed in the GUI. Parameters such as diameter, thresholds and image brightness & contrast can be selected in the toolbar to the left. For visualization purposes, the user can view the raw image solely, or overlaid with outlines and/or masks.

As evident from Figure 11, the model zoo is also presented in the GUI; the user may segment, or train a new model, with a model from the zoo, or alternatively load their own model into the GUI. An additional feature in Cellpose is that the network can predict which model from the zoo that is most appropriate for a specific image type. This will be utilized when benchmarking the performance of Cellpose.

Cell Diameter

As for diameter setting, it can either be chosen manually or alternatively one can let the model calibrate it prior to segmentation. In Cellpose, the diameter is defined as the diameter of a circle of equivalent area as the ROIs, as presented in Eq.2. The corresponding unit is specified in pixels, and the default diameter is set to 30 pixels.

$$d = 2R = 2\sqrt{\frac{A}{\pi}} \quad (2)$$

This definition is mainly suitable for the case of isotropic cell shapes, e.g., round cells such as endothelial cells, macrophages and fibroblasts (see Figure 2 for reference). Cross-sectioned CMs

would accordingly be appropriate here, as the majority of cells have a round-like shape. Setting an accurate cell diameter is crucial for an accurate model training and segmentation output, why this will be carefully elaborated upon as part of the model development. Importantly, when the diameter is set smaller than the actual size there is a risk that the model will over-split cells. Correspondingly, setting the diameter larger than true size will increase the risk for cell-merging [42].

Thresholds

As introduced in the section 2.3.1, Cellpose utilizes two threshold parameters for 2D segmentation; flow and mask (also known as the cell probability) thresholds. Both of these will be examined for optimizing the Cellpose model. The former refers to the maximum error of the flows for each mask; it works as a consistency control to make sure the that the mask output coincide with the corresponding masks found when running the dynamics. The flow threshold can be selected from zero and above (with a default of 0.4). Intuitively, increasing the threshold (i.e., increasing the tolerance for errors) will thus increase the number of ROIs. Likewise, decreasing it will return fewer ROIs - generally removing the more ill-shaped ones [42]. The cell probability threshold is the probability of a pixel belonging to an ROI. Here, the threshold ranges from approximately -6 to 6, given that the probability is expressed as the sigmoid function (centered at zero, also being the default threshold) presented in Eq.3. The corresponding graph is presented in Figure 12.

$$P(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

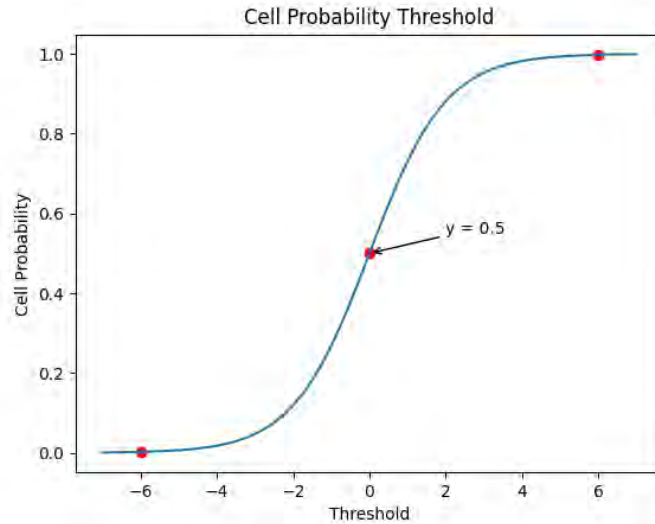


Figure 12: Cell probability as a function of threshold. Highlighted in the graph are the threshold boundaries ($x \in [-6, 6]$) and the default settings at $(0, 0.5)$.

Increasing the cell probability will, in contrast to the flow threshold, decrease the number of ROIs in the segmentation output. Similarly, decreasing it will return less ROIs, since the tolerance of probability is lower [42].

4.2.2 Omnipose

As the majority of the Omnipose algorithm builds upon Cellpose, most of above-mentioned features, such as thresholds and GUI procedures, are transferable to this section. The discrepancies will however be brought to attention in the following section. The training procedure is equivalent to Cellpose, however the GUI is slightly different from Cellpose - here illustrated in Figure 13.

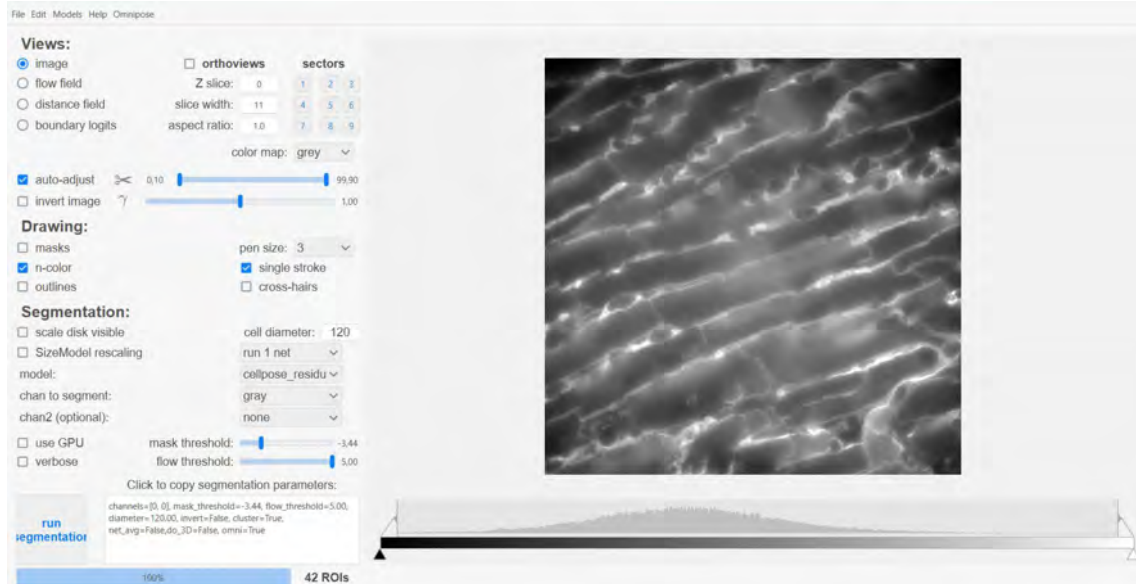


Figure 13: The GUI of Omnipose. Just as in Cellpose, visualization, manual segmentation and training can be performed in the GUI. Parameters such as diameter, thresholds and image brightness & contrast can be selected in the toolbar to the left. For visualization purposes, the user can view the raw image solely, or overlaid with outlines and/or masks. Additional adjustments to Cellpose, such as including N-colors and inverting the image, are possible in this GUI.

Apart from diverging in the actual GUI appearance, there are some additional settings to choose among in Omnipose to further adjust the output segmentation. These mainly include changing the foreground/background values of the image (i.e., the gamma-value), as well as using the so-called N-colors option. By implementing the N-colors option, one can easier distinguish and edit masks by only printing 4 colors.

Cell Diameter

The most prominent difference between Cellpose and Omnipose however, lies within the definition of the cell diameter. As Omnipose is developed as a tool for mainly segmenting bacteria, having markedly different geometry than what is considered in Cellpose, the diameter is defined accordingly. Thus, the idea of defining the diameter as the diameter of a circle with equivalent area will fail when introducing non-circular, anisotropic cells [56]. Instead, the diameter is expressed as specified in Eq.4, and ultimately the effective diameter in Eq.5.

$$\bar{\phi} = \frac{1}{\pi R^2} \int_0^{2\pi} \int_0^R (R-r)r dr d\theta = \frac{1}{\pi R^2} \left(\frac{\pi}{3} R^3 \right) = \frac{R}{3} \quad (4)$$

$$d = 2R = 6\bar{\phi} \quad (5)$$

Herein, $\bar{\phi}$ represents the mean of the distance field over the cell. A distance field represents a distance to the closest boundary pixel, which for bacteria will confine with its inherent thickness [56]. This field is averaged over an entire region (cell), which naturally will become proportional to the thickness of that region. The 'diameter' aspect is thus incorporated in Eq.4 as the diameter of an n-sphere considering the average of the distance field over the volume [42]. Just as in Cellpose, unit is specified in pixels and the default diameter is 30 pixels.

Rescaling & Resampling

An additional feature of Omnipose (and Cellpose to some extent) being considered in this project is the rescaling aspect of the algorithm. When prescribing a diameter in Cellpose, the input image will be rescaled to fit a default diameter of 30 pixels. Thus, if one assigns a diameter of 60 pixels, the rescaling factor of the image will be $30/60 = 0.5$ to accordingly fit a cell size that is manageable by the U-Net architecture [56]. This can be problematic for cells with more rod-like shapes; to fit a longitudinal cell shape into Cellpose's definition of the diameter (Eq. 2) one needs to set a very big diameter, which in turn will result in a rescaled image that might too small for Cellpose to segment. Cellpose is always running on rescaled images, however in Omnipose the user may disable this function. In this case, the images will be resampled to fit the original image-size before running the segmentation. This would in turn yield smoother output ROIs, however at a cost of being more time-consuming [56]. The aspect of resampling will thus be investigated as a parameter in the model development.

4.3 Model Development

As previously stated, the approach for achieving an adequate segmentation is to make two isolated models; one that covers the cross-section CMs and the smaller cells, and conversely one that covers the longitudinal CMs. The main reason for this is to enable fine-tuning of model parameters such as diameter and threshold parameters. The project will start by developing the cross-section model, mainly due to the fact that this model development is more straight forward than the corresponding longitudinal model; the cells are more conventionally shaped as compared to the longitudinal CMs. As the time-frame for this project is limited, the longitudinal model(s) will be initialized and-provided there is time remaining- optimized to some extent. For both Cellpose and Omnipose, the generic procedure for training a model- starting from a pretrained model- is as follows:

1. Run segmentation on a set of images by using a pretrained model from the zoo.
2. The errors made by the model is corrected in the GUI by manually drawing the missing and/or incorrectly segmented ROIs.
3. The new ground truth annotations are utilized to train the model in 1.
4. Step 2 and 3 is repeated.

This iterative process is repeated until a satisfactory accuracy is achieved. As an alternative to step 1, one can create ground-truth masks on the images and use these to train a model from scratch. As an alternative option to the GUI, training and segmentation can be initialized through the command line (using CPU), which is also a more time-efficient option. To save time when training new models, this will be the approach forward; the GUI will mainly be used for visualization and mask-correction purposes. Commands for training and segmentation are specified in the appendices.

Non of the models will be trained on cells that are not covered in the FOV, i.e., the cells around the edges of each FOV.

4.3.1 Over & Underfitting

One aspect of training the model is to make sure that it does not become overfitted. Overfitting is a direct consequence of training a model on sample data for too long, or alternatively if the model becomes too complex and begin to extract information from "noise" or other irrelevant information from the input dataset [59]. If the noise or other irrelevant data were to be memorized by the model there is a high risk that it will be unable to sufficiently generalize new datasets. Consequently, when the model is provided with new data it will be prone to perform poorly on the classification or prediction tasks that it was initially intended for. Common characteristics of an overfitted model is low error rates and high variance. A way to assess whether a model has become overfitted is to introduce the model to a "test" dataset (being a fraction from the training set, set aside for this purpose) and investigate the quality of its prediction; if the training data has a low error rate whereas the test set has a high one, the model is likely overfitted [59]. Conversely, there is also a risk that the model is underfitted. This is the case when the model has not been trained for long enough, or if the introduced variables are not sufficient to determine a significant relationship between input and output variables [59]. In contrast to overfitting, the main indication of underfitting is that the model is demonstrating a high bias and a low variance upon generating its predictions. Albeit, the consequence of underfitting a model is the same as overfitting; the model generalizes poorly to unseen data. The main challenge is thus to find a balance between the two extremities and to accordingly achieve a model which is liable to establishing a dominant trend and being broadly applicable to new datasets [59].

The practical implications for avoiding these issues are twofold. Firstly, to circumvent overfitting the models, a diverse training dataset will be used for the training. This implies that images from various experiments- with different image qualities and morphologies- will be included when developing the models. If needed, a pretrained model from the Cellpose model zoo can be utilized as a starting point to further avoid overfitting the data. Secondly, underfitting will be avoided by keeping record of the model development and making sure that the segmentation outputs are not indicating a high bias and low variance when introducing the model to unseen images. Another important aspect to consider when training the models concerns how the training data are included between consecutive training rounds. Notably, if retraining a model with the exact same set of images as in the prior training, the network will be biased towards these images in any consecutive training. Conversely, if a pretrained model is trained on a completely new set of images, without including the previous training data, all previous images will instead be downweighed. Consequently, when developing the cross-section and longitudinal models, the approach will be to include *all* images from past training rounds in each consecutive training. Moreover, to maximize the impact that any segmented images make to a training set, the image area containing cells will be maximized too (by either cropping images, or simply selecting FOVs containing as many cells as possible).

4.3.2 Cross-Section Model

For developing a segmentation model for cross-section cells, Cellpose 2.0 will be the strategy forward. Herein, FOVs exclusively containing the cross-sectional parts of the tissue, together the smaller cells, are selected. Images from different imaging rounds are included to ensure a comprehensive development of the model-training. As the morphology of a healthy heart potentially will be markedly

different from a diseased one, images covering both states will be included in the development too; as the end goal is to be able to study how the MERFISH data is altered during disease, the aim is to generate a model applicable to all relevant morphologies.

4.3.3 Longitudinal Model

For generating a model appropriate for the longitudinal sections of the CMs, both Cellpose 2.0 and Omnipose will be used to generate two separate models for comparison. Omnipose is adapted considering its alternative definition of cell diameter - a parameter that potentially will be more promising for the purpose of longitudinal CMs. Cellpose will however still be evaluated. This is mainly due to the fact that Cellpose indeed has been trained on highly diverse data - including non-biological and e.g., dendrites - why it is arguable that the algorithm may work sufficiently well for the data of concern in this project too. Moreover, from a more high-level perspective, using Cellpose for both segmentation models would presumably reduce the need for any additional implementations or adjustment to the overall MERFISH-analysis pipeline. Each model will be trained and optimized based on the same training data; the output segmentation from respective segmentation will be juxtaposed to evaluate the most promising algorithm to move forward with. In terms of generating the ground-truth masks for training purposes, all smaller cells and cross-section CMs will be neglected to make sure the models are trained on exclusively segmenting longitudinal CMs.

Stitching & Flat-field Correction

Longitudinal CMs are substantially larger than remaining cells in the cardiac tissue. In fact, as evident from Figure 9 CMs often extend over several field of views (FOVs) (each FOV being the size of 2048x2048 px, i.e. 219 μm). Consequently, it is not possible to draw more than a few cells per FOV, which is insufficient when training a model. Hence, in contrast to the cross-section model (where there already is a sufficient number of cells to segment within a single FOV), there is a need for additional image processing. The strategy will here be to generate training images composed of stitched FOVs, which accordingly would meet the requirement of approximately 100 ground-truth masks per FOV. What is moreover evident from Figure 9 is the fact that the edges surrounding each FOVs are darker than the rest of the image. As the segmentation algorithm is based on pixel intensity, this might be especially problematic when stitching together FOVs; there will be darker borders at each conjunction, aggravating the model's ability to achieve an accurate segmentation. Flat-field correction (FFC), being a method for improving the image quality and generating a more uniform background. This is achieved by correcting e.g., discrepancies in light sensitivity between pixel sensors of the camera, differences in illumination between different FOVs, or alternatively the variations in transmission of light through the camera lens (giving rise to the vignetting effect as prominent in depicted images). Consequently, FFC will be incorporated into the stitching code with the aim of circumventing these undesired image distortions.

4.4 Supplementary Software for Image Processing

Complementary software tools for extracting and processing the heart tissue images include image processing in Fiji and some python-based coding in Jupyter Notebook [60, 61]. Fiji is an image processing distribution of the open source software ImageJ, specifically used for processing biological images [39]. Codes for stitching and performing FFC of images will be written in Jupyter Notebook.

5 Results & Discussion

This section provides the results gathered from generating and improving the cross-section and longitudinal models respectively. Section 5.1 and 5.2 deliver the development and progression of the cross-section and longitudinal models respectively. Before developing the models, the performance of default models in Cellpose and Omnipose was established and evaluated as a benchmark for any subsequent progression. For all models, additional threshold optimization was performed to allow for further improvement without implementing more training. Lastly, a shorter section demonstrating the overall segmentation performance is presented in section 5.3. A discussion concerning the development and progression of the models will follow each section.

5.1 Cross-section Model

This section presents the results from training a Cellpose model intended for cell segmentation of the cross-section CMs and other smaller, round-shaped cardiac cells (such as fibroblasts and immune cell). Included herein are the benchmark of the performance of Cellpose’s default model, the training specifics, evaluations of model performances, and lastly the threshold optimization.

5.1.1 Benchmarking

To evaluate the cross-section model development, the segmentation performance of one of Cellpose’s default models was benchmarked. Herein, the Cellpose network was fed with an image representative for the cross-section areas of the tissue to let it predict the style of the image. Based on the predicted style, the network suggested the most appropriate model from the model zoo. This came to be CPx; a pretrained model which is the most general model from the zoo; it has been trained on the most diverse set of training data- including non-biological features [47]. The fact that the network predicts CPx to be the most promising model for segmentation yet highlights the fact that the cardiac tissue is highly complex and not a straight-forward procedure. In addition to predicting the most suitable default segmentation model, the network was set to calibrate the mean diameter of the cells in the image as well.

Table 1: Results from segmenting images of cross-section regions of the cardiac tissue using the default model CPx in Cellpose. Included are the calibrated diameter used, the average percentage of masks removed and the mean accuracy of the segmentation. Segmentation was performed using default thresholds, i.e., flow threshold 0.4 and cell probability 0.

Model	Diameter (px)	Inaccurate ROIs (%)	Mean accuracy (%)
CPx	57.2	78.9	13.2

The performance of CPx was quantified and is presented in Table 1. As evident, the mean accuracy (defined as in Eq.1) of the segmentation is 13.2%, using a calibrated diameter of 57.2 pixels. The average percentage of removed masks per segmentation was almost 80%. To further illustrate the performance of the CPx model, one of its segmentations is featured in Figure 14. Paired with one of the output segmentations of CPx is the same image but with the masks considered accurate from the output; inaccurate, segmentations are here removed. Notably, the result emphasizes the need for generating a new model with the machine learning HITL approach; an accuracy of around 13% is far from sufficient to achieve a segmentation feasible of covering the main part (being the cross-section regions of the tissue) of the cardiac tissue.

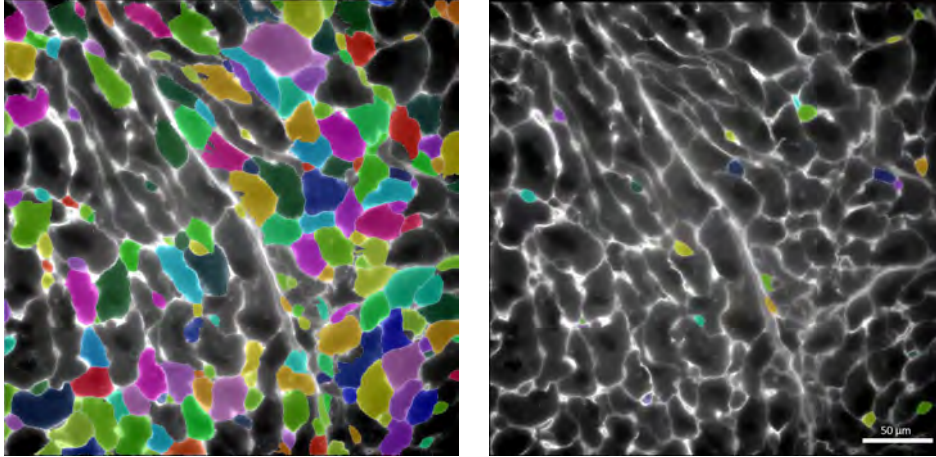


Figure 14: Example of segmentation of cross-section CMs using the default model CPx in Cellpose. Left image is the output segmentation from CPx. The right image illustrates the masks left after removing all inaccurate segmentations.

5.1.2 Model Training & Performance

The cross-section model was trained from scratch for three consecutive training rounds, each model being named model version 1-3 respectively. Specifics of the executed model training, including the number of FOVs, the total number of annotated masks, and the average number of masks per FOV, are summarized in Table 2.

Table 2: Training data for developing the cross-section model in Cellpose. Included for each model version 1-3 are the number of images and mask per training round and the average number of masks per image.

Model version	# training images*	# training ROIs**	Av. # of ROI/FOV
1	12	1254	193
2	12 + 10	1236	124
3	12 + 10 + 20	2404	120

* Only including additional images from previous training round ** Masks from previous training not included

As previously described, for each consecutive training all training data from previous rounds are included. This is done to avoid any downweighing of previous data when introducing new training. Consequently, the number of training masks and images specified in Table 2 only include the *additional* number introduced. All training masks were generated in the Cellpose GUI, and the training was initiated from the command line. The exact commands, including the specific training settings (all default parameters) are presented in Appendix A. A selection of ground-truth masks used for training are featured in Figure 15. Evidently, not all cells are annotated; the strategy for generating ground-truth masks was to solely annotate the cells with clear distinct boundaries to ensure that the model is trained accurately.

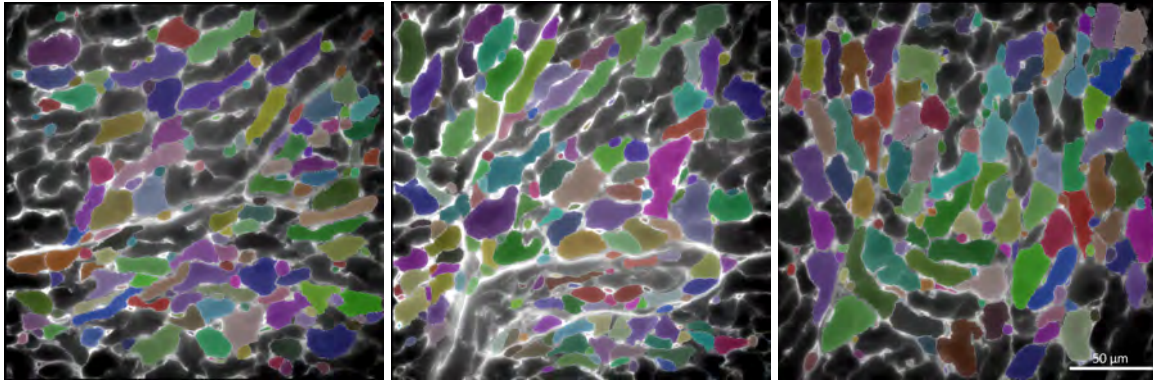


Figure 15: A selection of ground-truth masks for training a cross-section Cellpose model.

Examples of segmentations performed by model version 1-3 are featured in Figure 16. Juxtaposed with the output segmentations are the masks left after removing all incorrectly segmented masks. All segmentations evaluated in here are performed with default thresholds (i.e., 0.4 and 0 for flow and cell probability threshold respectively).

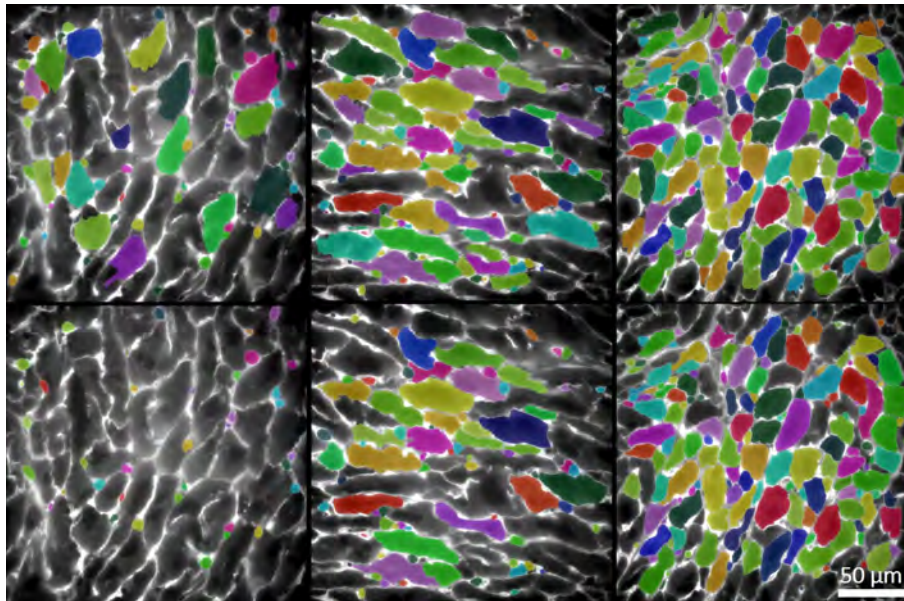


Figure 16: Segmentation from each model version (1-3) of the cross-section models. The upper row are the output segmentations from version 1-3 (left to right) respectively. Similarly, the bottom row are the remaining masks after removing the inaccurate segmentations from the output.

As evident, there is a prominent difference between the consecutive training rounds and their respective performance. After the first training round, there are still no accurate segmentation of larger cells- just as for the default CPx model. Nevertheless, a greater fraction of segmented cells are kept after inspecting the segmentation and removing inaccurate segmentations, suggesting a higher accuracy. Upon initiating the second training however, the performance is markedly improved. Not only is a larger portion of the FOV segmented, but the majority of the larger cells are in fact segmented. Furthermore, comparing the output segmentation with its corresponding revised

segmentation, the majority of these larger cells are indeed also correctly segmented. There are however still some inaccuracies in this model version, emphasizing the need of consecutive training. Inspecting the third, and last, training of the cross-section model, the performance is even stronger than its progenitor. Herein, almost the entire FOV is segmented, with the exception of the edges around the image. This exception is however expected due to two reasons; (1) the brightness of the FOV is always lower around the edges due to variations in light transmission through the camera lens, thus obstructing the overall segmentation feasibility; and (2) the model is simply not trained on segmenting cells around the edges, as it is not desired to generate half cells when applying the model to segment the entire cardiac tissue. Comparing the segmentation in Figure 16 before and after removing incorrect masks in version 3, the difference is minuscule. The masks being removed in this round are mainly due to some incomplete segmentations; oversegmentation, cell-merging and cell-splitting were rarely encountered.

The above-mentioned findings are further corroborated with the quantitative evaluation of the different model performances, here presented in Table 3. Included are the diameters used for segmentation, the average number of segmented masks per FOV, the average percentage of removed masks, and lastly the mean accuracy of the segmentation.

Table 3: Performance of cross-section models, versions 1-3. Included are the diameter used, the average number of ROIs, the average percentage of masks removed and the mean accuracy of the segmentation. The performances are contrasted to the default CPx model.

Model version	Diameter (px)	Av. # ROIs	Inaccurate ROIs (%)	Mean accuracy (%)
Default CPx	57.2	140	78.9	13.2
1	79.4	119	50.9	48.5
2	100	156	40.8	78.8
3	120	155	17.1	89.4

A pronounced trend between the consecutive versions and the ratio of inaccurate segmentation is apparent: the percentage of inaccurate segmentation decreases from almost 80% inaccurate masks for the default model, to approximately 17% for model version 3. This trend suggests that the output masks from version 3 are more precise than previous models, which is further validated by the substantial increase in mean accuracy (increasing from 13.2% to 89.4%, corresponding to an almost 6-folded increment). In order to apply this model on the cardiac tissue however, it is not sufficient to have a model that is accurate but not comprehensive; in order to implement the segmentation model and perform any downstream analysis of the MERFISH data, there is a decisive need for a model that allows for segmentation with full coverage of the tissue. The accuracy must thus be contrasted with the number of masks for each FOV, being an indication of the respective segmentation coverage. As evident from the average number of masks per FOV in Table 3, there is a prominent improvement in coverage between version 1 and 2, whereas version 3 has the same coverage as version 2. Note however that the amount of inaccurate masks are markedly reduced between these models, suggesting that the ultimate coverage is still greater. What is further apparent from the last column of the table is that the model accuracy is not improving to the same extent between consecutive training rounds; the improvement in accuracy stagnates between version 2 and 3. This suggests that the model may not be gaining from additional training, being one of the main reasoning behind why no further training was performed.

Yet another aspect of the model performance is the input for cell diameter. Evident from Table 3, the diameter chosen for segmentation varied between versions. For segmentation with CPx and version 1, the calibrated diameter was used. For version 2 however, the diameter was increased to

100 as it gave rise to its most optimized segmentation outputs. In contrast, 120 pixels was chosen for the final, version 3 model. The choice of 120 pixels was based on a qualitative review of the output segmentations for different input diameters. An illustration of how the segmentations differs for the same image, just by altering the diameter, is delineated in Figure 17. As evident, the segmentation output is markedly different depending on the diameter set. In line with what is stated in section 4.2.1, upon decreasing the diameter (from calibrated diameter to 30 px in this case), only very few of the larger cells are segmented; and those who are segmented are highly incomplete. Moreover, a number of lines are appearing on the larger cells. Notably, only the smallest cells are segmented, however not to a great extent; there are still a lot of absent segmentations for the smaller cells as well.

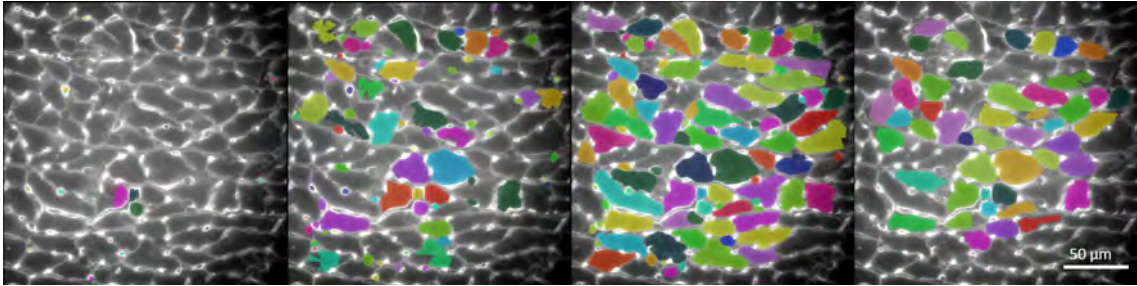


Figure 17: A representation of the differences in segmentation output as a function of cell diameter for cross-section model version 3. From left to right: 30 px, 75.9 (calibrated), 120, 300. Corresponding number of output segmentations were 75, 96, 112, 51 ROIs respectively

As compared to the case of a 30 px diameter, setting the diameter to what is calibrated by the network (in this case 75.9 px) yields a more accurate and comprehensive segmentation of the small cells. Another significant discrepancy between 30 and 75.9 px is the fact that larger cells are segmented to a significantly higher degree. Albeit, these segmentations are still not accurate; as evident, there are a lot of incomplete segmentations, indicating the need for increasing the diameter further. Increasing the diameter to 120 px returns the most complete segmentation performance. Herein, the model manage to cover segmentation of a broad range of cell sizes, including both the smallest and the largest cells in the image- without compromising the accuracy of the segmentation. Moreover, the number of incomplete segmentations are almost negligible. Setting a larger diameter than 120 px was shown to generate fewer and less accurate segmentations. To illustrate this, a diameter of 300 px is presented in Figure 17 (diameters between 120 and 300 px was investigated as well, demonstrating the same trends but with less pronounced discrepancies. See Figure B1 in Appendix B for comparison). Evidently, the performance degrades upon increasing the diameter beyond 120 px, which is also clear from the significant decrease in number of masks (112 against 51 masks for 120 and 300 px respectively). In contrast to 120 px, 300 px also yields some cell-merging and oversegmentation - both being factors indicating a diameter set too large.

5.1.3 Threshold Optimization

In this section, threshold parameters were adjusted to examine whether it was possible to further improve the performance of model version 3 *without* introducing any additional training. Herein, the thresholds of concern are the flow and cell probability threshold respectively, being introduced in section 4.2.1.

Flow Threshold

Results from varying the flow threshold on the same set of images using cross-section model version 3 are presented in Table 4. Results include the average percentage of inaccurately segmented masks, the mean accuracy and the increase/decrease in number of masks compared to the default flow threshold. As evident, the greatest accuracy (89.4%) and the second to smallest fraction of inaccurately segmented masks is achieved when using the default threshold. Notably, even though a flow threshold of 0.1 yields the smallest fraction of inaccurate masks (9.16%), the mean accuracy remains the smallest, which is in line with theory; lowering the threshold will yield fewer masks yet the majority will be accurate. This is further highlighted by inspecting the decrease in number of segmented masks, being a decrease of more than half the number of masks as compared to the default. A lower threshold than the default is thus not sufficient to generate an applicable model.

Table 4: Results from varying the flow threshold for cross-section model version 3. Included are the average percentage of inaccurate cells, the mean accuracy and lastly the change in number of accurately segmented masks compared to the default settings. Cell probability is herein set to 0 (default).

Flow threshold	Inaccurate masks (%)	Mean accuracy (%)	$\frac{\Delta ROI_i}{\#ROI_{\text{default}}}$ (%)
0.1	9.16	37.8	-57
Default (0.4)	17.1	89.4	0
0.7	32.6	80.2	-8.9
1	47.9	68.3	-22

Increasing the flow threshold above the default setting is instead increasing the average number of inaccurate masks, however again decreasing the mean accuracy. The predominant change in segmentation output between default and higher threshold is essentially an increased number of incomplete masks, mainly in the form of lines or small masks. An example of how a threshold of 1 differ from the default is delineated in Figure 18. Herein, apart from missing some of the larger cells, there is a large number of additional lines scattered over the image. Consequently, the default flow threshold remains the optimized parameter for the cross-section model version 3.

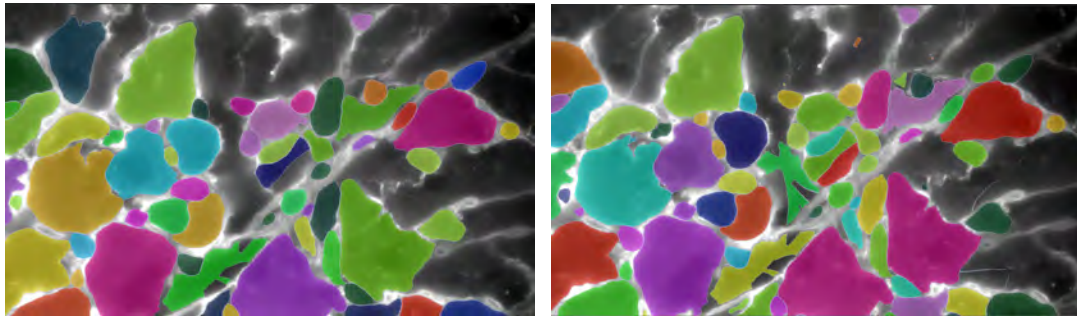


Figure 18: Comparison of segmentations from cross-section model version 3 using a flow threshold of 0.4 (default, left) and 1 (right) respectively.

Cell Probability Threshold

Results from varying the cell probability threshold on the same set of images using cross-section model version 3 are presented in Table 5.

Table 5: Results from varying the cell probability threshold for cross-section model version 3. Included are the average percentage of inaccurate cells, the mean accuracy and the change in number of accurately segmented masks in relation to default settings. Flow threshold is set to 0.4 (default).

Cell probability threshold	Inaccurate masks (%)	Mean accuracy (%)	$\frac{\Delta ROI_i}{\#ROI_{\text{default}}}$ (%)
0 (Default)	17.1	89.4	0
-0.5	18.4	88.6	-0.21
-1	21.5	91.2	4.1
-1.5	21.6	84.9	-4.7
-2	35.7	69.6	-19

Evident from Table 5, the mean accuracy increases upon decreasing the cell probability threshold. It then culminates at -1 with a mean accuracy of 91.2% and then decreases when lowering the threshold further. In terms of the fraction of inaccurate masks, this is lowest for the default model (with 17.1%) and increases with decreasing threshold. However, when contrasting the fraction of inaccurate masks to the mean accuracy, it becomes clear that the optimized cell probability threshold still is -1; the accuracy is the highest, and the number of correctly segmented masks are actually higher than the default segmentation.

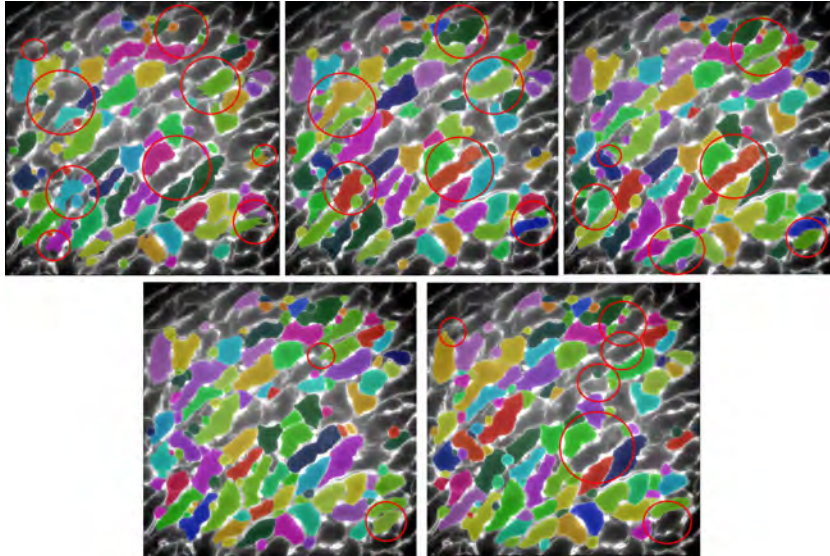


Figure 19: Comparison of segmentations from cross-section model version 3 when varying the cell probability threshold. From left to right: threshold (upper row:) 0, -0.5, -1, (lower row:) -1.5, -2. Discrepancies between consecutive different thresholds are highlighted by the red circles in each FOV.

Figure 19 illustrates some qualitative discrepancies between outputs of different cell probability thresholds. Highlighted with red circles are some of the differences between the various threshold settings. Upon lowering the threshold from default to -0.5, the segmentation improves in regards to two aspects; (1) incomplete masks become more complete; and (2) absent segmentations are now being segmented to some extent. Lowering the threshold further to -1 demonstrates the same

trend. Upon lowering the threshold to -1.5, however, will not generate any markedly improvement. A few masks become more complete, but the predominant difference is the addition of lines and in some cases oversegmentation. In contrast, when segmenting with a threshold of -2 fewer masks are generated, which is highly undesired. Consequently, considering the findings from above, a flow and cell probability threshold of 0.4 and -1 respectively is suggested to be settings which achieves an optimized segmentation without introducing any further training. However, as an additional validation, the thresholds are simultaneously reviewed to confirm that there isn't any conceivable combination of thresholds that would be more promising. Figure 20 features a qualitative evaluation of the different combination of flow and cell probability thresholds. Highlighted in red is the segmentation with default thresholds.

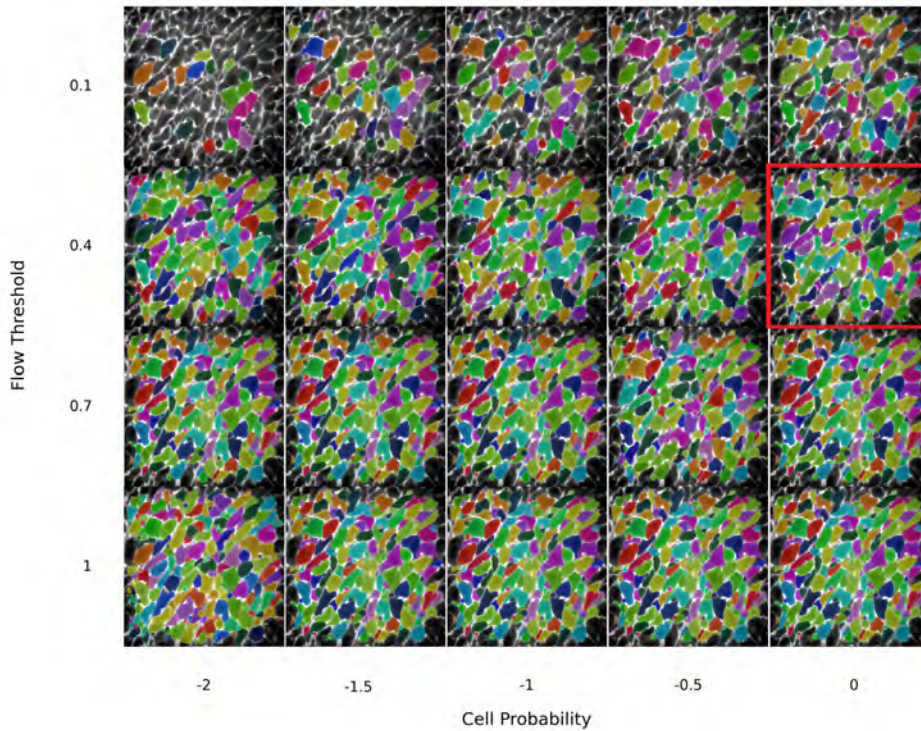


Figure 20: Segmentation performance of cross-section model version 3 as a function of different flow and cell probability thresholds. Default segmentation is highlighted in red.

Supplementary information to Figure 20 is summarized in Table 6, where the number of output masks are stated for each threshold combination. Evidently, all segmentations generated with a flow threshold above 0.4 will generate too many incorrect masks, regardless of chosen cell probability threshold. Correspondingly, all segmentations with a flow threshold of 0.1 will not generate sufficient numbers of masks. Ultimately, the most optimal threshold settings are thus with a flow threshold of 0.4 and - as previously examined- a cell probability threshold of -1.

Table 6: Number of output masks as a function of different combinations of threshold settings. Default settings are highlighted in bold numbers.

Flow/Cell prob.	-2	-1.5	-1	-0.5	0
0.1	20	38	52	68	90
0.4	195	195	197	197	193
0.7	235	224	221	215	204
1	431	332	271	240	225

In summary, worth highlighting is the fact that after training and optimizing the cell diameter and threshold settings, the cross-section model ultimately demonstrates a mean accuracy of 91.2%, which is a great improvement from both the default CPx segmentation and the performance after only a single training round. Consecutive training could certainly be performed, however due to limitations in time and the fact that a high mean accuracy was achieved, the development of a longitudinal model is from here on the top-priority.

5.2 Longitudinal Model

This section presents the results gathered from training a Cellpose and Omnipose model, intended for segmentation of the longitudinal CMs in the cardiac tissue. The approach for training longitudinal CMs are different from the cross-section regions in the heart in the sense that the raw microscopy images must be processed before generating any training data. The process of stitching and flat-field corrected (FFC) FOVs will thus firstly be presented. Subsequently, benchmarking and model development are presented for Cellpose and Omnipose respectively. Due to limitation in time, the main purpose of this section was to investigate which of the machine learning networks- Cellpose or Omnipose- was deemed as the most promising approach to move forward with; the work of performing further training and optimization of a longitudinal model will be continued at Kosuri Lab post termination of this thesis. Consequently, the evaluations in this section will exclusively be more qualitative as compared to the cross-section model.

5.2.1 Stitching and Flat-field Correction

To generate training data feasible for developing and training a longitudinal model, consecutive FOVs (with a size of 2048x2048 px) were stitched together in chunks of 3x3 FOVs (corresponding to a size of 6144x6144 px) based on the respective coordinates of the FOV in the tissue. A stitched image is presented in Figure 21.

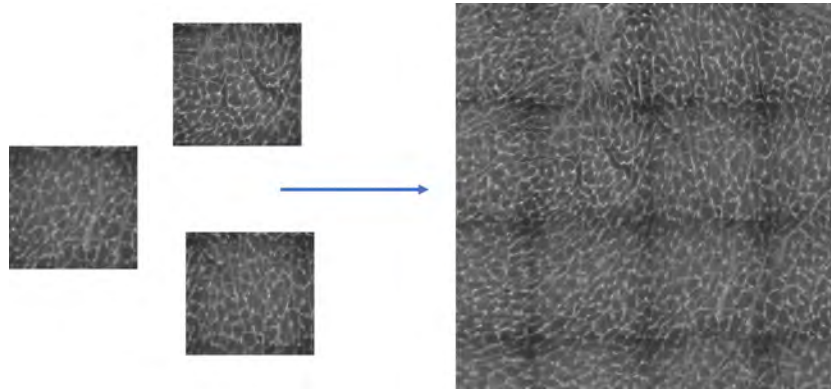


Figure 21: Single FOVs are stitched together in 3x3 chunks to a total size of 6144x6144 px.

The darker borders around the edges of each FOV, brought to attention in previous sections, is herein even more tangible. As the end-goal for segmenting the cardiac tissue is to segment the entire tissue in one go, not FOV by FOV, dark borders in the image is highly undesired. Furthermore, as concluded in section 5.1, the cross-section model does in general not perform well in segmenting the darker areas of the images. Thus, to maximize the impact of the training as well as avoiding any distortions of the longitudinal model training and performance, the stitched chunks were FF corrected. A collection of the resulting stitched and FFC images are featured in Figure 22.

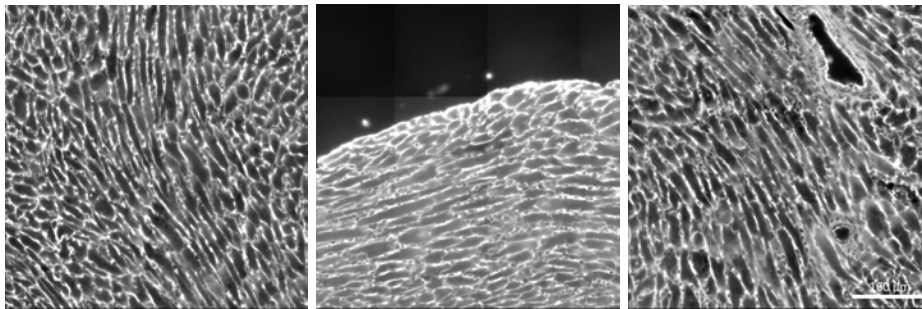


Figure 22: FF Corrected and Stitched FOVs.

Examples of ground-truth masks generated for the longitudinal models are featured in Figure 23. As evident from presented training images, not all longitudinal masks are annotated. Apart from the half cells, this is mainly due to uncertainties in how the boundaries of the cells are located; the cell boundary staining are in some cases distorted, especially in the tight junctions between consecutive longitudinal CMs. This is undoubtedly one of the major challenges in terms of segmentation.

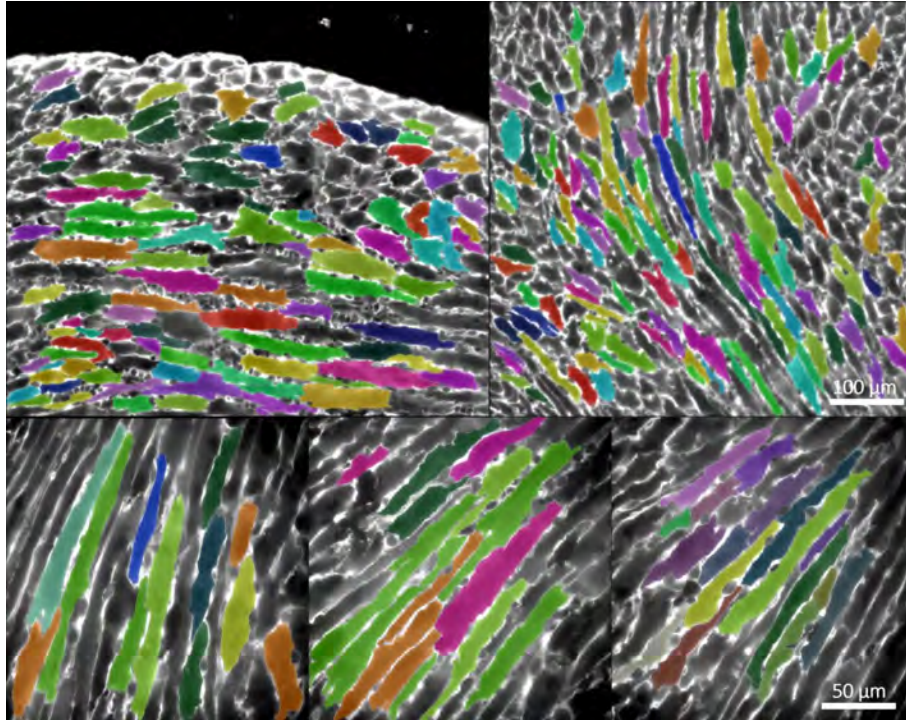


Figure 23: A selection of ground-truth masks for the longitudinal model. Stitched and FFC images in the top row, and single FOVs in the bottom.

To allow for a fair comparison between the Cellpose and Omnipose performances, both models were trained on the same ground truth masks. In addition to the stitched FOVs, single FOVs are included in the training data as well. The number of training images together with the total number of masks, both for the stitched and single FOVs, are compiled in Table 7.

Table 7: Training data for generating longitudinal models. Included are the number of images and total number of masks for both stitched and single FOVs.

	# Images	# Masks
Stitched FOV	10	939
Single FOV	6	82

5.2.2 Cellpose Model

This section is concerned with exploring the plausibility of applying the Cellpose network to the longitudinal regions of the heart. Included is a benchmark for the default models of Cellpose, along with the model development and training. As previously stated, there will be no quantitative assessment of the generated model, only a qualitative evaluation.

Benchmarking

Before evaluating the Cellpose default model performance, the cross-section model generated in the previous section was adopted to further validate that an additional model in fact is required to achieve accurate segmentation of the longitudinal regions. One resulting segmentation is depicted in Figure 24, segmented with a diameter of 347 pixels, as suggested by the network. The dark area at the bottom of the stitched image descends from an out-of-focus FOV.

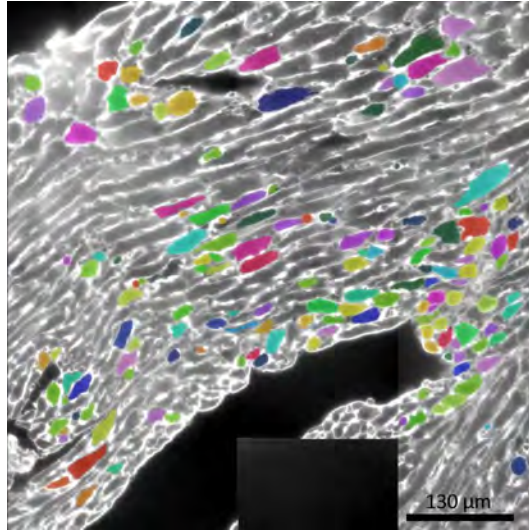


Figure 24: Cross-section model version 3 segmentation applied on a longitudinal region of the heart. Diameter set to 347 px, default thresholds.

Evidently, the cross-section Cellpose model is indeed not sufficient to segment the longitudinal CMs; the only generated masks are of the cross-section CMs and smaller cells, despite the fact that the diameter was set to almost three times larger. This is however a favourable feature in terms of model selectivity towards the cross-section CMs, as it suggests a minimized amount of overlap between the models. To further explore the performance of a default model in Cellpose, the same procedure as in section 5.1 was performed; the network ran to predict the style of an image, and from there suggest the most suitable default model. As expected, CPx was again chosen. Moreover, the network predictions suggested a diameter of 347 pixels. One resulting segmentation output is presented in Figure 25.

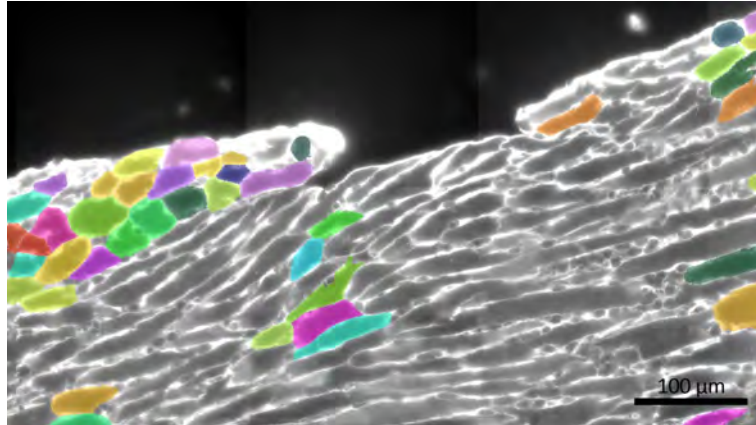


Figure 25: Benchmark for Cellpose default model CPx on segmenting the longitudinal regions of the tissue. Diameter calibrated and set to 347 px.

Evident from Figure 25, the CPx model performed well on segmenting the longitudinal regions of the tissue. Notably, only the cross-section CMs are segmented, and not with a great accuracy. This once again highlights the need for implementation of a new model trained with the HITL approach.

Model Training & Performance

Just as for the cross-section model, the Cellpose longitudinal model was initially trained from scratch with all the masks specified in Table 7, i.e., both stitched and single FOVs. Training commands and settings are compiled in Appendix A. One of the output segmentations from this model is presented in Figure 26.

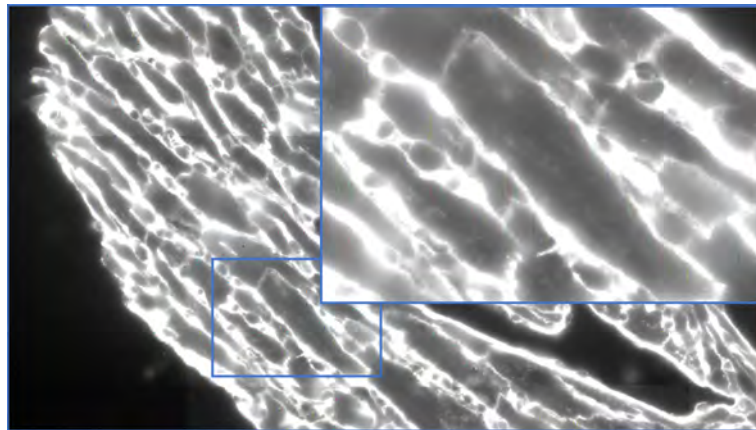


Figure 26: Segmentation generated from a longitudinal Cellpose model trained from scratch. The upper right corner is a close-up of the area surrounded in blue. Diameter set to 347 px, default thresholds.

Conspicuously, the output segmentation from training a longitudinal model from scratch only generates lines - in this case 3002 lines in one stitched FOV with the size 6144x6144 pxs (i.e., more ROIs than what is included in the *entire* training data). The reason for this may be two-folded;

(1) The Cellpose network has never been trained on any morphologies similar to the longitudinal regions in the heart; and (2) as previously discussed, the definition of the cell diameter is highly incompatible with the shape of the longitudinal CMs.

In an attempt of still adopting the Cellpose network to the longitudinal CMs, a new approach was taken. Herein, CPx was trained on the same training data to explore whether a pretrained model could perform better after introducing longitudinal training data. CPx was chosen as the pretrained model considering it was predicted as the most promising model by the network. In this approach, training was performed in two consecutive steps, wherein CPx firstly was trained on the single FOVs specified in Table 7, and then in a second training based on both the single FOVs and the stitched FOVs (specified in the same table). For comparison, Figure 27 delineates the difference in segmentation output between CPx default model, CPx trained on single FOVs and lastly CPx trained a second time on both single FOVs and stitched FOVs.

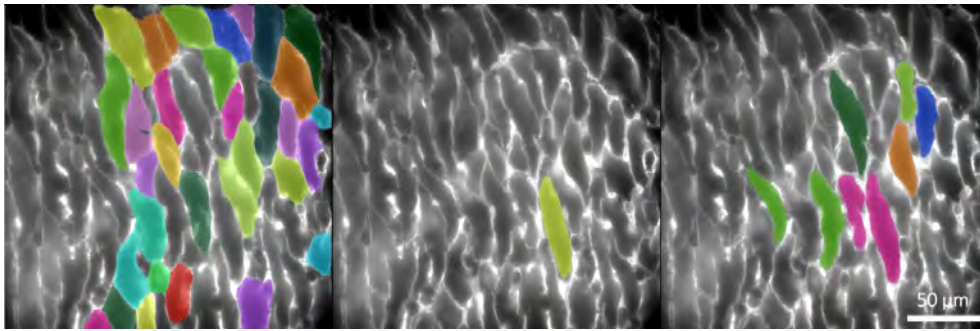


Figure 27: Training progress of longitudinal Cellpose model. From left to right: CPx default segmentation, 1st training on single FOV, 2nd training on single FOV and stitched FOVs. Default threshold and diameter of 352 px as suggested by the network.

The number of masks segmented was drastically reduced between the default CPx segmentation and after its first training. Note however that the segmentation from the default model is highly inaccurate, as compared as the single mask segmented after the first training round. Upon training the model a second time, the segmentation output was altered again; the model now segmented more cells, still fewer than CPx, but considerably more accurate. The generated model after two training rounds was then further reviewed to discern whether changing the flow and cell probability threshold would have a positive impact on the segmentation. Results in terms of segmentation outputs for the same FOV is presented in Figure 28. Evidently, there is no significant change in segmentation when altering the flow threshold. However, decreasing the cell probability will indeed improve the segmentation to some extent; upon decreasing this threshold, more masks are found *and* previously incomplete masks become more complete (in line with the observations discerned in section 5.1). Note, however, that the overall segmentation is not accurate and still suffers from absent segmentations, cell-merging and oversegmentation.

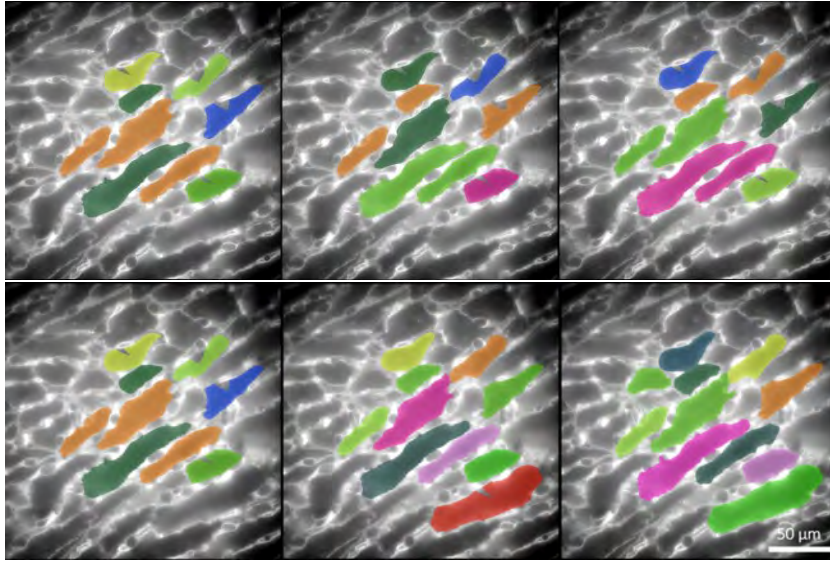


Figure 28: Changing flow threshold (upper row) between 0.4, 0.5, 0.6 and cell probability (lower row) between 0, -0.2, -0.6 threshold CPx-based longitudinal model. 340 px diameter.

Furthermore, Figure 29 showcases the difference in segmentation of a stitched image using default and -1 cell probability threshold respectively. Notably, the segmentation demonstrates an encouraging result; more cells are segmented for the lower threshold (275 vs 186 ROIs) and the masks are more complete.

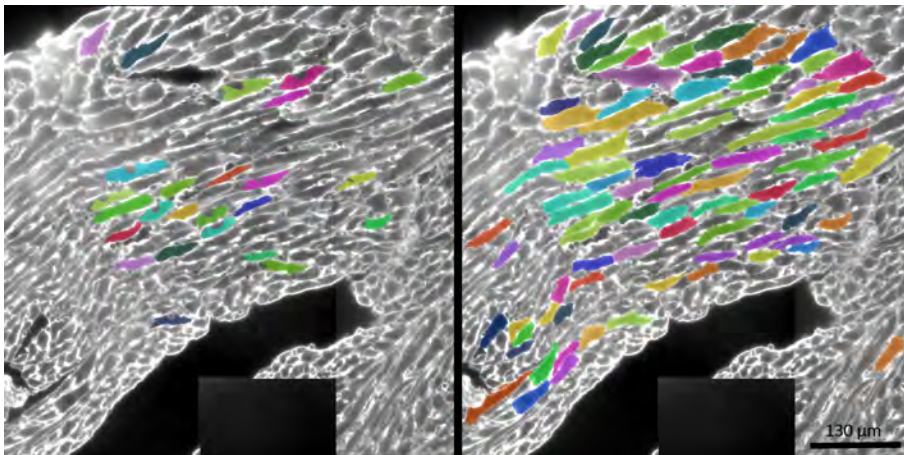


Figure 29: Default cell probability threshold (left) is contrasted with a -1 threshold (right) for the longitudinal Cellpose model.

Just as in Figure 28, Figure 29 exposes the inaccurate segmentation performance of this model even further, as only a fraction of the masks would be considered accurate when performing correction. In contrast to the cross-section model, discerning the most optimal threshold and diameter settings for this longitudinal model is highly restricted; as the accuracy of the segmentation is close to zero for all thresholds, one simply cannot distinguish the optimal threshold at this point. Moreover, after initializing further training to the model, it is possible that the optimal threshold will not stay

the same. Nevertheless, the cardinal inference of these result is that a longitudinal model generated with Cellpose indeed demonstrates a promising option to proceed with, and that the performance will be markedly improved upon lowering the cell probability threshold..

What is moreover apparent from the segmentation in Figure 29 is that the longitudinal CMs rarely are segmented over the border between FOVs in the stitched image. Upon examining the stitched and FFC images closer, one can discern some miss-alignments between certain FOVs, which in turn gives rise to some distorted cells. This may be due to drifts throughout the imaging procedure, obstructing the assignment of coordinates. If revisiting the theoretical section 4.2.1 concerning how the vector flow fields of Cellpose are generated, a plausible consequence of this artefact can be derived. As the Cellpose algorithm turns to heat diffusion simulations to generate the flow fields, wherein the heat source at the center of a cell exerts “heat” to its surrounding neighboring pixels at every iteration, while assigning all pixels outside of the mask are assigned to zero, the boundaries of the cell mask will be “leaky”. Consequently, in the cases where the stitched image is somewhat distorted, there is a risk that the heat does not dissipate to the furthest corners of the cell. What may furthermore be a concerning feature of the stitched images, is the relatively sharp line between the FOVs, being a direct consequence of the minor differences in image brightness between consecutive FOVs. These two aspects are thus likely to obstruct the output segmentations of Cellpose. To further optimize the training impact, as well as to refine the segmentation quality, one should therefore resolve these artefacts.

5.2.3 Omnipose Model

This section is concerned with exploring the plausibility of applying the Omnipose network to the longitudinal regions of the heart as an alternative approach to Cellpose. Included is a benchmark for the default models of Omnipose, along with the model development and training.

Benchmarking

Figure 30 presents the benchmark segmentation performed by one of the default models of Omnipose, named `bact_phase_omni`. Similar to Cellpose’s CPx model, `bact_phase_omni` is one of the most general and diversely trained models from the Omnipose model zoo. Figure 30 demonstrates segmentation on a stitched and FFC image. The cell diameter of 150 pixels was decided from the average width of a longitudinal CM, provided that the $0.107\ \mu\text{m}$ per pixels. Both threshold parameters are here set to its default values (0.4 and 0 for flow and cell probability threshold respectively). Evident from Figure 30, the default segmentation in Omnipose is strikingly different from the corresponding one in Cellpose. As compared to Cellpose’s CPx model, `bact_phase_omni` segments with less selectivity and indeed outputs more oblong-looking masks. Moreover, even fibroblasts and other smaller cells are herein segmented; in general, the segmentation from `bact_phase_omni` covers a larger area of the images than CPx, albeit with less accuracy (see Figure 31 for a close-up of the segmentation output). As previously discussed, any discrepancies between the two model outputs can be derived from the different characteristics of their respective training data, as well as the definition of cell diameter.

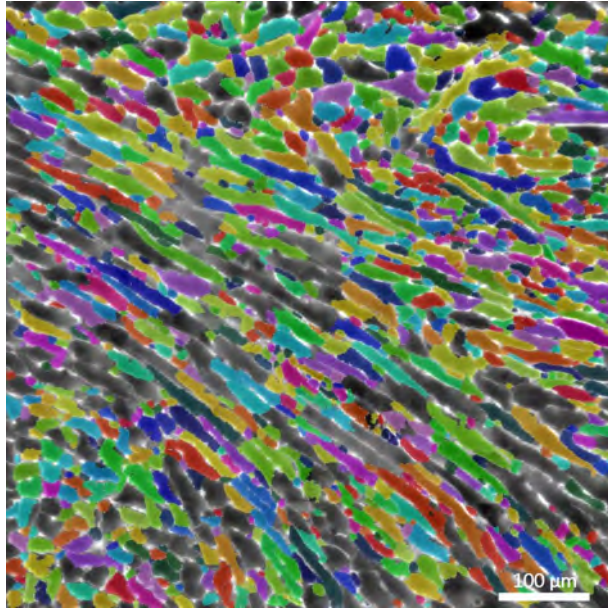


Figure 30: Segmentation performance of Omnipose’s default model `bact_phase_omni`. Default thresholds, diameter set to 150 px.

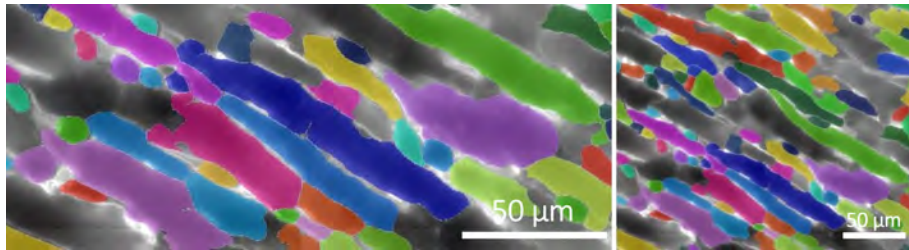


Figure 31: Close-up of a default segmentation in Omnipose using the `bact_phase_omni` model.

Model Training & Performance

Just as for the longitudinal Cellpose model, a longitudinal Omnipose model was initially trained from scratch on all ground truth masks from Table 7. Training commands and settings are compiled in Appendix A. One of the output segmentations from this model, after optimizing the flow and cell probability threshold in the same manner as for Cellpose, is presented in Figure 32. Notably, an omnipose model could in fact be trained from scratch, as compared to the corresponding Cellpose model. Moreover, the segmentation performance is greatly improved upon introducing training to the Omnipose network; the accuracy is improved whilst simultaneously maintaining the tissue-coverage. More of the longitudinal CMs are also segmented over the smaller cells. Nevertheless, the segmentation accuracy is still poor as it (mainly) demonstrates cell-merging and oversegmentation.



Figure 32: Resulting segmentation from longitudinal Omnipose model trained from scratch on single and stitched FOVs. Cell probability threshold set to -3, flow threshold at 0 (default). Diameter 150 px.

5.2.4 Cellpose VS Omnipose

A comparison between the performance of the longitudinal models generated in Cellpose and Omnipose is here displayed in Figure 33. As previously discussed, the segmentations are markedly different from each other. The main difference is the selectivity; the Omnipose model segments substantially more masks, however less selectively than the corresponding Cellpose model. As full coverage is one of the cardinal features requested when generating a model applicable for analyzing MERFISH data of an entire heart, the most promising model to move forward with is deemed to be the Omnipose model; upon training the Omnipose model, it demonstrates a more promising development – becoming more specific yet keeping broad coverage just after introducing the same set of training data. More training should however be introduced to both models to draw more rigorous conclusions.

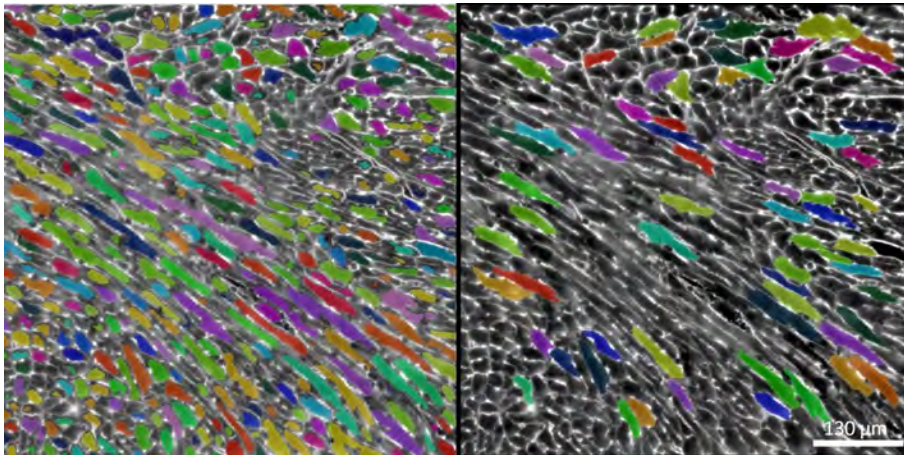


Figure 33: Segmentation comparison between model generated in Omnipose (left) and Cellpose (right) respectively.

5.3 Overall Performance

In an attempt to give insights into how the overall segmentation of the heart tissue would play out with the current state of the models, this section features some combined segmentations on stitched and FFC FOVs (from both the cross-section and a longitudinal model), as well as a segmentation of an entire heart tissue section using the cross-section model only. Even though the Omnipose model is currently deemed as the most promising longitudinal model to move forward with, for visualization purposes, its segmentation is not specific enough to combine with the cross-section model. Instead, the cross-section model is combined with the longitudinal Cellpose model. A selection of these combined segmentations are featured in Figure 34. Figure 35 display sections of a full-scale segmentation on one entire z-plane of the heart tissue using the cross-section Cellpose model.

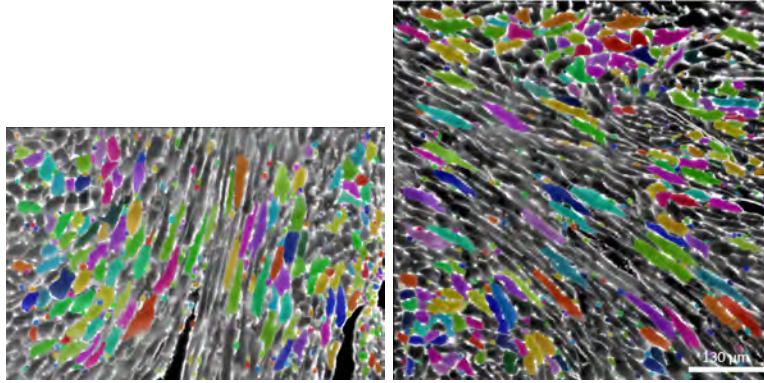


Figure 34: Merged segmentation from cross-section and longitudinal models, both generated in Cellpose.

Evident from Figure 34, the segmentation is currently not specific enough to cover the entire tissue section, however is substantially improved from how the benchmark, default, models perform, especially in terms of the cross-section segmentation. This is further corroborated by examining the sections of full-scale cross-section segmentation in Figure 35. Notably, ignoring the longitudinal regions of the heart, the majority of the tissue is herein segmented. One can however discern lines between the segmentations, being especially clear in the left image of the figure. This is a direct consequence of the tissue has been segmented one FOV at the time; the cross-section model has not been trained on segmenting the edges of each FOV, and does not perform well in the darker non-FFC areas of the FOVs. If the segmentation were to be performed on the a FFC version of an entire tissue, as opposed to just one FOV at the time, the model would most likely be segmenting even more masks.

Overall, the current segmentation performance is indeed promising; even though the longitudinal model(s) are lacking in accuracy, the majority of the heart can as of now be segmented with high accuracy. Once the longitudinal model has been introduced to more training, the entire heart tissue will be segmented with high coverage and accuracy. Nevertheless, as the cross-sectional regions make up the majority of the heart, one may already be able to apply the model and start deducing some general features from the MERFISH data- especially in the healthy heart, were there are no fibrosis and less irregular cell morphologies. Importantly, to be able to draw any valid conclusions from the MERFISH data of how the organization of cardiac cells are altered during disease, it is crucial to ensure that the final segmentation model can segment the vast majority of the cells, including the

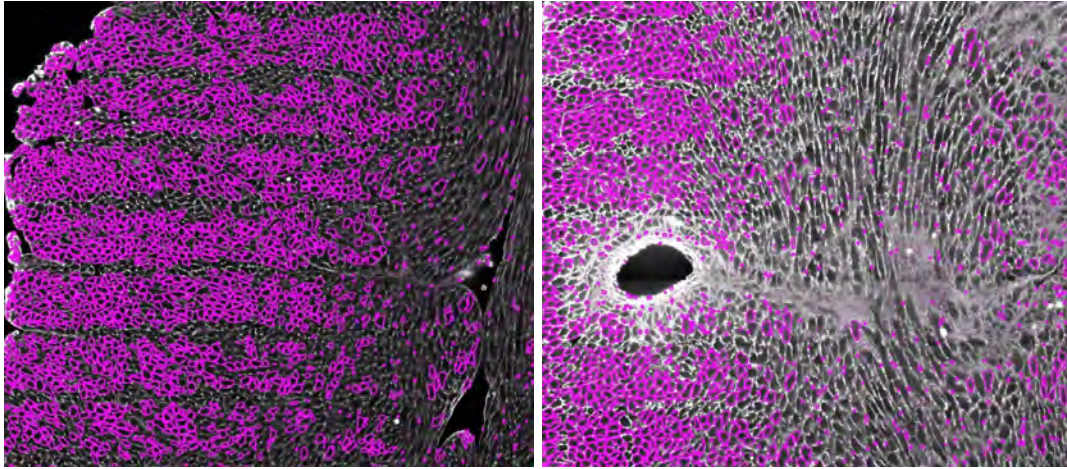


Figure 35: Close-ups of the full-scale cross-section segmentation of the cardiac tissue. Unpublished data, *Liu et.al.*

fibrotic regions and hypertrophic cells. The diseased heart is further challenging since in many cases the boundary staining is more irregular, especially in the tight junction between the longitudinal CMs.

6 Conclusions & Final Remarks

A cross-section model was developed in Cellpose from scratch and improved by implementing 3 consecutive training rounds. After 3 training rounds, a final mean accuracy of 89.4% was achieved when segmenting images with default threshold settings and a diameter of 120 pixels. Upon changing the flow and cell probability threshold to 0.4 and -1 respectively, the mean accuracy was further improved to 91.2%, this without implementing any additional training. This result suggests that the generated model is a promising approach for segmenting the cross-section regions in the heart, thus paving the way towards achieving a full-scale segmentation of the tissue. Further training could naturally be performed, however considering that the progression curve stagnated between the second and third training, and that the model already achieves a mean accuracy of 91.2%, it is deemed as efficient enough to be able to explore further features of the MERFISH data. Moreover, one must decide on a limit for when the model is “accurate enough”; to achieve a 100% accuracy is close to impossible, and there are yet a lot of analysis that can be performed with a 91.2% accuracy.

For the longitudinal region, models were generated using both Cellpose and Omnipose algorithms to identify the most promising approach to move forward with. The cellpose model was developed from the pretrained CPx model, whereas the Omnipose was successfully developed from scratch. As the longitudinal CMs extend over several field of views (FOVs), consecutive FOVs were stitched together before generating ground truth masks for training the models. These stitched FOVs were thereafter flat-field corrected (FFC) to remove the darker edges around each FOV. Due to the differences in training data, network construction and definition of cell diameter, the two algorithms generated remarkably different segmentation outputs. Cellpose generated fewer, yet more accurate masks. Omnipose instead offered more coverage yet with less specificity and accuracy upon being introduced to the same training data. Possible artefacts obstructing the segmentation performance might be derived from some miss-alignments and darker lines between the FOVs, which is why it is encouraged to remove these before performing any future training of the model(s). As one of the main requested attributes of the segmentation is to generate full coverage of the tissue, since ultimately transcripts of the entire heart are to be analyzed, Omnipose was deemed as the most promising algorithm to move forward with. Notably, the specificity of the model was greatly improved upon introducing the same training data. More training should however be performed to both models before excluding Cellpose completely.

6.1 Future Prospects

Certainly, there is a lot of work to be done to further improve the segmentation and make it more rigorous; the work conducted herein will be carried on by the Kosuri Lab post termination of the thesis. As previously stated, the primary focus for the future is to establish a more accurate and specific segmentation model for the longitudinal CMs. Whether this will be in terms of performing more training to the initialized Omnipose model (or possibly the Cellpose model), make alterations to the algorithm code, or even adapt a completely new algorithm/approach, is subject to future evaluation. In addition to improving the longitudinal model further, the stitching and FFC should be perfected to remove the artefacts brought to attention in section 5.2.2. Moreover, a challenge remains in segmenting cells in fibrotic regions of the diseased heart, where the boundary staining is particularly distorted (see Figure 36 for an example).

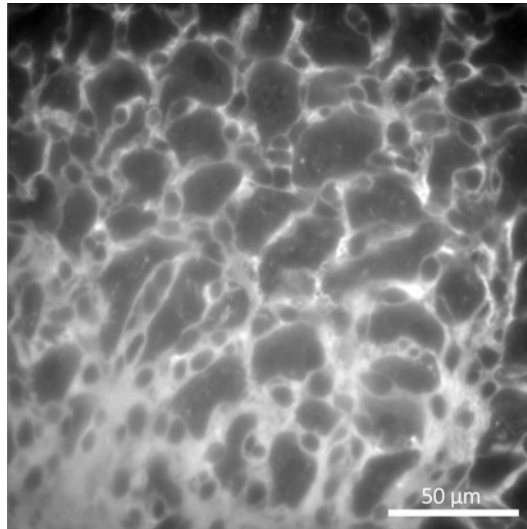


Figure 36: Microscopy image of a diseased heart, with the white area being cardiac fibrosis.

Finally, the ultimate goal would be to achieve an accurate 3D segmentation, i.e., identifying cells throughout several z -planes of the tissue. Challenges herein lies withing accurately assigning masks from different z -planes to the same cell, being particularly challenging for the irregular shaped cardiac cells - especially the CMs.

References

- [1] Gianluigi Savarese and Lars H Lund. “Global Public Health Burden of Heart Failure”. In: *Cardiac Failure Review* 03 (2017), p. 7. DOI: 10.15420/cfr.2016:25:2. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5494150/>.
- [2] Monika Litviňuková et al. “Cells of the adult human heart”. In: *Nature* 588 (Sept. 2020), pp. 1–7. DOI: 10.1038/s41586-020-2797-4. URL: <https://www.nature.com/articles/s41586-020-2797-4>.
- [3] Torsten Doenst, Tien Dung Nguyen, and E. Dale Abel. “Cardiac Metabolism in Heart Failure”. In: *Circulation Research* 113 (Aug. 2013), pp. 709–724. DOI: 10.1161/circresaha.113.300376. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3896379/>.
- [4] Sigolène M. Meilhac and Margaret E. Buckingham. “The deployment of cell lineages that form the mammalian heart”. In: *Nature Reviews Cardiology* 15 (Nov. 2018), pp. 705–724. DOI: 10.1038/s41569-018-0086-9. URL: https://www.nature.com/articles/s41569-018-0086-9?WT.feed_name=subjects_heart-development#Sec14.
- [5] Li Wang et al. “Single-cell reconstruction of the adult human heart during heart failure and recovery reveals the cellular landscape underlying cardiac function”. In: *Nature Cell Biology* 22 (Jan. 2020), pp. 108–119. DOI: 10.1038/s41556-019-0446-7. URL: <https://pubmed.ncbi.nlm.nih.gov/31915373/> (visited on 04/28/2021).
- [6] Michelle M. Kittleson et al. “Identification of a Gene Expression Profile That Differentiates Between Ischemic and Nonischemic Cardiomyopathy”. In: *Circulation* 110 (Nov. 2004), pp. 3444–3451. DOI: 10.1161/01.cir.0000148178.19465.11. (Visited on 04/17/2023).
- [7] Rongxin Fang et al. “Conservation and divergence of cortical cell organization in human and mouse revealed by MERFISH”. In: *Science* 377 (July 2022), pp. 56–62. DOI: 10.1126/science.abm1741. (Visited on 11/28/2022).
- [8] Kok Hao Chen et al. “Spatially resolved, highly multiplexed RNA profiling in single cells”. In: *Science (New York, N.Y.)* 348 (Apr. 2015), aaa6090. DOI: 10.1126/science.aaa6090. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4662681/>.
- [9] *KOSURI LAB*. KOSURI LAB. URL: <https://kosurilab.com/main#heartatlas> (visited on 04/18/2023).
- [10] Ludvig Larsson, Jonas Frisé, and Joakim Lundeberg. “Spatially resolved transcriptomics adds a new dimension to genomics”. In: *Nature Methods* 18 (Jan. 2021), pp. 15–18. DOI: 10.1038/s41592-020-01038-7.
- [11] Filippos Triposkiadis et al. “The Interventricular Septum: Structure, Function, Dysfunction, and Diseases”. In: *Journal of Clinical Medicine* 11 (June 2022), p. 3227. DOI: 10.3390/jcm11113227.
- [12] Virpi Talman and Riikka Kivelä. “Cardiomyocyte—Endothelial Cell Interactions in Cardiac Remodeling and Regeneration”. In: *Frontiers in Cardiovascular Medicine* 5 (July 2018). DOI: 10.3389/fcvm.2018.00101. URL: <https://dx.doi.org/10.3389/fcvm.2018.00101>.
- [13] Vincent F. M. Segers, Dirk L. Brutsaert, and Gilles W. De Keulenaer. “Cardiac Remodeling: Endothelial Cells Have More to Say Than Just NO”. In: *Frontiers in Physiology* 9 (Apr. 2018). DOI: 10.3389/fphys.2018.00382.
- [14] Thomas King. *Elsevier’s Integrated Pathology E-Book*. Saintt Louis Elsevier Health Sciences, 2006.

- [15] Marek Michalak and Luis B. Agellon. “Stress Coping Strategies in the Heart: An Integrated View”. In: *Frontiers in Cardiovascular Medicine* 5 (Nov. 2018). DOI: 10.3389/fcvm.2018.00168. (Visited on 12/30/2021).
- [16] Shunsuke Funakoshi et al. “Generation of mature compact ventricular cardiomyocytes from human pluripotent stem cells”. In: *Nature Communications* 12 (May 2021). DOI: 10.1038/s41467-021-23329-z. (Visited on 04/29/2022).
- [17] Colby A. Souders, Stephanie L.K. Bowers, and Troy A. Baudino. “Cardiac Fibroblast”. In: *Circulation Research* 105 (Dec. 2009), pp. 1164–1176. DOI: 10.1161/circresaha.109.209809. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3345531/>.
- [18] Andrea Trost et al. “Pericytes in the Retina”. In: *Advances in Experimental Medicine and Biology* 1122 (2019), pp. 1–26. DOI: 10.1007/978-3-030-11093-2_1. URL: https://link.springer.com/chapter/10.1007%2F978-3-030-11093-2_1 (visited on 12/17/2019).
- [19] Chenglong Xia et al. “Spatial transcriptome profiling by MERFISH reveals subcellular RNA compartmentalization and cell cycle-dependent gene expression”. In: *Proceedings of the National Academy of Sciences* 116 (Sept. 2019), pp. 19490–19499. DOI: 10.1073/pnas.1912459116.
- [20] Bruce Alberts et al. *An Overview of Gene Control*. Nih.gov, 2016. URL: <https://www.ncbi.nlm.nih.gov/books/NBK26885/>.
- [21] Viktor Petukhov et al. “Cell segmentation in imaging-based spatial transcriptomics”. In: *Nature Biotechnology* 40 (Oct. 2021), pp. 345–354. DOI: 10.1038/s41587-021-01044-w. (Visited on 02/23/2023).
- [22] “Method of the Year 2020: spatially resolved transcriptomics”. In: *Nature Methods* 18 (Jan. 2021), pp. 1–1. DOI: 10.1038/s41592-020-01042-x.
- [23] Xiaowei Zhuang. “Spatially resolved single-cell genomics and transcriptomics by imaging”. In: *Nature Methods* 18 (Jan. 2021), pp. 18–22. DOI: 10.1038/s41592-020-01037-8. URL: <https://www.nature.com/articles/s41592-020-01037-8.pdf> (visited on 08/16/2021).
- [24] Vivien Marx. “Method of the Year: spatially resolved transcriptomics”. In: *Nature Methods* 18 (Jan. 2021), pp. 9–14. DOI: 10.1038/s41592-020-01033-y. URL: <https://www.nature.com/articles/s41592-020-01033-y>.
- [25] Michaela Asp et al. “A Spatiotemporal Organ-Wide Gene Expression and Cell Atlas of the Developing Human Heart”. In: *Cell* 179 (Dec. 2019), 1647–1660.e19. DOI: 10.1016/j.cell.2019.11.025. URL: <https://www.sciencedirect.com/science/article/pii/S0092867419312826>.
- [26] Abdul Rauf Shakoori. “Fluorescence In Situ Hybridization (FISH) and Its Applications”. In: *Chromosome Structure and Aberrations* (2017), pp. 343–367. DOI: 10.1007/978-81-322-3673-3_16. URL: https://link.springer.com/chapter/10.1007/978-81-322-3673-3_16.
- [27] A. M. Femino. “Visualization of Single RNA Transcripts in Situ”. In: *Science* 280 (Apr. 1998), pp. 585–590. DOI: 10.1126/science.280.5363.585. (Visited on 05/06/2020).
- [28] J.R. Moffitt and X. Zhuang. “RNA Imaging with Multiplexed Error-Robust Fluorescence In Situ Hybridization (MERFISH)”. In: *Visualizing RNA Dynamics in the Cell* (2016), pp. 1–49. DOI: 10.1016/bs.mie.2016.03.020. (Visited on 02/23/2023).
- [29] Gal Haimovich and Jeffrey Gerst. “Single-molecule Fluorescence in situ Hybridization (sm-FISH) for RNA Detection in Adherent Animal Cells”. In: *BIO-PROTOCOL* 8 (2018). DOI: 10.21769/bioprotoc.3070.

- [30] Ed Lein, Lars E. Borm, and Sten Linnarsson. “The promise of spatial transcriptomics for neuroscience in the era of molecular cell typing”. In: *Science* 358 (Oct. 2017), pp. 64–69. DOI: 10.1126/science.aan6827. URL: <https://www.science.org/doi/10.1126/science.aan6827>.
- [31] Changqing Yi et al. “Microfluidics technology for manipulation and analysis of biological cells”. In: *Analytica Chimica Acta* 560 (Feb. 2006), pp. 1–23. DOI: 10.1016/j.aca.2005.12.037. (Visited on 04/07/2023).
- [32] Zhenzhou Wang. “Cell Segmentation for Image Cytometry: Advances, Insufficiencies, and Challenges”. In: *Cytometry Part A* 95 (Dec. 2018), pp. 708–711. DOI: 10.1002/cyto.a.23686. (Visited on 05/10/2020).
- [33] Rohan Lewis. *The Definitive Guide to Cell Segmentation Analysis*. Biodock, Mar. 2021. URL: <https://blog.biodock.ai/definitive-guide-to-cell-segmentation-analysis> (visited on 02/27/2023).
- [34] Chee-Huat Linus Eng et al. “Transcriptome-scale super-resolved imaging in tissues by RNA seqFISH+”. In: *Nature* 568 (Apr. 2019), pp. 235–239. DOI: 10.1038/s41586-019-1049-y. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6544023/>.
- [35] Betty I. Tarnowski, Francis G. Spinale, and James H. Nicholson. “DAPI as a Useful Stain for Nuclear Quantitation”. In: *Biotechnic Histochemistry* 66 (Jan. 1991), pp. 296–302. DOI: 10.3109/10520299109109990.
- [36] Jeongbin Park et al. “Cell segmentation-free inference of cell types from in situ transcriptomics data”. In: *Nature Communications* 12 (June 2021). DOI: 10.1038/s41467-021-23807-4. (Visited on 02/23/2023).
- [37] Kyuseok Im et al. “An introduction to Performing Immunofluorescence Staining”. In: *Methods in molecular biology (Clifton, N.J.)* 1897 (2019), pp. 299–311. DOI: 10.1007/978-1-4939-8935-5_26. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6918834/>.
- [38] Lucas R Smith and Elisabeth R Barton. “SMASH – semi-automatic muscle analysis using segmentation of histology: a MATLAB application”. In: *Skeletal Muscle* 4 (2014), p. 21. DOI: 10.1186/2044-5040-4-21. (Visited on 04/19/2019).
- [39] Caroline A Schneider, Wayne S Rasband, and Kevin W Eliceiri. “NIH Image to ImageJ: 25 years of image analysis”. In: *Nature Methods* 9 (June 2012), pp. 671–675. DOI: 10.1038/nmeth.2089.
- [40] Erlend Hodneland et al. “CellSegm - a MATLAB toolbox for high-throughput 3D cell segmentation”. In: *Source Code for Biology and Medicine* 8 (Aug. 2013). DOI: 10.1186/1751-0473-8-16. (Visited on 04/30/2019).
- [41] Anne E Carpenter et al. “CellProfiler: image analysis software for identifying and quantifying cell phenotypes”. In: *Genome Biology* 7 (2006), R100. DOI: 10.1186/gb-2006-7-10-r100. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1794559/>.
- [42] Carsen Stringer et al. “Cellpose: a generalist algorithm for cellular segmentation”. In: *Nature Methods* 18 (Jan. 2021), pp. 100–106. DOI: 10.1038/s41592-020-01018-x. URL: <https://www.nature.com/articles/s41592-020-01018-x> (visited on 03/26/2021).
- [43] Steve Eddins. *The Watershed Transform: Strategies for Image Segmentation*. MathWorks, 2002. URL: <https://www.mathworks.com/company/newsletters/articles/the-watershed-transform-strategies-for-image-segmentation.html>.

- [44] Steve Eddins. *The Watershed Transform: Strategies for Image Segmentation*. Mathworks, 2002. URL: <https://www.mathworks.com/company/newsletters/articles/the-watershed-transform-strategies-for-image-segmentation.html>.
- [45] Tomas Vicar et al. “Cell segmentation methods for label-free contrast microscopy: review and comprehensive comparison”. In: *BMC Bioinformatics* 20 (June 2019). DOI: 10.1186/s12859-019-2880-8. (Visited on 08/07/2020).
- [46] Ewa Pietka. “48 - Image Standardization in PACS”. In: *Handbook of Medical Imaging*. Ed. by ISAAC N. BANKMAN. Biomedical Engineering. San Diego: Academic Press, 2000, pp. 783–801. ISBN: 978-0-12-077790-7. DOI: <https://doi.org/10.1016/B978-012077790-7/50056-4>. URL: <https://www.sciencedirect.com/science/article/pii/B9780120777907500564>.
- [47] Marius Pachitariu and Carsen Stringer. “Cellpose 2.0: how to train your own model”. In: *Nature Methods* 19 (Nov. 2022). DOI: 10.1038/s41592-022-01663-4. (Visited on 11/21/2022).
- [48] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully Convolutional Networks for Semantic Segmentation”. In: *arXiv (Cornell University)* (Nov. 2014). DOI: 10.48550/arxiv.1411.4038. (Visited on 04/28/2023).
- [49] IBM. *What are Convolutional Neural Networks? — IBM*. www.ibm.com. URL: <https://www.ibm.com/topics/convolutional-neural-networks>.
- [50] *What is a Convolutional Layer?* Databricks. URL: <https://www.databricks.com/glossary/convolutional-layer>.
- [51] *Imbalanced Data*. Google Developers, July 2022. URL: <https://developers.google.com/machine-learning/data-prep/construct/sampling-splitting/imbalanced-data>.
- [52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *arXiv:1505.04597 [cs]* (May 2015). URL: <https://arxiv.org/abs/1505.04597v1>.
- [53] G. LI et al. “Segmentation of touching cell nuclei using gradient flow tracking”. In: *Journal of Microscopy* 231 (July 2008), pp. 47–58. DOI: 10.1111/j.1365-2818.2008.02016.x. (Visited on 11/30/2021).
- [54] *Representation — Machine Learning Crash Course*. Google Developers, July 2022. URL: <https://developers.google.com/machine-learning/crash-course/representation/video-lecture>.
- [55] CloudFactory. *Human in the Loop: Accelerating the AI Lifecycle*. Cloudfactory. URL: <https://www.cloudfactory.com/human-in-the-loop> (visited on 02/27/2023).
- [56] Kevin J. Cutler et al. “Omnipose: a high-precision morphology-independent solution for bacterial cell segmentation”. In: *Nature Methods* 19 (Oct. 2022), pp. 1438–1448. DOI: 10.1038/s41592-022-01639-4. (Visited on 11/30/2022).
- [57] Uwe Schmidt et al. “Cell Detection with Star-convex Polygons”. In: *arXiv:1806.03535 [cs]* 11071 (2018), pp. 265–273. DOI: 10.1007/978-3-030-00934-2_30. URL: <https://arxiv.org/abs/1806.03535>.
- [58] Kaiming He et al. “Mask R-CNN”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2018), pp. 1–1. DOI: 10.1109/tpami.2018.2844175. (Visited on 03/22/2019).
- [59] *What is Overfitting?* www.ibm.com. URL: <https://www.ibm.com/topics/overfitting>.
- [60] Johannes Schindelin et al. “Fiji: an open-source platform for biological-image analysis”. In: *Nature methods* 9 (2012), pp. 676–82. DOI: 10.1038/nmeth.2019.

[61] Jupyter. *Project Jupyter*. Jupyter.org, 2019. URL: <https://jupyter.org/>.

Appendix A

Model training parameters

Cross-section Cellpose Model

Version 1:

- **Pretrained Model:** None
- **Learning Rate:** 0.1
- **Weight Decay:** 0.0001
- **Number of Epochs:** 100
- **Channels:** 0

Version 2:

- **Pretrained Model:** Version 1
- **Learning Rate:** 0.1
- **Weight Decay:** 0.0001
- **Number of Epochs:** 100
- **Channels:** 0

Version 3:

- **Pretrained Model:** Version 2
- **Learning Rate:** 0.1
- **Weight Decay:** 0.0001
- **Number of Epochs:** 100
- **Channels:** 0

Longitudinal Cellpose Model

Test 1:

- **Pretrained Model:** None
- **Learning Rate:** 0.1
- **Weight Decay:** 0.0001
- **Number of Epochs:** 100
- **Channels:** 0

Version 1:

- **Pretrained Model:** CPx
- **Learning Rate:** 0.1
- **Weight Decay:** 0.0001
- **Number of Epochs:** 100
- **Channels:** 0

Version 2:

- **Pretrained Model:** Version 1
- **Learning Rate:** 0.1
- **Weight Decay:** 0.0001
- **Number of Epochs:** 100
- **Channels:** 0

Longitudinal Omnipose Model

Version 1:

- **Pretrained Model:** None
- **Learning Rate:** 0.1
- **Weight Decay:** 0.0001
- **Number of Epochs:** 500
- **Channels:** 0
- **Diameter:** 150

Version 2:

- **Pretrained Model:** Version 1
- **Learning Rate:** 0.1
- **Weight Decay:** 0.0001
- **Number of Epochs:** 500
- **Channels:** 0
- **Diameter:** 150

Appendix B

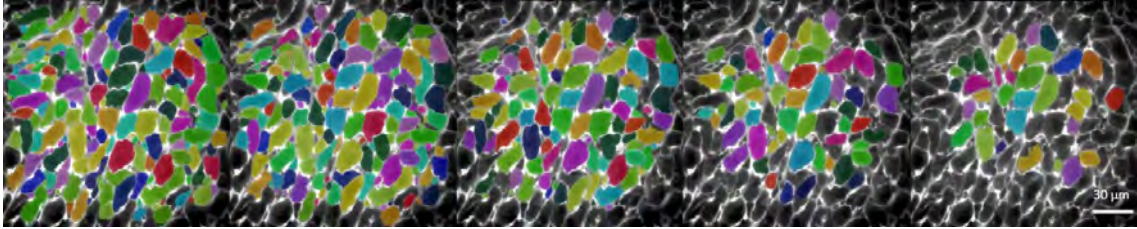


Figure B1: Segmentations generated from cross-section model version 3, varying diameter between 120 and 300 px (from left to right: 120, 150, 200, 250, 300 px). The number of masks for each diameter was (from left to right) 201, 160, 107, 70 & 36.