

Anni Nieminen

**IMAGE AND VIDEO PROCESSING
METHODS FOR STUDYING
hiPSC-DERIVED CARDIOMYOCYTE
BIOMECHANICS**

Bachelor's thesis
Faculty of Medicine and Health Technology
Birhanu Belay
May 2023

ABSTRACT

Anni Nieminen: Image and video processing methods for studying hiPSC-derived cardiomyocyte biomechanics
Bachelor's thesis
Tampere University
Biotechnology and Biomedical Engineering
May 2023

Cardiovascular disease (CVD) is the most common cause of death globally. Human induced pluripotent stem cell-derived cardiomyocytes (hiPSC-CMs) provide a suitable model for studying CVD and developing new treatments. Using hiPSC-CMs in heart disease modelling has several benefits compared to using other cell types such as embryonic stem cells or animal cells. Many cardiovascular diseases are known to affect cardiomyocyte behaviour including their biomechanics. Therefore, studying the contractile machinery and force production of hiPSC-CMs is central in CVD research.

Microscopy imaging methods are required for scientists to view cells, their contractile structures and study their force production. Namely fluorescence microscopy and traction force microscopy -based imaging methods are often used in studying cellular biomechanics. To quantify and extract information from microscopy images and videos, image and video processing methods are required. Therefore, image and video processing methods are a central part of studying hiPSC-CM biomechanics and CVD.

The ever-increasing amount of imaging data requires high-throughput, fast, accurate and automated image, and video processing methods. The aim of this thesis is to provide a review of the state-of-the-art image and video processing methods and tools that can be used for studying hiPSC-CM biomechanics. The focus of this thesis is on machine learning -based and fully automated image and video processing methods. Moreover, this thesis acts as an introductory guide of the available methods and tools for researchers interested in image and video processing for studying hiPSC-CM biomechanics.

This thesis is divided into three main sections. In the first section the working principles of fluorescence microscopy and traction force microscopy are introduced in addition to their applications in studying hiPSC-CM biomechanics. In the second and third sections image and video processing methods and tools for fluorescence microscopy and traction force microscopy are introduced respectively.

Keywords: image processing, video processing, hiPSC-derived cardiomyocyte, biomechanics, machine learning

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

TIIVISTELMÄ

Anni Nieminen: Kuvan- ja videonkäsittelyn menetelmiä hiPSC-johteisten sydänlihassolujen biomekaniikan tutkimuksessa

Kandidaatintyö

Tampereen yliopisto

Bioteknologian ja biolääketieteen tekniikan koulutus

Toukokuu 2023

Sydän- ja verisuonisairaudet ovat maailmanlaajuisesti merkittävien kuolinsyy. Ihmisen indusoiduista pluripotenteista kantasolu-johteisista soluista (engl. human induced pluripotent stem cell, hiPSC) voidaan erilaistaa sydänlihassoluja. hiPSC-johteiset sydänlihassolut (engl. hiPSC-derived cardiomyocyte, hiPSC-CM) ovat osoittautuneet hyödyllisiksi sydänsairauksien mallinnuksessa. Niiden käyttöön liittyy useita etuja verrattuna esimerkiksi alkion kantasoluihin tai eläinperäisiin sydänlihassoluihin. Sydänsairaudet vaikuttavat sydänlihassolujen käyttäytymiseen ja niiden biomekaniikkaan. Tämän vuoksi sydänlihassolujen voimaantuottavien rakenteiden ja voiman tuottamisen tutkiminen on keskeistä sydänsairauksien mallinnuksessa ja uusien hoitojen kehityksessä.

Jotta soluja, niiden voimaa tuottavia rakenteita ja voimantuottoa voidaan tutkia, tarvitaan mikroskooppikuvantamista. Erityisesti fluoresenssimikroskopia ja siihen perustuvat muut kuvantamismenetelmät kuten traktiomikroskopia (engl. traction force microscopy, TFM) ovat osoittautuneet hyödyllisiksi hiPSC-CM-solujen biomekaniikan tutkimuksessa. Kuvista tai videoista saadun informaation kvantifoimiseksi puolestaan tarvitaan erilaisia kuvan- ja videonkäsittelyn menetelmiä. Tämän vuoksi kuvan- ja videonkäsittelyn menetelmät ovat keskeinen osa hiPSC-CM-solujen biomekaniikan ja sydänsairauksien tutkimusta.

Jatkuvasti kasvava kuvantamisdatan määrä vaatii suurikapasiteettisia, nopeita, tarkkoja ja automatisoituja kuvan- ja videonkäsittelyn menetelmiä. Siksi tämän kirjallisuuskatsauksen tavoitteena on esitellä sekä huippuluokan että uusimpia kuvan- ja videonkäsittelyn menetelmiä hiPSC-CM-solujen biomekaniikan tutkimukseen keskittyen automatisoituihin ja koneoppimiseen perustuviin menetelmiin. Lisäksi tämä työ toimii oppaana saatavilla olevista kuvan- ja videonkäsittelyn menetelmistä sekä työkaluista hiPSC-CM-solujen biomekaniikan tutkimuksesta kiinnostuneille tutkijoille.

Tämä työ jakautuu kolmeen pääosaan. Ensin lukijalle esitellään työn taustaa hiPSC-CM-soluista sekä kahdesta mikroskooppikuvantamismenetelmästä. Ensin käsitellään fluoresenssimikroskopian ja sitten traktiomikroskopian toimintaperiaatteita ja sovelluksia hiPSC-CM-solujen biomekaniikan tutkimuksessa. Seuraavassa osiossa perehdytään fluoresenssimikroskopiaalle soveltuviin kuvan- ja videonkäsittelyn menetelmiin ja työkaluihin keskittyen esikäsitelyyn, segmentointiin ja kvantitatiiviseen analyysiin. Kolmannessa osiossa esitellään traktiomikroskopiaalle soveltuvia kuvan- ja videonkäsittelyn menetelmiä ja työkaluja keskittyen traktiovoimien määrittämiseen.

Avainsanat: kuvankäsittely, videonkäsittely, hiPSC-johteiset sydänlihassolut, biomekaniikka, koneoppiminen

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck –ohjelmalla.

PREFACE

This thesis was written for the Computational Biophysics and Imaging Group at Tampere University. I would like to thank my supervisor Birhanu Belay for his support and guidance throughout the writing process. In addition, I am grateful to the Computational Biophysics and Imaging Group, especially group leader Jari Hyttinen, for granting me the opportunity to write this thesis.

Tampere, 3.5.2023

Anni Nieminen

CONTENTS

1. INTRODUCTION	1
2. BACKGROUND	3
2.1 hiPSC-derived cardiomyocytes	3
2.2 Bright-field imaging	5
2.3 Fluorescence microscopy imaging	6
2.4 Traction force microscopy	8
2.5 Image processing tools	10
3. IMAGE PROCESSING METHODS FOR FLUORESCENCE MICROSCOPY	11
3.1 Pre-processing.....	11
3.1.1 Deconvolution	11
3.1.2 Other pre-processing methods.....	13
3.2 Machine learning based segmentation methods	14
3.2.1 Convolutional neural networks	16
3.2.2 U-net.....	18
3.2.3 Mask R-CNN.....	21
3.2.4 YOLOv4.....	22
3.3 Quantitative analysis	24
3.4 Sarc-Graph	26
4. IMAGE PROCESSING METHODS FOR TRACTION FORCE MICROSCOPY ...	28
4.1 Traction reconstruction.....	28
4.1.1 Fourier transform traction cytometry.....	28
4.1.2 Other traction reconstruction methods	29
4.2 Available image processing tools	30
4.2.1 Traction reconstruction tools for ImageJ	30
4.2.2 Traction reconstruction tools for MATLAB	31
4.2.3 Bayesian Fourier transform traction cytometry	31
5. CONCLUSIONS AND FUTURE PROSPECTS	34
REFERENCES.....	36

LIST OF SYMBOLS AND ABBREVIATIONS

BEM	boundary element method
BFTTC	Bayesian Fourier transform traction cytometry
CM	cardiomyocyte
CNN	convolutional neural network
CVD	cardiovascular disease
ESC	embryonic stem cell
FEM	finite element method
FTTC	Fourier transform traction cytometry
hiPSC	human induced pluripotent stem cell
hiPSC-CM	human induced pluripotent stem cell-derived cardio- myocyte
iPSC	induced pluripotent stem cell
OOP	orientational order parameter
PIV	particle image velocimetry
PSF	point-spread function
R-CNN	region based convolutional neural network
ReLU	rectified linear unit
RoI	region of interest
TFM	traction force microscopy
3D	three-dimensional
\otimes	convolution

1. INTRODUCTION

Cardiovascular disease (CVD), including cardiomyopathies is the most common cause of death globally [1]. The treatment, medication, and mortality due to CVD causes a tremendous financial and health burden. For example, in the United States the direct cost of CVD was 251.4 billion US dollars in 2018 – 2019 [1]. Although research on CVD and new treatments have made progress, using traditional animal or cancer cell -based disease models have their limitations. Human induced pluripotent stem cell cardiomyocytes (hiPSC-CMs) in many ways provide a better alternative for studying CVD. [2]

Induced pluripotent stem cell (iPSC) technology was first introduced by Shinya Yamanaka and Kazutoshi Takahashi in 2006. It allows for any somatic cell to be artificially reprogrammed into an embryonic-like pluripotent state. The main advantages of iPSC-technology are that iPSCs can differentiate into any cell type, regenerate infinitely providing an endless supply of iPSCs and they contain the genome of the donor. Human induced pluripotent stem cells (hiPSCs) refer to iPSCs from a human donor, therefore containing a human-genome. Due to the previously mentioned advantages of iPSC-technology, hiPSCs have a wide range of applications and have allowed for revolutionary developments in biomedical research; they can be used *in vitro* disease models, drug discovery and testing. [3], [4]

The main function of the heart muscle is to pump blood and provide a sufficient oxygen supply to all organs via blood circulation. Therefore, biomechanics is one of the most central factors governing heart function. Several cardiomyopathies are known to affect the contractile forces exerted by cardiomyocytes (CMs), leading to abnormal heart function. Such conditions can be caused by morphological or orientational changes in the CM contractile structures. Morphology is the study of size, shape, and other structural features. In addition to cardiomyopathies, several drugs are known to affect heart function and CM contractile structures. Therefore, studying CM biomechanics is an integral part of heart disease research and developing new treatments. [3], [5], [6]

Most cells, including hiPSC-CMs, cannot be seen with the naked eye due to their microscopic size and therefore microscopy imaging is needed to examine them. This literature review provides a brief overview of some of the most common imaging methods that are

used to image hiPSC-CMs and study their biomechanical properties, namely fluorescence microscopy and traction force microscopy.

In order to perform quantitative analysis and extract useful information from images and videos, they often need to be processed in several ways. Image and video processing are the processes performed on images or videos to produce an enhanced version of them or to extract useful information from them. The aim of this literature review is to provide an overview of the newest and state-of-the-art image and video processing methods that can be used for studying hiPSC-CM biomechanics. Moreover, this thesis acts as an introductory guide on the available image and video processing methods and tools for researcher interested in studying hiPSC-CM biomechanics.

This literature review is divided into five chapters. In the second chapter the necessary background for the topic of this thesis is covered; hiPSC-CMs are introduced in depth, the working principles of fluorescence microscopy and traction force microscopy, and some of the most used image processing tools are introduced. In chapter three some of the most central image processing methods and tools suitable for fluorescence microscopy imaging, including pre-processing methods, segmentation algorithms and quantitative analysis. Section four focuses on the most central image processing methods and tools used to analyse traction force microscopy images, with a focus on traction reconstruction. Finally in chapter five conclusions are provided with discussion of future prospects.

2. BACKGROUND

In this chapter the necessary background for the topic of this thesis is provided. First hiPSC-CMs are introduced in depth. In addition, some of the most used imaging methods for studying hiPSC-CM biomechanics; bright-field-imaging, fluorescence microscopy and traction force microscopy are introduced, and their working principles are explained. Finally, some of the most used image and video processing tools in the biomedical field and related concepts are introduced.

2.1 hiPSC-derived cardiomyocytes

After iPSC-technology was first discovered by Shinya Yamanaka and Kazutoshi Takahashi in 2006, hiPSC-technology and the use of hiPSC-CMs have become increasingly popular in CVD research. Due to the many beneficial properties of hiPSCs such as their ability to continuously replicate and differentiate into cardiomyocytes, hiPSC-CMs have become useful for many applications in CVD research. hiPSC-CMs are used in CVD-models and researching new treatments for CVDs. In this chapter hiPSC-CMs will be introduced more in depth; the benefits and drawbacks of using them are discussed and they are compared to mature cardiomyocytes. In addition, their most central contractile structures and biomechanical properties are introduced. [3]

hiPSC-CMs can be reprogrammed from any somatic cells, most often skin cells, from a human donor. The donor cells are first treated with transcription factors that revert the cells back to an undifferentiated pluripotent state. This state is achieved, when the transcription factors influence, how DNA is packed and expressed in the cells. After this, the cells possess properties similar to those of embryonic stem cells (ESCs); they can differentiate into any cell type and divide endlessly. To obtain hiPSC-CMs from hiPSCs, a differentiation protocol that mimics the most central signalling pathways in embryonic cardiac specification must be followed. [2], [5]

Alternatives to using hiPSC-CMs include using animal cells, adult CMs, immortalized cell lines or using ESCs. In many cases hiPSC-CMs provide a better alternative compared to using other cell types. The main advantage of hiPSC-CMs compared to adult CMs is that they can reproduce indefinitely unlike non-proliferative adult CMs. In addition, mature CMs are known for being challenging to culture successfully. While hiPSCs are similar to ESCs in several ways, using hiPSCs involves far fewer ethical barriers than using

ECSs obtained from human embryos. Another advantage of hiPSC-CMs is that they contain the genome of the donor. Using animal cells in the research of human diseases has its limitations. Reaction to treatment can be species specific, which complicates using animal-based models in drug testing and disease modelling. An immortalized cell line that models key cardiac physiological properties has not been developed, causing immortalized cells to be a poor option in CVD research. Moreover, using hiPSC-CMs provides a way to obtain an unlimited supply of human cardiomyocytes that are phenotype specific, meaning that they contain the exact genome of the donor. This means that, using cells from a donor, who suffers from a CVD produces affected hiPSC-CMs, which can then be used to study the disease mechanism, research new treatments or even study personalized treatment options. [2], [5], [6]

Although hiPSC-CMs address many of the downfalls of using other cell types in CVD research, they have some drawbacks as well. One of the drawbacks of iPSC-technology in general is that a large proportion of the donor cells do not reach the pluripotent state and become induced pluripotent stem cells [5]. One of the major issues specifically with hiPSC-CMs is their immaturity. They tend to have an immature, embryo-like, physical structure, electrophysiological and biomechanical properties. Therefore, hiPSC-CMs do not provide a flawless model of the behaviour and properties of mature CMs, which limits their applicability in disease modelling and drug testing. [2]

The maturity of hiPSC-CMs depends on their culturing time. New ways of culturing hiPSC-CMs to allow the development of more mature structures and properties are currently under research. However, there are still many differences between mature CMs and hiPSC-CMs. Generally, hiPSC-CMs resemble embryonic or fetal cardiomyocytes both structurally and functionally. For example, their shape is more rounded compared to mature CMs, their contractile machinery is underdeveloped, and their calcium transients and electrophysiology differ. [2], [7]

Contracting is one of the main functional properties of all muscle cells including cardiomyocytes. Their contractile machinery refers to the cellular structures that take part in contraction. The main contractile structures in CMs, are sarcomeres comprised of actin and myosin filaments. A sarcomere is formed of band-like structures including m- and z-lines, giving CMs a striated-like appearance (Figure 1). Z-lines are responsible for connecting individual sarcomeres to each other forming structures referred to as myofibrils. In Figure 1, the contractile machinery of cardiomyocytes is illustrated.

The maturity of hiPSC-CMs is strongly related to the organization level of their contractile machinery. The immaturity of hiPSC-CM contractile machinery can be seen for example

in low myofibril density, low sarcomere, and myofibril orientation and poor z-line alignment. The unorganized nature of the hiPSC-CM contractile machinery leads to a significantly lower ability to produce contractile force compared to mature CMs. hiPSC-CMs are also known to contract in a spontaneous manner instead of in co-ordination with neighbouring cells like mature CMs. [2], [5]

In addition to maturity, other factors such as heart diseases and drugs are known to affect the contractile machinery and force production of CMs. They can affect the morphology or the alignment of the contractile machinery leading to impaired force production and heart function. For these reasons studying the biomechanical properties of CMs is an integral part of CVD research and developing new treatments. [3], [5], [6]

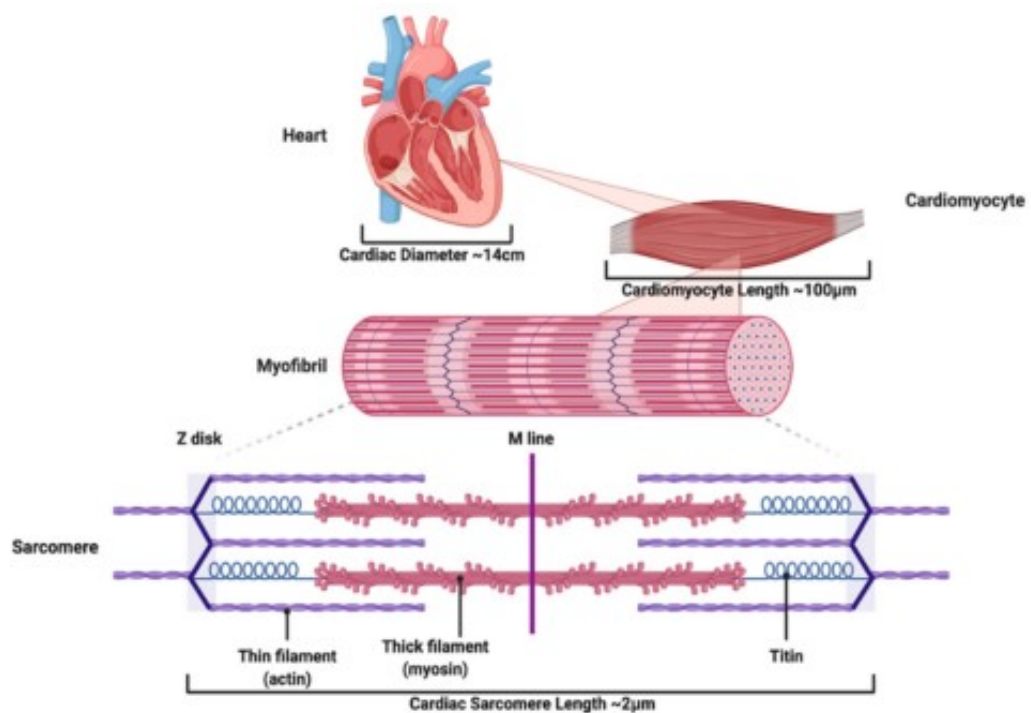


Figure 1. Illustration of cardiomyocyte contractile machinery showing sarcomere, myofibril, and z-line structures. Adapted from [9].

2.2 Bright-field imaging

Microscopy imaging is required for researchers to be able to view cells and analyse their properties and behaviour. Therefore, microscopy imaging is an integral part of CVD research. In this chapter one of the simplest methods of microscopy imaging called bright-

field imaging is introduced as background for the following imaging methods, fluorescence microscopy and traction force microscopy.

An optical microscope consists of two types of components: imaging and illumination components. It uses visible light and a collection of lenses to provide a magnified image of the specimen. Although an optical microscope allows cells to be magnified up to 1000 times, it has a limited resolution of 200 nm due to the diffraction limit of visible light. [5], [10]

Bright-field imaging is an illumination technique used in optical microscopy. In bright-field imaging the specimen is illuminated with white light from below. The light is transmitted through the sample and through an appropriate set of lenses to focus the image on to the detector. Contrast in the image is a result of attenuation of the light that has passed through the specimen. However, cells and cellular structures are usually transparent, which exhibits as poor contrast in the image. Due to a lack of contrast, it may be difficult to distinguish cells and cellular structures from one another and from the background. For this reason, dyes are often used to provide better contrast. Alternatively, other imaging methods developed to provide enhanced contrast such as fluorescence microscopy, can be used. A scheme of a bright-field microscope and some of its main components is shown in Figure 2. [5], [10]

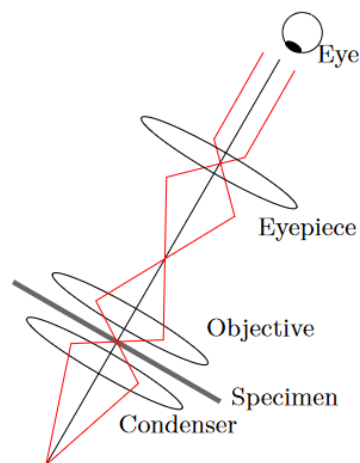


Figure 2. Scheme of a brightfield microscope and its main components. Adapted from [11].

2.3 Fluorescence microscopy imaging

Fluorescence microscopy imaging is a popular imaging method used in biological applications that addresses the lack of contrast in traditional optical microscopy such as

bright-field microscopy. Fluorescence microscopy is based on the use of fluorophores, which are chemicals that absorb light and as a result of excitation re-emit photons of a lower wavelength. Fluorescence microscopy imaging is especially useful when the goal is to image cellular structures with high accuracy and specificity. Therefore, fluorescence microscopy imaging has many applications in studying hiPSC-CM biomechanics. It is especially useful in analysing the morphological features of the cellular contractile machinery for drug testing, disease modelling and analysing the maturity of the cells. [5], [10]

A fluorescence microscope is similar to a traditional optical microscope with a few modifications. A mercury lamp is used to illuminate the sample with UV-light and excite the fluorophores. The UV-light falls on an excitation filter, which allows only specific wavelengths of light to pass through; the wavelengths that excite the fluorophores. Then the UV-light is reflected by a dichroic mirror onto the specimen. The fluorophores in the sample react to the UV-light and produce a fluorescent signal as a result of excitation. The signal passes through an emission filter, that only allows the wavelength of the fluorescent signal to pass through, before reaching the detector. A scheme of a fluorescence microscope showing its main components can be seen in Figure 3. [10]

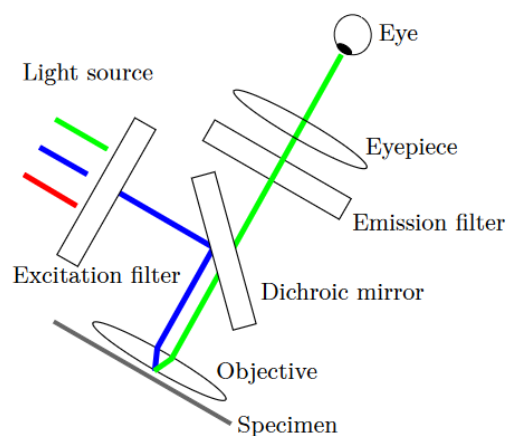


Figure 3. Scheme of a fluorescence microscope. Adapted from [11].

One way to further improve the resolution of fluorescence microscopy is to use confocal imaging (Figure 4). Confocal imaging is a form of fluorescence microscopy. It uses two pinholes to limit the excitation and detection beams of light in order to suppress any out-of-focus light. The final images are less out of focus and blurry than traditional fluores-

cence microscopy images and allow the detection of more detailed structures. In confocal imaging a laser is used to illuminate the sample instead of a mercury lamp. Another difference compared to so-called wide-field microscopy is that the entire sample cannot be viewed at once. To gain spatial information, various scanning methods are used; either the sample or the laser beam can be moved. Confocal imaging also allows the acquisition of three-dimensional (3D) images, when the sample is scanned in the z-plane in addition to x- and y-planes. Figure 4 illustrates the power of confocal microscopy by comparing a confocal image to a widefield image of the same specimen. [5], [10]

Another advantage of fluorescence microscopy imaging is its specificity. Fluorophores are often coupled to antibodies when imaging biomedical samples such as hiPSC-CMs. This allows the fluorophores to bind to certain molecules such as nucleic acids or proteins within the cells. This way different cellular structures can be viewed with high specificity and detail. [10]

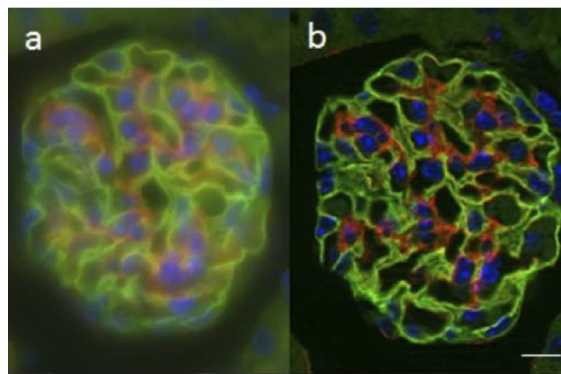


Figure 4. A comparison of confocal and widefield microscopy images of the same specimen to illustrate the power of confocal microscopy. a) Widefield imaging provides a much blurrier image that shows significantly less details. b) Confocal microscopy provides a sharper image with much more visible details. Adapted from [12].

2.4 Traction force microscopy

Traction force microscopy (TFM) is a microscopy imaging method that can be used to measure the contractile forces of a cell or multiple cells, called tractions. For this reason, TFM is a central imaging method used for studying cellular biomechanics. A traction force is a local force per unit area, that all adherent cells exert on their microenvironment. Tractions are related to several properties of the cellular microenvironment. The maturity of the cell and whether it is affected by a disease, or a drug may also affect the strength of tractions it exerts. TFM has several applications, it can for example be used to study

the mechanical properties of hiPSC-CMs in different microenvironments, at different maturity levels, when treated with a drug or affected by a heart disease. [13], [15]

Most often TFM is used to image a singular cell, but it is also possible to image a system of multiple cells together. The sample cell or cells are seeded on to a transparent substrate, such as a hydrogel, embedded with fluorescent beads. As the cells contract, they exert forces on the substrate and the fluorescent beads move as a result. This system is coupled to a fluorescence microscopy setup to track the movement of the beads in time and location. Computing the displacement field can be achieved based on comparing a “stressed state” image, where the cells are contracting and a reference image, where the cells have been removed from the substrate. From the measured displacement field the traction force field can be further solved. Figure 5 illustrates the possible outputs of a TFM image processing pipeline, including a displacement field and a traction intensity map. [14], [16]

In order to successfully compute the traction field, the mechanical characteristics of the substrate material must be well-known. Materials such as polyacrylamide and silicon-gels are often used as the substrate material. These materials are a popular choice because they exhibit linear-elastic behaviour, making traction field computation feasible. In addition, their mechanical properties are not known to be affected by biochemical factors. Their stiffness can also be widely varied depending on the goals of the study. [14], [16]

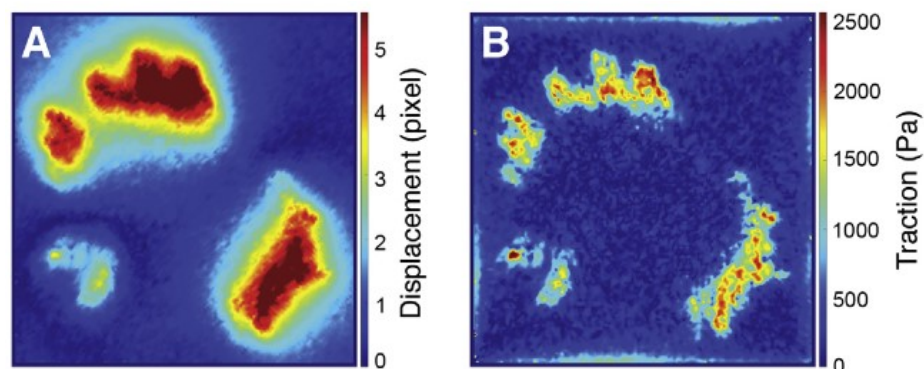


Figure 5. Example outputs of a TFM image analysis pipeline performed with a MATLAB-based software package, TFM, developed by DanuserLab (<https://github.com/DanuserLab/TFM>, accessed 8.4.2023) a) Displacement field measured in pixels. b) A map of traction intensities measured in Pascals. [16]

2.5 Image processing tools

Many types of tools have been developed for image processing. Here some of the most used tools in biomedical image processing will be introduced in addition to some related terminology. Some of the most popular image processing software for biomedical image processing include ImageJ and Fiji. Programming languages such as MATLAB, Python and Java are also used.

ImageJ is a Java based open-source image processing software designed for processing and analysing scientific images. It includes a variety of predefined functions but also allows the user to further extend the program's abilities with macros and plugins, which is one of the reasons it has gained much popularity in scientific image processing. [17]

A macro is a compact program containing a sequence of operations. A macro can be saved on a computer disk and be rerun at any time. In image processing a macro would include the operations performed on an image such as filtering or adjusting saturation. Macros provide an easy way to process a set of images identically and automate this process. [17]

Plugins are a way to further extend the capabilities of ImageJ. A plugin is a program that works in collaboration with another software, namely ImageJ. They allow the user to design their own image processing algorithms outside of the native capabilities of ImageJ. Fiji is a distribution of ImageJ that includes most of the core functionalities of ImageJ, in addition to many additional plugins. Instead of having to individually download plugins, Fiji provides a comprehensive bundle of them in one package. None of the plugins introduced in this thesis are readily available in Fiji. However all the ImageJ plugins including DeconvolutionLab2, U-net segmentation plugin SarcOptiM and TFM by Tseng et al. are available to be downloaded for both ImageJ and Fiji. [17]

Different programming or scripting languages such as Python, Java, JavaScript, and MATLAB can be used to create macros or plugins for image processing software such as ImageJ [17]. Moreover, programming languages such as Python and MATLAB can also be used on their own to design original image processing algorithms and software packages. In addition, namely MATLAB has built-in toolboxes and functions designed for image processing [18]. Many libraries containing image processing and machine learning functions have also been created for Python including OpenCV and Scikit-Image. Therefore, programming languages can provide a valuable way to extend existing image processing software with macros and plugins, but they can also be useful image processing tools on their own, providing endless ways to customize an image processing pipeline.

3. IMAGE PROCESSING METHODS FOR FLUORESCENCE MICROSCOPY

In this chapter some of the most integral and newest image and video processing methods and tools for fluorescence microscopy images, that can be used to study the hiPSC-CM contractile machinery and biomechanics, are covered. These methods include pre-processing methods, machine learning -based segmentation algorithms and quantitative analysis.

3.1 Pre-processing

Pre-processing methods are image processing operations that are applied before further processing. The goal is to enhance an image for further processing steps in order to guarantee more accurate results. In addition to making images more visually pleasing for the viewer, their quality should be made optimal for further analysis. The line between pre-processing and further processing steps is not perfectly clear and the definition may vary based on the source. In this thesis image processing steps applied before segmentation and quantitative analysis are considered pre-processing steps. There is no specific pre-processing pipeline that can be applied for all images or all imaging methods. However, the aim of this chapter is to briefly introduce some of the most common pre-processing operations used for fluorescence microscopy imaging keeping in mind the application of studying hiPSC-CM biomechanics.

3.1.1 Deconvolution

Deconvolution is a central image pre-processing method especially useful in fluorescence microscopy imaging including confocal imaging. Although confocal imaging produces higher resolution images than widefield microscopy, its resolution is still limited due to the diffraction limit of light. The microscope object lenses reduce the contrast of small features causing them to appear blurry and out-of-focus. In other words, the imaging components convolve the image. Deconvolution provides a way to further enhance the image resolution and counter the convolution effect caused by microscope components. The effect of convolution and deconvolution are illustrated in Figure 6. [10]

Convolution is a systematic error meaning that it is native to the imaging setup and the error is identical in images taken with the same setup. Convolution is characterized by the point-spread-function (PSF) of the optical system. The PSF of a system can either

be measured by imaging an ideal point-like light source or estimated with various methods including mathematical modelling. The final image H taken with an optical system is the convolution (\otimes) of the real image G and the PSF of the system with noise η added to it:

$$H = G \otimes PSF + \eta \quad . \quad (1)$$

When the PSF is known, its effect can be reversed with deconvolution. This step is crucial to obtain accurate results in further processing steps. As seen in Figure 6 deconvolution has a profound effect on the image resolution; the deconvolved image has much higher resolution showing more details compared to the same image without deconvolution. [10]

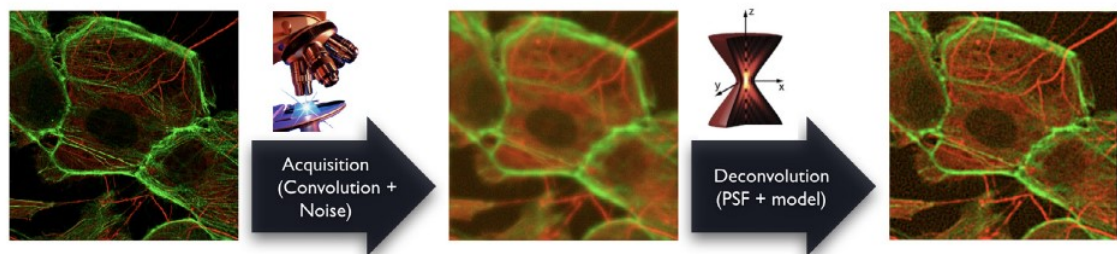


Figure 6. The effect of convolution and deconvolution on a fluorescence microscopy image. Convolution caused by the PSF of the imaging system causes the image to appear blurry. When deconvolution is performed on the image, its resolution is visibly increased. [19]

Solving Equation 1 in the Fourier frequency space is simple if the noise η is not accounted for. However, in real-world applications noise is always present in images. Therefore, more complex algorithms are required to perform deconvolution. Many types of deconvolution algorithms have been developed; including Wiener filtering, iterative algorithms such as the Richardson-Lucy algorithm and so-called blind deconvolution algorithms for when the PSF is unknown. The choice of algorithm depends on the user requirements and computational resources, since running complex algorithms especially for large 3D data sets can be computationally exhaustive in terms of memory usage and running time. One of the most prevalent deconvolution algorithms for confocal data include the Richardson-Lucy algorithm. [10]

Many deconvolution software packages including both commercial and open-source packages exist. For example, ImageJ has freely downloadable plugins specifically de-

signed for deconvolution such as DeconvolutionLab2 (<https://github.com/Biomedical-Imaging-Group/DeconvolutionLab2>, accessed 30.4.2023). It is an open-source Java-based software. It includes several different deconvolution algorithms such as the Richardson-Lucy algorithm and many others. It is compatible with ImageJ, Fiji, and MATLAB. An advantage of the software package is that it makes several algorithms available for the user to experiment with. Furthermore, extensive documentation of the software package is provided in an article by the developers. [19]

3.1.2 Other pre-processing methods

The aim of most pre-processing operations used for fluorescence microscopy images is to reduce noise and improve the foreground signal to make the images more visually appealing, guarantee accurate results in further processing steps and make a series of images more uniformly comparable. Some commonly used pre-processing steps in addition to deconvolution include background removal, enhancing contrast and applying filters. Most image processing software, including ImageJ and Fiji, have these functionalities integrated in them or available in the form of plugins [17]. Figure 7 illustrates the effect of using a few simple pre-processing operations on a fluorescence microscopy image of hiPSC-CMs with fluorescently stained z-lines. The pre-processed image is visibly enhanced compared to the original one. [8], [20]

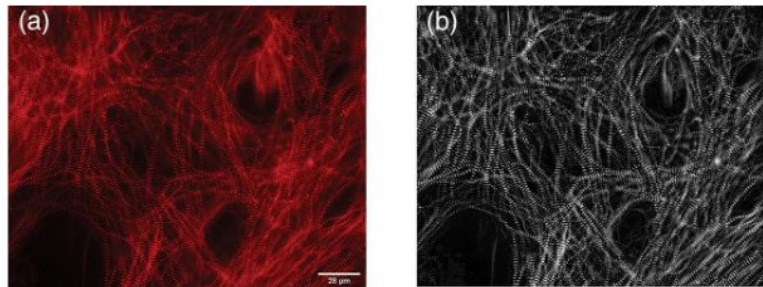


Figure 7. The power of pre-processing methods. a) An unedited fluorescence microscopy image of hiPSC-CMs with fluorescently stained z-lines. b) The same image after pre-processing including contrast enhancement and background removal. [8]

Contrast is the difference in luminance or hue. Contrast enhancing operations are used to increase the contrast between foreground and background signals and enhance the perceptibility of the fluorescently dyed structures such as z-lines. Furthermore, contrast enhancement can enhance a relatively weak fluorescent signal, making it easier to threshold for segmentation purposes. [8]

Background removal or subtraction is the operation of identifying the foreground and background signals and removing the background signal. This is often done because the background of the image is uneven, which can negatively affect further analysis steps. In short, background subtraction can be performed in order to remove low frequency signals in the background and enhance the fluorescent foreground signal. [8]

Some noise is always present in images. Noise is the presence of artifacts such as random brightness and colour variation in an image or video caused by camera components. Fluorescence signals are often low in intensity and noise can make it difficult to detect the fluorescent signal in the presence of substantial noise. Noise can be removed in a pre-processing operation called filtering. To remove noise, lowpass, blurring, filters are used. However, a lowpass filter will impair details due to its blurring effect. To achieve a satisfactory result, the filter should be adjusted in a way where it effectively removes noise but does not negatively impact details in the image. Moreover, detail-preserving low-pass filters such as the adaptive median filter can provide a good compromise between noise removal and detail preservation. [8], [10]

3.2 Machine learning based segmentation methods

After pre-processing has been completed, further processing steps such as segmentation can be conducted. Segmentation refers to the process of partitioning an image down into subgroups, which are referred to as image segments. It allows different structures to be identified from images. Practically any fluorescently dyed structure can be identified or in other words segmented from a fluorescent microscopy image. [21]

Segmentation is a crucial image processing step when studying hiPSC-CM biomechanics. When parts of the contractile machinery of hiPSC-CMs are fluorescently dyed, these structures can be identified from the images with segmentation algorithms and their properties can be further studied with quantitative analysis.

Many types of segmentation algorithms have been developed over time. Some of the simplest and most commonly used segmentation methods include thresholding methods such as Otsu's method. The disorganized nature of sarcomere structures in hiPSC-CMs provides an additional challenge to accurate segmentation [8]. In addition, increasingly large amounts of imaging data are continuously being acquired. Manual segmentation is burdensome, slow, and not always accurate. The huge amounts of data available require fast and highly accurate high-throughput segmentation methods. [8]

Deep learning methods, especially convolutional neural networks (CNNs) have proved to not only be fast but provide accurate segmentation results. CNNs have been used in

many types of segmentation tasks including biomedical applications such as cell and nuclei segmentation [22], [23]. A comprehensive summary of biomedical applications that the CNN architectures introduced in this thesis have been used for is presented in Table 1.

To the knowledge of the author, CNNs have not yet been used in the specific task of segmenting structures of the contractile machinery of hiPSC-CMs from fluorescence microscopy images. However, CNNs have potential to provide a faster, high-throughput and more accurate way to segment hiPSC-CM contractile structures. Using CNNs for this purpose has been mentioned by Cao et al. as a future step their group will explore [8]. In the following chapters the working principle of CNN-based segmentation and a few state-of-the-art CNN architectures that have potential to perform well in this task are introduced including U-net, Mask R-CNN and YOLOv4.

Table 1. A summary of biomedical applications U-net, Mask R-CNN and YOLOv4 CNN-architectures have been used for.

CNN architecture	Application	Imaging method	Reference
U-net	ISBI challenge 2012, segmentation of neuronal structures	electron microscopic stacks	[23]
U-net	ISBI challenge 2015, cell segmentation	light microscopy	[23]
U-net	2018 Kaggle Data Science Bowl cell nuclei segmentation contest	brightfield and fluorescence microscopy	[22]
U-net	Cell segmentation	2D fluorescence, differential interference contrast, phase-contrast, bright-field microscopy, and 3D bright-field image stacks	[24]
U-net	Neurite segmentation	electron microscopy stacks	[24]

CNN architecture	Application	Imaging method	Reference
U-net	Multiorgan segmentation	computed tomography	[25]
Mask R-CNN and U-net ensemble model	Nuclei segmentation	fluorescence, brightfield and histology imaging	[26]
Mask R-CNN	Skin lesion segmentation	dermoscopic imaging	[27]
Mask R-CNN	Multiorgan segmentation	computed tomography	[28]
Mask R-CNN	Breast cancer detection, segmentation, and classification	ultrasound	[29]
YOLOv4	Melanoma lesion detection	dermoscopic imaging	[30]
YOLOv4	Pulmonary nodule detection	computed tomography	[31]

3.2.1 Convolutional neural networks

Convolutional neural networks (CNNs) are a deep learning architecture that has proven to be especially useful in image processing and specifically object identification and image segmentation. CNNs were first introduced in the 1990's. CNN architecture was inspired by the visual perception of living creatures. They make highly accurate object identification and image segmentation possible with little to no image pre-processing. Originally CNNs did not provide satisfactory results because of a lack of available training data and computing power. However, in the past decade CNNs have gained traction due to an increase in these resources. [32]

Usually three types of basic components can be found in a CNN: convolutional, pooling, and fully connected layers. Each layer consists of several nodes also known as neurons. Neurons are connected amongst each other and between layers. Each neuron has a threshold value called an activation value. If the output of the neuron exceeds its activation value, it activates and sends information to the next layer. [32]

The purpose of convolutional layers is to learn features of the input image. An individual convolutional layer is formed of multiple convolution kernels, which compute feature maps. One kernel can for example detect vertical lines in the input image and another horizontal lines. As the depth of the network increases the feature detection becomes progressively sophisticated detecting increasingly abstract features. For example, going from detecting shapes, to detecting facial features, an entire face and finally identifying a person in the input picture. [32]

The output feature maps of convolutional layers are sensitive to the location of the feature in the input image. To address this and reduce the computational power required for further operations, feature maps are downsampled meaning that their resolution is lowered. Pooling layers are responsible for the downsampling. They either perform average or maximum pooling. In short average pooling summarises the average presence of a feature whereas maximum pooling summarises the most activated presence of a feature. [32]

After multiple consecutive convolutional and pooling layers there may be a fully connected layer. A fully connected layer is usually the last layer in a CNN before an output layer. Its purpose is to execute high-level reasoning, which is achieved by connecting all the neurons of the previous layer to its own neurons. However, a fully connected layer is not necessary since it can be replaced by a 1×1 convolutional layer. The final layer of a CNN is the output layer, which produces the output of the network. The output can be for example a classification prediction, or in a segmentation task it can be a segmentation mask alongside classification predictions. [32]

Two types of CNN-based image segmentation methods exist: semantic segmentation and instance segmentation. Semantic segmentation classifies every individual image pixel into one or more classes, each being a real-world object. Instance segmentation on the other hand identifies every instance of an object in the image and separates it with a bounding box. In addition, it classifies each individual pixel within these boxes. The main difference between instance segmentation and semantic segmentation is that instance segmentation is able to separately identify and separate each instance of an object. [33]

Object detection algorithms can also be used in image segmentation to identify regions of interest for further segmentation. Object detection is a simplified version of instance segmentation. It does not produce a segmentation mask as an output, instead it only produces bounding boxes around recognized objects and the predicted class for each object. An illustration of the different image classification techniques is shown in Figure 8. [30], [33]

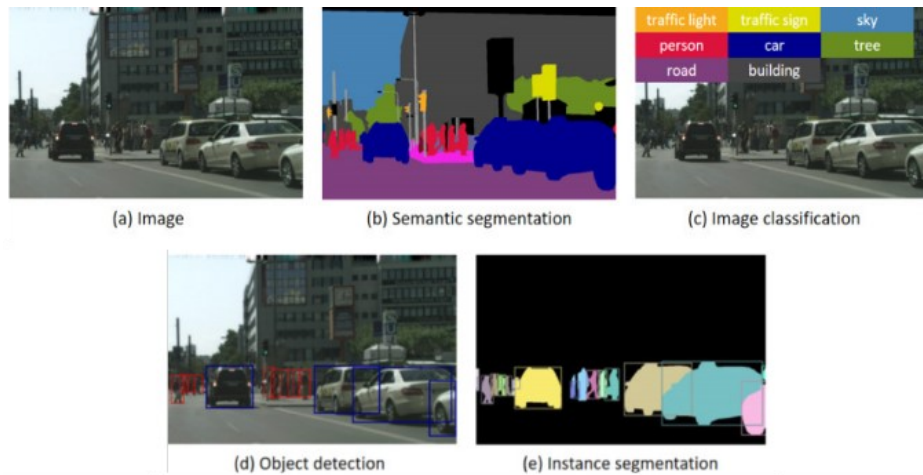


Figure 8. Illustration of different image classification and segmentation types. a) The original image has not been classified. b) Semantic segmentation classifies each individual pixel. c) Image classification assigns one or more class labels to the input image. d) Object detection identifies each object and marks its location with a bounding box. e) Instance segmentation identifies each instance of an object in the input image, marks its location with a bounding box and classifies the object pixels. Adapted from [33].

For studying the contractile machinery of hiPSC-CMs both instance and semantic segmentation algorithms have the potential to produce highly accurate segmentation masks. For segmentation tasks, where identifying each individual fluorescently dyed structure is necessary, instance segmentation algorithms would provide a viable option. Object detection algorithms could be used for example to detect individual z-line structures as regions of interest for further image segmentation. Although, CNN-based segmentation methods have shown great potential in many biomedical applications, it should also be noted that these algorithms have not yet been explored for hiPSC-CM contractile structure segmentation. Therefore, further research is required to compare their performance to traditional segmentation algorithms and amongst each other in this task. [32]

3.2.2 U-net

U-net is a semantic segmentation CNN architecture designed for biomedical applications including several types of microscopy imaging data such as bright-field and fluorescence microscopy images. It was developed at the University of Freiburg and published in 2015. The source code including the original trained networks for U-net are freely available online. In addition, a freely available ImageJ plugin for cell detection, segmentation and counting with U-net has been published (<https://sites.imagej.net/Falk/plugins/>, accessed 30.4.2023) [24]. U-net was developed to allow accurate image segmentation with a low

amount of training data, which is made possible by using data augmentation techniques. U-net has since performed well in many types of segmentation tasks. In 2015 it won the ISBI cell tracking challenge with a large margin. In addition, an enhanced U-net -based CNN architecture outperformed other implementations in the 2018 Kaggle Data Science Bowl cell nuclei segmentation contest [22]. [23]

U-net forms a U-shaped network consisting of a contracting and expansive path also known as encoder and decoder structures. The left, contracting, path receives the input image and extracts useful information from it using convolutional layers. The right, expansive, path upsamples feature maps by performing up-convolutions. After this a concatenation with the corresponding feature maps from the contracting path follows in a process known as skip-connections. Finally, as an output the network produces a segmentation mask. [23]

When unpadded convolutions are used, the output image is smaller than the input image and therefore some data is lost in the process. To avoid the loss of data, a technique called overlap tile strategy is used. The pixels of the entire image are classified tile by tile, where a tile is a small part of the input image. At the image borders the tiles are extrapolated by mirroring the image as illustrated in Figure 10. [23]

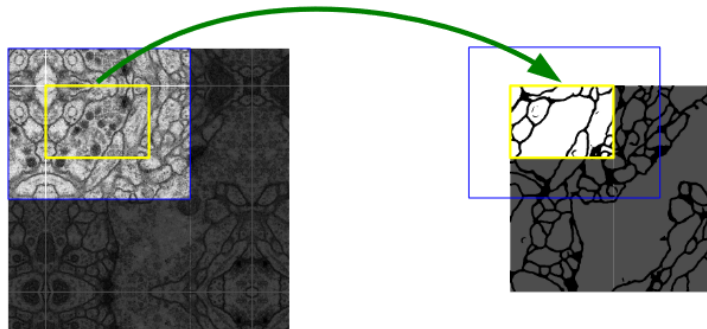


Figure 10. Illustration of the overlap tile strategy. To perform segmentation in the area highlighted by a yellow box, the pixels in the blue box (input image tile) are required as an input to not lose data in the segmentation process. [23]

The left, contracting, path is a classic convolutional network consisting of recurrent 3×3 convolutions. After each convolution there is a rectified linear unit (ReLU). An ReLU is an activation function, that introduced nonlinearity to the network and helps it learn complex features. An activation function takes the output from a node and turns it into an input for the next node. The ReLU returns the output if it is positive and otherwise it returns zero.

The ReLu is followed by a 2 x 2 maximum pooling operation to down-sample the input image. The amount of feature channels is doubled in every down-sampling operation. [23]

In the expansive path each step is comprised of an up-sampling of the feature map, which is followed by a 2 x 2 up-convolution, each halving the amount of feature channels. After this a skip-connection follows. Skip-connections take a feature map from the contracting path and feed it straight to the corresponding part of the expansive path. In other words, skip-connections take features learned in the contracting path and use them to perform localization in the expansive path. Each skip-connection is followed by two 3 x 3 convolutions and an ReLu. The last layer consists of a 1 x 1 convolution for mapping each feature vector to a class. In total U-net incorporates 23 convolutional layers. The U-net network architecture is illustrated in Figure 9. [23]

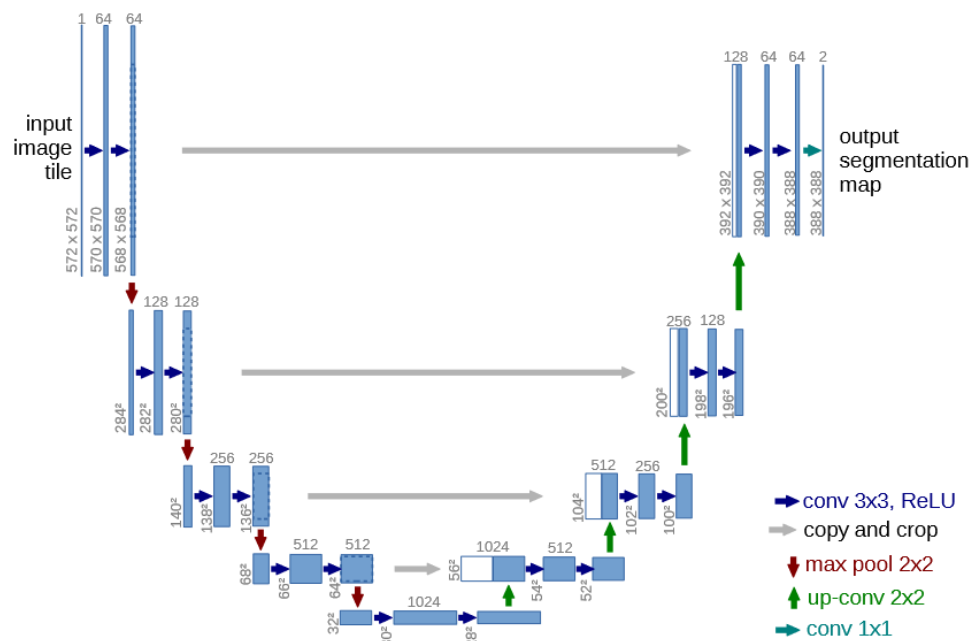


Figure 9. U-net network architecture. The x- and y-size of the image tile is marked at the bottom of the blue boxes. Each of the blue boxes depict a feature map with several channels. The number of channels is marked on the top side of every box. White boxes on the other hand depict copied feature maps and arrows depict each of the operations. [23]

3.2.3 Mask R-CNN

Mask R-CNN is an instance segmentation CNN architecture. It was not initially designed for biomedical applications, but it has been widely used in the biomedical field for example for skin lesion segmentation and nuclei segmentation [26], [27]. It has gained much popularity due to its high accuracy, simplicity, and efficiency. It is able to recognize each instance of an object or several different types of objects in an image, separate them with bounding boxes and produce high-quality segmentation masks. Mask R-CNN was first published in 2018. The Python implemented source code is freely available online. [34]

Mask R-CNN has been used to detect skin lesions in a three-stage segmentation pipeline with highly accurate results, outperforming previously used state-of-the-art lesion detection methods [27]. In addition, a network combining Mask R-CNN and U-net has been used for nuclei segmentation, where the predictions of both U-net and Mask R-CNN were used as input in an ensemble model. The ensemble model had a slightly higher precision of 1 % compared to the results obtained with U-net and Mask R-CNN separately. [26]

To find each individual instance of an object in an image, several candidate object locations also known as proposals must be processed and finally refined when an object is found. The amount of region proposals also known as regions of interest (RoI) may end up being exceedingly high causing this approach to be computationally demanding and not realistic in most real-world applications. [35]

To reduce the amount of selected Rols, region based convolutional neural networks (R-CNNs) were introduced. R-CNNs are based on selective search that is used to extract a certain number, up to 2000, proposal regions from the input image. These regions are reshaped and merged together, forming a smaller image and then fed into a CNN that performs object classification within the identified Rols. Although, an R-CNN only considers approximately 2000 Rols, it is still rather slow. [35]

To produce classification and detection results faster, an algorithm called Fast R-CNN was developed. Fast R-CNN is similar to R-CNN in many ways. However, instead of having to feed 2000 region proposals to the CNN, in the Fast R-CNN architecture an entire image is used as input for the CNN network. The input image is processed within the network using convolutional and pooling layers, finally producing a feature map as an output. The feature maps of corresponding Rols, computed with the selective search algorithm, are reshaped in a RoI pooling layer. This is done so that feature maps can further be fed into a fully connected layer. Finally, as an output the network produces class predictions and offset values for every individual RoI. The offset values are the four

coordinates determining the location, in other words the bounding box, of a recognized object. [36]

An algorithm called Faster R-CNN was developed to avoid the slow process of selective search used in both R-CNN and Fast R-CNN. Similarly, to Fast R-CNN an entire image is fed into a CNN that provides a feature map as an output. However, the most significant difference between Fast R-CNN and Faster R-CNN is that, instead of using the selective search algorithm, another CNN, called a region proposal network is used to predict region proposals. [37]

Mask R-CNN was built as an extension of Faster R-CNN. The main difference between Mask R-CNN and the previous R-CNN-based algorithms is that Mask R-CNN produces a binary segmentation mask for every RoI in addition to the class predictions and bounding box off-sets. The previous R-CNN version Fast R-CNN performed coarse spatial quantization within the RoI pooling layer, causing inaccuracies in the outputs. To address this problem, Mask R-CNN introduced a quantization-free layer called RoI-align, that improved mask accuracy up to 10-50 %. The Mask-RCNN framework is illustrated in Figure 11. [34]

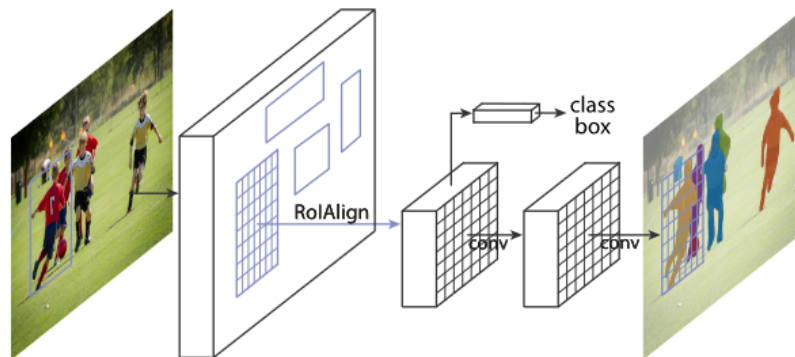


Figure 9. An illustration of the Mask R-CNN network architecture. [34]

3.2.4 YOLOv4

YOLOv4 is an object detection CNN architecture belonging to the YOLO-family. It is a continuation of the previous YOLO-version, YOLOv3. The name “YOLO” comes from the phrase “You only look once”, due to the one stage detection approach used in YOLO-algorithms contrary to the two-stage object detection approach used for example in Mask R-CNN. YOLO was not originally developed for biomedical applications. However, it has produced promising results in biomedical image segmentation. For example, in 2020 it

was used to detect melanoma lesions in a three-stage image segmentation pipeline with a higher accuracy than state-of-the-art CNN-based methods [30]. YOLOv4 was published in 2020 and open-source code for the implementation is freely available online. [38]

The main principle of YOLO-algorithms is that an entire input image is fed into the network, where one CNN is used to predict bounding boxes and class probabilities. The input images are split by YOLO into an $S \times S$ grid. Within each grid cell, N bounding boxes are predicted in addition to confidence for the boxes and C class probabilities. Each grid cell detects the objects, which centre is located in that cell. The confidence value expresses, how likely it is that an object is found in that bounding box and how accurate the box is. Only one set of class probabilities C is predicted for each grid cell even if multiple bounding boxes are located in that cell. Final predictions are based on the computation of a class-specific confidence score for every bounding box. An illustration depicting the working principle of YOLO image detection methods can be seen in Figure 12. [38]

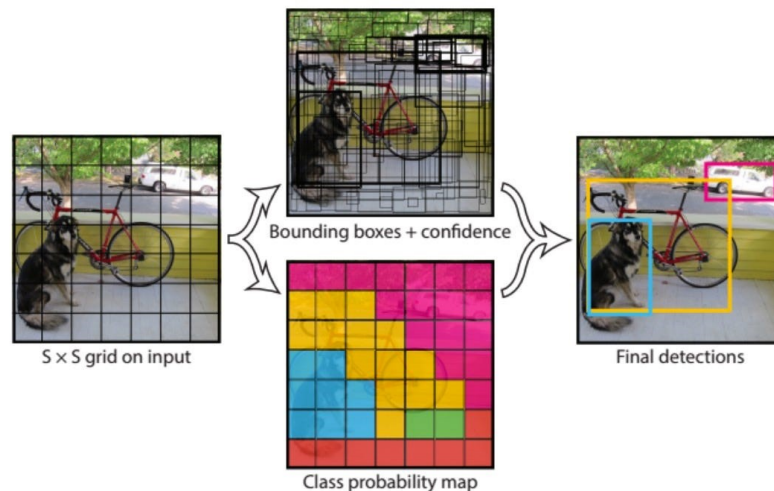


Figure 10. An illustration of the working principle of YOLO image detection. [39]

An advantage of YOLO algorithms is their speed. Another advantage of YOLO algorithms compared to region proposal -based algorithms is that they mistake fewer background areas as objects, due to their one stage detection approach. In addition, they are highly generalizable making them easy to apply to new types of inputs. However, due to the spatial constraint of the algorithms they have difficulties in localizing all objects, especially ones that are small in size. [39]

The most significant difference in YOLOv4 compared to YOLOv3, the previous YOLO version, is that it can be trained using a single GPU, making the algorithm more accessible. In addition, the YOLO-model was made more robust by applying a set of data augmentation techniques called “bag of freebies” and “bag of specials” by the authors. Bag of freebies -methods are data augmentation methods, which are applied to the training data set. These methods include for example scaling, flipping, rotating, adding noise, and adjusting saturation. The purpose of these methods is to increase variation in the training data and thus improve the model robustness. Bag of specials techniques on the other hand refer to post-processing methods that significantly increase object detection accuracy. [38]

3.3 Quantitative analysis

After the structures of interest have been segmented, further analysis such as phenotype measurements can take place. Different measurements are used for different purposes such as studying the maturity or cardiotoxic effects on the contractile machinery of hiPSC-CMs. In this chapter some of these quantifiable parameters are briefly introduced.

In 2015 Pasqualini et al. proposed 11 different parameters to describe sarcomere structure and combined them with machine learning algorithms to measure cardiomyocyte maturity in mature CMs as well as hiPSC-CMs. These metrics include sarcomere length, total energy, sarcomeric energy, sarcomeric packing density, orientational order parameter (OOP), sarcomeric OOP, non-sarcomeric OOP, z-discs relative presence, weighted OOP, coverage quality control and coherency quality control. OOP is a parameter ranging from 0 to 1. Its value increases as the sarcomeric organization increases. The main principle of determining these structural features is based on using histogram operations and Fourier transformation. Seven out of the eleven metrics developed by Pasqualini et al. are illustrated in Figure 13. To obtain more detailed descriptions of these parameters the reader is referred to the publication by Pasqualini et al. 2015. [40]

Metrics and meaning	Low value	High value
1. Sarcomere length : What is the distance between two adjacent Z-disks?		
2. Total Energy : How much non-homogeneous signal is there in the immunograph?		
3. Sarcomeric Energy : How much immunosignal is spaced a sarcomere length apart?		
4. Sarcomeric Packing Density : What fraction of the total signal is periodically arranged?		
5. Orientalional Order Parameter (OOP) : How well-oriented are the cytoskeletal elements?		
6. Sarcomeric OOP : How well-oriented are the Z-disks?		
7. Non-sarcomeric OOP : How well-oriented are the other cytoskeletal elements?		

Figure 11. Illustration of seven out of the eleven sarcomere structure metrics developed by Pasqualini et al. in 2015. Adapted from [40].

In 2018 a software package for studying sarcomere structural organization called SOTA was developed. SOTA stands for “SarcOmere Texture Analysis”. It is based on using algorithms known as Haralick texture features used for measuring image texture based on pixel-intensity. The pixel-based approach of SOTA was used to reduce potential bias in previous methods, including Pasqualini et al. 2015, where sarcomere structures were analysed based on manually selected rectangular sub-regions of images. The MATLAB-based implementation is freely available on the developer’s GitHub page (<https://github.com/saucermanlab/SarcOrgTextureAnalysis>, accessed 30.4.2023). [41]

Another MATLAB-based, freely available, computational analysis tool for segmenting and studying sarcomeric z-lines, called ZlineDetection, was published by Morris et al. in 2020 (<https://github.com/Cardiovascular-Modeling-Laboratory/zlineDetection>, accessed 30.4.2023). ZlineDetection allows the fully automated segmentation and analysis of muscle sarcomeric z-lines. Moreover, ZlineDetection provides several different metrics for studying z-line architecture including z-line length, z-line OOP and z-line fraction. [42]

In 2019 another open-source MATLAB-based software tool called SarcTrack was published. Sarc-Track is available on the developer’s GitHub-page (<https://github.com/HMS-IDAC/SarcTrack>, accessed 30.4.2023). For the first time it allowed the tracking of individual fluorescently dyed hiPSC-CM sarcomeres based on video footage. Moreover, SarcTrack provides segmentation and several analysis functionalities. It can be used for example to determine sarcomere displacement between consecutive image frames as

well as sarcomere contraction and relaxation parameters. The SarcTrack algorithm is based on modelling sarcomeres using so-called Morlet wavelets. The developers successfully used SarcTrack to assess the effects of compounds known to alter hiPSC-CM contractile function. [43]

Many other tools and measures for the quantitative analysis of hiPSC-CM contractile structures exist in addition to the ones mentioned in this chapter. For example, an ImageJ plugin called SarcOptiM allows the measurement of sarcomere length using Fourier transformation [44]. Also for example in an automated image processing pipeline for sarcomere structure analysis to study cardiotoxicity in hiPSC-CMs published in 2022 by Cao et al., metrics such as z-line fraction area were used. [8]

3.4 Sarc-Graph

Sarc-Graph is one of the most recent developments in quantitative analysis tools for analysing fluorescence microscopy images and videos of hiPSC-CM. It is an open-source software developed for segmentation, tracking and analysis of sarcomeres specifically in hiPSC-CMs. It was developed to analyse video microscopy footage of hiPSC-CMs with fluorescently labelled z-lines. It allows the tracking and analysis of individual sarcomere structures instead of providing a single average metric per image. In addition, it provides analysis tools beyond the scope of previous sarcomere tracking software SarcTrack. The goal of Sarc-Graph is to provide a way to study the link between hiPSC-CM morphology and contractile function. Sarc-Graph was published in 2021 and its Python-based implementation has been made freely available online (<https://github.com/elejeune11/Sarc-Graph>, accessed 30.4.2023). [45]

Sarc-Graph includes functions for segmenting sarcomeres and z-lines and performing quantitative analysis. In addition, it can be used to track z-lines and sarcomeres in actively beating hiPSC-CMs. Sarc-Graph also provides tools for data visualization. Furthermore, it allows fully automated quantitative analysis of the contractile behaviour of hiPSC-CMs similarly to TFM-based approaches. However, unlike TFM it does not allow direct measurement of contractile forces. One of the main advantages of Sarc-Graph compared to other analysis software mentioned in the previous chapter is that it allows the tracking of changes in the motion and length of single sarcomeres between image frames. In addition, Sarc-Graph can quantify the direction and magnitude of average cell contraction. [45]

Sarc-Graph allows the user to perform segmentation, tracking and analysis functions without much knowledge about the software or underlying algorithms. On the other hand,

more advanced programmers can use the freely available open-source code and make modifications depending on their needs. Due to the modularity of the software, it is easy to pick and choose different functionalities such as segmentation and use it separately or modify it. An interesting option could be using CNN-based segmentation methods and combining them with Sarc-Graph tracking and analysis functions. Overall, Sarc-Graph provides accurate segmentation results with a short runtime and little to no parameter tuning but supports user-made modifications making it a suitable option for both beginners and experts in the field of image and video analysis. [45]

4. IMAGE PROCESSING METHODS FOR TRACTION FORCE MICROSCOPY

In order to study the forces hiPSC-CMs exert on their environment, TFM-based methods are required. The main image processing challenge in TFM is determining the traction force field based on the measured deformation field. In this chapter a few methods to solve this problem are introduced in addition to available image processing tools. For a more detailed review of TFM imaging methods and image processing methods the reader is referred to the literature review by Zancla et al. 2021 [16].

4.1 Traction reconstruction

4.1.1 Fourier transform traction cytometry

When using a deformation assay in TFM, the traction field can be solved based on the displacement field of fluorescent beads caused by cellular force production. Together with the knowledge of the mechanical properties of the substrate material, namely its Young's modulus and Poisson's ratio, the traction field can be solved using the inverse Boussinesq solution. An issue with this approach is that the displacements of fluorescent beads are not only determined by local tractions. This is because, point-like traction forces are known to cause displacements in a wide spatial area. This causes the Boussinesq solution to become an ill-posed inverse problem, where any errors in the displacement field will cause a magnified error in the traction force field, resulting in having to conduct computationally tedious matrix operations. [14], [16]

In the inverse problem cell traction forces t are described in terms of the known displacement field u . The displacement field u is the convolution of Green's function G and traction force t . Green's function describes the impulse response of the system as a result of a point force. The problem can be solved using a convolutional Fredholm integral as follows:

$$u(r) = G \otimes t = \int G(r - r')t(r')dr', \quad (2)$$

where u is the displacement at $r = (x,y,z)$ and t is the traction acting at $r' = (x',y',z')$. [14], [16]

To produce a faster and more computationally efficient solution, Butler et al. developed a method called Fourier transform traction cytometry (FTTC) and were the first to find a solution to the Fredholm integral [46]. FTTC is based on the realization that Equation 2

can be solved in the Fourier frequency space utilizing convolution theorem. According to convolution theorem the Fourier transform of a convolution becomes a product, allowing the Fredholm convolutional integral to be solved in the Fourier space as follows:

$$T(k) = \frac{U(k)}{G(k)},$$

where T , U and G are the Fourier transforms of the traction, displacement, and Greens' function, respectively. k denotes the spatial frequency. A traction force field solved with FTTC by Butler et al. is illustrated in Figure 14. [14], [16], [46]

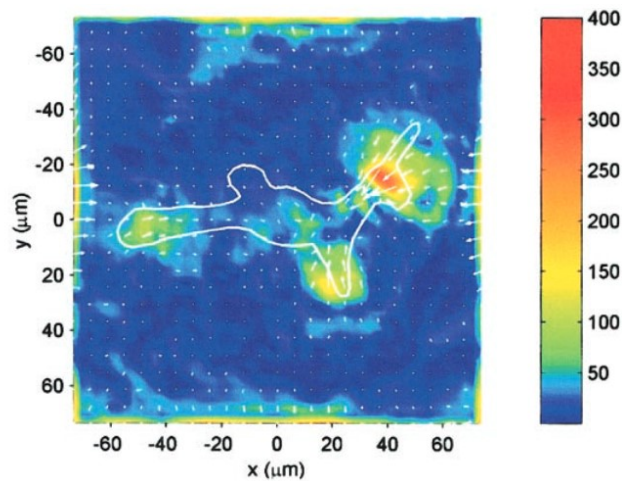


Figure 12. Traction force field computed with FTTC by Butler et al. based on a displacement field of fluorescent beads in TFM. The white line highlights the borders of a cell. Arrows indicate the relative magnitude and direction of tractions. Background colours illustrate the traction magnitude in Pascals. [46]

FTTC has become the most popular way to solve the traction field -inverse problem, for which it provides a computationally efficient solution. However, FTTC is easily affected by noise in the displacement field. Different denoising and regularization strategies have been introduced to counter this problem. These methods aim to remove as much noise as possible, while leaving the traction signal untouched. An optimal regularization parameter can be selected for example based on the Bayesian theory or graphically from a so-called L-curve. [14], [16]

4.1.2 Other traction reconstruction methods

Several other methods for solving the traction force field have been developed in addition to FTTC. These methods include for example boundary element method (BEM), finite element method (FEM) and micro patterning.

BEM is a numerical method for producing solutions to integral equations. The principle of using BEM to solve Equation 2 is that the integral is discretised using a computation mesh with nodes close enough to perform interpolation. The performance of BEM surpasses that of FTTC but the computational load of performing BEM is higher, while both methods are prone to misjudging tractions at adhesion sites, estimating them to be lower than they really are. [16]

FEM is another numerical technique that can be used to find approximate solutions to integral equations and partial differential equations. FEM attempts to break the problem down into ordinary differential equations to simplify it. These equations are solved using numerical integration. The problems are solved locally in a finite number of subregions, called elements. An advantage of FEM is that it works well with complex geometries. Therefore, it has been used especially in 3D-TFM applications. [16]

In micro patterning the substrate used in TFM is marked with fluorescent proteins. These dot markers can be used to solve displacements and traction forces. In micro patterning the forces can be modelled with the displacement of singular points, significantly simplifying the solution of traction forces. It allows the automated real-time measurement of tractions. In addition, the original MATLAB code for this method has been made open-source and freely available for researchers to use [47]. [16]

4.2 Available image processing tools

Several image processing tools including software packages, plugins and macros specifically designed for traction reconstruction, have been made available. In this chapter some of these tools for ImageJ and MATLAB are introduced.

4.2.1 Traction reconstruction tools for ImageJ

Tseng et al. published an image processing plugin series for ImageJ allowing the entire traction reconstruction process to be executed in ImageJ. It has been made freely available on the developer's GitHub-page (https://github.com/qztseng/imagej_plugins). The package consists of three plugins called Template matching, PIV and FTTC. [48]

The template matching plugin executes alignment between the “stressed-state” and reference images to compensate for possible experimental inaccuracies. The displacement field is computed iteratively using a method called particle image velocimetry (PIV). From which the traction force field is solved using the previously described FTTC-method. The FTTC-plugin requires the displacement field solved by PIV, substrate mechanical parameters, image spatial calibration data and a regularization parameter as input. [16], [48]

The traction reconstruction tool by Tseng et al. does not allow the user to select a custom regularization strategy, nor does it direct the user in regularization parameter selection. To automate the calculation process of the plugins by Tseng et al. and solve the mechanical energy accumulated in the substrate gel due to deformation, a macro for ImageJ was later published in a detailed how-to guide. [49]

4.2.2 Traction reconstruction tools for MATLAB

Several TFM image processing tools have been developed in MATLAB. Danuser et al. published a freely accessible software for displacement field measurement and traction reconstruction (<https://github.com/DanuserLab/TFM>). The displacement field is quantified with image-based tracking. Traction is determined based on deformation of the substrate gel in reference and stressed states. A regularization method called L1-regularization is used by default. Furthermore, the software supports BEM and FTTC traction reconstruction algorithms based on L2-regularization. The software tool allows creation of traction magnitude heatmaps and traction vector fields. [16] No article has been published by Danuser et al. describing the software tool, but the algorithms used by Danuser et al. are described in detail in a publication by Han et al. [50]. In addition, software tools for analysing three- and four-dimensional TFM have also been developed for MATLAB [51], [52].

4.2.3 Bayesian Fourier transform traction cytometry

Recently in 2020 Huang et al. published a MATLAB software package for the reconstruction of traction forces. The software package is freely available online (https://github.com/CellMicroMechanics/Easy-to-use_TFM_package, accessed 30.4.2023). It is described in detail in the article published by Huang et al., which includes a user manual for the software package in the supplementary materials. The aim of Huang et al. was to create an easy-to-use software for traction reconstruction that allows automated noise removal. Therefore, detailed knowledge of the algorithmic or mathematical background of traction reconstruction is not necessary for the user. The graphical user-interface is shown in Figure 15, illustrating its simplicity. [53]

The method for traction reconstruction developed by Huang et al. was named Bayesian Fourier transform traction cytometry (BFTTC). BFTTC uses FTTC for traction reconstruction and Bayesian regularization for denoising, making the algorithm both computationally fast and robust. The software offers the option of using standard L2 regularization, where the user must provide a regularization parameter as input, in addition to BFTTC

regularization, which automatically determines the optimal regularization parameter based on the displacement data and noise. The Bayesian regularization method proposed by Huang et al. was adapted from real-space Bayesian regularization to the Fourier space in order to achieve higher robustness and computational speed. Readers interested in the mathematical background of the method are referred to the article by Huang et al. [53]

The program requires deformation data as input. The deformation data can be determined using other software packages. The standard deviation of noise can be provided as an input parameter to the program. Alternatively, the program can determine this parameter based on a user-defined image area. Mechanical properties of the substrate material including Young's modulus and Poisson's ratio also need to be provided as input to the program. A user guide for more detailed instructions for using the program is provided by Huang et al. in the supplementary materials of their article. [53]

BFTTC was tested using both experimental and synthetic data showing that BFTTC can handle large measurement noise. Although the accuracy of the method was not improved by a large margin compared to other L2 regularization-based methods, it made the automated selection of the regularization parameter possible. Furthermore, BFTTC proved to be a computationally fast method, only taking seconds to a few minutes to be completed. BFTTC combines the benefits of accuracy, fast computation time and robustness to an easy-to-use user interface, making it a great option for researcher not well-versed in the mathematical background of traction reconstruction. [53]

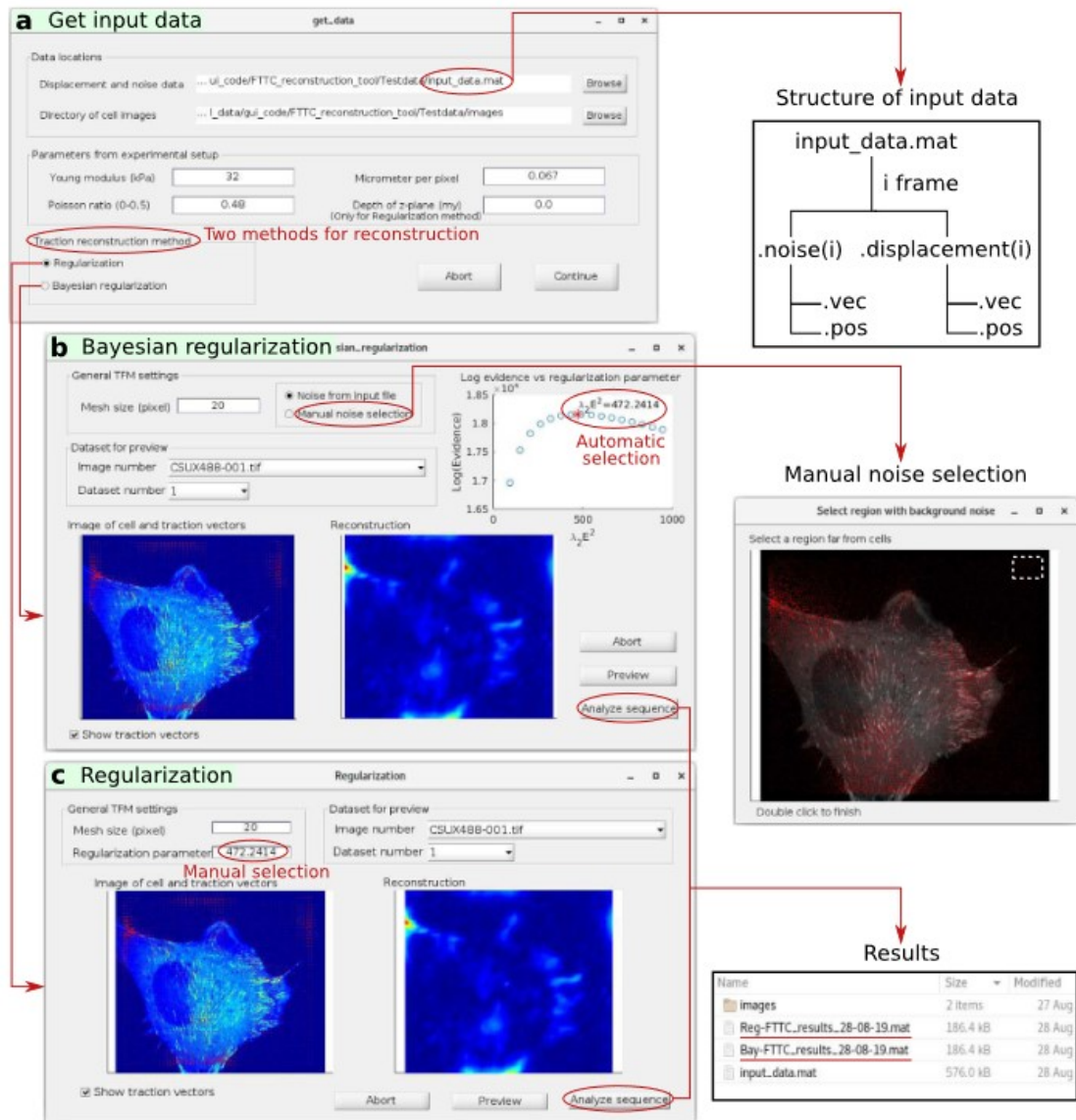


Figure 13. The graphical user interface of the BFTTC program developed by Huang et al. [53]

5. CONCLUSIONS AND FUTURE PROSPECTS

The aim of this thesis was to provide an overview of image and video processing methods and tools available for studying hiPSC-CM biomechanics and function as an introductory guide for researchers interested in the topic. As illustrated by this thesis, many types of image processing tools and methods are available. The recent trend in image and video processing is clearly headed towards automated and machine learning -based approaches, due to the need for high-throughput and high-accuracy methods. Manual image and video processing is time-consuming and often requires expertise in the underlying mathematics. In addition to saving time, fully automated, high-throughput methods make image and video processing more accessible to all researchers.

To study the contractile machinery of hiPSC-CMs, fluorescent microscopy is the widely adopted imaging technique. A central image processing problem in fluorescence microscopy is the segmentation of fluorescently dyed structures. For which many types of segmentation algorithms and techniques have been developed. The ever-increasing amount of available imaging data requires automated and robust segmentation algorithms. In addition, the unorganized nature of hiPSC-CM contractile structures poses an additional challenge to accurate segmentation [8].

In several biomedical applications machine learning -based segmentation algorithms have been adopted for segmentation. To the best of the author's knowledge, they have not yet been used for studying hiPSC-CM contractile structures. However, using them for this purpose should be further explored as proposed by Cao et al. [8]. A few state-of-the-art CNN architectures are introduced in this thesis including U-net, Mask R-CNN and YOLOv4. In the future, these methods should be compared amongst each other and with traditional image segmentation algorithms to determine their potential in studying the hiPSC-CM contractile machinery.

In addition to segmentation, quantitative analysis is needed to study the morphological features of hiPSC-CM contractile machinery or their kinematic behaviour. Several metrics and software tools have been developed for this purpose including the state-of-the-art software analysis tool Sarc-Graph published recently in 2021. Sarc-Graph for the first time allows the automated tracking and analysis of individual sarcomere structures in beating hiPSC-CMs with very little parameter tuning required. In addition, it provides multiple functions for example for sarcomere and z-line segmentation as well as quanti-

ifying hiPSC-CM contraction. In the future it would be interesting to compare the capabilities of CNN-based segmentation methods to Sarc-Graph segmentation. Moreover, CNN-based segmentation methods could be combined with the analysis functions provided by Sarc-Graph. [45]

To explicitly study the contractile forces exerted by hiPSC-CMs, TFM-based imaging is needed. The most central image processing problem in TFM is the reconstruction of traction forces. Several algorithms and image processing tools have been developed for this purpose and for different platforms including ImageJ and MATLAB. One of the newest traction force reconstruction software, BFTTC published in 2020 by Huang et al, for the first time allows the automated denoising of TFM data [53]. However, it requires the usage of another software tool for determining the deformation field, which is required by BFTTC as input. The available traction reconstruction tools still lack a fully automated option that can perform both deformation field computation and traction reconstruction.

REFERENCES

- [1] C. W. Tsao *et al.*, “Heart Disease and Stroke Statistics-2023 Update: A Report From the American Heart Association,” *Circ. N. Y. N.*, vol. 147, no. 8, pp. e93–e621, 2023, doi: 10.1161/CIR.0000000000001123.
- [2] K. Zhu, X. Bao, Y. Wang, T. Lu, and L. Zhang, “Human induced pluripotent stem cell (hiPSC)-derived cardiomyocyte modelling of cardiovascular diseases for natural compound discovery,” *Biomed. Pharmacother.*, vol. 157, p. 113970, Jan. 2023, doi: 10.1016/j.biopha.2022.113970.
- [3] J. Li *et al.*, “hiPSC-Derived Cardiac Tissue for Disease Modeling and Drug Discovery,” *Int. J. Mol. Sci.*, vol. 21, no. 23, pp. 8893–, 2020, doi: 10.3390/ijms21238893.
- [4] K. Takahashi and S. Yamanaka, “Induction of Pluripotent Stem Cells from Mouse Embryonic and Adult Fibroblast Cultures by Defined Factors,” *Cell*, vol. 126, no. 4, pp. 663–676, Aug. 2006, doi: 10.1016/j.cell.2006.07.024.
- [5] B. Alberts, *Essential cell biology*, Fifth edition. New York: W.W. Norton & Company, 2019.
- [6] D. Sinnecker, K.-L. Laugwitz, and A. Moretti, “Induced pluripotent stem cell-derived cardiomyocytes for drug development and toxicity testing,” *Pharmacol. Ther.*, vol. 143, no. 2, pp. 246–252, Aug. 2014, doi: 10.1016/j.pharmthera.2014.03.004.
- [7] A. Di Baldassarre, E. Cimetta, S. Bollini, G. Gaggi, and B. Ghinassi, “Human-Induced Pluripotent Stem Cell Technology and Cardiomyocyte Generation: Progress and Clinical Applications,” *Cells*, vol. 7, no. 6, p. 48, May 2018, doi: 10.3390/cells7060048.
- [8] L. Cao, L. Schoenmaker, S. A. Ten Den, R. Passier, V. Schwach, and F. J. Verbeek, “Automated Sarcomere Structure Analysis for Studying Cardiotoxicity in Human Pluripotent Stem Cell-Derived Cardiomyocytes,” *Microsc. Microanal.*, vol. 29, no. 1, pp. 254–264, Feb. 2023, doi: 10.1093/micmic/ozac016.
- [9] C. Tharp, L. Mestroni, and M. Taylor, “Modifications of Titin Contribute to the Progression of Cardiomyopathy and Represent a Therapeutic Target for Treatment of Heart Failure,” *J. Clin. Med.*, vol. 9, no. 9, pp. 2770–, 2020, doi: 10.3390/jcm9092770.
- [10] U. Kubitscheck, *Fluorescence Microscopy - From Principles to Biological Applications (2nd Edition)*. Newark: John Wiley & Sons, 2017.
- [11] A. Maier, *Medical Imaging Systems An Introductory Guide*, 1st ed. 2018. in *Image Processing, Computer Vision, Pattern Recognition, and Graphics*, no. 11111. Cham: Springer Nature, 2018. doi: 10.1007/978-3-319-96520-8.
- [12] J. Jonkman and C. M. Brown, “Any Way You Slice It-A Comparison of Confocal Microscopy Techniques,” *J. Biomol. Tech.*, vol. 26, no. 2, pp. 54–65, 2015, doi: 10.7171/jbt.15-2602-003.
- [13] M. Wheelwright, Z. Win, J. L. Mikkila, K. Y. Amen, P. W. Alford, and J. M. Metzger, “Investigation of human iPSC-derived cardiac myocyte functional maturation by single cell traction force microscopy,” *PLoS ONE*, vol. 13, no. 4, p. e0194909, Apr. 2018, doi: 10.1371/journal.pone.0194909.
- [14] C. M. Cuerrier and A. E. Pelling, *Cells, forces, and the microenvironment*. Boca Raton, Florida: CRC Press, 2015. doi: 10.1201/b18184.
- [15] L. R. Sewanan and S. G. Campbell, “Modelling sarcomeric cardiomyopathies with human cardiomyocytes derived from induced pluripotent stem cells,” *J. Physiol.*, vol. 598, no. 14, pp. 2909–2922, 2020, doi: 10.1113/JP276753.
- [16] A. Zancla, P. Mozetic, M. Orsini, G. Forte, and A. Rainer, “A primer to traction force microscopy,” *J. Biol. Chem.*, vol. 298, no. 5, p. 101867, May 2022, doi: 10.1016/j.jbc.2022.101867.
- [17] J. M. M. Pérez and J. Pascau, *Image processing with ImageJ: discover the incredible possibilities of ImageJ, from basic image processing to macro and plugin development*, 1st ed. Birmingham: PACKT Publishing, 2013.

- [18] "Image Processing Toolbox." <https://se.mathworks.com/products/image.html> (accessed Apr. 07, 2023).
- [19] D. Sage *et al.*, "DeconvolutionLab2: An open-source software for deconvolution microscopy," *Methods*, vol. 115, pp. 28–41, Feb. 2017, doi: 10.1016/j.ymeth.2016.12.015.
- [20] L. Cao, A. D. van der Meer, F. J. Verbeek, and R. Passier, "Automated image analysis system for studying cardiotoxicity in human pluripotent stem cell-Derived cardiomyocytes," *BMC Bioinformatics*, vol. 21, no. 1, p. 187, Dec. 2020, doi: 10.1186/s12859-020-3466-1.
- [21] P.-G. Ho, Ed., *Image Segmentation*. InTech, 2011. doi: 10.5772/628.
- [22] F. Long, "Microscopy cell nuclei segmentation with enhanced U-Net," *BMC Bioinformatics*, vol. 21, no. 1, p. 8, Jan. 2020, doi: 10.1186/s12859-019-3332-1.
- [23] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *arXiv.org*, 2015.
- [24] T. Falk *et al.*, "U-Net: deep learning for cell counting, detection, and morphometry," *Nat. Methods*, vol. 16, no. 1, pp. 67–70, 2019, doi: 10.1038/s41592-018-0261-2.
- [25] X. Dong *et al.*, "Automatic multiorgan segmentation in thorax CT images using U-net-GAN," *Med. Phys. Lanc.*, vol. 46, no. 5, pp. 2157–2168, 2019, doi: 10.1002/mp.13458.
- [26] A. O. Vuola, S. U. Akram, and J. Kannala, "Mask-RCNN and U-Net Ensembled for Nuclei Segmentation," *IEEE*, 2019, pp. 208–212. doi: 10.1109/ISBI.2019.8759574.
- [27] F. Bagheri, M. J. Tarokh, and M. Ziaratban, "Skin lesion segmentation based on mask RCNN , MULTI ATROUS FULL-CNN , and a geodesic method," *Int. J. Imaging Syst. Technol.*, vol. 31, no. 3, pp. 1609–1624, Sep. 2021, doi: 10.1002/ima.22561.
- [28] J.-H. Shu, F.-D. Nian, M.-H. Yu, and X. Li, "An Improved Mask R-CNN Model for Multiorgan Segmentation," *Math. Probl. Eng.*, vol. 2020, pp. 1–11, Jul. 2020, doi: 10.1155/2020/8351725.
- [29] J.-Y. Chiao, K.-Y. Chen, K. Y.-K. Liao, P.-H. Hsieh, G. Zhang, and T.-C. Huang, "Detection and classification the breast tumors using mask R-CNN on sonograms," *Medicine (Baltimore)*, vol. 98, no. 19, p. e15200, May 2019, doi: 10.1097/MD.00000000000015200.
- [30] S. Albahli, N. Nida, A. Irtaza, M. H. Yousaf, and M. T. Mahmood, "Melanoma Lesion Detection and Segmentation Using YOLOv4-DarkNet and Active Contour," *IEEE Access*, vol. 8, pp. 198403–198414, 2020, doi: 10.1109/ACCESS.2020.3035345.
- [31] S. Mei, H. Jiang, and L. Ma, "YOLO-lung: A Practical Detector Based on Improved YOLOv4 for Pulmonary Nodule Detection," *Piscataway: IEEE*, 2021, pp. 1–6. doi: 10.1109/CISP-BMEI53629.2021.9624373.
- [32] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, May 2018, doi: 10.1016/j.patcog.2017.10.013.
- [33] S. Hao, Y. Zhou, and Y. Guo, "A Brief Survey on Semantic Segmentation with Deep Learning," *Neurocomputing*, vol. 406, pp. 302–321, Sep. 2020, doi: 10.1016/j.neucom.2019.11.118.
- [34] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020, doi: 10.1109/TPAMI.2018.2844175.
- [35] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation".
- [36] R. Girshick, "Fast R-CNN," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 1440–1448. doi: 10.1109/ICCV.2015.169.
- [37] Shaoqing Ren, Kaiming He, R. Girshick, and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.
- [38] A. Bochkovskiy, W. Chien-Yao, and M. L. Hong-Yuan, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv.org*, 2020.
- [39] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *IEEE*, 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

- [40] F. S. Pasqualini, S. P. Sheehy, A. Agarwal, Y. Aratyn-Schaus, and K. K. Parker, "Structural Phenotyping of Stem Cell-Derived Cardiomyocytes," *Stem Cell Rep.*, vol. 4, no. 3, pp. 340–347, Mar. 2015, doi: 10.1016/j.stemcr.2015.01.020.
- [41] M. D. Sutcliffe, P. M. Tan, A. Fernandez-Perez, Y.-J. Nam, N. V. Munshi, and J. J. Saucerman, "High content analysis identifies unique morphological features of reprogrammed cardiomyocytes," *Sci. Rep.*, vol. 8, no. 1, Art. no. 1, Jan. 2018, doi: 10.1038/s41598-018-19539-z.
- [42] T. A. Morris *et al.*, "Striated myocyte structural integrity: Automated analysis of sarcomeric z-discs," *PLoS Comput. Biol.*, vol. 16, no. 3, pp. e1007676–, 2020, doi: 10.1371/journal.pcbi.1007676.
- [43] C. N. Toepfer *et al.*, "SarcTrack," *Circ. Res.*, vol. 124, no. 8, pp. 1172–, 2019, doi: 10.1161/CIRCRESAHA.118.314505.
- [44] C. Pasqualin, F. Gannier, A. Yu, C. O. Malécot, P. Bredeloux, and V. Maupoil, "SarcOptiM for ImageJ: high-frequency online sarcomere length computing on stimulated cardiomyocytes," *Am. J. Physiol. Cell Physiol.*, vol. 311, no. 2, pp. C277–C283, 2016, doi: 10.1152/ajpcell.00094.2016.
- [45] B. Zhao, K. Zhang, C. S. Chen, and E. Lejeune, "Sarc-Graph: Automated segmentation, tracking, and analysis of sarcomeres in hiPSC-derived cardiomyocytes," *PLoS Comput. Biol.*, vol. 17, no. 10, pp. e1009443–e1009443, 2021, doi: 10.1371/journal.pcbi.1009443.
- [46] J. P. Butler, I. M. Tolić-Nørrelykke, B. Fabry, and J. J. Fredberg, "Traction fields, moments, and strain energy that cells exert on their surroundings," *Am. J. Physiol.-Cell Physiol.*, vol. 282, no. 3, pp. C595–C605, Mar. 2002, doi: 10.1152/ajpcell.00270.2001.
- [47] S. R. Polio, K. E. Rothenberg, D. Stamenović, and M. L. Smith, "A micropatterning and image processing approach to simplify measurement of cellular traction forces," *Acta Biomater.*, vol. 8, no. 1, pp. 82–88, 2012, doi: 10.1016/j.actbio.2011.08.013.
- [48] Q. Tseng *et al.*, "Spatial organization of the extracellular matrix regulates cell–cell junction positioning," *Proc. Natl. Acad. Sci. - PNAS*, vol. 109, no. 5, pp. 1506–1511, 2012, doi: 10.1073/pnas.1106377109.
- [49] E. Paluch, "Measurement of cell traction forces with ImageJ," in *Biophysical Methods in Cell Biology*, in Biophysical methods in cell biology, vol. 125. United States: Elsevier Science & Technology, 2015, pp. 269–287. doi: 10.1016/bs.mcb.2014.10.008.
- [50] S. J. Han, Y. Oak, A. Groisman, and G. Danuser, "Traction microscopy to identify force modulation in subresolution adhesions," *Nat. Methods*, vol. 12, no. 7, pp. 653–656, 2015, doi: 10.1038/nmeth.3430.
- [51] J. Barrasa-Fano, A. Shapeti, Á. Jorge-Peñas, M. Barzegari, J. A. Sanz-Herrera, and H. Van Oosterwyck, "TFMLAB: A MATLAB toolbox for 4D traction force microscopy," *SoftwareX*, vol. 15, pp. 100723–, 2021, doi: 10.1016/j.softx.2021.100723.
- [52] J. Toyjanova, E. Bar-Kochba, C. López-Fagundo, J. Reichner, D. Hoffman-Kim, and C. Franck, "High resolution, large deformation 3D traction force microscopy," *PLoS One*, vol. 9, no. 4, pp. e90976–e90976, 2014, doi: 10.1371/journal.pone.0090976.
- [53] Y. Huang, G. Gompper, and B. Sabass, "A Bayesian traction force microscopy method with automated denoising in a user-friendly software package," *Comput. Phys. Commun.*, vol. 256, pp. 107313–, 2020, doi: 10.1016/j.cpc.2020.107313.