The Optical Stretcher: Towards a Cell Sorter Based on High-Content Analysis

Dissertation

zur Erlangung des akademischen Grades Doktoringenieur (Dr.-Ing.)

vorgelegt an der
Technischen Universität Dresden
Fakultät Maschinenbau und Verfahrenstechnik

eingereicht von

Christoph Faigle
geboren am 17. Juli 1985 in Bonn

Betreuende Hochschullehrer:
Prof. Dr. Thomas Bley
Prof. Dr. Jochen Guck
Prof. Dr. Roberto Osellame

Dresden, den 4. November 2015

Kurzfassung

Die mechanischen Parameter einer biologischen Zelle sind relevante Indikatoren für ihre Funktion oder für Krankheiten. Beispielsweise sind gewisse Krebszellen verformbarer als gesunde Zellen. Die Herausforderung besteht darin, Methoden zur Messung dieser Parameter zu entwickeln, die das System Zelle nicht beeinflussen. Der *Optical Stretcher* ist ein mikrofluidisches System, das einzelne suspendierte Zellen kontaktfrei mithilfe von Lasern verformt, um deren viskoelastische Eigenschaften zu ermitteln. Der Vorteil hierbei ist, dass im Gegensatz zu Standardmessmethoden der Molekularbiologie die Zellen nicht aufwendig mit zusätzlichen Markern versehen werden müssen. Grundversionen des Optical Stretchers existieren schon seit einigen Jahren, die es ermöglichen, homogene Zellpopulationen zu vermessen. Jedoch konnte man bis anhin nur Durchschnittswerte für jede Population errechnen. Um inhomogene Zellpopulationen zu charakterisieren, ist es jedoch nötig, jede einzelne Zelle zu betrachten und zusätzliche Parameter, seien es mechanische oder optische, wie der Brechungsindex, zu messen.

In dieser Arbeit werden verschiedene Erweiterungen des Optical Stretchers aufgezeigt. Erstmalig wird ein Ablauf vorgestellt, der u.a. mittels eines verbesserten Bildanalysealgorithmus mechanische Daten in Echtzeit analysieren kann. In Kombination mit Messungen des optischen Brechungsindex können so einzelne Zellen detaillierter als früher charakterisiert werden. Zudem ist es nun möglich, interessante Subpopulationen zu extrahieren, um sie daraufhin beispielsweise mit bekannten molekularbiologischen Verfahren näher zu untersuchen. Weiterhin werden je nach Aufgabenbereich verschiedene neuartige Geräte zur Zellmessung, basierend auf mikrofluidischen und optischen Überlegungen, vorgestellt. Das Grundkonzept ist dabei jeweils ein mikrostrukturierter Chip, der in ein handelsübliches Mikroskop integriert werden kann. Diese Chips bieten die Möglichkeit, gemessene Zellpopulationen nach ihren mechanischen Eigenschaften zu trennen. Es wird aufgezeigt, wie diese Trennung inklusive mathematischer Klassifizierung vonstattengeht. Exemplarisch wurden diese Methoden mit Zellarten unterschiedlicher mechanischer Eigenschaften getestet, um ihre Anwendbarkeit in der Praxis zu prüfen. So wurden Einzelzellen in ihre jeweilige Herkunftspopulation sortiert. Diese neuen Methoden bieten die Möglichkeit eines vielseitigen Geräts, das in der biophysikalischen Forschung Anwendung finden kann.

Abstract

The mechanical parameters of biological cells are relevant indicators of their function or of disease. For example, certain cancerous cells are more deformable than healthy cells. The challenge consists in developing methods that can measure these parameters while not affecting the cell. The *Optical Stretcher* is a microfluidic system that deforms single suspended cells without contact using lasers and determines the cells' viscoelastic properties. The advantage compared to standard methods of molecular biology is that cells do not need to be treated with additional markers. Basic versions of the Optical Stretcher have existed for some years. These allow the measurement of homogeneous cell populations. Up until now, it was only possible to calculate average population values of compliance. To characterize inhomogeneous populations however, it is necessary to consider each single cell and measure additional mechanical or optical parameters such as the refractive index.

This work highlights various extensions of the Optical Stretcher. A novel procedure, including an improved image processing algorithm, is presented to analyze mechanical data in real time. In combination with measurements of the optical refractive index, single cells can now be characterized in more detail. Moreover, it is now possible to extract interesting subpopulations that can be further examined with molecular biology techniques. Depending on the intended purpose, novel devices for cell measurements, based on microfluidic and optical considerations, are presented. The fundamental concept involves microstructured chips that can be integrated into a commercial microscope. These chips offer the possibility of separating measured cell populations according to their mechanical properties. This separation, including mathematical classification, is demonstrated. These methods are tested with cell types of differing mechanical properties to prove their applicability in practice. Single cells are sorted into their respective population of origin. These novel methods offer the possibility of a versatile device to be applied in biophysical research.

Contents

1 Introduction

The Optical Stretcher is a device that manipulates biological cells by trapping and deforming them with light. It is a tool that offers rheological measurements without touching cells.

To put the work presented in this dissertation in its context, it is advisable to gain an overview over the work done on the Optical Stretcher over the years and put the current challenges into perspective. A short timeline of the development of the Optical Stretcher with special consideration of the technical advances is therefore in order.

Arthur Ashkin is usually credited as the inventor of the laser beam trap in 1970. He employed two divergent laser beams from an Argon laser to accelerate and trap latex spheres using only the radiation pressure of light [1]. He also invented the optical tweezers, which trap objects using a focused laser beam [2]. The optical tweezers went on to become a very successful tool for biophysics, while the dual beam laser trap was initially relegated to a niche existence. In 1993, a dual beam laser trap built with optical fibers was used to trap yeast cells [3]. The Optical Stretcher itself was invented by Jochen Guck in 2000 when red blood cells were stretched using a dual beam trap at 785 nm with a Ti-Sapphire laser [4]. Subsequent publications [5, 6] described the theoretical physics in more detail and proposed an algorithm for image analysis. Different cell lines were compared and showed different deformations, proving that certain cell lines can be distinguished by their deformation. Most importantly, it was shown that the optical stretching does not affect the cell's viability. The setup for this work, subsequently called *open setup*, was simple: two optical fibers were placed on a microscope slide, positioned against a backstop in such a way that the cleaved fiber ends faced each other, and then fixed with a drop of glue. A droplet of medium was then placed so that the trapping region was filled, and cells were pipetted into this medium. With luck, cells passed the trapping region while settling and were trapped.

Obviously, this setup was cumbersome and had low throughput, so a new system was devised by Bryan Lincoln to increase throughput: the *Microfluidic Optical Stretcher*, consisting of a single square glass capillary connected to a microfluidic pump system and clamped on a photolithographically structured glass slide in such a way that the optical fibers, while still facing each other, no longer were in touch with the medium [7, 8]. The laser used was a 1064 nm Ytterbium fiber laser. The main advantage was that the setup did not have to be newly built for each experiment and could be easily flushed to clean it. This led to completely new possibilities for the Optical Stretcher: cell throughput reached 50-100 cells per hour, could easily be controlled by a microfluidic system and the imaging quality was much improved with respect to the open setup. Soon, this setup became and still is the stable workhorse for many an experiment.

Using this setup, the Optical Stretcher was used to probe biological questions extensively. For example, it was shown that cancer cells differ in their mechanical phenotype from healthy cells [9, 10, 11, 12],

that differentiation of myeloid precursor cells leads to a change in function and stiffness [13, 14] and that the mechanical phenotype of suspended stem cells does not change over cultivation time [15]. Further research considered the effect of cytoskeletal proteins on cellular stiffness [16, 17], as well as the effect of calcium influx on cellular contraction [18, 19] and the deformation of giant unilamellar cesicles in the Optical Stretcher [20, 21, 22].

The effect of heating on cellular behavior was another important question under investigation, particularly since, as part of the light energy gets absorbed by the medium surrounding the cell in the Optical Stretcher, cells will experience a temperature increase of their environment which can both affect their stiffness [23, 24, 25, 22, 26] as well as their viability [27].

Over the years, the theoretical framework of the physics behind the Optical Stretcher was extended both concerning the interaction of light with the cell [28, 29, 30, 31] as well as the mechanical properties of the cell itself [32, 33, 34, 35, 36, 37, 38].

Nonewithstanding the success of the single glass capillary setup, other technical approaches have been taken to improve handling, simplify use or extend the functionality of the Optical Stretcher. For example, a design that incorporates optical fibers into a pre-molded PDMS chip [39] was tested. Rather than using optical fibers as waveguides, a significantly new approach was introduced by Roberto Osellame's group by writing waveguides into materials such as bulk glass via femtosecond laser induced fracturing [40] and structuring microfluidic channels into the bulk glass by subsequent hydrofluoric etching [41, 42, 43, 44]. A similar approach is to use photolithography in deep UV to write waveguides into polymers [45] and one can anticipate an Optical Stretcher could be fabricated using this method. These approaches have the advantage that not only is the channel geometry within the bulk material freely designable in three dimensions, but the alignment of waveguides to the microfluidic channels is near-perfect and robust without any moving parts. Functionality can be extended by improving the cellular throughput by aligning cells along the microfluidic channels so they are all trapped. This can be done by acoustophoretic prefocussing [46, 47].

As one can see, there is a wide range of research that benefits from the Optical Stretcher. The following section will describe the biological structures that affect the cell's mechanical properties and describe how the optical stretcher can be used to obtain data on celllular mechanics.

1.1 Physics behind the Optical Stretcher

1.1.1 The eukaryotic cell

Cells are the building blocks of life and are the smallest entity within an organism that can be said to be "alive". There exists a wide variety of cell types, all with their specific function. The eukaryotic cell (Figure 1.1) itself is a highly complex system comprised of a myriad of components [48]. In this work however it suffices to see the cell as made up from certain distinct components: the cell membrane, composed of phospholipids and forming the outer boundary of the cell, the cytoskeleton, providing the cellular structure, and the nucleus. As by far the largest contribution to the global mechanical properties of the cell come from the cytoskeleton, it is worth taking a closer look at.
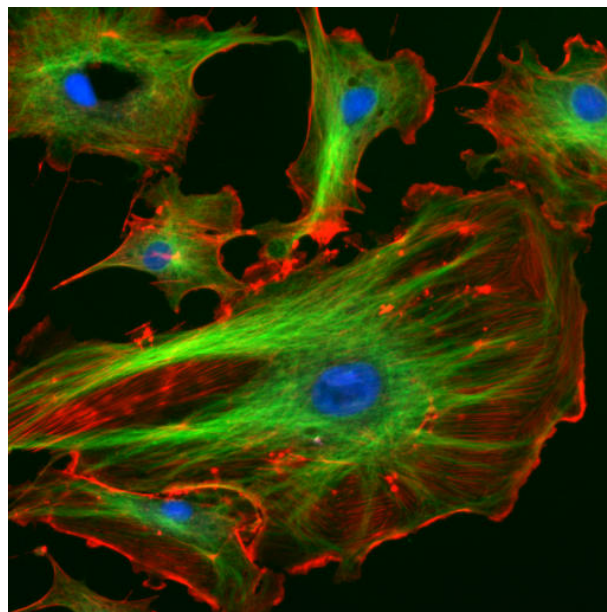
Figure 1.1: Adherent eukaryotic cells. The nucleus is stained in blue, microtubuli in green, actin filaments in red. Image taken from the ImageJ package, public domain.

The cytoskeleton, as well as being a part of cellular signaling pathways and acting as a route for intracellular transport of vesicles and organelles, plays an important role not only by providing a passive scaffold for the internal organization of the cell, distributing forces across the entire cell, but also as an active protagonist, moving and shaping the cell through force generation during migration and cell division. The cytoskeleton itself consists of polymer filaments bound by crosslinkers that constantly assemble and disassemble, forming a dense, but always changing, network. The three main polymer components are actin filaments, microtubules and intermediate filaments.

Actin polymerizes to stiff filaments of 7-9 nm diameter, which in turn can form fibers or branched networks. Actin plays a large role in cell migration within adherent cells and is usually found in front of the cell in form of stress fibers connected to the focal adhesion points, where the cell adheres to the extracellular substrate. These fibers interact with motor proteins such as myosin to contract the cell, thereby moving it. In suspended cells, actin usually forms a shell called actin cortex on the inner face of the cell

membrane.

Microtubules, hollow tubes of the protein tubulin of 23 nm diameter, are the stiffest and longest components of the cytoskeleton, forming filaments that can span the whole cell. Unconstrained, they can grow up to 50 μm long. Apart from forming an integral part of the structural rigidity of the cell, they are involved in intracellular transport as well as chromosome separation during cell division. The microtubules are almost inextendable, with a purported elastic modulus similar to plexiglas [49].

Intermediate filaments such as keratin have an average diameter of 10 μm. Compared to microtubules and actin filaments, they are much more deformable. However, they also resist tensile stress without breaking [50].

While the cytoskeletal network is composed of such diverse components, globally the cell displays a passive viscoelastic behavior as well as an active response to externally induced stresses. The next section describes how such a behavior can be measured by using an Optical Stretcher.

1.1.2 Deformation of a dielectric material through optical force

To build a simple version of the Optical Stretcher, all that is needed are two perfectly aligned Gaussian beam sources facing each other. Any dielectric object between these sources will be trapped and deformed purely by application of laser light. How does this work?

In the Optical Stretcher, light from a laser source passes through a cell. How does the passage of light lead to a deformation of the cell? The propagation of light through a dielectric object such as a cell can be described theoretically by two regimes that are distinguished by the ratio of light wavelength λ and the radius r of the object. The *Rayleigh regime* [6] describes the situation where the object is much smaller than the wavelength ($r \ll \lambda$), while the *ray optics regime* describes the opposite case, ($r \gg \lambda$). As the radius of the cell in the Optical Stretcher (5 - 10 μm) is larger than the wavelength of the laser light (800 - 1000 nm), the simpler ray optics approach can be used to describe the effect of the light on the cell. It must be noted that additional effects such as interference, scattering and diffraction within the object are not taken into account by this simple approach. These effects are included when the effect is described by the Mie solution of the Maxwell equations [28]. The Mie approach covers both Rayleigh and ray optics regimes and also offers a solution for the intermediate regime $r \approx \lambda$. The agreement between ray optics and Mie theory is very close [29], justifying the ray optics simplification for explanation's sake.

In ray optics, a light beam is decomposed into individual beams of photons with separate intensity, momentum and directions $\vec{e}_i$. In a uniform medium, these beams propagate in a straight line in direction $\vec{e}_m$. Each photon in the beam is described by a certain momentum $\vec{p}_m$ which is proportional to the energy $E = \frac{hc_m}{\lambda}$ and the speed of light c_m of the medium it travels in, which itself depends on the medium's refractive index n_m

$$\vec{p}_m = E\frac{n_m}{c}\vec{e}_m.$$

When the photon beam meets the interface between media of different refractive indices n_1 and n_2 (for example between the cell medium and the cell, $n_1 < n_2$) at an angle θ, the beam's momentum changes both in direction according to Snell's law $n_1 \sin\theta = n_2 \sin r$ and in magnitude due to the refractive index change according to the previous momentum equation. Usually, a small amount of light is reflected at the interface. The reflection and transmission coefficients $R(\theta)$ and $T(\theta) = 1 - R(\theta)$ can be calculated by Fresnel equations. However the contribution of the reflection to the total momentum change is usually very small. This occurs because the difference of refractive indices is usually not that great. Absorption of light within the cell is neglected as cells are relatively transparent to light in the near-infrared region around 1000 nm.

The change in momentum Δp_i of a light ray passing through a particle can be described as

$$\Delta\vec{p}_i = \vec{p}_1 - \vec{p}_2 + \vec{p}_r$$

with $p_1 = En_1/c$, $p_2 = T(\theta)p_1 n$ and $p_r = R(\theta)p_1$ with $n = n_2/n_1$.

Because of the necessary conservation of momentum, the particle is subject to a restoring force F_i in the direction normal to the interface, pointing away from the medium with the higher refractive index, i.e. the particle [51].

$$\vec{F}_i = \frac{\Delta\vec{p}_i}{\Delta t}$$

The same procedure obviously occurs in the opposite direction when the ray of light passes the second interface, the back of the cell. The force at the back of the cell then faces in the direction of the beam propagation and is larger than the force acting at the first interface [5]. Summing up the forces of the individual rays over all incident angles at the interface leads to a resulting net force acting on the particle with perpendicular *scattering* and *gradient* components [52].

The scattering component is

$$F_{scattering} = \frac{n_m}{c}\int_0^{\pi/2} d\theta \int_0^{2\pi} d\phi I(\theta, \phi, r)r^2, sin(\theta)cos(\theta)$$
$$\times (1 + Rcos(2\theta) - \frac{T^2[cos(2\theta_0 - 2\rho) + Rcos(2\theta)]}{1 + R^2 + 2Rcos(2\rho)},$$

where θ and ϕ are the incident angles of light in the x- and y-planes, ρ is the reflection angle and $I(\theta, \phi, r)$ the light intensity which in the real life case is directly proportional to the laser power. The scattering component pushes the particle in the direction of the beam propagation (because the momentum change in the back is larger than in the front), while the gradient force pulls the particle towards the direction of the highest intensity of the laser beam (Figure 1.2).
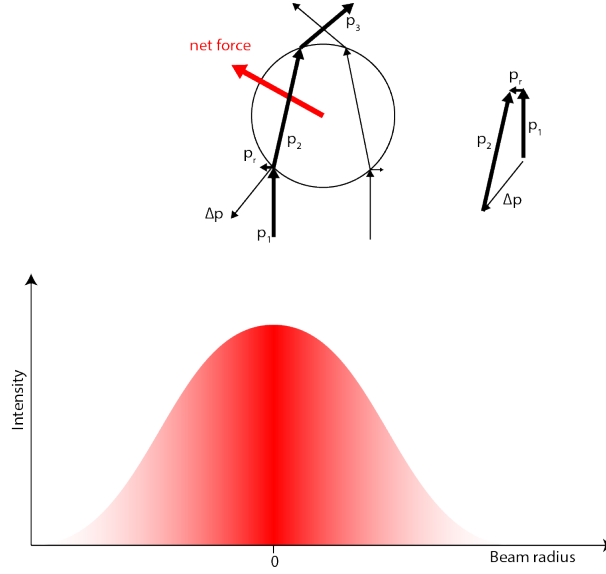
Figure 1.2: Scheme of momenta of two rays with different intensities acting on a particle. The red
arrow indicates the net force acting on the particle's center.

If two identical laser beam with a Gaussian profile (TEM$_{00}$-mode) are perfectly aligned facing each
other so that their beams propagate in opposite directions, gradient forces on the particle in the middle
are added and the scattering forces on the center will cancel. The particle will therefore position itself
in the center of the two light sources (with a caveat that is mentioned below). There will be no net force
acting on the particle's center of mass at this point. However, there still will be forces acting on the
particle's interfaces. A local stress σ is applied

$$\sigma_i = \frac{|\vec{F_i}|}{A}.$$

The peak stress along the beam axis ($\theta = 0$) is the largest and can be approximated by

$$\sigma_0 \approx \frac{n_m I}{c}(2 - R(\theta) + R(\theta)^2)(1 - \frac{n_{cell}}{n_m})$$

A spherical, non-rigid particle will then elongate into an ellipsoid shape until the restoring force, de-
pending on the particle's inner mechanical parameters, counterbalances the optical forces. Typically, the
forces acting on the particle surface are about an order of magnitude larger than the net force acting on
the center of mass. In the Optical Stretcher, the particle is a biological cell, and the elongation, tracked
over time upon application of laser power, gives rise to a characteristic deformation profile.

There is an additional consideration to take into account in the real, physical device. A particle in the
optical trap will not necessarily have a stable, centered position between two Gaussian laser beams as
mentioned above. In fact, the equilibrium position of a particle between two Gaussian beams depends on
the size of the beam waist [53]. For a particle to reach a stable equilibrium position, the particle diameter
must be smaller than the beam width at the equilibrium position. As the beam width $w(z)$ of an optical
fiber in a medium can be calculated by

$$w(z) = w_0 \sqrt{\frac{(\lambda z)^2}{(n_{med}\pi w_0)^2} + 1},$$

the distance from the beam waist to the particle should typically be larger than 80 μm for typical values ($n_{med} = 1.335$, $w_0 = 3.1\mu m$, $\lambda = 1064$ nm, cell diameter 5 μm). If this distance is too short, particles will start oscillating back and forth between two equidistant positions along the beam axis.

1.1.3 From optical force to viscoelastic parameters

Optical Stretcher measurements result in a deformation curve. This deformation can be related to the mechanical strain $\epsilon(t)$ by measuring the change of the cell's major axis a length over time, which is normalized by its mean length in rest a_0.

$$\epsilon(t) = \frac{a(t) - a_0}{a_0}$$

However, deformation is not a physical parameter that describes the cell's intrinsic mechanical properties. Therefore, the deformation must converted to obtain viscoelastic parameters. The viscoelastic deformation can be described by the *tensile creep compliance* [35, 34]

$$J(t) = \frac{\epsilon(t)}{\sigma_0 F_G}$$

where σ_0 is the optical stress mentioned in the previous section, calculated by the Lorenz-Mie model in [28]. The term F_G is a geometric correction factor that accounts for the distribution of optical stress dependent on the size of the cell [33]. However, this correction term only affects the magnitude of the measured compliance, not its dynamic behavior.

To extract viscoelastic parameters from the cell's compliance curve, various fitting models have been proposed. The main model used is the linear viscoleastic model. Usually, these models try to fit a combination of mechanical elements such as springs and dashpots to imitate the cellular behavior [35, 54, 55, 14] under stress. However, a more universal model used partially in this work appears to be the power law model [15, 14, 56]. It involves fitting the following equation to the curve obtained during application of the optical force (Figure 1.3).

$$J(t) = J_0(\frac{t}{\tau_0})^\beta$$

β, the *fluidity* describes the viscoelastic behavior of the cell, where $\beta = 0$ is perfectly elastic behavior, while $\beta = 1$ is completely viscous behavior. For suspended biological cells, β ranges from 0.3 to 0.6 [56, 24]. τ_0 is 1s, while the compliance J_0 is the inverse of the cell's Young's modulus.

The main consideration to take into account for these models is that while they fit the measurement of mechanical responses well, the parameters used in the fit do not correspond to any real structure within the cell since they assume to the cell to be a homogeneous material. This makes predictions for cellular changes impossible. Also, they do not take into account active responses by the cell, such as cellular reorganization in answer to the applied stress. This can especially occur over longer time scales.
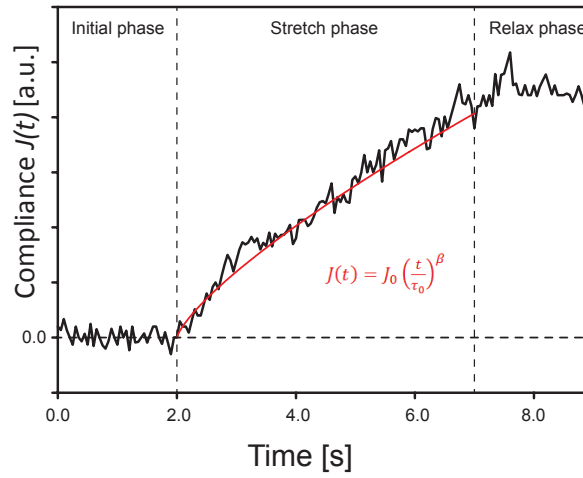
Figure 1.3: A single-cell deformation curve fitted with the power law. Using power law fitting (red), a deformation curve can be fitted during stretch phase to obtain viscoelastic parameters.

There exists a multitude of more in-depth models which take into account internal cell structure. Exhaustive overviews can be found in [55] and [57].

1.2 Motivation for new studies

The Optical Stretcher has proven itself a capable tool for biophysical research with an extensive theoretical underpinning. The work described in this thesis aims to extend the Optical Stretcher's capabilities by new designs and new workflows. Three main issues are identified:

- **Obtaining single cell data in real time:** In previous work, experimentally acquired images are analyzed post-experiment to obtain deformation data. This data is averaged to obtain a population's deformation. The results are then combined with refractive index measurements on other systems to give rise to compliance curves. Can these measurements be included into one Optical Stretcher system in such a way that physical data is obtainable for every single cell during the measurement itself?

- **Classifying single cells and sorting them:** Previously, to determine differences between cell populations, each population was measured sequentially. After the experiment, measured cells were lost for further use within the microfluidic network. Can the information gathered in such a way be used in a novel microfluidic device that can determine the type of an individual cell from a mixed population and sort the cell in such a way that mixed populations are separated and can be retrieved?

- **Rare cell measurements:** Microfluidic solutions often need copious amounts of connective tubing. Cells tend to get lost within this tubing, which is not a problem for cell samples involving tens

of thousands of cells. However, if only one specific cell from a rare sample is to be analyzed, can a device be designed to move this cell to and extract it from an Optical Stretcher? Furthermore, given that such cells are rare, can additional parameters apart from the deformation alone be used to classify them in such a device? How are such devices comparable to other, established Optical Stretcher devices?

The next chapters will try to answer these questions in detail and describe how novel devices are comparable to established OS devices. Chapter 2 introduces a new, robust and fast image analysis algorithm for cell contour detection in real time, and explains how different data can be combined to obtain viscoelastic parameters of the cell in question. Statistical methods are introduced that serve as a basis for classification of cells into different populations. Chapter 3 presents a novel monolithic device that is capable of high-throughput measurements of cells and that can be used to classify and sort a mixed population of treated and untreated suspension cells. Chapter 4 considers a simple but robust device that can be used to measure and sort individually selected cells. It combines the measurements of cell deformation with individually obtained refractive index measurements to present a possible multivariate classification where each cell's individual mechanics are considered. Then, all devices are briefly compared to see if measurements correspond. The final chapter provides a summary of the main conclusions of this work and proposes an outlook into the future potential of the Optical Stretcher.

2 High content data acquisition and analysis

The aim of this chapter is to present the additions made to the well-known process of Optical Stretcher experiments. This chapter will describe what is needed to obtain high content data from such experiments. It starts out with the software that controls the physical part of the experiment: the stretching of the cell, which is captured by a camera. This stretch needs to be analysed, ideally in real time by a sophisticated image processing software. Additionally, the refractive index of the cell has to be acquired in a separate step. The refractive index can be used as a parameter to describe the cell, but is also needed for tje calculation of the global geometric factor which relates the cell's deformation to its compliance. Finally, once a cell has been measured and it is possible to describe it via certain parameters such as its compliance, its radius and its refractive index, it can be put in relation with already known cell populations and potentially be classified as a member of these populations.

Previously, the Optical Stretcher was only one, though fundamental, part of an array of tools that were needed to characterize cell populations. A typical experiment consisted of a main part that measured the axial strain $\epsilon(t)$ of a number of cells of a certain population on the Optical Stretcher. However, since the measured cells were then lost for further measurements, in a second part of the experiment a tool such as the DHM (Section 7.1.3.3) was needed to measure the average refractive index of cells of a similar population. Finally, the average refractive index n, the average cell radius r, the distance between the optical fibers d as well as the stretching laser power P were used as the inputs for a MATLAB program that calculated the aforementioned global geometric factor GGF. In order to give the *creep compliance* $J(t)$, the axial strain $\epsilon(t)$ is divided by this factor

$$J(t) = \frac{\epsilon(t)}{GGF(n,r,d,P)}.$$

By using an average global geometric factor, information about the individual cell is lost. Assume, for example, one cell has a low actual compliance and a high refractive index while another cell has a high actual compliance, but a low refractive index. The physical forces acting on these cells will be different due to the change in refractive index, but to the experimenter, they will appear to have similar deformation curves. If these curves are divided by an average geometric factor, also the cells' assumed compliance will appear to be similar. In contrast, if the global geometric factor were recomputed for each cell individually, the cells' different physical parameters could be taken into account. For example, the range of compliance within populations might differ much more than previously thought or different subpopulations might exist that were previously not detected.

What is thus needed is a workflow that enables the user to collect all relevant data on one device. Ideally, this occurs in real time, with the cell in question still available for the user to manipulate, for example sorting the cell out for subsequent measurement or recultivation.

2.1 Main software control

In order to simplify the update of the main program for later users, and because the hardware does not allow external synchronization of cameras and laser, it was decided to modularize the program so that modules can run independently of each other. The heart of the software is a finite state machine that describes the various laser states. The state machine was the chosen approach after noticing queueing problems in simpler event-based structures, i.e. commands that would appear during the execution of one event would be queued and performed after the current event had finished running, even if they contradicted the previous event. This is not possible in a state machine. The state machine used here is a *Mealy state machine* [58] (Figure 2.1), which is dependent both on the current state and the input.

Through the modular setup, additional external devices such as the laser driver, the SID4Bio camera or the microfluidic pump are controllable. The laser driver is controlled via USB by an external data acquisition (DAQ) card inside the laser housing. The laser needs a so-called *heartbeat* in order to work, so in the background of the main program a constant 1 Hz signal is sent to the laser to keep it running. Communication via USB is very fast. The delay in laser rise time is due only to the laser hardware itself (Figure 2.2 a). The monochromatic camera is also connected via USB and runs independently of the laser via NI-VISA. To synchronize image acquisition to the rise in laser power, after the **Initial** state the program waits until a new image frame is available (Figure 2.2). This waiting time is dependent on the camera frame rate. Once a new frame is available for acqusition, the lasers are set to the *stretching* value. This ensures that the rise in power, which takes about 40ms to go from the initial value to the next value, is always synchronized to start exactly when a frame is acquired. If the frame were acquired at a time point during the power rise, the laser power for this frame would be somewhere in between the *trapping* and the *stretching* value and would be unknown. During analysis, this could shift deformation (as a function of laser power) vs. time curves by some milliseconds and make deformation profiles of cells less comparable.

The software also offers the possibility to control a microfluidic pump (via USB) and the SID4Bio camera (via FireWire) without recourse to the proprietary software of the manufacturer. Phase images require a manual change of phase plate to brightfield and a change of the light path on the microscope. Then, a phase imaging module can be opened where images of cells and the corresponding reference image are acquired, analyzed and saved as a phase image. The phase image is subsequently measured in an external program [59] (Section 7.1.3.3. Fluorescence images are acquired by manually changing the light path of the microscope from transmitted to reflected light and then saving a single image in the microscope. Phase contrast image analysis is done by sending each acquired frame into a custom DLL integrated into the Labview program. The user has the possibility to help analysis by inputting certain parameters for contour recognition. This way the user can see the impact of his parameters on the cell before even starting an experiment, potentially saving time. While running the step stretch profile, each fiber's laser power, the timing of the frame rates and the detected minor and major axes of the cell are saved to text files for subsequent processing by the user.
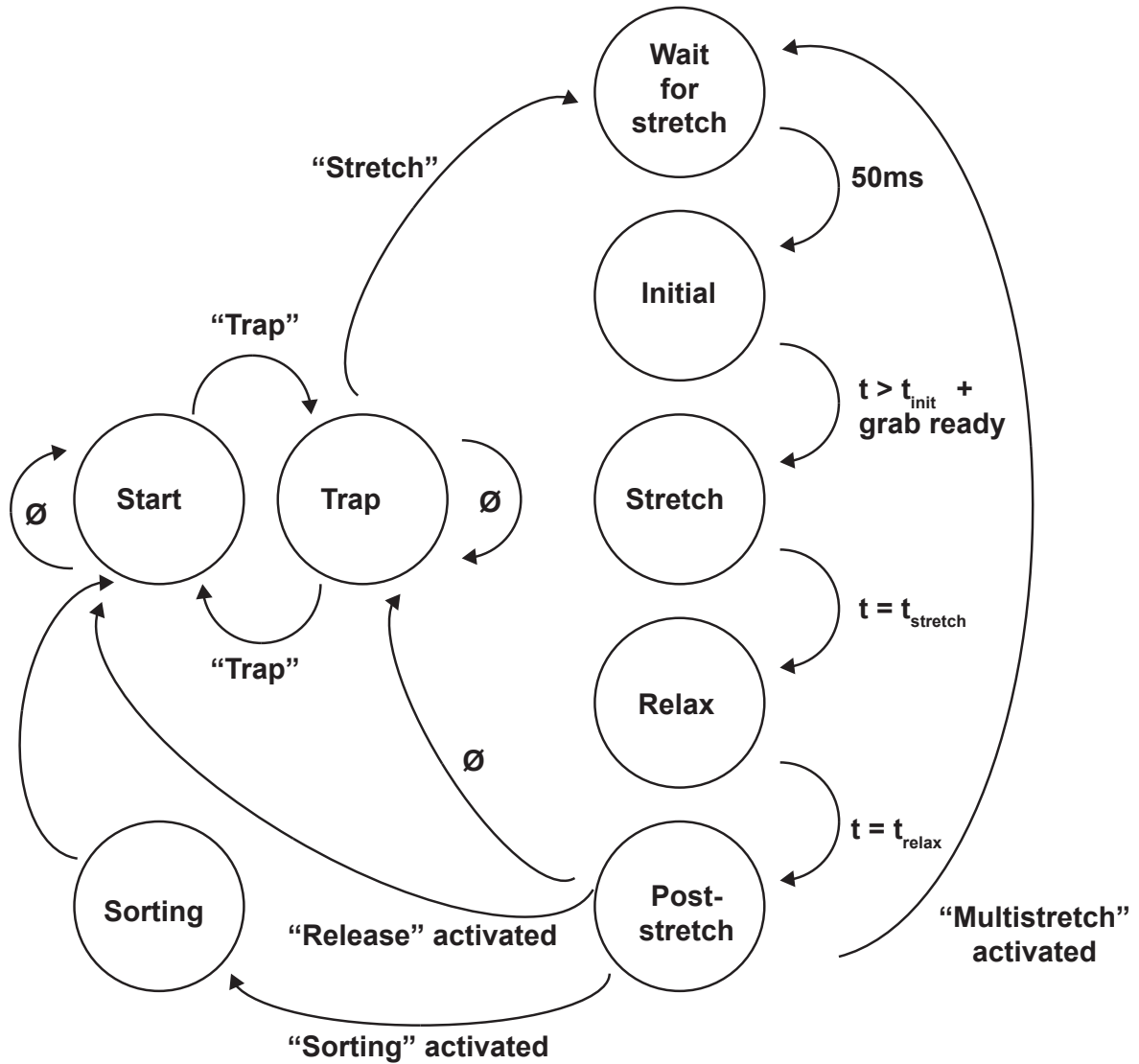
Figure 2.1: Control state machine. Start: The program starts and the laser power is at 0W. **Trap:** Laser power rises to the *trapping* power. **Wait for stretch:** Arrays and text files for saving the information after the measurement are initialized. After 50 ms: **Initial:** Image frames are grabbed and saved, while the lasers stay at initial (trapping) power until the specified initial time has passed. Once a new frame is available, the program moves on to the next step. This step synchronizes the rise of the power function to the next grabbed image frame. **Stretch:** Laser power rises to the *stretching* power. Save frames. Continue after specified amount of time. **Relax:** Power reduces back to the *trapping value*. Save frames. Continue after specified amount of time. **Post-stretch:** Save all information in text files. Reset all relevant variables to make machine ready for next stretch. If *multistretch* is activated, go to *Wait for stretch*. Else, go to *Start* or *Trap*. **Sorting:** If *sorting* is activated, a popup screen asks the user to choose the sorting direction. The cell is then pushed to this side by turning on only one laser side for 10s.
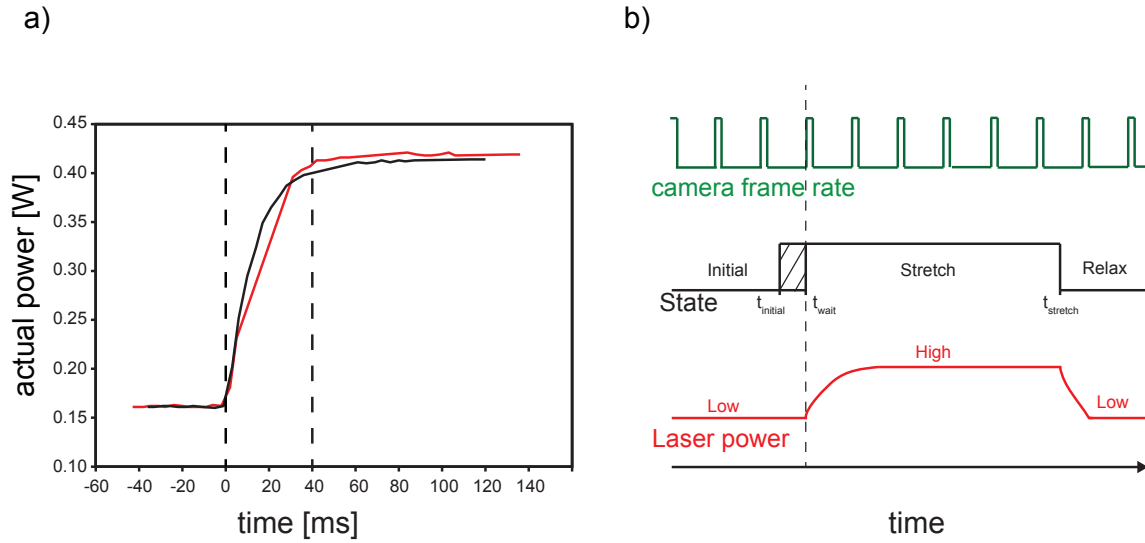
Figure 2.2: Laser synchronization. a) From the command to raise the power from 0.1W to 0.5W (nominal) at $t = 0$ to their actual output, both lasers (Channel 1 black, Channel 2 red) need about 40ms. b) The laser waits for the camera to be ready to grab a new frame upon changing the Labview state from **Initial** to **Stretch** before switching powers.

2.2 Phase contrast imaging

A cell has a different optical behavior than its surrounding medium. If it is illuminated by a white light source, the transmitted light will have a small *phase* and *amplitude* change compared to unimpeded light. However, these changes are typically too small to lead to a good contrast. To improve contrast, which is needed for the detection of cell contours, the phase change must be increased which will result in a detectable brightness change. For this, phase contrast microscopy is used (Figure 2.3 a). Illuminating light that passes through an annular aperture in the condenser of the microscope can only illuminate the specimen in a certain angle. All light that is not scattered by the specimen, i.e. the cell, falls onto a phase ring in the objective of the microscope. This light is dampened in amplitude and phase-shifted by 90°. Light that is scattered and refracted by the specimen, called foreground light, is typically phase-shifted by approximately -90°and enters the objective at a slightly different angle, not passing the phase ring. Background and foreground light now interfere with a phase shift of approximately 180° which is visible as a high contrast between the object and its background. In practice, some diffracted light, usually of a lower order and thus corresponding to larger objects such as the membrane or the nucleus of the cell, also passes through both the phase ring and the unimpeded region of the objective. By interference with itself by a phase shift of 90°, a bright halo appears around the cell (Figure 2.3 b). While this typically is an unwanted effect in phase contrast microscopy, it is useful in this case since this halo is used for the detection of the cell boundaries.

a)

Illuminating light

Condenser annulus

Condenser
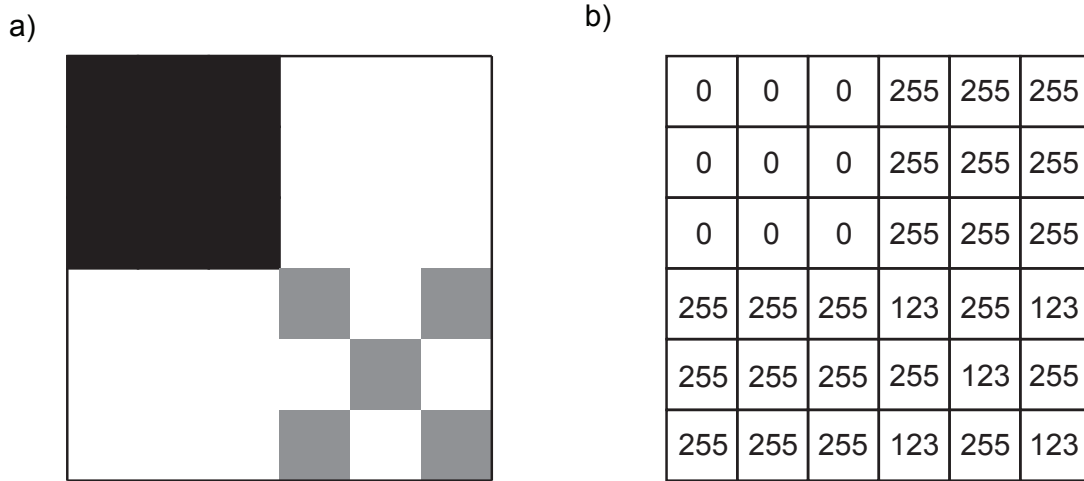
Cell

Objective

Phase shift ring

Image plane

b)
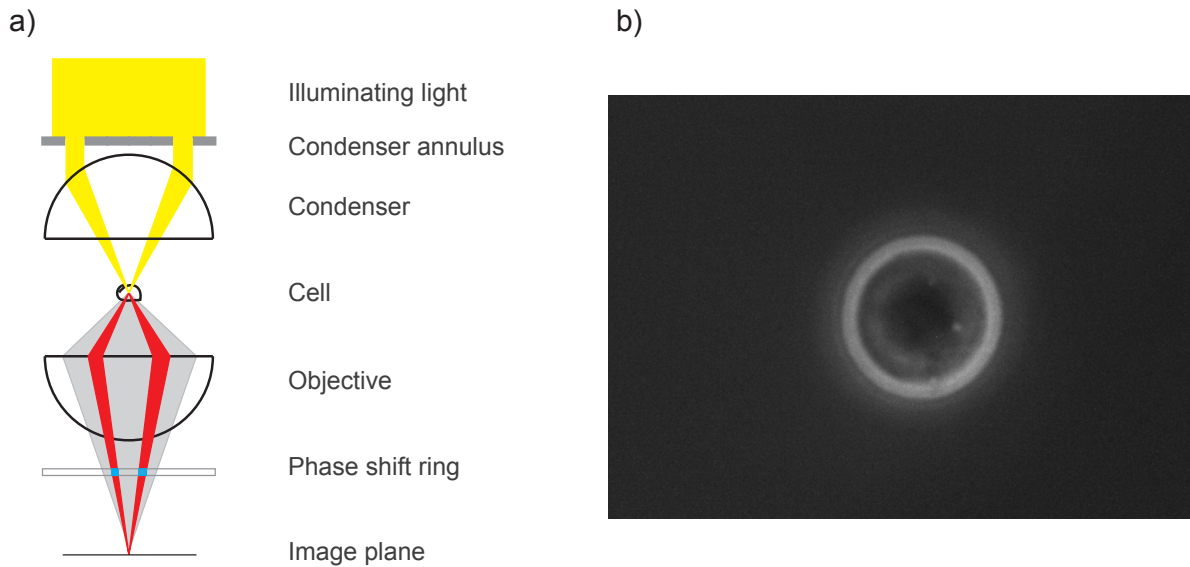
Figure 2.3: Phase contrast microscopy. a) Beam path of a microscope in phase contrast mode. Illuminating light (yellow) passes through the condenser annulus before scattering (grey) on the cell, which changes its phase by -90°. Unscattered background light (red) passes through the phase ring in the objective, shifting it by 90. Fore- and background light interfere at the image plane with a phase difference of 180, leading to high contrast. b) A cell imaged with phase contrast microscopy. The typical halo around the cell is clearly visible.

2.2.1 Real-time image analysis

Ideally, the acquired phase contrast image has a bright, continuous halo around the cell boundary. This boundary is easy to identify for the human eye, but to calculate each cell's major and minor axis, this boundary of the cell needs to be detected by software. The challenge hereby is that these halos often have blurred edges, are not uniform in brightness, and are noisy, i.e. have bright or dark speckles which can come from broken camera pixels. As always, good image acquisition, using Köhler illumination, is the best prerequisite for good analysis, but sometimes there is little one can do to improve images in the microscope. Previously, an analysis program using proprietary Labview code was used for analysis after measurement. In order to allow a direct measurement *on the fly*, i.e. in real time, the analysis needed to be ported to a faster programming language that is directly integrable into Labview via DLL. Although there exists a variety of techniques to extract features from images [60, 61, 62, 63, 64, 65], it was decided to focus on three types of methods, namely morphological operations, edge detectors and shortest path algorithms. These are briefly introduced with explanatory images.

As a preliminary, it is useful to recall some terms in image processing. A computer does not see an image as a shape but simply as a *matrix* of so-called *pixels*, numerical values that are used to represent the brightness of the image area (Figure 2.4). The matrix size depends on the total number of pixels and their arrangement and is subsequently defined as n rows by m columns. The *gradient* of an image describes the change in pixel value from one pixel to the next. Typically, it is calculated by describing the pixel in question as a function of its neighboring pixels. Mathematically, this amounts to convolving

the image matrix with a smaller matrix the size of the neighborhood in question. For a cell image, ideally, the gradient at the cell's border (from dark background to high cell foreground) is very large. High gradients alone are not sufficient though, since camera *noise* can create high gradient spots which are not part of the border or create blurred spots on the border with gradient values too low to be detected. Thus the challenge consists of finding as many connected high-gradient spots on the image as possible that form a closed boundary. Finally, *computational complexity* is a classifier of the amount of processing power needed for an algorithm or method. In this case, one is interested in how the computation time scales with the increase in image size $n \times m$ and gives an upper or worst case bound for the run time. It is described by using *big O notation* where $O(f(nm))$ means that in the worst case, the algorithm will have a run time of order $f(nm)$. As the actual run time is highly dependent on the computer used, computational complexity is more a theoretical descriptor of an algorithm than an actually measurable parameter.

a)   b)

0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
255	255	255	123	255	123
255	255	255	255	123	255
255	255	255	123	255	123

Figure 2.4: Image to matrix conversion. a) A grayscale image. b) The same image written as a matrix of its pixel, i.e. brightness values (black = 0, white = 255).

2.2.1.1 Morphological operators

Morphological operators [66] operate on each pixel of an image, convolving it with a simple pre-defined matrix called a *structuring element* or *kernel*. This kernel typically comes in a 3x3 square matrix. Depending on the type of operation, the area of the image underneath the kernel is processed in different ways. In grayscale images such as the ones obtained in an OS experiment, the main possible operations can be described as:

1. *Erosion*, which causes dark areas in the image to get larger

2. *Dilation*, which causes bright areas in the image to get larger

3. *Opening*, which removes small bright spots

4. *Closing*, which removes small dark spots

In this case, the use of morphological operators ensures that the halo around the cell is entirely closed, as this is not necessarily the case upon image acquisition. First, a contour of the cell is acquired by automatic thresholding, which divides an image into a dark background and a bright foreground (Figure 2.5 a). Next, the image is repeatedly dilated until the contour is closed, i.e. the contour circularity which is determined by the ratio of contour circumference to area, is larger than a pre-defined value. (Figure 2.5 b). The image is subsequently eroded the same number of times it was dilated (Figure 2.5 c). The resulting contour is smoothed and overlaid on the original image (Figure 2.5 d). Obviously, this is a rather haphazard method that is highly dependent on the input image. Automated thresholding such as Otsu's method [67] can fail when the image is too noisy or the background values are to similar to the foreground. On the other hand, this method needs no user input, is very fast (on the order of a few milliseconds) with a computational complexity of $O(nm)$ (i.e. one matrix operation for each image pixel) per morphological operation, and could be used for fast real-time processing were it more foolproof.
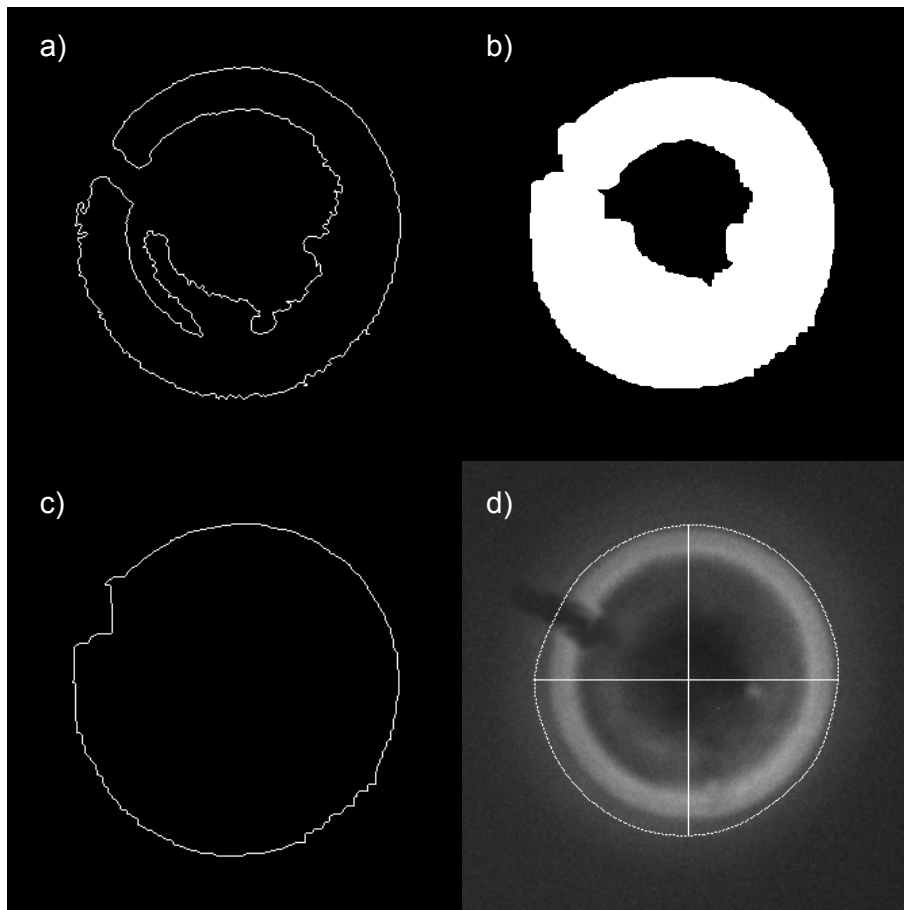

Figure 2.5: Cell contour detection with morphological operators. a) The first detected contour is not circular. b) The contour is dilated until the circularity is high enough. c) The contour then is eroded as many times as it was dilated. d) The smoothened final contour does not ideally detect the cell boundary.

2.2.1.2 Canny edge detection

The Canny edge detector [68] is one of the best known edge detectors. An *edge* is defined as a point in an image where the intensity of a pixel changes much with respect to its neighbors, i.e. the gradients are high. This can be used to identify features. This edge detection method is known as a robust edge detection algorithm which is based on three criteria:

1. Good detection: every edge point in the image should be actually detected

2. Good localization: the detected edge should be as close to the actual edge as possible

3. Only one point should be detected per edge.

Essentially, the image is convolved with an *edge detection* matrix, giving rise to a gradient matrix. Then, the Canny algorithm uses two intensity thresholds. All pixels with gradient values above the high threshold are marked as an edge, while all below the lower threshold are rejected and set to a value of zero. Canny himself suggests the upper threshold be two to three times the size of the lower threshold. Gradient values in between the thresholds are only considered to be part of an edge if one of their neighbors is above the upper threshold. In principle the gradient is highest on the border of a cell halo, so all points detected should lie on this border. However, Canny's algorithm can be problematic because it requires user interaction for choosing the correct thresholds, which can change under different illumination conditions and affects the result dramatically (Figure 2.6). Consequently, the Canny algorithm often detects edges which are not necessarily part of the border that is desired, making more user interaction necessary to specify where this border should lie.
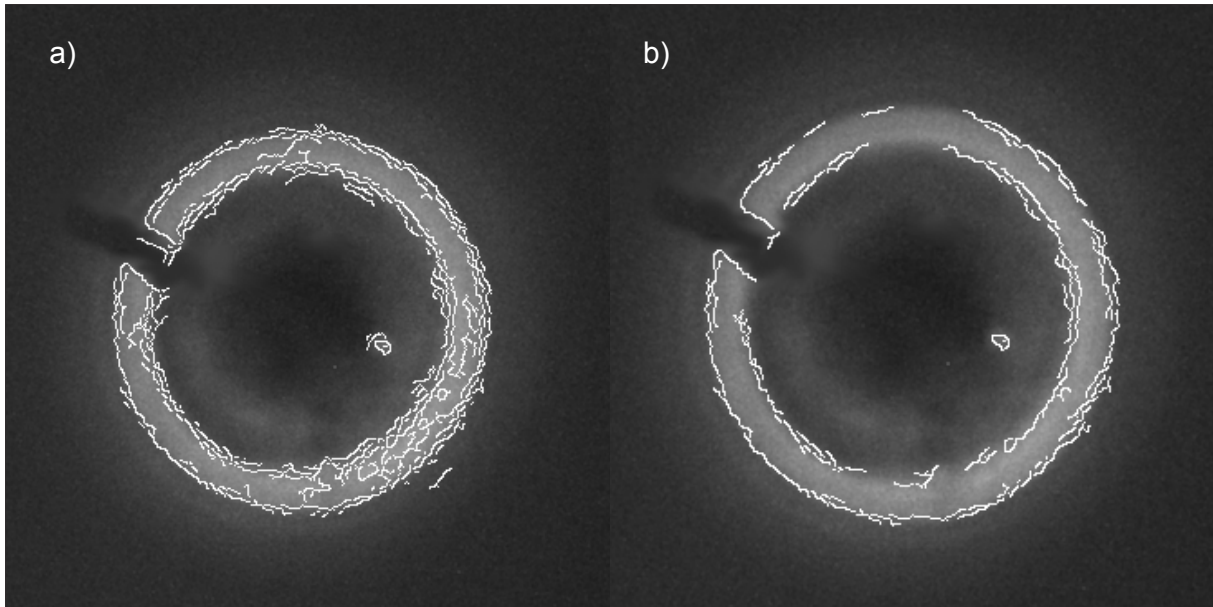

Figure 2.6: Canny edge detection with different settings. a) Detected edges with an upper and lower threshold of 60 and 20, respectively. b) Detected edges with an upper and lower threshold of 90 and 30, respectively.

2.2.1.3 Shortest Path Algorithm

Both morphological operations and Canny edge detection treat an image locally; that is, they convolve every single pixel by its neighborhood, not taking into account the whole image. In contrast, the *shortest path* algorithms presented here treat the whole image globally. Describing the concept of shortest paths as a metaphor, imagine someone wanting to cross a mountain and valley range. He will try to find the path that minimizes the required energy to pass, that is, the one that involves spending the least energy in total. He will only pass via local obstacles, like high mountains, if in the end they lead him to a valley he can traverse easily, resulting in a low total energy expenditure.

The shortest path algorithms presented here do not work on the image of the cell itself, but on its gradient image calculated by convolving the image with a simple filtering function. The high gradient points of an image, i.e. the points of highest brightness in the gradient image, can be represented as depressions in a terrain. The higher the gradient, the lower the terrain. Mathematically, this amounts to simply inverting the gradient image to obtain the terrain. The challenge now consists of finding a circular (i.e. with the same beginning and end point) path that goes along the lowest gradients points in a cell image. Mathematically, one needs to find a path that minimizes the sum of gradient values along the steps it takes before finally ending up at its starting point. One could imagine that this shortest circular path is just a single-step path from and to itself, so there is an added constraint that needs to be addressed. A simple way of ensuring that a path is more than one step long is to convert the problem into a rectangular shortest path problem by transforming the circular search region into a rectangular graph by cutting it radially and making sure that every angle is passed once. Thus, the shortest path will go from one side of the rectangle to the other (Figure 2.7). In fact, the generally used term *circular shortest path* is inaccurate, since not a circular, but a rectangular shortest path is obtained.
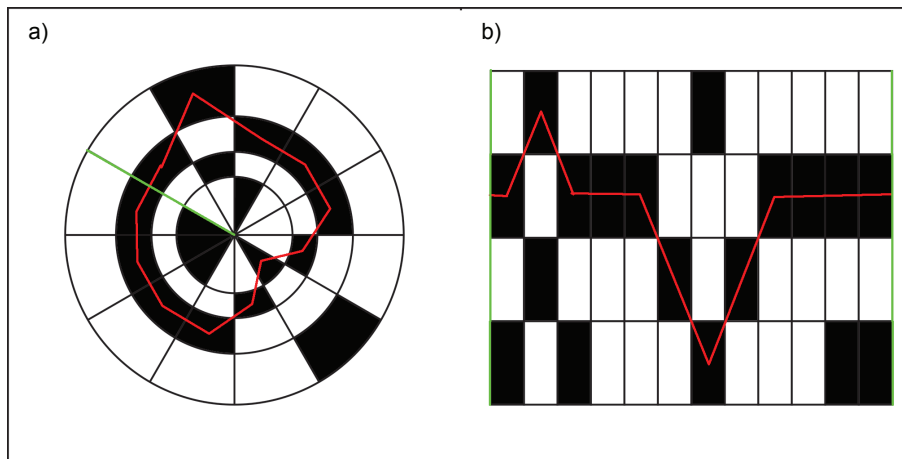
Figure 2.7: Shortest circular path. a) A circular path goes through a region. This region is cut by the green line to allow remapping. b) The same region is remapped into polar coordinates, starting from the cut. The shortest path runs from left to right, starting and ending in the same row.

The problem can now be respecified: On a rectangular map of pixel gradient values, a path which min-

imizes the sum of passed pixel values has to be found that goes from the starting pixel to the end pixel. Constraints are:

1. The path cannot go in loops, it is *acyclic*.

2. Each point has only three potential neighbors in the column to the right, it cannot step backwards.

3. The path has to end on the same row it began, which is due to its actual circularity.

There are many methods proposed to obtain shortest paths [69, 70]. One of the most famous algorithms is Dijkstra's algorithm [71]. This algorithm makes use of a priority-based decision structure. Its extension, the A*-algorithm [72], makes use of a heuristic to determine the most probable path continuation, saving on computation time. These algorithms originally come from graph theory but can be easily implemented when one considers the image as a graph where each pixel is a graph node. However, these algorithms, together with the constraints mentioned above, are bested in terms of computer run-time by a simple Viterbi algorithm [73], also known as *trellis* (Figure 2.8 a).

The simple version of this algorithm is dubbed *brute-force* (SPA-BF). Assume a *gradient image matrix I* of size $m \times n$, a *distance matrix d* and a *predecessor matrix pre* of the same size. Then, for every *node*, i.e. gradient pixel value, $I(a, 1)$ on the first column of the image, we can calculate a shortest path p to its corresponding end node $I(a, n)$ on the last row of the image by following algorithm:

Algorithm 1 Shortest Path Algorithm - Brute Force (SPA-BF)

1: **Initialization**:
2: *d(column(1))* $\leftarrow \infty$
3: *d(a,1)* $\leftarrow$ *I(j,1)*
4: **Propagation**:
5: **for** $row\ i \leftarrow 2, n$ **do**
6:     **for** $column\ j \leftarrow 1, m$ **do**
7:         $d(i, j) \leftarrow I(i, j) + min(d(i - 1, j - 1), d(i, j - 1), d(i + 1, j - 1))$
8:         $pre(i, j) \leftarrow row(min(d(i - 1, j - 1), d(i, j - 1), d(i + 1, j - 1)))$
9: **Backtracking**:
10: $p^n \leftarrow pre(a, n)$
11: **for** $row\ i \leftarrow n - 1, 1$ **do**
12:     $p^i \leftarrow pre(p^{i+1})$

For every pair of start and end nodes, a shortest path through the image is hereby calculated with the computational complexity $O(nm)$, which leads to a total complexity $O(nm^2)$, since for every shortest path (of which there are in total m) every single node in the image (of size nm) needs to be traversed. The shortest of all these shortest paths is the global shortest path through the image. This algorithm is called the *multiple search algorithm* [74].

Although the implementation of this algorithm is relatively straightforward, calculations on this order of complexity lead to a runtime of around 1200 milliseconds for a typical cell image of the Optical Stretcher, which is not fast enough for real-time detection. Added assumptions are needed to implement an improved version of the algorithm.

1. Since the start and end nodes lie on the same row, there is an upper bound to the number of nodes that need to be processed per shortest path.

2. If two shortest paths from different starting nodes a and b touch in a node x in the image, all shortest paths from the nodes between a and b will pass through x.
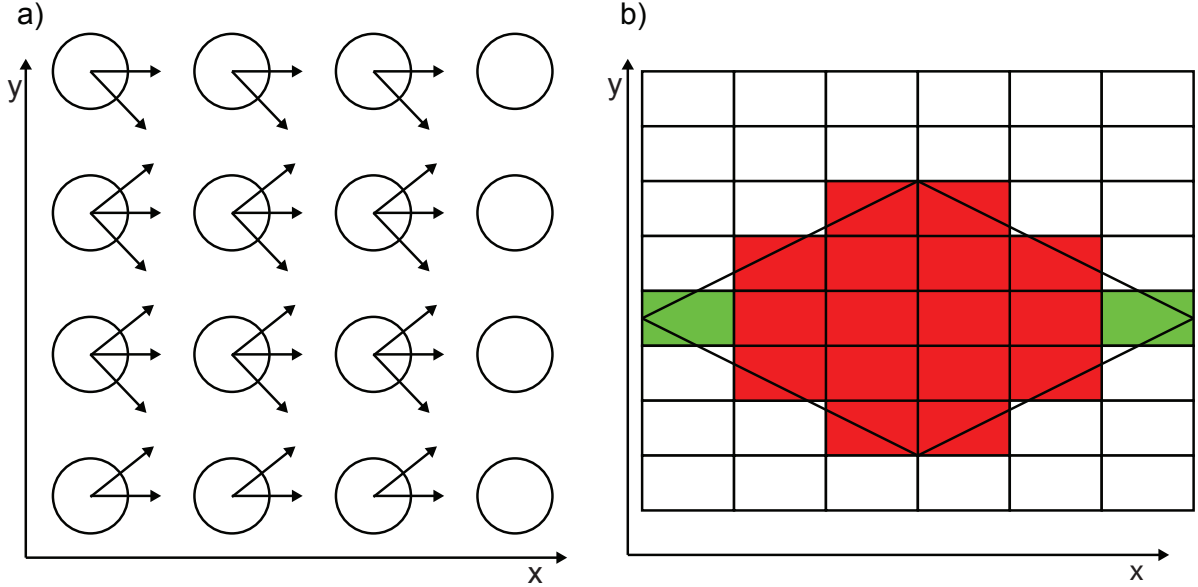
Figure 2.8: Trellis and rhomb constraint. a) A trellis-like structure. Each round node has three neighbors in the next column. b) Rhomb constraint: Starting and end node are in green. Since only three neighbors are allowed and the path is constrained, only nodes in red need to be tested for the path.

Proof of the first assumption: A *rhomb* shape (Figure 2.8 b) of traversed pixels is sufficient, since any node beyond the rhomb will not be close enough to the end or start node to reach it in the number of steps necessary due to the *no step back* constraint. The number of pixels that need to be traversed is thus reduced to maximally $0.5n^2$.

Proof of the second assumption: Let p_a and p_b be two shortest paths through the rectangular graph I, starting in a resp. b and ending in a' and b'. If p_a and p_b have two nodes x and y in common, they also also have a common subpath $s_a = s_b$ between x and y, because if $s_a < s_b$, then p_b does not have minimum cost, as its subpath s_b can be replaced by s_a.

Corollary: Shortest paths starting from differing nodes cross (i.e. touch at a single node and continue in differing directions) only once.

Proof continued: Any path p_c starting from a node c lying in-between a and b will by necessity touch p_a or p_b, if these touch. This means that p_c will have a common subpath with p_a and p_b. This subpath s_c will definitely pass through x and y, else p_c would cross p_a or p_b twice. This means that all shortest paths starting between a and b will have a common subpath that passes through x and y. This also means the shortest path of all shortest paths between a and b will pass through x.

Since the original image is circular, the rectangular graph can be shifted. This means a part of the left side of the rectangle can be cut and appended to the right side without changing the distance of the shortest path. This corresponds to cutting the circular original region radially at another angle. Since x is part of the global shortest path, the cutting and appending shifts the graph in such a way that x is the new

starting node of the shortest path algorithm. Calculating the shortest path from x to its corresponding end node results in a global shortest path for all starting points between a and b, the region A_{ab}.

This property can be used to significantly reduce calculations by a recursive algorithm. If two paths with differing starting points touch, there is no more need to calculate the paths of the nodes that lie between these starting points. If they do not touch, the path in the center between them is calculated to see if it touches one of them (see Figure 2.9 for a graphical explanation). This can be repeated until either all nodes have had their shortest path calculated, as in the brute-force method, or all starting regions of the image have a shortest path.

Algorithm 2 Shortest Path Algorithm - Recursive

 1: **Initialization**:
 2: *Choose two starting points a(0,1) and b(n,1)*
 3: *Perform rhomb-constrained SPA-BF with a and b*
 4: **Recursion**:
 5: **if** $\exists s_a = s_b, \, starting \; in \; x$ **then**
 6:     *Perform shift and SPA-BF with x*
 7:     *mark A_{ab} as checked*
 8: **else**
 9:     *Perform SPA-BF with c = (b-a)/2*
10:     *goto* **Recursion**

Once the shortest paths for all beginning nodes or regions have been calculated, the shortest global path can easily be chosen. In most cases considered in this work, the global shortest path is calculated within the first few iterations. For i iterations of the algorithm, the worst-case complexity is $O(nm(2^i + 1))$. A typical image needs 50 milliseconds to calculate, a 20-fold improvement over the brute force case. It can be thus used for real-time acquisition at a frame rate of up to $20Hz$. The algorithm can be sped up even more by not continuing to expand all nodes with a distance from the start that is larger than the straight line path from the start node to the end node, since any shortest path can be no longer than the straight line path, as this would then be the shortest path, or by using a *branch-and-bound* [75] approach.

For a typical cell, the process is this (see Figure 2.10):

1. In order to find the approximate center c of the cell, the image is thresholded automatically using Otsu's algorithm [67] and its first mathematical moment, a weighted average of pixel intensities, is calculated, which leads to the centroid.

2. A circle of user-specified radius r and center c is remapped to a rectangle.

3. The rectangle is filtered with a Scharr filter [76] and then inverted so the highest gradient has the lowest brightness.

$$G_{Scharr} = \begin{bmatrix} -3 & 0 & 3 \\ -10 & 0 & 10 \\ -3 & 0 & 3 \end{bmatrix}$$

4. Initialization: The first two boundary paths are calculated (rhomb-constrained SPA-BF). If they do not touch:
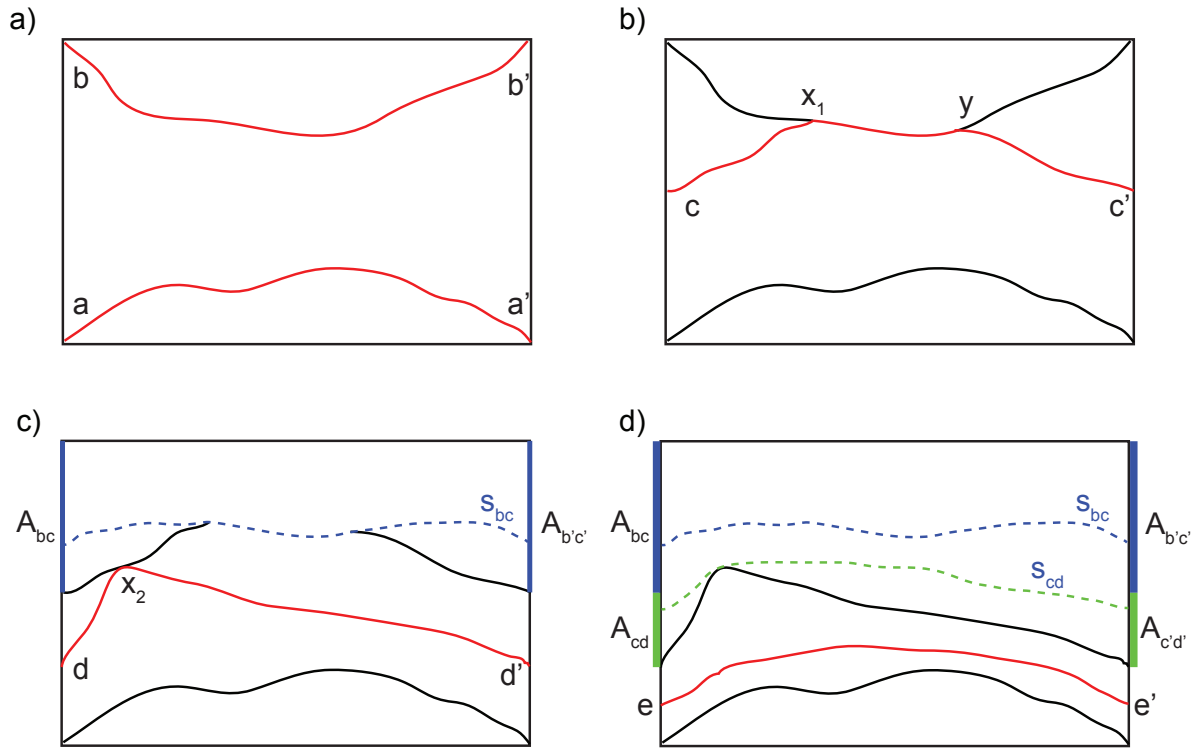
Figure 2.9: Recursive shortest path algorithm. a) **Initialization step:** Constrained shortest paths are calculated for graph boundary nodes a and b. b) **Recursion step 1:** As s_a and s_b do not touch, a new path at $c = (b - a)/2$ is added. c) **Recursion step 2:** s_b and s_c touch in x. The image is shifted and a regional shortest path s_{bc}, going through x is calculated. This is the shortest path for the region A_{bc}. A new path at $d = (c - a)/2$ is added. d) **Recursion step 3:** s_c and s_d touch, so a new regional shortest path s_{cd} is calculated for region A_{cd}. A new path at $e = (d - a)/2$ is added. Recursion continues.

5. Recursion step: The path of the nodes equidistant of the known paths are calculated. If it touches one of the boundary paths, this region's shortest path is recalculated. Repeat this step until all possible start and end nodes resp. regions are connected by a shortest path.

6. After all possible shortest paths have been calculated, the smallest, thus global shortest path is chosen and remapped to cartesian coordinates. It is the cell contour.

7. Using the Levenberg-Marquardt [77] algorithm, an ellipse is fitted to the contour. The major and minor axis of the ellipse are used for the deformation data that is required for an optical stretch measurement.
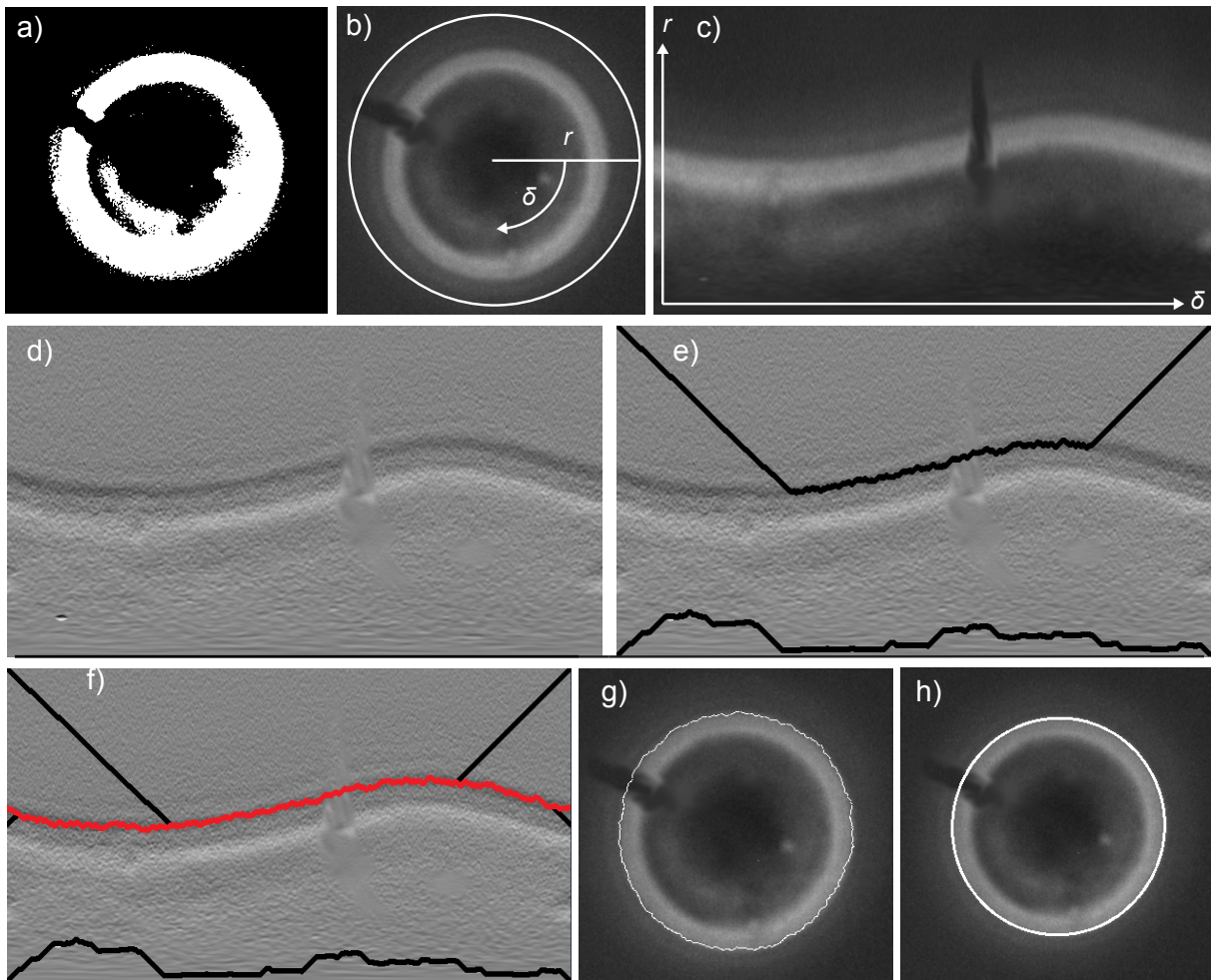
Figure 2.10: Recursive shortest path algorithm on a cell image. a) Automatic thresholding to find the approximate cell center c. b) The image is cropped by a circular region with radius r around c. c) The circular region is remapped in polar coordinates r and δ. d) The remapped image is filtered by a Scharr filter. e) Initialization: The boundary shortest paths are calculated. f) Recursion step: A new path between the previous ones is calculated. As it touches one of the other paths, a common shortest path (red) is calculated. g) The global shortest path is remapped to cartesian coordinates. h) An ellipse is fitted through the global shortest path.

This recursive shortest path algorithm can deal with a wide range of cell images, also those that are

blurred or noisy (Figure 2.11).
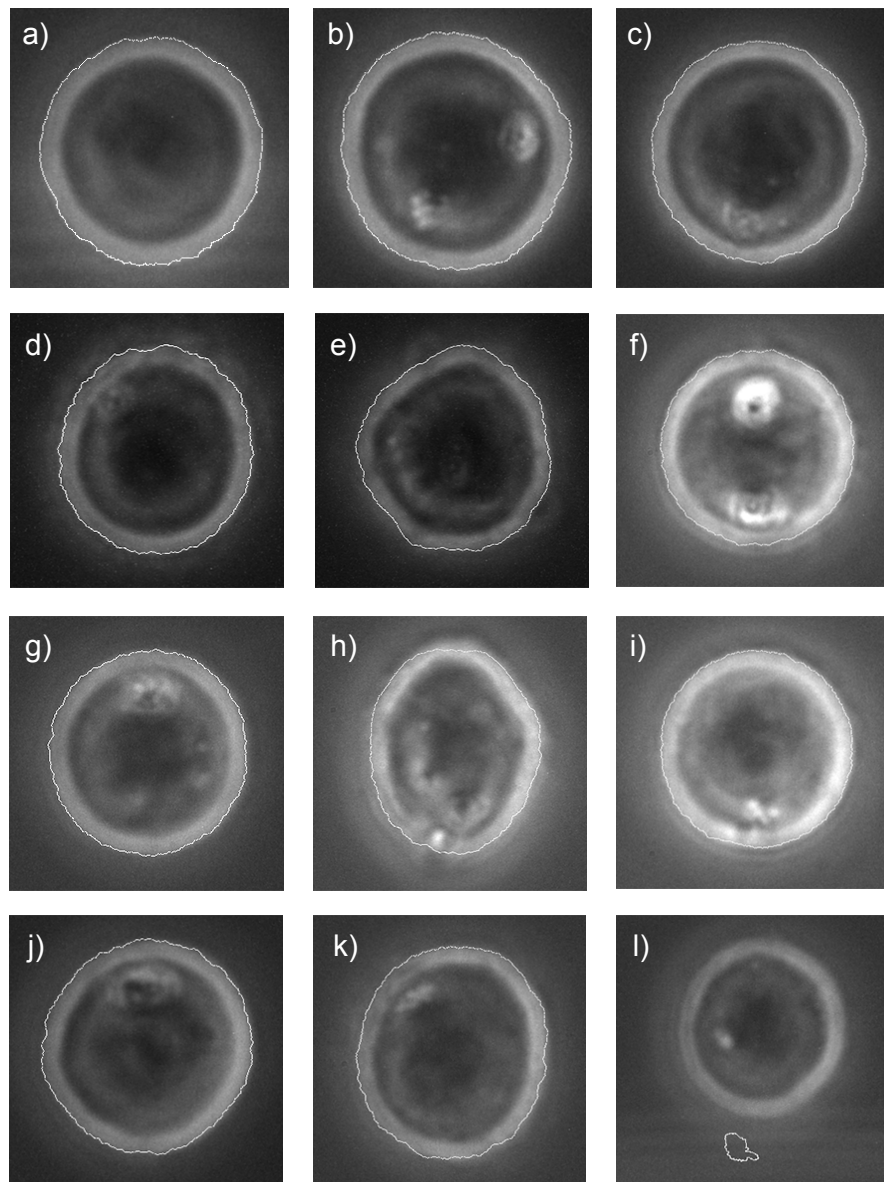
Figure 2.11: Cell boundaries detected with the recursive shortest path algorithm. a) - k) Cellular boundaries are detected for a wide range of cell shapes and illuminations. l) The effect of bad centroid choice: Failed automatic thresholding leads to the assumed center of the cell actually being outside the cell. The algorithm thus fails.

The benefits and drawbacks of the various segmentation methods are summarized as follows:

Method	Quality of detection	User input	Speed per cell	Complexity
Morphological Operations	bad	low	$\approx 2ms$	$O(nm)$
Canny Edge Detection	user dependent	high	$\approx 10ms$	$O(nm\,log(nm))$
Brute-force shortest path	very high	low	$\approx 1200ms$	$O(nm^2)$
Recursive shortest path	very high	low	$\approx 50ms$	$O(nm(2^i + 1))$

2.3 Refractive image acquisition

The refractive index of a cell not only plays a role in the calculation of optical forces acting on a cell in the Optical Stretcher [78, 6, 28, 29, 30], but also acts itself as a possible marker for diseases [79, 80, 81, 82] or cellular function [83, 84]. As mentioned at the beginning of the chapter, refractive index measurements were formerly done on a separate setup, the DHM [59], which made it impossible to relate an individual cell's refractive index to its deformation. The advent of the SID4Bio camera now makes it possible to do refractive index measurements on cells while they are in the Optical Stretcher setup. For this, it is sufficient to attach the camera to a free camera port on the microscope.

A detailed explanation of the digital holographic microscope and the algorithm that calculates the refractive index can be found in Section 7.1.3.3.

The SID4Bio camera differs from the DHM setup in significant ways. The fact that the DHM uses a coherent laser beam to measure phase (from which the refractive index is calculated as described in the Methods chapter) means that defocusing does not play a role in the measurement because the laser light passing through the sample is in parallel. In contrast, the SID4Bio uses a standard microscope light source, which follows the usual beam path. The light from the light source is focused onto the sample just as in standard imaging, because the phase information is only extracted at the modified Hartmann mask which is directly in front of the camera detector chip. This means that slight defocusing can change the measured refractive index dramatically. Additionally, the focal points of the standard light path and the SID4Bio are slightly different, leading to intrinsic defocusing when changing light paths in the microscope.

A point of interest is the effect of unwanted debris or other cells in the reference image on the calculation of the apparent refractive index of a cell. As mentioned, to calculate the refractive image of a cell, the SID4Bio camera needs an empty *reference image*, i.e. one without the object in question. This image is then subtracted from the *object image* of the cell in question, which ideally results in a phase image of a cell with a completely empty background. The background image will give information about the phase change of the light path of the microscope, while the image of an object taken will include information about the light path as well as the optical path of the object itself. Subtracting the reference from the object thus results in a phase image that only shows the phase change (which is given by *optical path thickness*) due to the object, which is used to calculate its refractive index. In theory, it would be sufficient to only subtract the background within an area of interest around the object. However, the SID4Bio camera appears to be hardware coded to do the following: after subtracting the reference from the image, the resulting image is offset in such a way that the phase information starts at 0. This means that if there are artifacts like other cells, dirt etc. on the reference image that are not present on the object, the optical path thickness of the phase image at those spots where those artifacts were is not negative, but zero. Consequently, the optical path of the actual object appears larger now since it has added thickness due to the added offset. These artifacts can be seen as *ghosts* in the resulting phase image. This highly affects the calculation of the refractive index by making it appear much larger (Figure 2.12). While a ghost of the size of a cell is easily detectable and can be dealt with by taking a new, empty reference, smaller and

less detectable objects can insidiously change the measured refractive index of a cell ever so slightly.

To counteract the effect of artifacts, the phase unwrapping algorithm program contains a ramp and offset correction algorithm [59] which fits the known background with a linear fit in x and y-directions. This generates an artificial background image which is a ramp with the slope and offset of the fit. This artificial image is subtracted from the object image and can correct for uneven backgrounds. However, if the artifact is too close to the object this correction will not be able to significantly improve results.
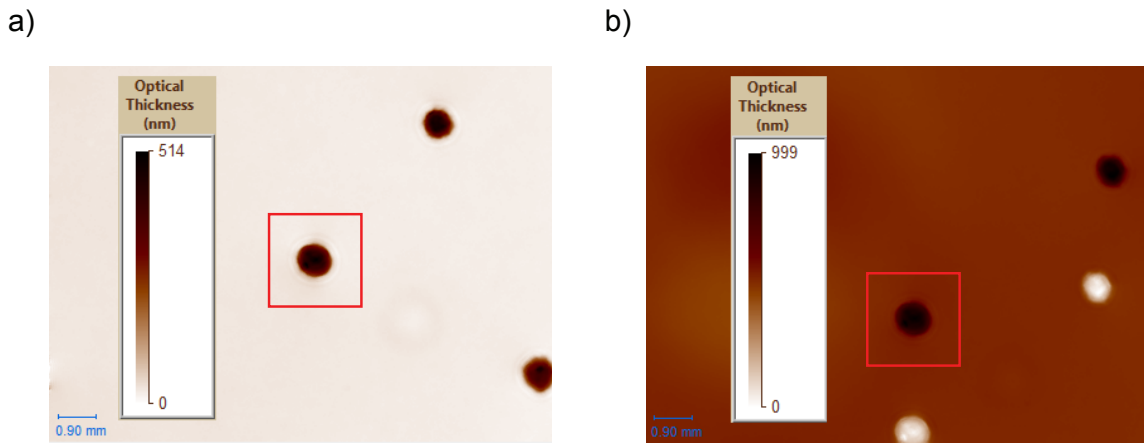
Figure 2.12: Effect of *ghost* artifacts on refractive index measurement. a) Phase image of a cell (marked by a red box) that was taken with an empty reference. The cell's apparent refractive index is 1.378 (background corrected 1.37493). b) Phase image of of the same cell, with itself as a slightly shifted ghost in white. Note the higher optical path thickness which results in an apparent refractive index of 1.437 while a background correction yields a refractive index of 1.37684.

Figure 2.13 a shows the effect of defocusing on the refractive index measurements with no background correction, Figure 2.13 b with background correction. Here, individual cells were imaged once in focus and once again after shifting the focus by 3μm upwards. For each cell and focal height, corresponding reference images were taken by shifting the microscope stage so that the image was free of cells. It appears that defocusing has a highly significant effect on the calculation of the refractive index. However, this defocusing actually mainly occurs because of bad background correction. The artifacts and inhomogeneities in bad background images confuse the algorithm that detects the cells' shape and lead it to believe the radius of the cell is larger. Accordingly, there appears to be a trend towards higher apparent cell radii and lower refractive indices, which makes sense because the cell's optical material would appear to be spread over a larger area/volume and thus be less dense. However, the changes in refractive index are sometimes quite drastic. The problem is additionally compounded by the fact that cells are living material and can change or move over time, also over the time needed to readjust the focus during the measurement. Utmost attention must therefore be placed on correctly obtaining a background image.
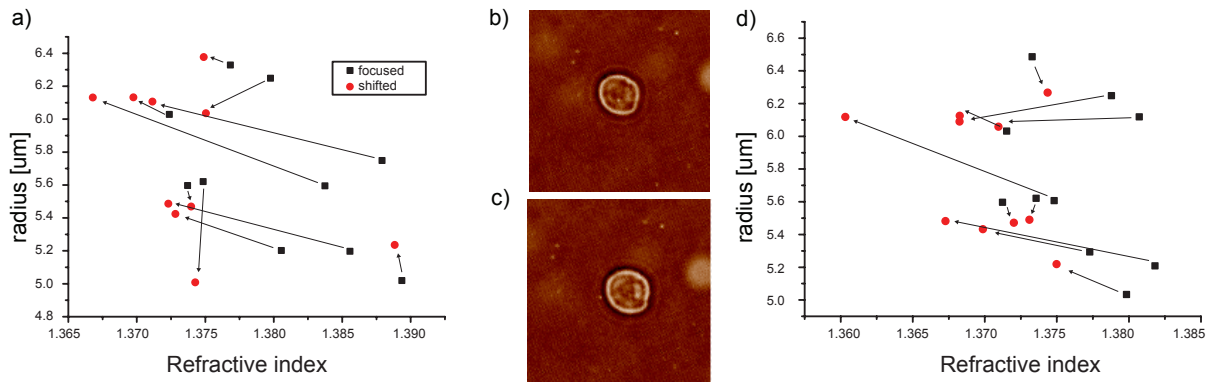
Figure 2.13: Effect of defocusing on refractive index measurements. a) Individual cells' refractive index is measured at focus f = 0 μm (black) and once again, shifted upwards by 3 μm (red). The arrowhead lines relate the measurements to each other. The background is not corrected. b) A cell in focus as seen by the SID4Bio camera. c) The same cell as in b) defocused by 3 μm upwards. d) The same cells as in a) with corrected background. Although measured refractive indices seem closer, artifacts still play a significant role.

If background correction works, the shape of the surrounding medium does not affect the result of the refractive index measurement. For normal image acquisition, the curvature of the surface of the medium plays a role in imaging as the curved interface between medium and air acts as a lens, which typically leads to blurring or shifting of the halo in the phase contrast case. This appears not to be the case for the SID4Bio camera because curvature effects are taken into account through the subtraction of the reference image. This was tested by measuring cells both in a very curved droplet of medium as well as on a confined volume that was large enough to have a flat surface in the middle (Figure 2.14 a).

Finally, stretching the cell itself might have an effect on the refractive index. One could imagine photo-toxic effects [85], or heating and denaturing of intracellular proteins that do not affect viability enough to start apoptosis. A quick experiment (Figure 2.14 b) where cells' refractive indices were measured before and after stretching indicates that there is no big change. However, these results should be taken with a grain of salt due to the very small sample size and artifact effects that are not detectable by eye.
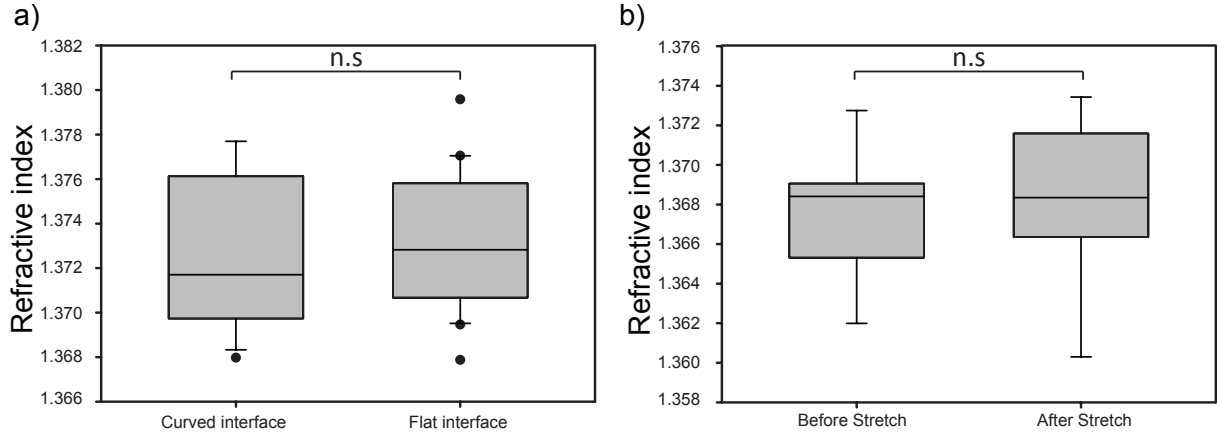
Figure 2.14: Effect of surface curvature and stretching on measured refractive index. a) Comparison of the effect of air-medium interface curvatures on HL60 cellular refractive index.
b) Stretching of Jurkat cells for 5s at 0.5 W per fiber does not appear to change the cellular refractive index significantly.

2.4 Global geometric factor computation

By using the image analysis program on each saved image frame during a cell stretch and by relating the obtained ellipse major and minor axes to the timing of the frames, a *deformation curve* is obtained. To relate this curve to mechanical parameters by fitting mechanical models, it must be transformed into a physical parameter first. For this, it is normalized by the average axis length a_0 during the initial resting phase of the stretch. This is the *time-varying axial strain*:

$$\epsilon(t) = \frac{a(t) - a_0}{a_0}$$

From this strain, the *creep compliance* of the cell over time can be calculated by

$$J(t) = \frac{\epsilon(t)}{\sigma_0 \, F_G}$$

The product of σ_0, the *optical stress* acting on the cell, and the *geometric factor* F_G [33] is the *global geometric factor* (GGF). The GGF is a complicated function of laser power, fiber distance, the cell's radius and the refractive indices of medium and cell. The software for the calculation of the global geometric factor is written in MATLAB, dates from 2010 [29], and takes between 5 to 30 minutes per calculation. Obviously this kind of time frame is not helpful for analysis of cells that are still in the system, and therefore, in the past, a global geometric factor was calculated for a cell population. For analysis in the system and per cell, actual calculation of the global geometric factor was not possible however.

To overcome this difficulty, a simple approach is to interpolate the cell's individual global geometric factor from a multidimensional grid of GGFs that are calculated ahead of time. In this case, for computational reasons, the laser power is fixed, while ideally it would be another independent variable of the

function that calculates the GGF. The current independent variables are

- *Fiber distance* d being the distance from one fiber end to the center of the cell, ranging from 50 μm to 150 μm in 10 μm steps,

- *Cell radius* r, the initial average axis length a_0, ranging from 5 to 10 μm in 0.5 μm steps,

- *Cellular refractive index* n_{cell} as measured in the system by the SID4Bio camera, ranging from 1.365 to 1.385 in 0.01 steps

It is worthwhile taking a closer look at the GGFs calculated and their dependence on the input parameters. Figure 2.15 a) shows that the GGF is less dependent on the refractive index than intuitively anticipated. In fact, it appears almost linear. This is surprising, because the force acting on the cell in the optical trap comes precisely from the difference between the refractive indices of the medium and the cell. In fact, since the optical stress acting on a particle can be calculated with geometric optics [8]:

$$\sigma_0 = \frac{n_{med} I_0}{c}(2 - R + R^2)(\frac{n_{cell}}{n_{med}} - 1), \ \ R = (\frac{n_{cell} - n_{med}}{n_{cell} + n_{med}})^2,$$

with I_0 the laser intensity, c the speed of light in a vacuum and n_{med} the refractive index of the medium (typically 1.335 for PBS), this indicates that the term R, the reflection at the interface of cell to medium, hardly plays a role in determining the optical stress. Presumably the linearity disappears once the cell's refractive index is much closer to that of the medium, as it is known experimentally that objects with similar refractive indices as the medium cannot be trapped in the optical trap.

Figure 2.15 b) shows a linear dependency of the GGF on the cell radius with a lower slope at higher fiber distances. This is likely due to the linear growth of the laser beam width ω over distance z [7] (Figure 2.15 d) with

$$w(z) = w_0 \sqrt{\frac{(\lambda z)^2}{(n_{med}\pi w_0)^2} + 1}$$

where λ is the laser wavelength and n_{med} the refractive index of the medium. The larger the beam diameter, the larger the cell radius can be for the cell to be completely within the laser beam. If the beam width is smaller than the cell diameter, the cell will not be entirely affected by the beam, which could lead to reduced optical stress and consequently reduced GGF. For very large fiber distances, this implies that the cell is entirely inside the laser beam no matter its diameter, which relates to the near-constancy of the GGF.

Figure 2.15 c shows an exponential dependency of the GGF on fiber distance. In general, this makes sense because the power per unit area decreases quadratically with the fiber distance due to beam width expansion.

Calculating the global geometric factor for each combination of values took a total of over six days of non-stop computation. These 330 values were stored as a three-dimensional grid, where each axis represents one input parameter and accordingly, the axis coordinate represents the input parameter value. This grid will be called *sample* grid. If the user, from the refractive index measurement and the cell deformation curve, obtains input parameters that lie in between the parameters used for calculation,
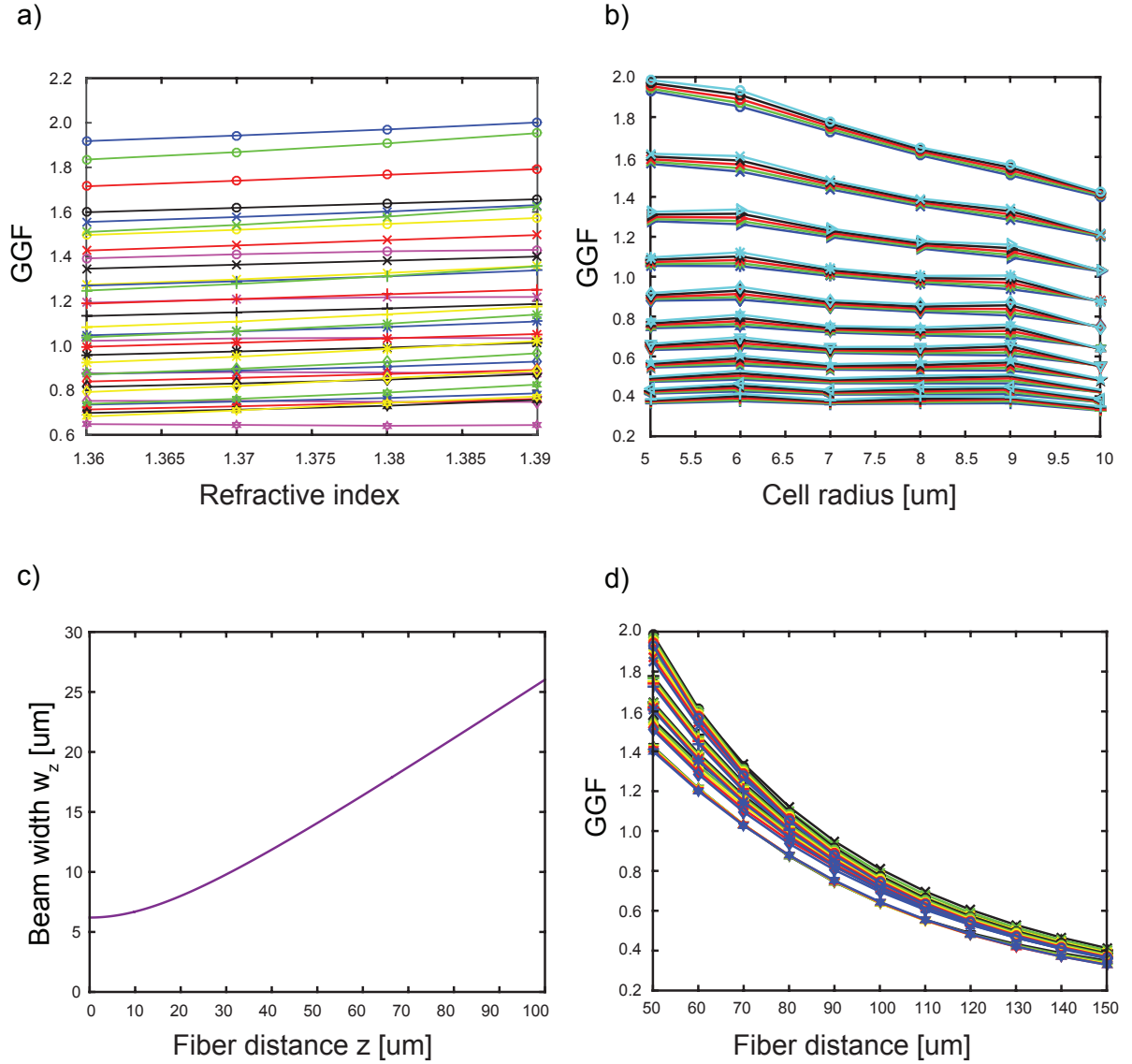
Figure 2.15: Calculation of the global geometric factor for laser power P = 0.75 W a) Global geometric factor as a function of refractive index. Different colors denote fiber distances (top = 50 μm, bottom = 100 μm), different symbols denote cell radii (top = 5 μm, bottom = 10 μm). b) Global geometric factor as a function of cell radius. Different colors denote refractive indices (top = 1.385, bottom = 1.365), different symbols denote fiber distances (top = 50 μm, bottom = 150 μm). c) Global geometric factor as a function of distance of fiber end to cell center. Different colors denote cell radii (top = 5 μm, bottom = 10 μm), different symbols denote refractive indices (top = 1.385, bottom = 1.365). d) Laser beam diameter in function of distance to fiber end. The starting beam diameter is 6.2 μm in accordance to manufacturer's values. λ = 1060 nm, $n_{med} = 1.335$.

the corresponding GGF can be interpolated from the sample grid. The interpolation used is *simple Kriging* [86], which is a method that calculates an interpolated value as a linear combination of sample values and corresponding *weights*. It assigns an optimal weight to each sample value by calculating the distance between the sample grid coordinates and the coordinates of the interpolated point as well as the distances between the sample coordinates. This means that closely clustered samples have less weight, otherwise their clustering would lead to a bias of the interpolated value towards the cluster.

X_0 is the value that is interpolated from the j known samples X_j of the function that calculates the GGFs, with the sample coordinates in three dimensions being the input values cell radius, refractive index and fiber distance.

$$X_0 = \sum_{j=1}^{n} \lambda_j X_j$$

where λ_j denotes the *kriging weight* or interpolation coefficient, which weights the contribution of each sample GGF to the interpolated GGF. In this case, the entire range of GGFs is used to calculate the interpolated value ($n = 144$). λ, the vector of kriging weights, is calculated by

$$\lambda = \mathbf{C}^{-1}\mathbf{b}$$

where C is the covariance matrix of the known input coordinates. Essentially, the covariance matrix describes all the distances between the sample coordinates. The element ij of the covariance matrix thus is the distance of element i to element j of the sample coordinates. b is also a vector of covariances which describe the distance of the interpolating coordinates (i.e. the parameters that are actually obtained during measurement) to all the sample coordinates. Essentially, λ is calculated with the samples in order to minimize the variance of the interpolated value with respect to the *actual* value at that position that is and remains unknown. This also means that Kriging interpolation only gives an estimated value that depends on the sample grid and not necessarily the value that would occur were one to recalculate the GGF function with the user input.

As a simple example, consider a 2×2 matrix

$$\begin{pmatrix} 3 & 4 \\ 13 & 14 \end{pmatrix}$$

which represents the sample values which were precalculated. These values can also be represented by their coordinates in two dimensions:

$$\begin{pmatrix} (1,1) \\ (1,2) \\ (2,1) \\ (2,2) \end{pmatrix} = \begin{pmatrix} 3 \\ 4 \\ 13 \\ 14 \end{pmatrix}$$

Now, the user wants to know the value at the coordinate (1.2,1.7). No calculated value is known at this point, so interpolation has to be used. The covariance matrix C of the distance between the sample coordinates and the covariance matrix b of the distance between this interpolating coordinate and the sample coordinates are thus

$$C = \begin{pmatrix} 0 & 1 & 1 & 2\sqrt{2} \\ 1 & 0 & 2\sqrt{2} & 1 \\ 1 & 2\sqrt{2} & 0 & 1 \\ 2\sqrt{2} & 1 & 1 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 0.728 \\ 0.361 \\ 1.063 \\ 0.854 \end{pmatrix}$$

This leads to the kriging weights

$$\lambda = \begin{pmatrix} 0.197 \\ 0.536 \\ 0.040 \\ 0.108 \end{pmatrix}$$

which are summed with the vector of the known values $(3, 4, 13, 14)^T$ to finally give the interpolated result 4.767.

Since this calculation only involves a few matrix multiplications, it can be done in real-time after acquiring the cell's deformation curve and results in an individual geometric factor that is used to determine a more accurate compliance curve.

2.5 Data analysis

Each cell has a set of parameters that define it individually, and a measurement of a cell will yield these parameters as was described in the previous sections. These parameters can be visualized as a point in n-dimensional space, where n is the number of independent variables and each dimensional axis corresponds to the value range of one these variables. A typical experiment will therefore lead to a point set in n dimensions. To detect possible subpopulations within the measured cells and correspondingly the point set, it can be split into subgroups by clustering techniques. The reverse is also possible. If there are already known subpopulations, an unknown cell can be classified as a member of a subpopulation by comparing its parameters to the subgroup's parameters.

Various clustering techniques, most notably the *k-means clustering* method [87] (Figure 2.16 a and b), which splits a point set into k subgroups by minimizing the distance of an object to its cluster's center, can be used to determine cell subpopulations.
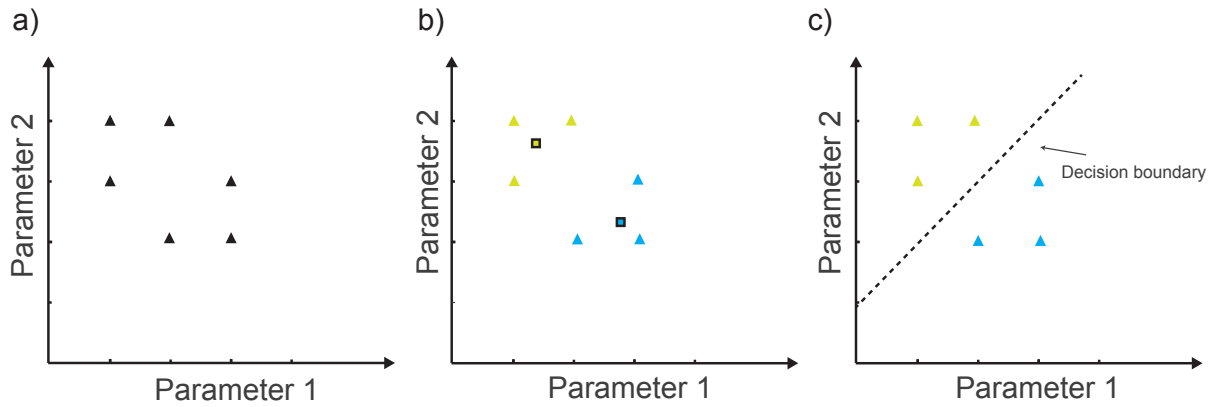
Figure 2.16: 2-D dataset classified by *k-means clustering*. a) A set of values with two parameters 1 and 2. b) Classification of data into two clusters using the k-means algorithm. Boxed points are the center of mass of each cluster. c) Linear discriminant analysis gives a boundary function $g(x)$ that can be used to classify new datapoints.

To classify individual cells into populations whose parameters are already known, the process is reversed. Subgroups of previous data have to be defined before each new cell in the trap is compared to the subgroups, upon which a decision is taken as to which subgroup the cell most likely belongs to.

2.5.1 Monovariate classification

Classification of a cell a, described by an independent parameter x, as a member of a class i from a total set of N known classes can be done by using *Bayes' theorem* [88]. If the data from the known experiments is normally distributed, a single boundary constant is sufficient to minimize the misclassification error. The probability that a cell described by the value x belongs to a certain class i can be written as

$$posterior = likelihood * \frac{prior}{evidence}$$

or

$$P(i \mid x) = p(x \mid i) \frac{P(i)}{\sum_{n=1}^{N} p(x \mid n)P(n)}$$

where *posterior* describes the probability that a cell with value x will be a member of class i. The *likelihood* $p(x \mid i)$ is calculated from the distribution of class i over value x (Figure 2.17 a) and describes the probability that a class i will have a cell of the value x in it. In this work, the likelihood is calculated prior to classification from experiments with a known cell population. The *prior* $P(i)$ describes the probability that the cell is a member of class i, regardless of the value of x (Figure 2.17 b). For equal cell concentrations per cell type, the prior accordingly is 0.5. *Evidence* describes the probability of the cell having a specific value x and depends on the distribution of x.

Now, a *risk function* R that describes the misclassification error when classifying a cell a with value x to class i can be calculated.

$$R(a_i \mid x) = \sum_{n \neq i}^{N} \lambda(a_i \mid x)P(n \mid x)$$

If one assumes that each misclassification is equally costly, the loss function $\lambda(a_i \mid x)$ equals 1 in case of misclassification and 0 in case of correct classification. In the two-class case with classes i and j, the risk function of a cell a that was classified to class i can be simplified to

$$R(a_i \mid x) = P(j \mid x)$$

Since one would like to classify cell a into the class with the least risk associated, one should decide to classify a cell as member of class i when $P(i \mid x)$ is larger than $P(w_j \mid x)$. It follows that if one writes this as the *minimum error discriminant function*

$$G(x) = \arg \max_{n} P(n \mid x)$$

it suffices to find the point x where $G'(x) = P(i \mid x) - P(j \mid x)$ changes its sign to find the boundary value for the classification regions. This can be done by a simple derivation of $G(x)$ (Figure 2.17 c). Now, all cells with values x below the boundary are classified to class i, while cells with values x above the boundary are classified to class j.

2.5.2 Multivariate classification

For classification of cells described by more than n variables, things become more complicated. The Bayesian approach can be used in a method called *linear discriminant analysis* (LDA). In principle, LDA assumes that likelihood functions $p(x \mid i)$ of the training set, i.e. the homogeneous cell populations, are normally distributed. Each training class i from a total of n_k classes with n_i data points x_i then has the *mean* and *covariance* parameters μ_i and $\Sigma_i = \frac{(x_i - \mu_i)^T (x_i - \mu_i)}{n_i - k}$. With normally distributed data, classification is optimal [89]. Often this normality assumption is violated, however. LDA has been shown to work well nonetheless [90].

The goal behind linear discriminant analysis is to maximize *between-class variance* while minimizing *within-class variance*. In layman's terms, one wants to find a boundary equation that cuts space into
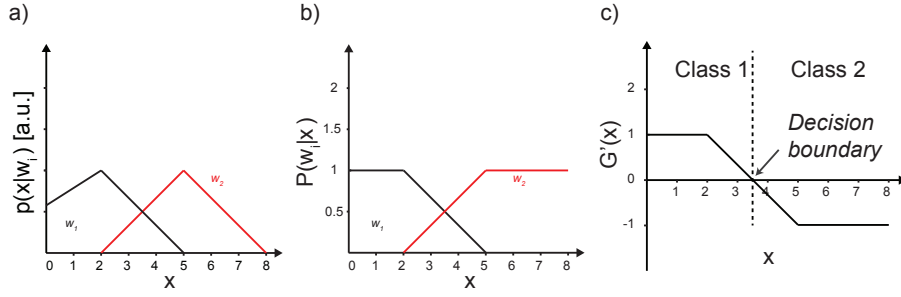
Figure 2.17: Classification of univariate data into two classes using Bayes' theorem. a) Likelihood functions of two training classes. The area under each curve is 1. b) Posterior probabilites of training classes. The prior is 0.5. c) Minimum error discriminant function $g(x)$. The classification boundary can be found at the zero crossing.

classes so that points in these classes are as closely clustered to their mean as possible, but as far away from the other classes mean as possible.

As mentioned, LDA assumes that training data is normally distributed. For a normal distribution of class i with the variable vector x in n dimensions the likelihood formula of the class is

$$f_i(x) = \frac{1}{(2\pi)^n \Sigma_i} \exp(-\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i))$$

Following Bayes' rule for optimal classification into k classes with prior $P(k)$

$$G(x) = \arg\max_k P(k \mid x),$$

where the posterior $P(k \mid x)$ is

$$P(k \mid x) = \frac{f_k(x)P(k)}{f(x)}.$$

$f(x)$ is independent of k and can be removed. Since the logarithmic function is strictly increasing, one can rewrite the minimum error discriminant as

$$G(x) = \arg\max_k \log(f_k(x)P(k)),$$

$$G(x) = \arg\max_k \left[-\log((2\pi)^n \Sigma_k) - \frac{1}{2}(x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) + \log P(k) \right]$$

For the two-class case with classes i and j, a cell with data x is assigned to class i if

$$-\log((2\pi)^n \Sigma_i) - \frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1}(x - \mu_i) + \log P(i) \; > \; -\log((2\pi)^n \Sigma_j) - \frac{1}{2}(x - \mu_j)^T \Sigma_i^{-1}(x - \mu_j) + \log P(j)$$

One more, very important assumption in LDA is that all covariance matrices Σ_i of each class are equal, $\Sigma_i = \Sigma$, $\forall\, i$. Σ is estimated in the two class case by:

$$\Sigma = P(i)\Sigma_i + P(j)\Sigma_j.$$

If the priors $P(i)$ and $P(j)$ are equal, this relation then simplifies to

$$(x - \mu_i)^T \Sigma^{-1}(x - \mu_i) \; > \; (x - \mu_j)^T \Sigma^{-1}(x - \mu_j)$$

.

The vector $x_c = (x_1, x_2)$ which solves the equation $P(i \mid x) \; = \; P(j \mid x)$ can be used to describe the wanted classification boundary:

$$\log \frac{P(i)}{P(j)} - \frac{1}{2}(\mu_i + \mu_j)\Sigma^{-1}(\mu_i - \mu_j)^T + x_c \Sigma^- 1(\mu_i - \mu_j)^T = 0$$

.

For two classes in two dimensions, the boundary function then has the form $g(x) = a_0 + a_1 x_1 + a_2 x_2$ with the coefficients.

$$a_0 = \log \frac{P(i)}{P(j)} - \frac{1}{2}(\mu_i + \mu_j)\Sigma^{-1}(\mu_i - \mu_j)^T$$

$$(a_1, \; a_2, \; \ldots, \; a_n) = \Sigma^{-1}(\mu_i - \mu_j)^T$$

As a very simple example, assume two classes of training data x_i and x_j in two dimensions (Figure 2.16 b). Each member of the class is described by a value of Parameter 1 and Parameter 2. If one wants find the optimal boundary between these to classes, one can see that just using the distance between the means along the parameter axes will not yield as good a result as if one placed the boundary exactly between the means. The goal is thus to find a boundary function that maximizes the distance between class mean while minimizing the distance of each class member to the class mean.

$$x_i = \begin{pmatrix} 1,3 \\ 1,4 \\ 2,4 \end{pmatrix}, \quad x_j = \begin{pmatrix} 2,2 \\ 3,3 \\ 3,2 \end{pmatrix}.$$

One can calculate the means and covariance matrices ($n = 3$ data points each, $k = 2$ dimensions)

$$\mu_i = (4/3 \;\; 11/3) \;\; \mu_j = (8/3 \;\; 7/3),$$

$$\Sigma_i = \frac{(x_i - \mu_i)^T(x_i - \mu_i)}{n - k} = \begin{pmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{pmatrix} \quad \Sigma_j = \begin{pmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{pmatrix}$$

The estimated covariance Σ is

$$\Sigma = \frac{1}{2}\Sigma_i + \frac{1}{2}\Sigma_j = \begin{pmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{pmatrix}, \quad \Sigma^{-1} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$$

Now, the coefficients a_n can be calculated by the formulas above:

$$a_0 = -\frac{1}{2}\begin{pmatrix} 4/3 + 8/3 & 11/3 + 7/3 \end{pmatrix}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}\begin{pmatrix} 4/3 - 8/3 \\ 11/3 - 7/3 \end{pmatrix} = -4,$$

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} 4/3 - 8/3 \\ 11/3 - 7/3 \end{pmatrix} = \begin{pmatrix} -4 \\ 4 \end{pmatrix}$$

The boundary between these classes can thus be described by the equation $-4 - 4x_1 + 4x_2 = 0$ (see Figure 2.16 c). It is possible to classify a new data point by simply inserting its values into the classifying relation. For example, the data point $x_{new} = (4, 3)$ solves

$$-4 - 4 * 4 + 4 * 3 = -8,$$

which is less than zero, so x_{new} is classified as belonging to class j.

The boundary function $x_2 = x_1 + 1$ splits the entire, in this case two-dimensional, parameter plane. To classify a cell into one of the classes, it is sufficient to know where the boundary lies. Now, different cell populations can be measured in different experiments, a boundary between those populations can be calculated, and another experiment can use this boundary to classify unknown cells.

2.6 Concluding remarks

The goal of this chapter was to show that it is possible to have continuous workflow from the moment a cell is placed into the Optical Stretcher. The cell is stretched and its deformation curve is calculated by a circular shortest path algorithm. The cell's refractive index is obtained by acquiring an image with a specialized camera. The combined information is used to calculate the geometric factor, a factor necessary for the determination of the compliance. The amount of time needed for all this is so short that the cell can be analyzed and classified into a known population while it is still in the Optical Stretcher. This offers hitherto unprecedented possibilities for experiments.

3 A monolithic chip for single cell sorting

As seen in the introductory chapter, the art of moving cells into the optical trap of the Optical Stretcher has evolved over time. From a droplet of liquid placed between two fibers, where cells sediment and occasionally float into the optical trap, control of cell movement has improved significantly, such as in a fixed glass capillary, where cells flow into the trap by manually applying a pressure differential between both ends, or even in a block of polydimethylsiloxane (PDMS) with structured microchannels, where flow is controlled by pumps.

These methods come with a drawback, however: they only offer limited functionality and cannot easily be adapted to new challenges. To put it in clearer terms, although a cell population can be stretched and measured, one cannot extract and use this same population subsequently. Either there is no possibility to extract cells at all or there is only one outlet, which means that the few hundred measured cells will subsequently be inextricably mixed with all other cells in the sample. In order to be able to extract measured cells, and only these, or possibly delve even deeper into the cell population and extract only the cells of interest that are part of the measured sample, multiple outlets, along with the ability to channel cells to leave via the required outlet, are needed. This calls for a redesign of the Optical Stretcher setup.

Why is sorting and extraction of particular cells interesting? Previously, a single Optical Stretcher experiment only measured homogeneous populations, which could then be compared to other measurements of different populations. Usually, a number of cells would be measured. After the actual experiments, their individual deformation would be calculated, "failed" stretches taken out of the data set and the stretch profiles of many cells averaged to give rise to a *deformation curve*, representing the entire population (Figure 3.1). Unfortunately, this does not do justice to heterogeneous populations. For example, what if there is a subpopulation within the measured population? It will be lost to the average. Even if a subpopulation is identified by chance, its measurement is already past. There is no possibility to retrieve the cells of interest and further determine what it exactly is that makes them different. What is needed is a method that sorts cells based on a defined parameter, and retrieves them.

Sorting of cells has been done before. One very well-known method is *FACS* (Fluorescence-activated cell sorting), which uses fluorescent markers in the cells to characterize them and which sorts by deflecting electrostatically charged droplets [91], but there are many more methods such as magnetic bead cell separation [92]. However, these methods need additional ingredients such as fluorescent markers or magnetic beads which could influence the cells' behavior [93, 94, 95, 96]. It would thus be interesting to have a device that can sort without needing markers.

As interest in cellular deformability as a description of cellular function has increased, so have the number of approaches proposed to sort heterogeneous cell populations based on this deformability. Most
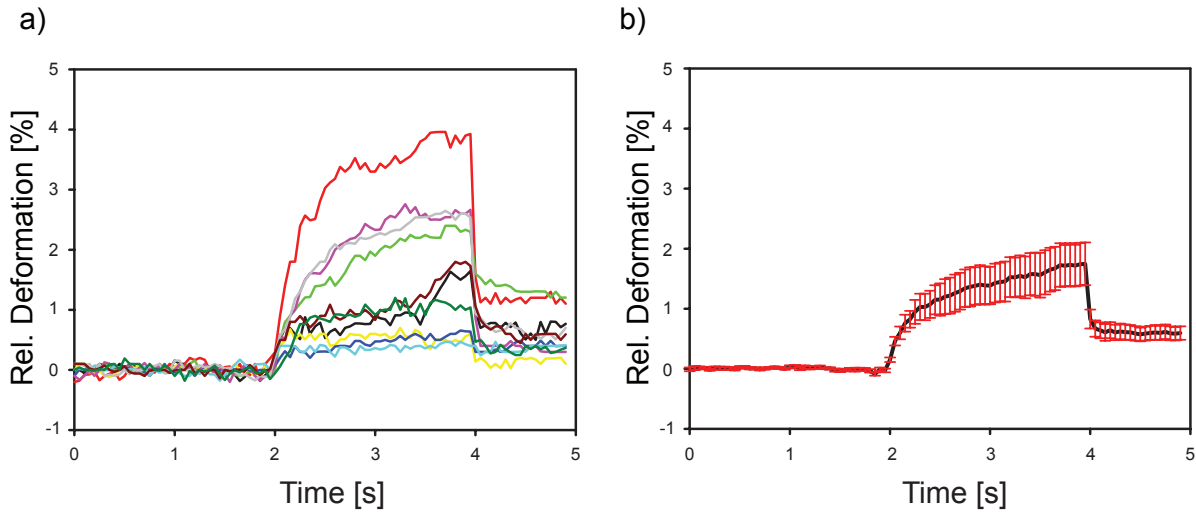
Figure 3.1: Individual cell stretches give rise to a population curve a) Ten individual cell stretches. b) Averaged deformation curve for ten cells.

of these are based on microfluidics, ranging from deterministic lateral displacement, where cells move along different streamlines depending on their size and deformability with respect to an array of posts on the chip [97, 98, 99], inertial microfluidics, where cells move along streamlines in specifically designed flow channels at high speeds [100, 101] or ratchet-based systems, where less compliant cells get stuck in gaps that are smaller than themselves [102, 103, 104, 105]. All of these methods have the disadvantage that they are *passive*. This means that during the design of a specific chip, certain cell properties, mainly their size and/or stiffness, have to be taken into account *a priori*, and that these methods usually only work well within a certain flow speed range. Most importantly, passive methods do not enable the user to change the parameters of sorting during an experiment or to reuse the same device for an entirely different type of experiment. Therefore, an active solution is needed, which allows the sorting parameters to be adjusted on the fly without any other change to the workflow such as the flow speed or the optical power.

These sorting methods obviously must have multiple outlets, but there is no reason not to also include multiple inlets to an Optical Stretcher sorting device. This would offer even more possibilities, making use of the laminar flow in the system. Side channels could create a sheath flow [106] that centers cells so that throughput is higher because more cells are trapped or chemicals could be added to create a chemical gradient [107], whose effect in varying concentrations could easily be tested. Simply said, more inputs and more outputs equal more possibilities of use, but also increased difficulty of flow control.

This chapter will report on the use of a chip to separate two distinct cell populations actively based on their cellular deformability without any additional markers. The work presented in this chapter was published in a peer-reviewed journal [108].

3.1 Design

To successfully build an Optical Stretcher, there are some necessary requirements which have to be taken into account. These can be categorized into three main points: optical manipulation, observation of cells, and the connection of the setup to the outside world.

3.1.1 Optical manipulation

The requirements in the category *optical manipulation of cells* are those that are most relevant to the function of the Optical Stretcher, i.e. the trapping and stretching of cells. To trap and stretch, laser light with a Gaussian beam profile needs to be delivered to the cell via a waveguide. Upon exiting this waveguide, the path of the beam to the cell must not be significantly deteriorated since any impediment of the beam will lead to lower trapping efficiency due to power loss or a less Gaussian beam shape. Any material between the waveguide end and the cell should have a high enough laser damage threshold to offer a long lifetime to the setup. Furthermore, since two counterpropagating laser beams are required for trapping and stretching, their axial alignment is a critical point. Finally, it is useful to adjust the height of the beams with respect to the microfluidic channel in which the cells will pass, thus ensuring that most of the cells passing will be trapped which raises the throughput.

To deliver light to a specific spot, optical fibers are usually used. Optical fibers let light propagate in one direction only by keeping it confined in a glass core with a high refractive index that is surrounded by a cladding with a low refractive index. Due to total internal reflection, the light cannot escape into the cladding and reflects along the core. Because of their very low power loss over large distances, optical fibers are mainly used in telecommunication, but can also be doped with rare earth elements and used as a fiber laser [109]. To obtain a Gaussian beam profile for stretching at the end of the waveguide, a single-mode fiber must be used since it will allow the fundamental transverse mode of the beam, which is the one with a Gaussian intensity profile, to pass. Commercially available fibers are cylindrical in shape and have a diameter of 125 micrometers for a laser wavelength of 1060 nanometers. Any design must therefore be based around this shape and a bulk material must have corresponding openings in which the fibers can lie.

How does one create structures in a bulk material? One option is to use a polymerizable material such as PDMS [39] that can be structured by casting on an appropriate master, but this comes with its own problems, most notably the low laser damage threshold. Instead, bulk glass is the material of choice because it offers a much higher damage threshold and stability. However, the structuring process is much more complex. Two pieces of bulk glass must be wet etched (see Methods, Section 7.1.1) which structures the glass, and thermally bonded to each other to seal the structures.

If only one piece of glass were etched, the waveguide channel would now also be hemispherical at the fiber end (Figure 3.2 a). A laser beam passing through a curved wall would be refracted upwards. More cells would pass underneath the beam, which reduces throughput and the Gaussian beam profile at the cell would be distorted. The solution to these problems consists in etching two separate glass pieces asymmetrically and bonding them thermally (Figure 3.2 b). The lower the fiber lies with respect to the

fluidic channel, the less curved the channel wall will appear to the beam. If the groove depths are chosen correctly, the fiber core is neither too high, which would reduce the throughput, nor too low, where the beam would be reflected off the bottom of the fluidic channel and possibly interfere with the trapping.
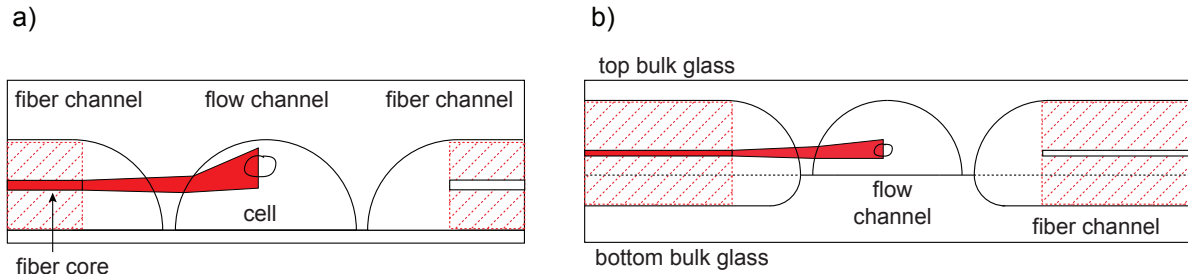
Figure 3.2: Asymmetric etching avoids beam path deflection a) One-sided etching alignment and beam path deflection. b) Asymmetric etching avoids obstacles in the beam path. [108]

Since the beam path, after exiting the optical fiber and passing through different materials and interfaces can be complex to determine, the optimal geometry can be calculated by using *finite-difference time domain* (FDTD) simulations. These were done in [108] by Estela Martín-Badosa.

Finally, the optical fibers have to be aligned exactly opposite from each other so that the optical stress profile acting on the cell is symmetric. When two glass pieces are bonded together, any slight misalignment of the fiber grooves of the upper and lower half can make it impossible to introduce the fibers into the chip, since there is not enough space. This is overcome by using three-point alignment technique: the upper channel is etched wider than the lower channel, giving it a flat surface. This ensures that the fiber sits securely in the lower glass half while the upper half has a tolerance of movement. The measures of the chip can be found in Figure 3.3

3.1.2 Observation of cells

Measuring the often very small deformation of cells makes it necessary for the image to be as undistorted as possible. Unfortunately, any rounded edge such as the etched fluidic channel leads to fuzziness in the image. To avoid this as much as possible the fluidic channel, in contrast to the optical channels, was only etched into one glass half, so that the thermally bonded second half of the bulk glass acts as a flat bottom for the optical channel, reducing the distortion.

3.1.3 Connection to the outside

The monolithic glass chip has two unconnected channel systems: one for fluidics, one for optics. The fluidic system must transport the cells from the sample reservoir to the trap and from there to the correct outlet. In order to connect sample and tubing to the glass chip without leakage, an 8-way connector is tightly clamped against the chip. This also allows access to the optical channels which aew meant as

a)   b)
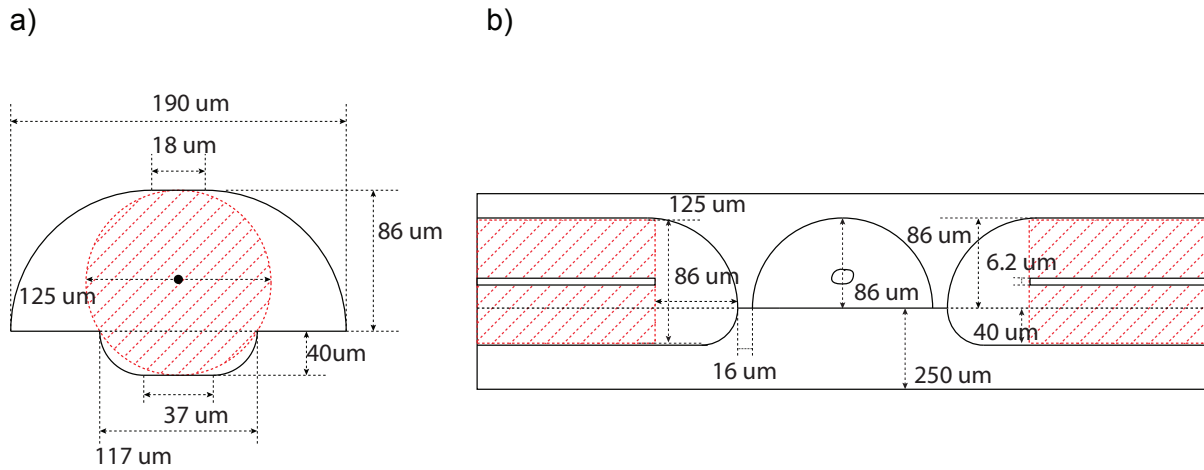
Figure 3.3: Measures of the final design a) Three-point fiber alignment. b) Side view. [108]

a point of introduction of refractive index matching gel or oil to decrease the refraction between fiber ends and channel walls. The fibers themselves are introduced on the side of the chip into fiber channels. By making the lower half of the chip somewhat larger than the upper half, the fiber can be laid into its corresponding groove and simply pushed in, instead of tediously trying to fit each fiber into a 125 micrometer wide hole. Finally, the chip is placed into an aluminum frame and fixed, adding stability. Subsequently, it can be placed on a microscope with the help of a custom frame (Figure 3.4).

The chip with three inlets and three outlets was designed by Franziska Lautenschläger [108] and was produced by Dolomite Microfluidics (Royston, UK).

a)   b)
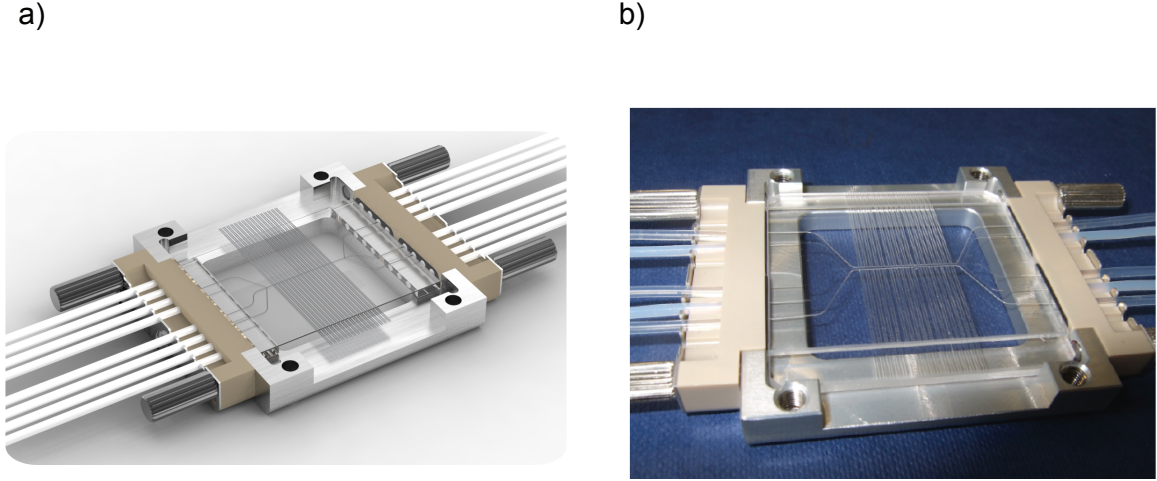
Figure 3.4: Produced monolithic chip a) CAD drawing of chip. Image courtesy of Dolomite Microflu-
idics Ltd. b) The finished chip in its 8-way connector and holder. One can see both fluidic
and optic channels.

3.2 Microfluidic constraints

Apart from the chip design itself, one must think about the workflow during a typical experiment. For
the control of flow of the different in- and outlets, a simple hydrostatic pressure system such as used for
the single capillary setup [7] is not enough. In order to determine what kind of pump to use to control
multi-channel fluidic devices, it is useful to calculate the fluidic resistance of the channels to determine
the pressure range that is applicable [110]. The pressure drop ΔP across a fluidic channel with area A
and length L is calculated, analogous to an electric circuit, as

$$\Delta P = R_h Q,$$

with

$$Q = vA,$$

where R_h denotes the *hydrodynamic resistance*, analogous to electric resistance, and Q the *flow rate*,
with the *flow velocity* v. Since the system should work in a laminar flow regime, the Reynolds number

$$Re = \frac{vL\rho}{\mu},$$

with ρ the *fluid density* and μ the *dynamic viscosity* of the fluid, should be smaller than 1000. We can
insert these values to get

$$\Delta P < \frac{1000 R_h A \mu}{L}.$$

The hydrodynamic resistance R_h of a circular channel with radius r is calculated as

$$R_h = \frac{8\mu L}{\pi r^4},$$

while that of a semicircular channel can be approximated [111] by

$$R_h = \frac{4}{3}\frac{\mu L}{\pi}\frac{13}{r^4}.$$

Since the fluidic resistance scales inversely quartic, obviously the fluidic resistance of small microchannels plays a much greater role than the attached large inner diameter tubing. Fluidic networks can be written analogous to electric circuits, so for calculation a microfluidic chip with one inlet and two outlets can be described as in Figure 3.5.
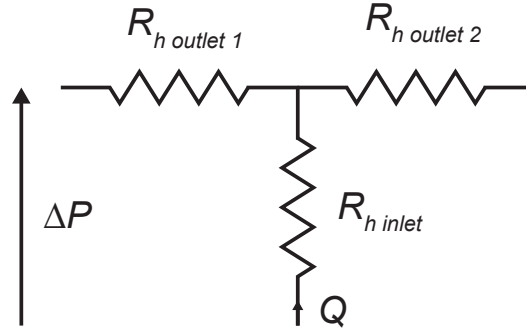
Figure 3.5: Equivalent circuit for a simple microfluidic chip. Each channel can be described as a resistor.

For a proof of principle, it is assumed that the channels all have the same dimensions and a typical $r = 50\mu m$, so the pressure is halved over the final channel. The maximum pressure for the laminar flow regime can thus be calculated for water ($\mu = 1mPas$, $\rho = 1kg/m^3$, $r = 50\mu m$) as

$$\Delta P < \frac{2000mu\pi r^2}{L\rho}\frac{8\mu L}{\pi r^4} < 6400Pa < 0.64Bar.$$

Any pump must thus be able to control pressure in channels with accuracy in the millibar range to guarantee laminar flow. Standard syringe pumps do not offer this kind of accuracy. For the multichannel fluidic chips, a pressure pump with multiple 0-25 mbar channels is used. This allows the creation of differential pressure over the chip. In fact, the pressure differences needed to control cell flow in this chip is within the microbar range. This pump also offers the possibility of purging at high pressure, which is useful for cleaning the fluidic channels.

In the proof of principle experiment, the pressure driven pump with multiple channels could easily be controlled via software. Only one inlet channel and three outlet channels were used, the other two inlet

channels were blocked since they were not needed. All open channels were connected to enclosed reservoirs via the 8-way connectors and microfluidic tubing. Cells were be placed in a reservoir connected to the inlet. Then they were made to flow into the tubing and inlet, while a counterpressure was applied to the two side outlets. This ensured that stray cells would not enter the side outlets which would later be used for extraction of sorted cells. If a cell had entered a side channel before the actual experiment, it would have distorted the results. In contrast to the previously used manual method which involved moving two filled reservoirs up and down in order to create a pressure difference between them, the pressure controlled pump could be set to zero flow instantly. This allowed the user to quickly start and stop flow and trap a cell without needing to adjust the pressure difference constantly.

Initially, the idea was to sort cells after measurement by changing the pressure acting on the outlets so that the cell in question would take a preferred exit. It was soon realized that the sorting region of the chip was too far away from the measurement and decision-making area to be imaged at the same time . Using pressure to sort was thus not an optimal solution. The flow profile of such a microfluidic channel being parabolic (Figure 3.6 a) with velocity v_x in x-direction

$$v_x = v_{max}(1 - \frac{r^2}{R^2}),$$

where v_{max} is the maximum speed in the center of the channel, cells in the center of the channel flow faster than those at the edges. In theory, after trapping in the optical trap, all cells should be centered, but in practice, less spherical cells do not align exactly in the center or heavier cells float towards the bottom of the channel faster than lighter ones. This means that some cells flow faster than others and overtake each other at some point. Since one cannot monitor the entire length from the optical trap to the outlets, it is no longer guaranteed that the correct cell flows to its prescribed exit. The solution to this problem is to add another step after stretching: turning off one or the other side of the lasers applies a scattering forward force [44] to the cell and pushes it to the left or right channel wall of the trapping region. The laminar flow regime ensures that all cells on a streamline actually flow into their corresponding outlet (Figure 3.6 b). This way, cells that are pushed onto the same streamline cannot overtake each other and it is guaranteed that all cells on the same streamline exit the chip along the prescribed outlet.

a)

b)
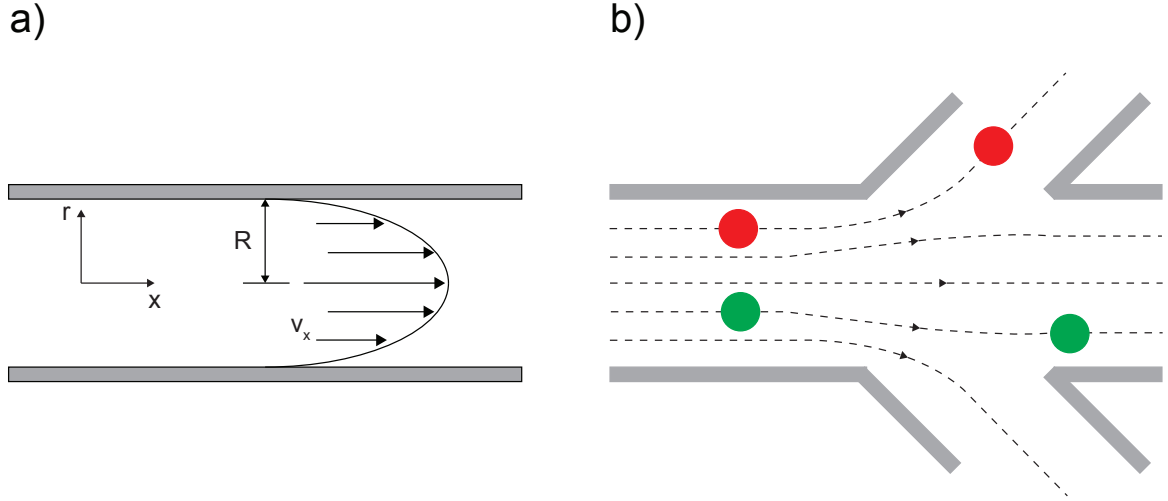
Figure 3.6: Parabolic flow and streamlines a) Parabolic flow in a cylindrical channel. b) Depending on the location of an object in the single channel, streamlines will carry it to different outlets.

3.3 Results

3.3.1 Design effects

The design elements previously mentioned each have an effect on the performance of the chip.

3.3.1.1 Fiber alignment and insertion

With every tested device, the three point alignment ensured that fiber alignment was near-perfect, which guaranteed that the optical stress profile acting on a cell in the trap remains symmetric. The height of the optical fiber core, 22.5 μm above the flow channel surface, was sufficient to trap practically every passing cell. However, a critical spot is the insertion of the optical fibers into their appropriate guiding channels. Since the opening is very small and the glass fiber is not robust, it can often break at the exit of the channel and prove impossible to remove, rendering this specific channel useless. Utmost care must thus be used when inserting the fiber into the device, and it must be secured so that it does not snap when moving the device. This can be done by placing modeling putty on top of the channel exit which holds the fiber in place.

3.3.1.2 Beam path deterioration

The beam path deterioration that was expected to be introduced by the curvature of the channel wall and the height of the laser fiber core with respect to the optical axis ($r = 22.5$ μm) proved to have little effect on the quality of the stretch. There are only rare incidences of cells not being able to be trapped. Comparison of the reflection coefficients at an interface calculated via the Fresnel equations

$$R_{\mathrm{s}} = \left| \frac{n_1 \cos \theta_{\mathrm{i}} - n_2 \cos \theta_{\mathrm{t}}}{n_1 \cos \theta_{\mathrm{i}} + n_2 \cos \theta_{\mathrm{t}}} \right|^2, \quad R_{\mathrm{p}} = \left| \frac{n_1 \cos \theta_{\mathrm{t}} - n_2 \cos \theta_{\mathrm{i}}}{n_1 \cos \theta_{\mathrm{t}} + n_2 \cos \theta_{\mathrm{i}}} \right|^2, \quad R = \frac{R_{\mathrm{s}} + R_{\mathrm{p}}}{2},$$

where θ_i and θ_t are the incoming and transmitted angle of the beam with respect to the interface and n_1 and n_2 are the indices of refraction of the material the beam exits and enters, respectively, show that the reflection coefficients at each interface are very slightly higher for the asymmetrically bonded chip (incident angle $\theta_i = arcsin(\frac{22.5\mu m}{86\mu m}) = 15.2°$)in comparision to the state-of-the-art device ($\theta_i = 0°$). For $n_1 = 1.4646$, the optical gel refractive index, and $n_2 = 1.5229$, the refractive index of borosilicate glass, the reflection coefficients differ by 0.0003 % which is neglegible for power loss.

However, the curvature of the channels and the additional material (optical gel and bulk glass) in the beam path do lead to an increase of the beam radius at the trap position. This can be shown by use of *ray transfer matrix analysis*, which takes into account all optical elements within the beam path (Figure 3.7 a). Each optical element is described in form of a matrix M. The first expositions [78, 112] refer to centered optical systems, using a 2x2 matrix system M with four coefficients $ABCD$. This formalism has been extended to a misaligned optical systems (i.e., the beam not running along the optical axis of the system) using a 3x3 formalism [113, 114]. Propagation through a homogeneous material of thickness z_{prop} is described as a matrix

$$M_{prop} = \begin{pmatrix} 1 & z_{prop} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

while a beam passing a curved interface with curvature radius R at a displacement r with respect to the optical axis of the system is described as a matrix

$$M_{interface} = \begin{pmatrix} 1 & 0 & 0 \\ \frac{(1-\frac{n_1}{n_2})}{R} & \frac{n_1}{n_2} & 0 \\ \frac{(n_1-n_2)r}{R} & 0 & 1 \end{pmatrix}.$$

The whole beam path from fiber end to cell then can be described by multiplying individual matrices in the reverse order of the appearance of optical elements of the chip:

$$M_{tot} = M_{medium} * M_{interface:glass-medium} * M_{glass} * M_{interface:gel-glass} * M_{gel} =$$

$$\begin{pmatrix} 1 & z_{med} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ \frac{(1-\frac{n_{glass}}{n_{med}})}{R} & \frac{n_{glass}}{n_{med}} & 0 \\ \frac{(n_{glass}-n_{med})r}{R} & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & z_{glass} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ \frac{-(1-\frac{n_{gel}}{n_{glass}})}{R} & \frac{n_{gel}}{n_{glass}} & 0 \\ \frac{-(n_{gel}-n_{glass})r}{R} & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & z_{gel} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} A & B & 0 \\ C & D & 0 \\ G & H & 1 \end{pmatrix}$$

with the distance values $z_{medium} = z_{gel} = 82 \ \mu m$, $z_{glass} = 24 \ \mu m$ and the refractive indices $n_{gel} = 1.4646$, $n_{glass} = 1.5229$, $n_{medium} = 1.335$.

Now, the *complex radius of curvature* $q(z)$ of a Gaussian beam is known [114] as

$$\frac{1}{q(z)} = \frac{1}{R(z)} \frac{2i}{\beta \omega_z}$$

with $R(z)$ the radius of curvature of the beam, ω_z the beam waist size at z and β the wave number

$$\beta_0 = 2\pi \frac{n_{gel}}{\lambda}$$

of the Gaussian beam at the input plane, i.e. the optical gel.

$u(z)$, the solution of the differential equation describing a plane wave propagating in z-direction [113], can be written in matrix form as

$$\begin{pmatrix} u(z) \\ \frac{u(z)}{q(z)} \\ S(z)u(z) \end{pmatrix} = \begin{bmatrix} A & B & 0 \\ C & D & 0 \\ G & H & 1 \end{bmatrix} \times \begin{pmatrix} u(0) \\ \frac{u(0)}{q(0)} \\ S(0)u(0) \end{pmatrix}$$

with S being the displacement of the beam center with respect to the optical axis of the system, so that $S(0) = r$. Dividing the second line of the equation by the first line, results in the Kogelnik [78] transform describing the propagation of the complex radius of curvature in z-direction.

$$q(z) = \frac{Aq(0) + B}{Cq(0) + D}$$

As one can tell, the parameters $S(z)$ and $u(z)$ do not play a role for the calculation of the complex radius of curvature. Solving $q(z)$ for an original displacement of $r = 22.5 \ \mu m$ and the optical element curvature $R = 86 \ \mu m$, as well as the refractive indices and propagation lengths already mentioned above, leads to a beam width value of $\omega_z = 15.62 \ \mu m$ at the center of the microfluidic channel where cells are trapped. This value is slightly larger than the beam width at the center of the state-of-the-art capillary design [7], which is around $12 \ \mu m$. However, since the area of the beam is practically doubled, this means the incident power per area is halved, an explanation for why more power than in the previous setup is needed to trap and stretch cells. This also means that fluctuations in the microfluidic flow which can occur when flow is not perfectly zeroed can push cells out of the trap more easily.

Another possible, much more critical beam path deterioriaton possibility arises from the fact that the the channels where the optical gel is inserted are open to the outside. This means dust or air bubbles can aggregate in the gel in the channels and settle between the fiber end and glass channel wall. If this dust is not detected and replaced by fresh optical gel, the dirt will heat up and rapidly expand, causing critical damage to the structure (Figure 3.7 b).

3.3.1.3 Observation quality

The optical quality is sufficient to obtain good images for contour segmentation. The curvature of the microfluidic channel wall and the consequent refraction of the illumination light seem to have little effect on image quality. When viewing cells at high magnification under phase contrast illumination, refraction is mainly visible as a slightly blurry halo around the cell (Figure 3.8 a). This blurriness is usually insignificant for boundary detection except for very small deformations. However, care must be taken to
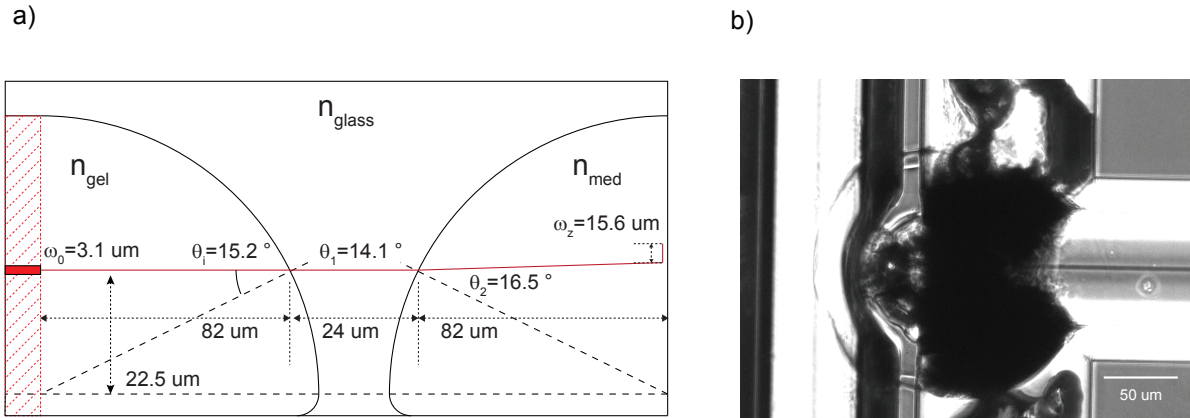
Figure 3.7: Beam path in monolithic chip and critical failure a) Beam path calculations for monolithic chip. b) Dirt in the optical gel within the chip can cause an explosive critical failure.

balance the laser power so that the trapped cell is centered exactly in the center of the fluidic channel, otherwise the change in illumination near the channel walls (Figure 3.8 b) will bring the halo of the cell out of the center and wreak havoc with the contour detection algorithm.
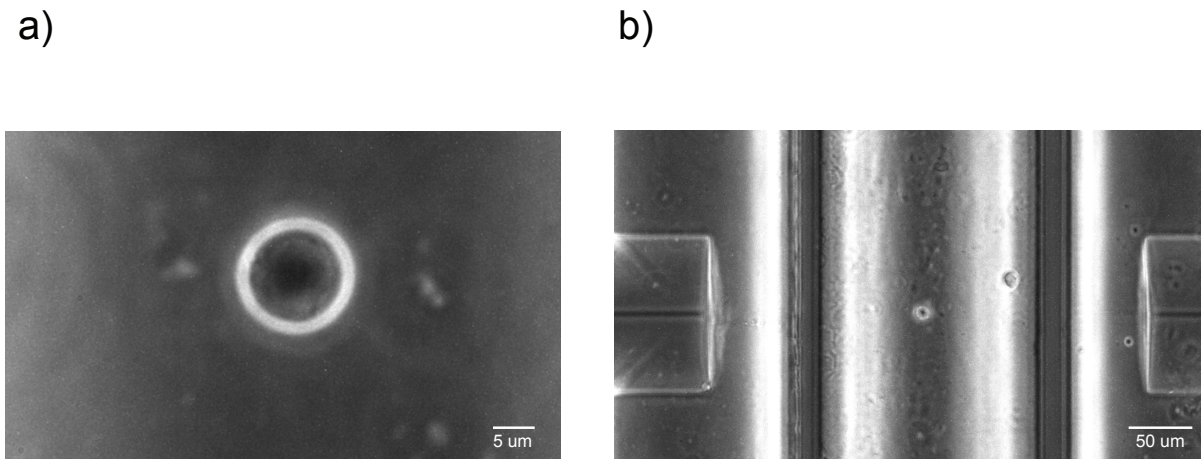
Figure 3.8: Optical quality of cells in monolithic chip a) A trapped cell viewed by phase contrast microscopy at 40x magnification. b) A cell viewed at 10x magnification. Note the optical effects near the channel walls due to the channel's curvature.

3.3.1.4 Microfluidic flow and extraction

Microfluidic flow is easily controllable using the microfluidic pressure pump. After sorting, laminar flow conditions ensure that cells remain on their streamline until they are separated at the junctions of the chip. However, extraction does not appear to be possible. Cells that are sorted are not retrievable in their destination reservoir. It is likely that the connections from the chip to the reservoir bring too much

dead volume into the system. Cells that leave the microfluidic chip can get lost in the region between the 86 μm radius and 250 μm tubing radius. Even though the tubing is clamped watertight to the chip via the 8-way connector so that no liquid can get lost, the seal region inside the 8-way connector does not guarantee that the tubing directly touches the chip, which adds a possible 800 μm radius sealing region. Here, laminar flow might be disturbed and cells may get stuck in corners or attach to the seal. In short, there is no simple way to extract such a small sample from the device.

3.3.2 Mechanical phenotyping of single populations

To prove the functionality of the monolithic chip, differing cell populations were measured individually as a preliminary experiment. In this case, the cell type was the human promyelocytic leukemia cell HL60/S4. Due to their spherical shape in suspension, they are optimal for studies in the Optical Stretcher and have been extensively used [24, 14, 17]. One population was untreated, while the other was treated with the cytoskeletal drug cytochalasin D. Cytochalasin D is a reversible actin polymerization inhibitor that disrupts the formation of actin filaments inside the cell [115]. It was thus expected that treated cells would appear more compliant due to their disrupted cytoskeleton.

3.3.2.1 Workflow

A typical experiment commences by filling the chip and connecting tubing with phosphate buffered saline upon which a vial containing the sample is connected to the microfluidic pump. The flow speed is controllable via this pump. Once a cell is near the trapping region, the lasers are turned on to a trapping power of 0.2 W per fiber while flow is set to zero. After trapping the cell, a wait for a few seconds is needed for the cell to align itself along its major axis. Then, the actual measurement begins. The cell is kept at trapping power for two seconds, then the power per fiber is raised to 1.2 W per fiber and held there for 2 seconds, after which the cell relaxes for one second at trapping power again. During the pre-experiments with homogeneous populations, cells are then released by turning off the lasers, while during the sorting experiments, power is turned off only on one side for 10 seconds, after which the cell is released and flows out. All cells were measured within the 2 h that followed taking them out of the incubator. One part of the cells was left untreated, the other was treated with 0.2 μM of the cytoskeletal drug cytochalasin D and incubated for 15 minutes at 37°C. In both cases, cell suspensions from both samples were then removed by centrifugation, upon which both samples were resuspended in *RPMI 1640* medium. Each sample was measured in the Optical Stretcher for 30 minutes each. Since the experiments were repeated three times each for the untreated and the treated cells, each group (treated/untreated) had a total number of around 100 cells. The average refractive index of treated and untreated cells was measured by using a digital holographic microscope, which showed that there was no significant difference in refractive index (untreated cells = 1.379 ± 0.0036, cytochalasin-D treated cells = 1.380 ± 0.0056, $p = 0.288$).

3.3.2.2 Viability of cells in the optical stretcher

To see whether cells remained viable in the Optical Stretcher, 10% Trypan blue solution was added to cell suspensions before insertion into the Stretcher [116]. After 30 minutes, observation of trypan blue exclusion was an indicator of cell viability. Cells with an intact plasma membrane exclude trypan blue,

while damaged and apoptotic cells have a detectable trypan blue uptake. This is visible by the cell's blue coloring (Figure 3.9 cannot show blue uptake due to monochrome camera). The measurements show that the optical stretching has no effect on trypan blue uptake ($n = 50$) in the time scale measured. However, it remains to be seen if the cell are actually viable in the long term, as there are definite effects on cells during stretching (as noted in Figure 3.9 b, where the cell remains more elliptic than an unstretched one). In future experiments, viability could be tested more precisely by extracting cells and recultivating them for proliferation experiments.
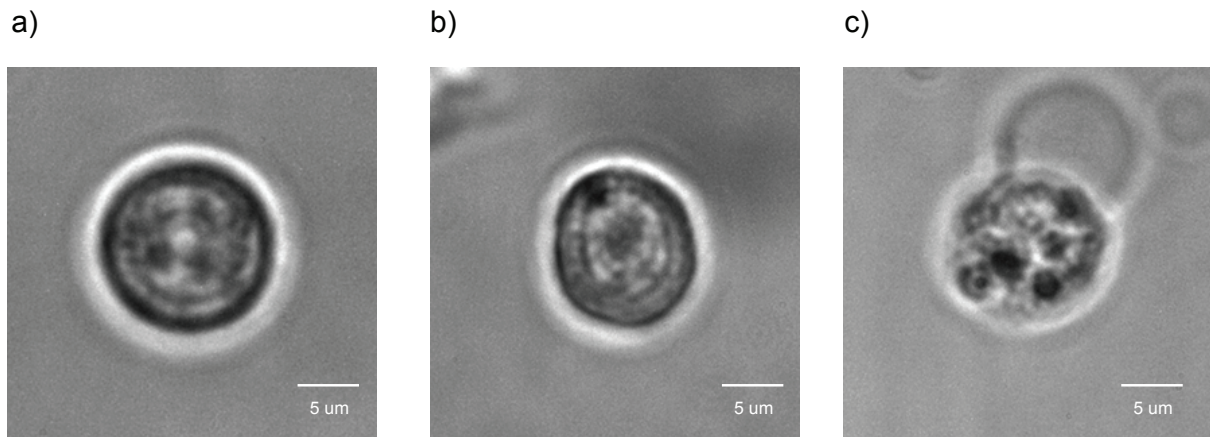
Figure 3.9: Viability of cells in monolithic chip. a) Cells remain viable after 30 minutes in the microfluidic channel without having been stretched. b) Cells remain viable after 30 minutes in the microfluidic channel, having been stretched with 1.2 W laser power. Note the more elliptic shape of the stretched cell. c) A control cell that was incubated in Ethanol shows signs of apoptosis. Since the camera is monochrome, blue uptake cannot be seen from these images.

3.3.2.3 Compliance distinguishes cell populations

To determine a boundary compliance that could be used to classify unknown cells into the *treated* and *untreated* group, each group had to be measured separately to extract the data relevant for grouping. In this case, it was decided that one parameter used to classify a cell would be the median compliance value of the final second of the stretch. This parameter has the following advantage: while this value is typically near the maximum compliance, it avoids the possibility of noise disrupting the data, which would happen if one chose a value at a specific time point. On the other hand, this parameter choice can lead to a loss of information about the entire stretch, for example modelled values such the viscoelastic response which carries information about the time-dependency of the stretch. Another possible parameter is the cellular radius, defined the major axis of the cell before stretching. The length might be larger for a treated cell, since the actin cytoskeleton might no longer able to keep the cellular structure tightly together.

As was mentioned in Section 2.4, to determine the compliance from the deformation information obtained from the experimental process, the global geometric factor is needed. This is a complex calculation which includes the distance of the fibers from the cell, the refractive index of the cell, and the cellular radius. The global geometric factor is known from the chip geometry, while the refractive index is acquired as an average from DHM measurements. The cellular radius differs from cell to cell. It was therefore decided to use a global geometric factor computed by averaging the geometric factors of the largest and smallest cells in total. The average GGF was 0.92, with a difference of 6% between the largest and smallest cell measured.

Measuring approximately 100 cells per population leads to a distribution of values that can be seen in Figure 3.10. As can be seen in Figure 3.11, there is a clear difference in compliance between cytochalasin-D treated cells and untreated cells, confirming the expectation. In contrast, the distribution of major axis lengths appears to be unrelated to the treatment, excluding this as a possible sorting parameter.
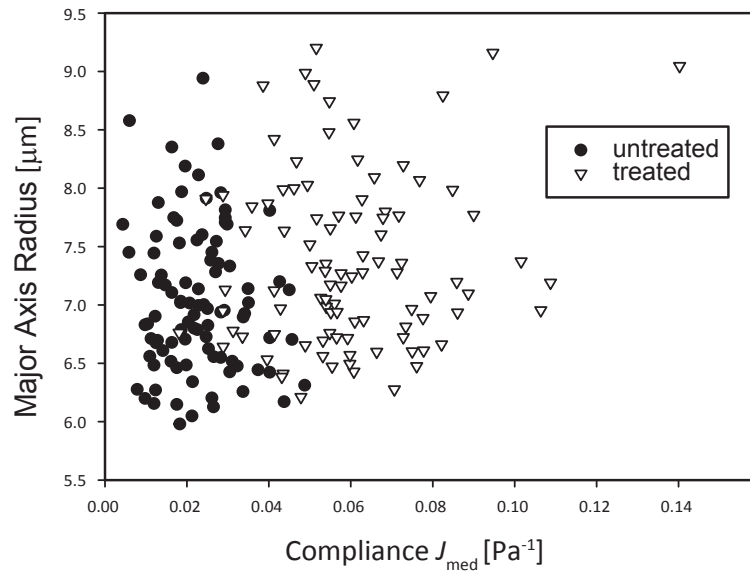
Figure 3.10: Compliance distinguishes cell populations Cytochalasin-D treated cells (white, $n = 97$) tend to be more compliant than untreated ones (black, $n = 95$), while they do not differ much in size.

In a next step, a boundary value for the compliance is calculated. This boundary will be used to determine whether an unknown cell belongs to Class I (untreated) or Class II (treated). To do this, Bayes' theorem for monovariate classification as detailed in Section 2.5 is used. Although the data is not normally distributed, Bayes' theorem works here. The *prior* $P(i)$, which is a measure of the probability that a cell belongs to a specific class i, is 0.5 for each class. This is because the concentration of cells in the mixed population is presumed to be a 50:50 concentration of untreated to treated cells. The *likelihood* $p(x \mid i)$ is known from the distribution of cells of the previous experiments with homogeneous populations (Figure 3.12 a).One can now calculate the posterior probability of a cell with a known compliance x

a)

b)
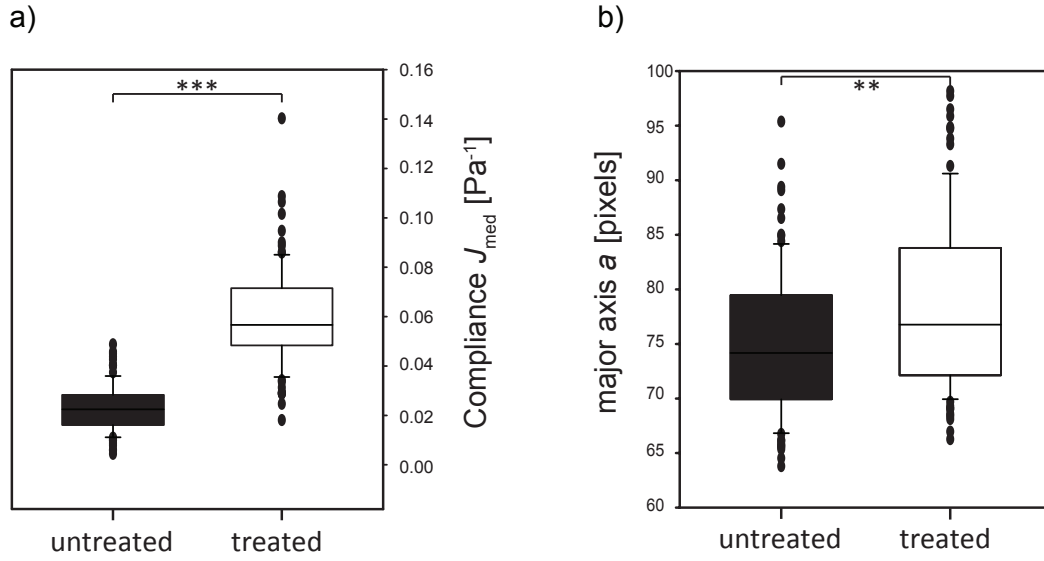
(Figure 3.12 b)

$$P(I \mid x) = p(x \mid I) \, \frac{P(I)}{p(x \mid I)P(I) + p(x \mid II)P(II)}$$

and find the zero crossing of the minimum error discriminant function

$$G(x) = P(I \mid x) - P(II \mid x)$$

.

The zero-crossing spot is the classification boundary value, in this case $J_B = 0.0402 \; Pa^{-1}$. Cells that have a compliance lower than J_B will be classified as untreated, while those with a compliance over J_B will be classified as treated.

3.3.3 Mixed populations can be sorted based on their mechanical phenotype

Using the now determined boundary compliance, a classification error can be estimated simply by counting all misclassified cells from the previous experiments and assuming that the *cost* of each misclassification is the same and thus not relevant. Thus

$$\frac{I_{misclassified} + II_{misclassified}}{\#cells} = \frac{8 + 10}{192} = 9.37\%$$

the misclassification is 9.37 % and one can expect around 10 % of unknown cells to be misclassified if the previously determined distributions hold true.

Using the classification boundary $J_B = 0.0402 \; Pa^{-1}$, a mixed population of cells was classified. For

a)

b)
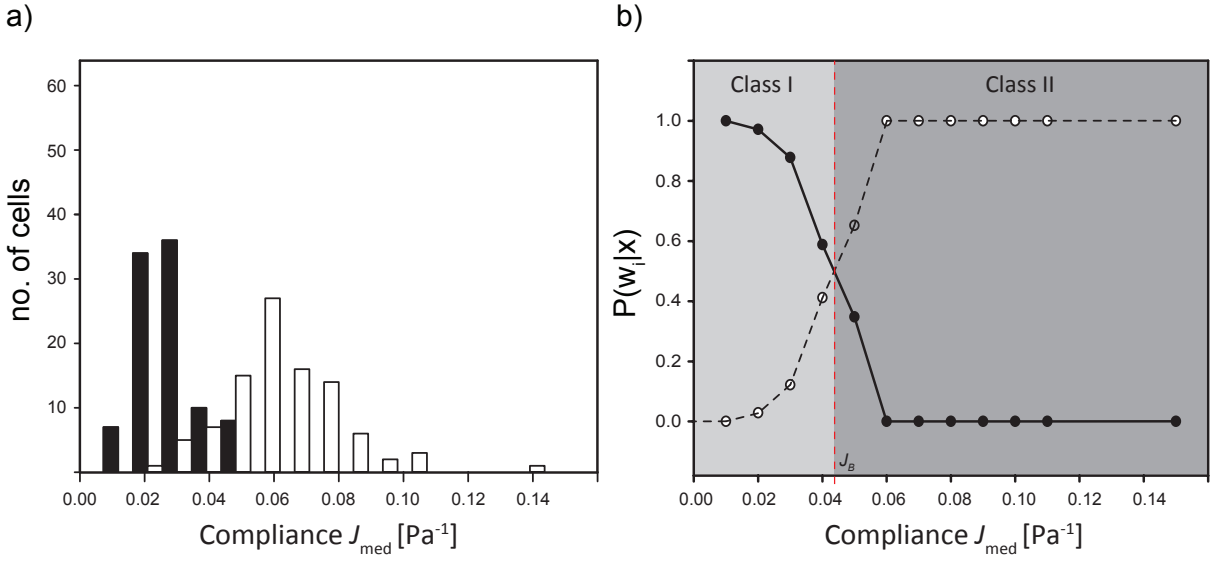
Figure 3.12: Determination of boundary compliance using Bayes' theorem a) Histogram of cellular compliance (black untreated, white treated) over number of cells. b) Posterior probability functions of cellular populations. A zero-crossing of the minimum error discriminant function is found at $J_B = 0.0402\ Pa^{-1}$ which is used as the classification boundary between Class I (untreated) and Class II (treated).

this, both treated and untreated cells were poured together in equal concentrations. Untreated cells were marked with a fluorescent dye as a control. This way, after classification, one can easily check if the classification was correct.

Then, the cells were measured and sorted. This included trapping a cell, turning off the flow, measuring the major axis elongation with an external Labview program, comparing the calculated compliance to the boundary value, imaging the cells' nuclear fluorescence for control purposes, turning off one laser for optical sorting of the cells and starting the flow again.

The results in Figure 3.13 show that the compliance distribution of the mixed population differs from what was expected during the preliminary experiments. For example, the range of compliances of untreated cells was larger than in the single untreated population. This naturally has an effect on misclassification, which was

$$\frac{8+6}{54} = 25.9\%.$$

This is likely due to the fact that that cytochalasin-D has a reversible effect. That is, over the course of the experiment, the cytochalasin-D present in the treated cells is released into the medium, affecting untreated cells which subsequently become more compliant.

Could the misclassification be improved if one had not normalized with an average global geometric factor? In a worst case scenario, assuming that all untreated cells had the GGF of the smallest cell, and the treated cells had the GGF of the largest cell, this would shift distributions slightly (by 6%) apart
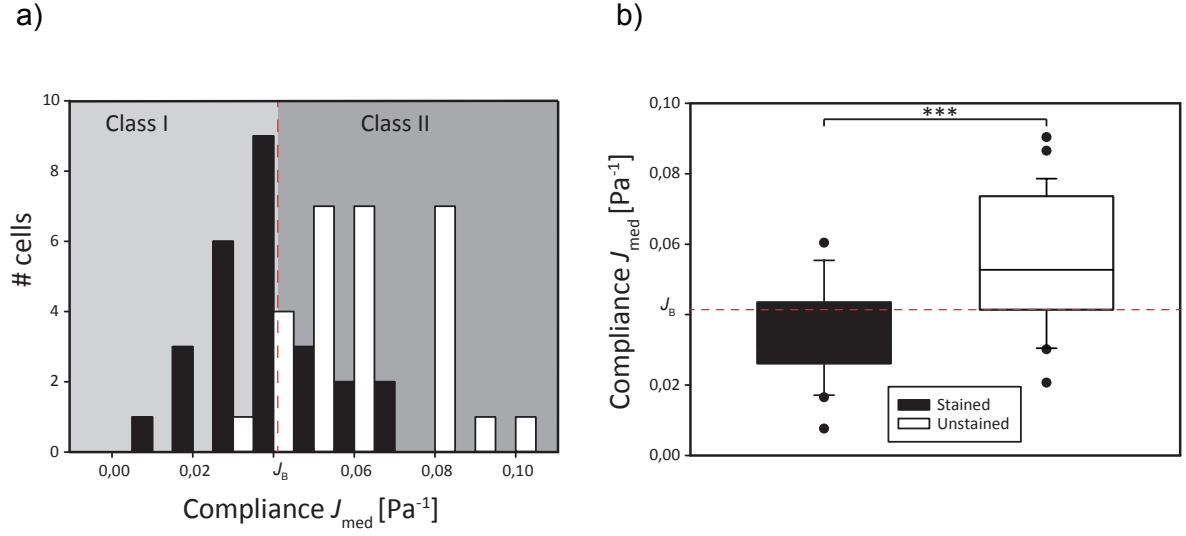
Figure 3.13: Classification of mixed population a) Histogram of cellular compliance (black untreated/fluorescent, white treated) over number of cells ($n = 54$). The compliance distribution of a mixed population overlaps more than in the previous, single population distribution. b) Stained (untreated) and unstained cells significantly ($p < 0.001$) differ in compliance.

from each other. The new boundary compliance would be $J_B = 0.044 \ Pa^{-1}$. This would decrease the expected classification error to

$$\frac{2 + 14}{192} = 8.33\%.$$

This new boundary compliance in turn leads to a hypothetical misclassification of the experimental data of

$$\frac{6 + 7}{54} = 24.1\%$$

of the mixed population.

3.4 Concluding remarks

This chapter discussed a device capable of classifying and actively sorting a mixed cell population by measurement of the cells' compliance. The device measures cells and sort them to specific outlets with the help of fluidic streamlines. It has two main advantages:

- Cells can be actively classified and sorted and sorting boundaries can be changed without redesign of chip or experimental flow

- Cells do not need any additional markers for classification

Additionally, fluorescent imaging can be used as a marker or control if it is required.

There are some caveats regarding this device. First, the fabrication of the device is complex and must be done by an external manufacturer. Next, very small deformations cannot be as easily detected as in smooth glass side channel surfaces with rectangular channels due to the wet etching process and the consequent curvature of channels. Additionally, optical fibers are farther away from the cell and the beam width is larger, which means that more power is needed for trapping and stretching.
One major drawback is the fact that the microfluidic connections are not capable of extracting sorted cells. The cell numbers measured during an experiment are very small, and the cells classified for extraction will be lost in the tubing and connections between measurement and receptacle.
Finally, the classification error is rather high. Using additional classification parameters and individual compliance calculations might improve the classification.

The next chapter will present a device that tries to overcome the above-mentioned limitations. It is simple to set up with low cost, it is able to extract individual cells, and it offers more classification parameters.

4 An Optical Stretcher setup for the accurate measurement of rare cells

All Optical Stretcher devices have to solve the common problem of getting the cells the user wants to have measured into the optical trap. While the very first Optical Stretcher let cells sediment in a droplet and at times accidentally trapped a cell that floated through the vicinity, the single glass capillary setup and the monolithic chip offer a more controlled way of moving cells in and out of the optical trap. Both of these devices need a large number of cells however, as many cells disappear within the tubing. This raises the question: What if the sample only consists of a few cells or what if one wants to measure a few select cells within a larger sample? These few cells would likely get lost within the connecting tubing, making a measurement impossible. Furthermore, what if one wishes to extract or separate a few select cells after the experiment? If they are moved to a common reservoir, the measured cells get lost within the multitude of their siblings.

A device is thus needed that can reliably move individually selected cells from a reservoir to an optical trap. Furthermore, it should then be possible to remove these cells into the individual reservoir of the user's choice in case they need to be reused, such as for recultivation, single-cell AFM measurements [117] or spectroscopy [118].

This chapter will present a very simple device which makes it practical and quick to build and use in any biophysical laboratory. It is based on the first design of the Optical Stretcher with some added features.

4.1 Design

The principal requirements for the design of an Optical Stretcher capable of handling and measuring a small number of cells are similar to those of other setups. As mentioned in Section 3.1, these requirements consist of the means to optically manipulate cells, the means to observe said cells and the connection of the setup to the outside world. A setup for rare cells must particularly take care not to lose any potentially measurable cells in the process.

In order to trap and stretch cells, a waveguide must be aligned in such a way that the fiber ends face each other exactly. In previous versions of the Optical Stretcher, this was achieved by inserting the fibers into specific grooves and holes to fix them tightly so that they could not move with respect to each other. While this ensured that they could not move out of position, the necessary superstructure of glass or PDMS limited user access to the trapping region. Cells were moved to the trap using microfluidic channels such as in [7, 41] and [108]. These channels, together with their microfluidic connectors, tubing and reservoirs, introduced a significant amount of dead volume into the system. This led to individual cells getting stuck on their way to the trap. While this is not a problem when measuring sample sizes of tens of thousands of cells, for rare cells that only number in the tens or hundreds, every cell counts.

This dead volume problem makes a redesign necessary. Since the option of microfluidic channels and pressure-driven delivery of cells from a reservoir is no longer available, the simplest solution is to manually transport each cell to be measured from a reservoir to the trapping region. The user can thus select an individual cell to be measured from a set of rare cells, move it to the trapping region, measure and analyze it and then return it to the reservoir or any other location of choice. Sorting of the measured cells can also be achieved by bringing measured cells to different collection spots.

What is thus needed is a flexible method of cell transport that is capable of dealing with a variety of reservoirs of rare cells. These reservoirs can range from a droplet of cell solution on a coverglass to a microtiter plate with single cells treated with different compounds in each well. In this case, an uncomplicated approach is the use of a micropipette connected to an pressure-driven air syringe. With the help of an electronic micropipette holder, cells can be sucked into the pipette at the reservoir, transported by moving the pipette along the stage of the microscope and inserted directly into the laser beam of the optical trap.

One might argue that the aspiration of cells into a pipette that is not much wider than the cell itself has an effect on cell function or viability due to the shear stresses acting on the cell. While there are reports of an effect on cellular function of long-term low shear stress [119, 120, 121] or of high stresses in short time scales [122], research in microfluidic channels of a similar size ($20\ \mu m$) suggests that viability of cells is not affected even for significantly higher flow rates ($0.2\ \mu l/s$) that lead to a shear stress on the order of 500 Pa [123]. In fact, as the velocity used for sucking cells into the micropipette is slower by multiple orders of magnitude, it can be safely assumed that the wall shear stress does not exceed 1 Pa. The stress acting on the cells during aspiration and release is thus likely lower than in the *in vivo* bloodstream [124, 125].

The optical trap region of the setup must offer access for the insertion of the cells by the micropipette.
As the user needs to be able to see the pipette and adjust the height of the micropipette with respect to
the laser in order to insert cells directly into the trap, at least one side of the trap must be open to access
from the outside. The trapping region is thus no longer entirely closed and any superstructure that clamps
optical fibers as mentioned before has to take this into account.

A straightforward approach is the use of an alignment setup similar to the original Optical Stretcher [3,
5, 6]. Here, optical fibers were aligned by placing them against a straight edge with the help of three-
dimensional micromanipulators. This straight edge typically consisted of a larger optical fiber that was
attached to the glass of the slide. Once the smaller optical fibers were tightly pressed against the back-
stop, the backstop's stiffness and the small distances between the fibers ensured that the optical fibers
were facing each other. The optical fibers could then be attached with glue and would no longer move.
Cells were then pipetted into a droplet of medium over the trapping region. Ideally, a cell floated into the
optical trap and was deformed.

This simple setup comes with some disadvantages, however. For example, the use of a backstop that is
very close to the optical fibers can adversely affect the image quality in phase contrast. Also, it is difficult
to judge the distance between fibers correctly by eye while aligning them. This distance highly affects
not only the laser power acting on cells during deformation but also equilibrium positions for objects
in a laser trap [53]. Finally, the backstop itself limits the accessibility to the trap, although only in one
direction.

To overcome these limitations, in the new setup a SU-8 photoresist-patterned coverglass (see Methods,
Section 7.1.1) was used. This coverslip was patterned using the chromium photomask that was already
in use for the closed Optical Stretcher [7]. The patterning on this chip offers two benefits: There are
aligned grooves in which to lay the optical fibers while building the system and there is an unpatterned
area of known size between these grooves. This area was previously used to hold the square glass capil-
lary in place in the closed Optical Stretcher but can now be used as a visual reference for fiber distance.
Since the grooves are slightly smaller than the diameter of the optical fibers, it is ensured that the fibers
touch the grooves in two spots and snap into place during alignment. If they are then weighted down and
attached using glue, they no longer can move and alignment is perfect (Figure 4.1 a).

The patterned chip sits on a custom microscope stage made by a 3-D printer (Figure 4.1 b). This offers
the advantage that additional slides used as cell input or output reservoirs can be added onto the same
system. This way, there is no cumbersome change of slides or stages. Since the 3-D printer is in-house,
any change in requirements can easily be realized in the matter of a few hours.
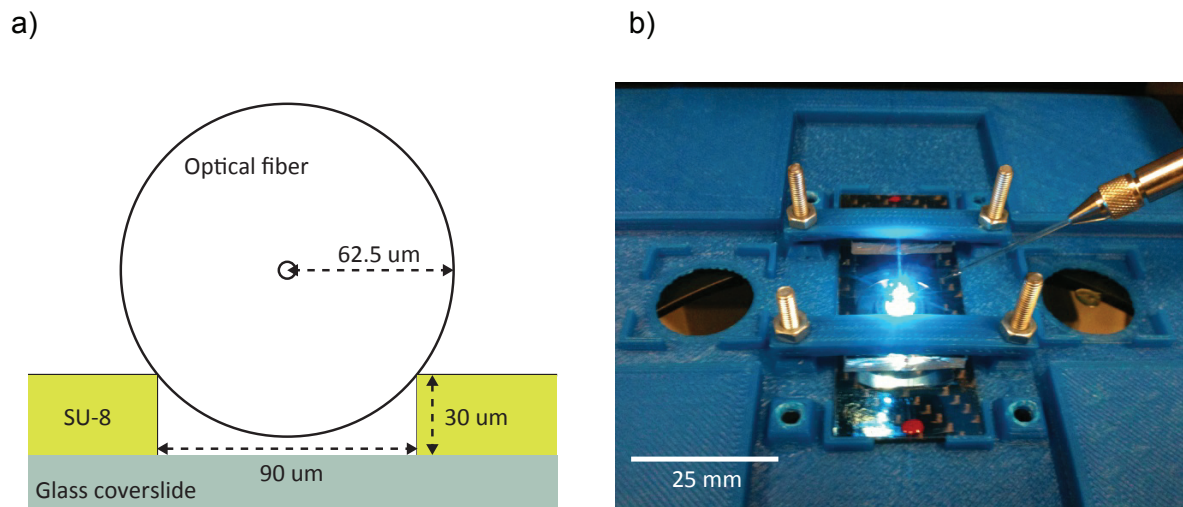
Figure 4.1: The open setup. a) Optical fibers, seen from the front rest on two edges of the structured photoresist. b) A patterned cover glass chip with barely visible optical fibers attached sitting on its 3-D printed stage holder.

4.2 Results

4.2.1 Workflow

A typical experiment starts with the building of the Optical Stretcher. For this, after stripping, cleaning and cleaving, optical fibers are placed close to their intended grooves on the photoresist patterned cover-glass (Menzel, Germany). Then, a weight, typically a screw nut, is placed on the fiber. This prevents the fiber from moving too much while the user can use a pair of tweezers to push the fibers into the groove along the whole length of the chip. When the fiber snaps into the groove, it becomes more difficult to move. Care must be taken nonetheless that both fibers are as well-aligned as possible across the length of the open area, which is typically 200 or 300 μm wide. Once both fibers are in place, they are fixed to the glass slide with drops of nail polish. The nail polish evaporates, gluing the fibers to the chip tightly. If needed, the chip can be reused after the experiment by pulling off the fibers, since the dried nail polish is removed along with the fibers, not leaving much residue. The chip must then be washed with deionized water and ethanol.

The chip with fibers attached is then placed into the 3-D printed stage holder. As the roughness of the 3-D print leads to the chip not lying smoothly on the surface and being unstable, a coverglass-sized thin carbon fiber plate (Conrad Electronics, Germany) with an appropriate opening for microscope objectives cut out is added. The chip now lies on it smoothly. If needed, additional square coverglasses can be added to the stage. These will act as input and output reservoirs for cells.

Phase contrast microscopy is highly dependent on the curvature of the air-medium interface through which the illumation of the microscope passes. This means that a very rounded droplet can easily de-

center the field diaphragm and shift the imaged halo of a cell in the trap, especially at high microscope magnifications. The use of a large O-ring that is filled with medium guarantees that the interface between air and medium will be reasonably flat in the center of the ring in order to improve image acquisition quality. An additional benefit is that due to the much larger volume of medium compared to a droplet, evaporation of the medium during the course of an experiment plays an insignificant role. Therefore, once the stage is placed on the microscope, a rubber O-ring (Conrad Electronics, Germany) with 25 μm diameter is placed on the chip, centered on the trapping area. As it is going to be filled with medium, it can be sealed with vacuum grease to ensure the medium does not disappear during the course of an experiment.

At this point, it is necessary to take a low magnification image of the aligned fibers. This image is used to calculate the distance of the optical fibers, which is needed to determine the global geometric factor.

A micropipette with an angled tip and an inner diameter of 19 μm (Eppendorf, Germany) is then inserted into the medium on the chip. The angle of 35°makes the pipette tip lie parallel to the imaging plane of the microscope. This renders the pipette tip visible in its entirety and facilitates control with an electronic micromanipulator system integrated in the microscope (InjectMan, Eppendorf, Germany). The user should take care not to fill the micropipette entirely as this can affect manual flow control. In fact, ideally only the 2 mm long tip of the pipette should be filled with medium so that the air-filled syringe (SAS Air Syringe, RI Lifesciences, UK) connected to the micropipette cannot apply too high pressures on the medium, improving handling of the cell, i.e. moving it in and out of the pipette.

Once the pipette tip is filled and pressures are equilibrated, a cell from the cell reservoir can be sucked into the pipette (Figure 4.2 a). This is done at 10x magnification. Ideally, one sucks in the cell until it is around 1 mm away from the pipette opening. This ensures that it will not be sucked out or in inadvertently when moving through air if the reservoir lies on another glass coverslip. Since the pipette will move first through the medium of the reservoir, then air, and then again the medium of the optical trap, slight pressure differences could induce a flow in the pipette and remove the cell if there is not a secure buffer distance. Interestingly, cells hardly ever seem to disappear on the way from a reservoir droplet to the O-ring of the trapping region. On the way back, cells tend to disappear more often. They are either pushed out of the pipette by the pressure change when reentering the droplet before being in focus or they attach to the air-medium meniscus inside the pipette, which destroys them and can lead to pipette clogging (Figure 4.2 c). However, in the majority of cases, cells survive their arduous journey to and from reservoirs if they are placed far enough inside the pipette.

Once a cell is safely positioned inside the micropipette, it is moved to the trapping region. By moving the pipette up and down on the optical axis with the lasers turned on and using the scattering of light on the pipette tip as a guide, one can determine the height of the pipette with respect to the optical axis. Then, it suffices to gently push the cell out of the pipette and trap it immediately without a great deal of effort (Figure 4.2 b).

Once a cell is captured in the trap, magnification is changed to 63x in brightfield imaging. Together
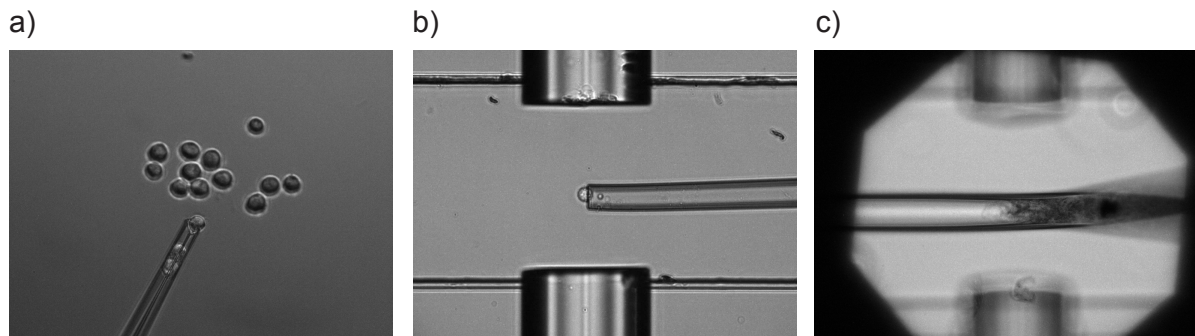
Figure 4.2: Single cell transport with micropipette. a) A single cell from a reservoir of cells is sucked into a micropipette. b) The cell is placed into the optical trap after placing the pipette at the necessary height. c) Care must be taken to remove debris from the pipette as it can lead to clogging.

with the integrated *optovar* magnification changer of 1.6x, this leads to an effective total magnification of 100x. Then, the SID4Bio module acquires an interferogram of the cell. High magnification is vital for the following reason: while the resolution of the SID4Bio interferogram is at most 1600x1200 pixels, the imaging area is much larger than the object which typically only has a diameter of around 160 pixels at 63x magnification. The cell in the resulting phase image thus only has a diameter of 40 pixels due to the quadrilateral shearing interferometer.

After interferogram aquisition, the magnification is changed to 40x (times 1.6x optovar) and phase contrast imaging. Then, the standard laser step is applied and deformation of the cell recorded. 40x magnification is necessary due to the thickness of the coverglass. The 40x objective has a larger working distance than the 63x objective. Even though both have a correction collar to take coverglass thickness into account, the 40x objective seems to offer a less blurry halo which is helpful for image acquisition.

After the recorded deformation, the pipette is brought into close proximity to the cell still captured in the optical trap. The cell can be easily sucked into the pipette which is then moved away from the trap. Then, the magnification is changed back to 63x (times 1.6x) brightfield imaging. Now, the reference image can be acquired with the SID4Bio module, and the phase image calculated. With the external program by Schürmann et al. [59], this phase image is used to calculate the cells' refractive index.

Finally, after finishing all data acquisition, the user can decide what to do with the cell in the micropipette. The cell can be placed elsewhere on the chip, moved back to the reservoir, or moved onto an output reservoir depending on the user's wishes. As one would expect, the throughput of such a device is on the

low end and amounts to 8-10 cells per hour of measurement.

4.2.2 Photoresist patterned coverglass fabrication

A detailed description of the fabrication of photoresist patterned coverglass can be found in Methods
(Section 7.1.1). For the purposes of the following discussion, a brief description is given here. A polymer
photoresist (SU-8) is spread onto a glass slide and illuminated with UV light through a structured mask
corresponding to the requested pattern on the glass slide. Illuminated areas of the photoresist crosslink.
Then, crosslinking is strengthened during a *soft bake*. A developer bath removes uncrosslinked photore-
sist. Finally, the photoresist is cured during a *hard bake* at high temperatures. After this process, the
glass slide has a superposed structured and stable layer of SU-8.

4.2.2.1 Hard bake introduces tensile stress into chip

If one follows the manufacturer's recommendations for hard bake, one soon realizes that the *hard bake*
step, in which the SU-8 structures are cured at 150 °C leads to problems. Although the SU-8 structures
are chemically resistant and very stable, the thermal stress occuring in the SU-8 during the high temper-
ature exposure leads to buckling of the SU-8. This happens because SU-8 has a different coefficient of
thermal expansion (coefficient of thermal expansion $\alpha_{SU-8} = 52 \; 10^{-6} \; K^{-1}$, $\alpha_{glass} = 3.3 \; 10^{-6} \; K^{-1}$)
than the coverglass underneath. During baking and subsequent cooling, both materials, attached to each
other, will expand and retract differently. Since the SU-8 expands and retracts more than the glass, its
tensile stress increases [126, 127] and thin chips tend to bend upwards.

This bending is visible with the naked eye for thin coverglasses such as sizes #1 and #1.5 (from 130 to
190 μm thickness) that are used as standard coverglasses. These thicknesses are recommended if fluores-
cent imaging is used because thin coverglasses introduce little *spherical abberation* which would result
in a loss of contrast and sharpness of the image. However, in the case of the open setup, the bending of
the glass slides of these thicknesses is so drastic that it cannot be used in an open setup, since the slide no
longer stably sits in the slide holder. Nevertheless, this type of slide can still be used in a closed setup [7].
There, the bending is overridden by the force exerted on the chip by the clamps.

For the open setup, initially it was decided to leave the post-developmental hard bake step out while
manufacturing chips. It appeared that the SU-8 structural integrity after the soft bake alone would be
sufficient and the addition of extra tensile stress could therefore be avoided.

4.2.2.2 Fiber alignment is dependent on quality of photoresist patterning

There are many factors that can result in unsharp edges in the patterned photoresist. These include over-
or underexposure during UV illumination, a too long or too short development time or a too rapid change
in temperature leading to internal stress in the system. All these factors have the effect that the edges of
the fiber grooves are no longer well-defined (Figure 4.3 a). This can affect the fiber alignment step that is
done by hand. Ideally, fibers should snap into place and then be aligned in the fiber grooves along their
axis, but this can be hard to achieve if the photoresist is not well-structured. Since the fibers are typically

aligned under a stereo microscope so that the field of view is large enough to see what is done, small
misalignments at the fiber tips are hardly visible until the fibers are fixed and glued and imaged with
an inverted microscope. It is also possible that fiber misalignment occurs after some immersion time
(Figure 4.3 b). This is evidenced by cells starting to periodically move along two different axes while
being optically trapped since the two beams emanating from the fiber ends no longer face each other.

a)   b)
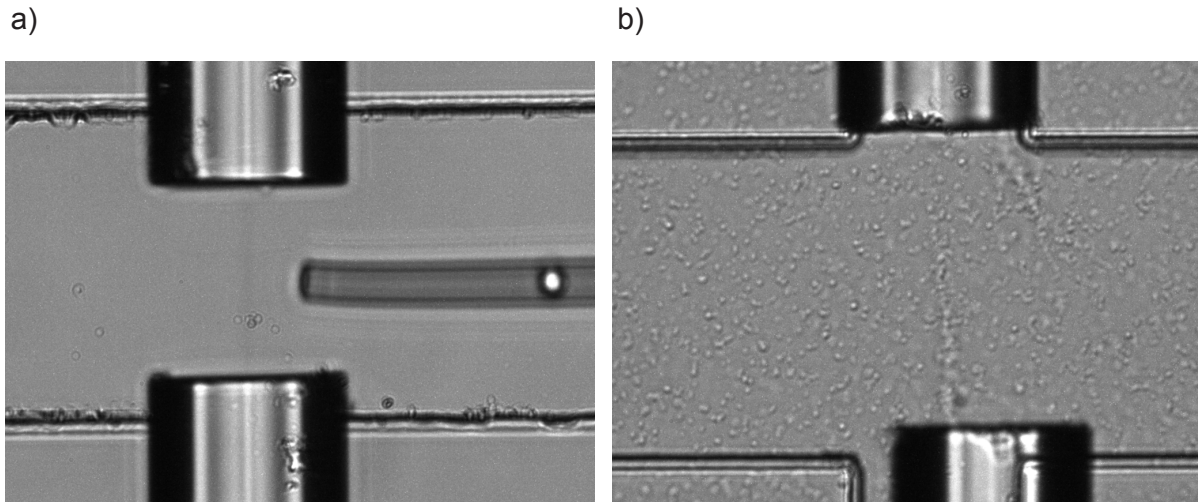
Figure 4.3: Fiber misalignment in open setup. a) Underexposure leads to poorly patterned structures
as evidenced by the horizontal edges of the channel. b) Fiber misalignment also occurs
during the experiment itself. Image courtesy of Saeed Ahmed.

4.2.2.3 Soft bake leads to SU-8 lift-off

Unfortunately soft baking does not appear to be sufficient to guarantee seamless use of the patterned chips
in an acqueous environment. Depending on the development time in a way not yet quite understood, it
appears that an acqueous medium such as water or PBS starts dissolving the links between coverslip
glass and SU-8 structures. This can happen in a space of time between ten minutes to hours. Once these
links are removed, the SU-8 starts floating and lifting off. If this happens during an experiment, the fibers
will no longer be aligned (Figure 4.4). In general, this limits measurement time to around 30-40 minutes
per experiment.

a)   b)
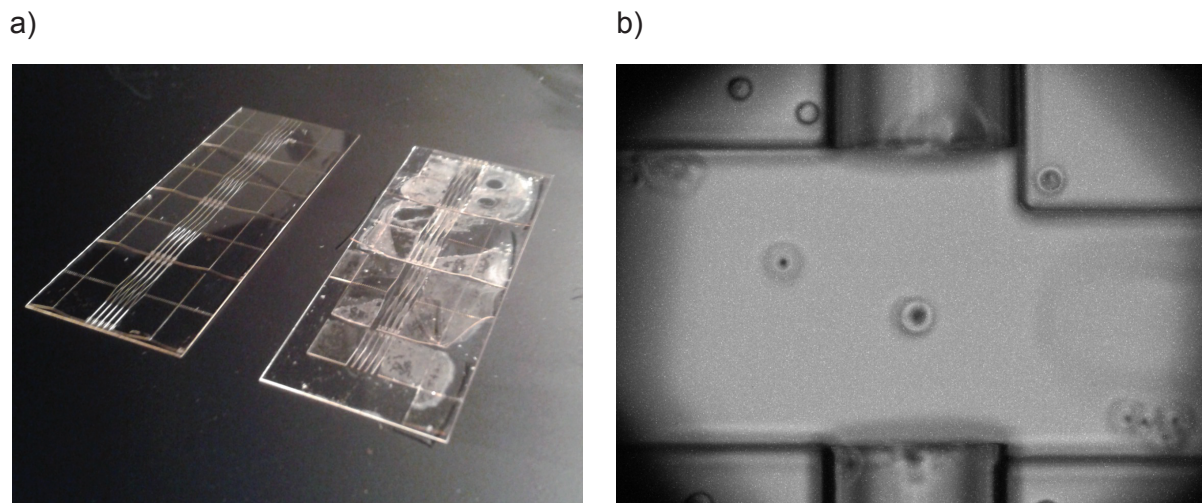
Figure 4.4: SU-8 pattern lift-off. a) Immersion in water leads to SU-8 lift-off. Left: unused patterned
chip. Right: Patterned chip after submersion in deionized water for two hours. b) Lift-off
removes structures in the chip and leads to fiber misalignment as evidenced by the structure
in the upper right corner of image that has moved downward.

4.2.2.4 The effect of laser power on coverglass bending

With cells easily pipettable into the optical trap and fiber alignment and lift-off problems under reasonable control, experiments can begin. However, cells that are in a stable position while being trapped appear to significantly defocus once the high laser power is turned on (Figure 4.5). This apparent defocusing seems to indicate that cells move upwards by up to 40 μm per minute of high-power (0.9 W per fiber) stretching, although their position stabilizes at some point.

Contrary to appearances however, it is not the cell that defocuses, it is the entire chip. Changing the laser power makes the center of the chip at the trapping region rise upwards after a short delay. Presumably, this is due to the heating by the laser beam. As the light exits the optical fiber, it expands in a conical shape that not only acts on the cell but also on the chip itself. This bending is not visible by the naked eye but severely effects image quality when measuring cell deformation. It is possible that the high power involved heats the SU-8 that already has a slight intrinsic stress from the patterning mentioned above. This heating will then make the SU-8 bend upwards towards its equilibrium position due to thermal expansion. The supposition that intrinsic stress is responsible for the bending is contradicted by the fact that bending also occurs for non-patterned (i.e. without a photoresist coating) glass that has a fiber glued to it. This phenomenon is interesting because the absorption spectrum of glass [128] suggests that at 1064 nm wavelength hardly any light is absorbed, so heating should be less than in water. Moreover, despite there being evidence of forces induced by the momentum change of the light leaving the optical fiber [129], in view of the forces required to bend the glass this is unlikely. While there have been reports [130] of controlled bending of thin borosilicate glass sheets at powers as low as 15 W, this bending occurs at 10.000 nm wavelength which has a higher absorption. [131] reports controlled bending of silicon structures at

a)   b)
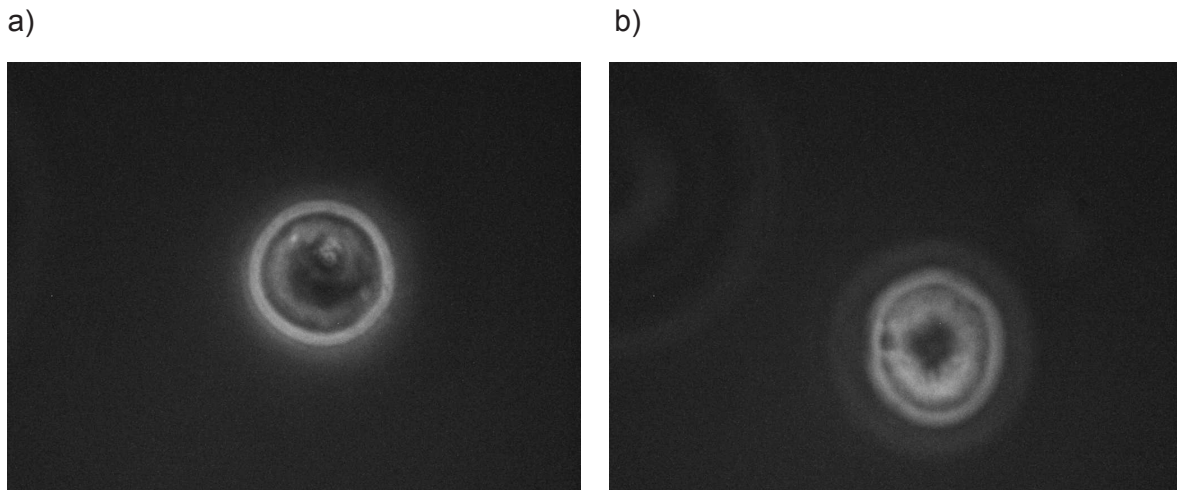
Figure 4.5: Apparent defocusing of cells at high laser powers. a) A cell is in a stable trap position at
0.2 W per fiber. The halo is well-defined. b) After ten seconds at 0.9 W per fiber, the cell is
severely defocused. Nevertheless, one can still see that it is deformed.

1064 nm wavelength and at slightly higher powers than the ones used in the Optical Stretcher. However,
pure silicon has an absorption coefficient an order of magnitude higher than glass (SiO_2). Possibly, the
nail polish used for gluing the fibers to the slide absorbs so much scattered light that heat is transferred
to the coverslip which makes it bend microscopically.

This bending appears to occur in every Optical Stretcher setup operating at 1060 nm wavelength, includ-
ing the closed microcapillary setup. However, tightly clamping the glass slide from the top does suppress
bending. As this is not an option for the open setup that needs to remain accessible, it was decided that
the final glass chip thickness must be a compromise between imaging quality and bending stiffness.
The glass thickness that was found to offer resistance both to the bending induced by high laser power
within time scales of 10-20 seconds acceptable for Optical Stretcher measurements as well as to the
intrinsic bend from the SU-8 hard bake is the non-standard #5 thickness (500 μm). Since glass of this
thickness does not bend when hard baking the photoresist pattern, the lift-off problem is solved at the
same time.

4.2.3 Measurements

With the above-mentioned fabrication issues solved, the setup can be used for data acquisition.

4.2.3.1 Choice of measured parameters

To characterize a cell, it makes sense to capture as many parameters as possible [132]. In this case, four parameters were chosen in order to acquire as much information as possible during an experiment.

1. **Refractive index:** The refractive index of the cell can not only be used to calculate the global geometric factor needed for the conversion of deformation to compliance but it can also serve as a possible stand-alone indicator of cell cycle [133, 134], apoptosis [135, 136] or disease [137, 81, 79, 138].

2. **Cellular radius:** The cellular radius which is defined as the pre-stretch major axis length of the cell is also necessary for the calculation of the global geometric factor. Cell volume, dependent on the radius, is known to be an indicator for apoptosis [139]. The cellular radius can also be an indicator for the cell cycle stage [140, 141] or could possibly be an indicator for disease [142].

3. **Last second stretch median:** As already used in [108], this value is the median compliance value of the last second of a stretch. It is a close indicator of the maximum compliance reached during a stretch. However, this value does not give any information about the time dependence of the stretch.

4. **Cell fluidity:** The power law model introduced in the introductory chapter can be fitted to an Optical Stretcher compliance curve. This power law model describes viscolelastic parameters [143] as

$$J(t) = J_0(\frac{t}{\tau_0})^\beta$$

   The exponent β indicates the cell fluidity or viscoelasticity of the cell, with $\beta = 0$ a purely elastic material and $\beta = 1$ a purely viscous fluid. Cell fluidity has previously been used to distinguish different cell types [15, 56, 14]. As it is a parameter that is derived from a fit over the whole stretch, it adds the time-dependency that the last second stretch median lacks.

4.2.3.2 Intra-setup variance

For every experiment in the open setup, experimental settings will necessarily be different. This difference is mainly a consequence of the distance between the fiber and the cell, since the global geometric factor decreases quadratically with distance as shown in Section 2.4, which will affect conversion of deformation to compliance. It therefore makes sense to compare individual measurements on open setup slides and see if the global geometric factor conversion holds.

Here, three sets of untreated HL60 cells were measured both for deformation and refractive index using the SID4Bio camera. Each measurement was done within an hour. The respective fiber-to-fiber distances were $d_1 = 180 \ \mu m$, $d_2 = 193 \ \mu m$, and $d_2 = 185 \ \mu m$. Failed cell stretchs (i.e. cells that

rotated or were not possible to analyze) were removed, leading to around 10 useful cell stretches per set
($n_1 = 11$, $n_2 = 9$, $n_3 = 8$). Results can be found in Figure 4.6.
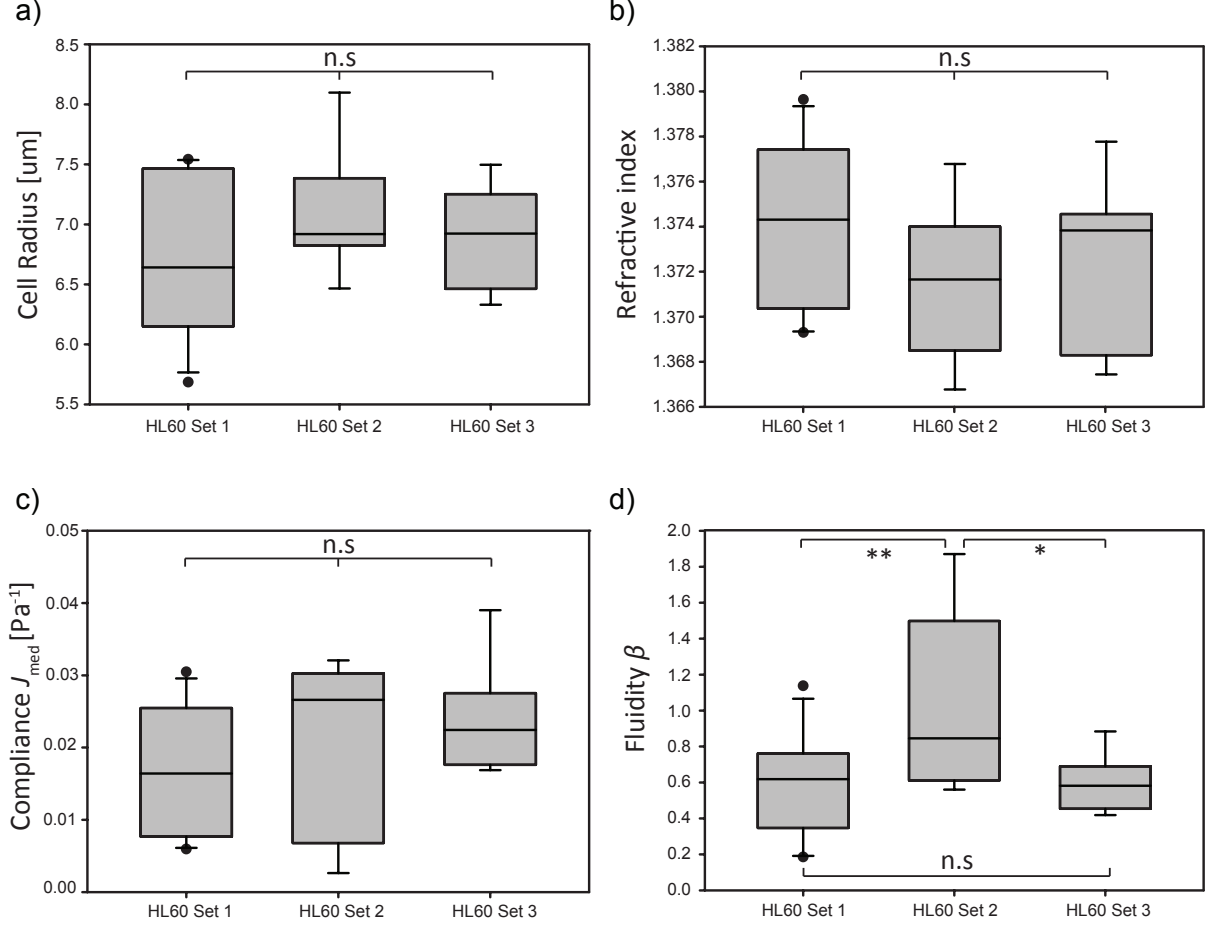
Figure 4.6: Comparison of HL60 cells measured in the open setup. Three individual measurements
of HL60 cells at 0.75 W power per fiber and fiber-to-fiber distances 1) 180 μm (n = 11), 2)
193 μm (n = 9), 3) 185 μm (n = 8). a) The measured cellular radius of each set of cells is
similar. b) Refractive indices of the measured cells. c) Median compliance of the last second
of the stretch. d) Cell fluidity β as obtained by fitting each stretch curve to the power law
model.

Detected cell radii (Figure 4.6 b) and normalized mean compliance (Figure 4.6 c) did not significantly
differ from measurement to measurement, adding further evidence to the hypothesis that different mea-
surements normalized by an individual global geometric factor lead to the same results.

The refractive indices of cells measured at different fiber distances did not significantly differ (Figure 4.6
c). The distribution of refractive indices from 1.368 to 1.378 corresponds to values of suspended cells
reported previously [84]. Possibly, the refractive index could be used as a parameter for characterization
of single cells; however the variance of the parameter is rather large so classification might only be useful
for cells that are known to differ considerably in refractive index in the diseased case, for example red

blood cells infected by *P. falsifarum* [81]. However, since in this study only the mean refractive index of a cell was measured, it is possible that by looking more closely at the refractive index distribution within the cell or nucleus, more information might be extractable [144, 145]

Finally, the fitted cell fluidity β (Figure 4.6 e) differs significantly for the furthest fibers (set 2). However, this could be due to more noisy stretches and the subsequent bad fitting, as set 1 and 3 do not significantly differ. While [17] and [56] report a temperature and power dependence of the cell fluidity, the high variance in this experiment is likely due to the low sample numbers.

Similar measurements can be done for Jurkat cells, a line of human T lymphocyte leukemic cells (Figure 4.7).
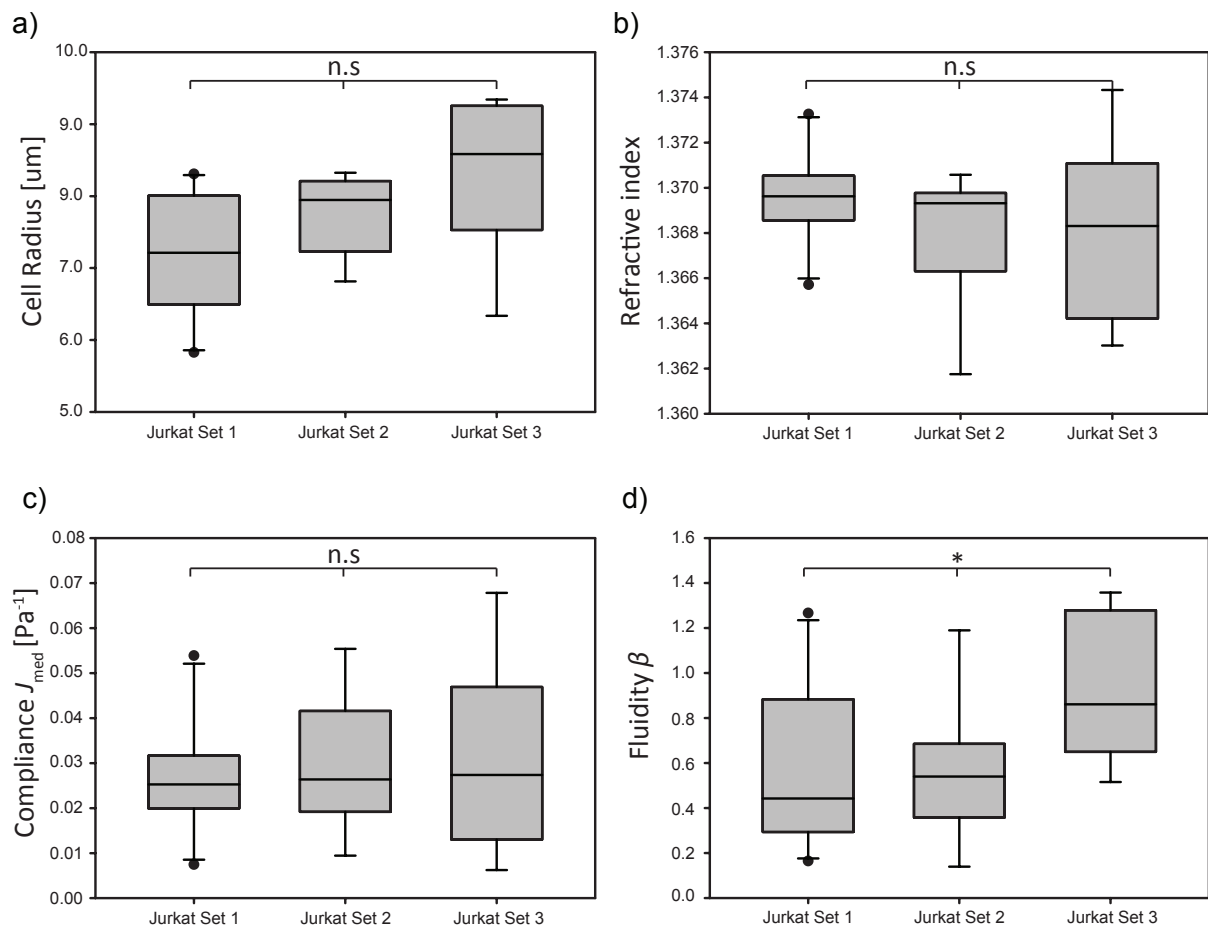
Figure 4.7: Comparison of Jurkat cells measured in the open setup. Three individual measurements of Jurkat cells at 0.75 W power per fiber and fiber-to-fiber distances 1) 243 μm (n = 10), 2) 280 μm (n = 9), 3) 248 μm (n = 9). a) The measured cellular radius of each set of cells is not similar. This is presumably experimenter's bias due to the selection of the most spherical cells for measurement. b) Refractive indices of the measured cells. c) Median compliance of the last second of the stretch. d) Cell fluidity β as obtained by fitting each stretch curve to the power law model.

As with HL60 measurements, refractive index (Figure 4.7 b) and compliance (Figure 4.7 c) of Jurkat cells did not significantly differ, while the cell fluidity (Figure 4.7 d), while differing significantly per set, did not appear to be related to the fiber distance. The cell radius (Figure 4.7 a) does not differ significantly.

A certain caveat is in order at this point. Around 10 measurements per experiment do not necessarily constitute a robust foundation for statistical analysis. The biological variance of the cells is large and there is an intrinsic experimental bias, as only cells that seem spherical from the outset are chosen for analysis and less spherical cells are ignored. Especially Jurkat cells tend to, on average, be more spheroid than spherical. Additionally, they are slightly more adherent to glass than HL60 cells. For this reason, Jurkat cells often have slight protrusions and are not as easily pulled into the micropipette from the reservoir slide. An interesting observation is that while most Jurkat cells start rounding up into a sphere while inside the micropipette and can be normally stretched in the optical trap, some appear to form roundish bleb-like protrusions along the optical axis under normal trapping stress. These blebs are then pulled out more compared to the rest of the cell during the stretch phase (Figure 4.8). This might be due to a rupture of the actin cortex at the spot of detachment [146].
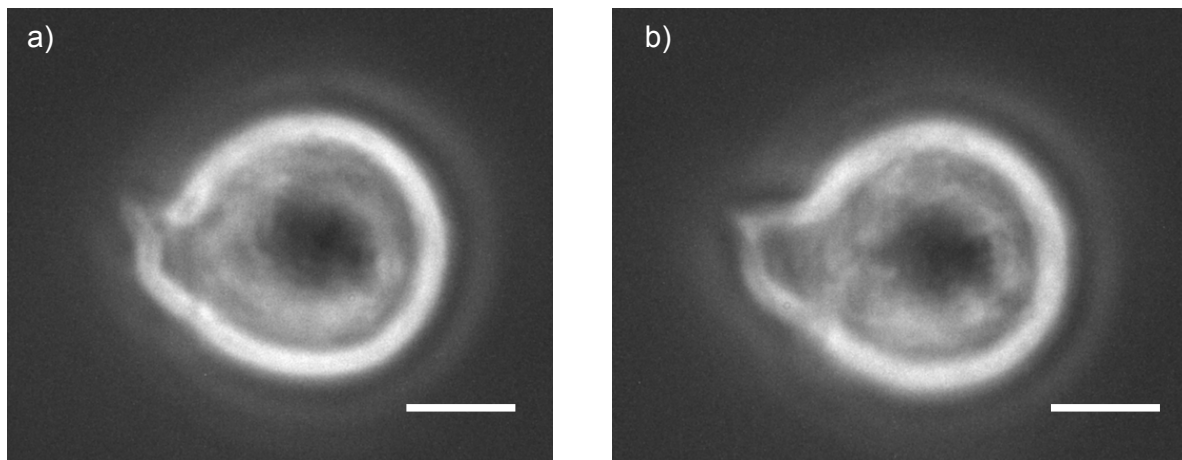
Figure 4.8: Protrusion of Jurkat cell a) A Jurkat cell that has formed a small protrusion along the optical axis undergoing trapping stress (0.2 W). b) The same cell at the end 5 second stretch at 0.75 W. Note that while the extrusion has grown, the rest of the cell appears to remain similar to its state at the beginning. This cell is not usable for compliance measurements due to its decidedly non-spherical shape. Scale bar 5 μm.

As predicted in Figure 2.15, the interpolated global geometric factor depends mainly on the fiber distance. This corresponds to the impression one gets from directly calculating the global geometric factor (see Section 2.4). Fitting the relationship of the fiber distance to the interpolated geometric factor with an inverse quadratic curve (Figure 4.9 a) shows the fitting equation is a polynomial with both an inversely linear and an inversely quadratic relationship. The inversely quadratic relationship of the optical stress

depends on the laser beam width. By plotting the measured deformation versus the refractive index of the cell, it is visible that the effect of the refractive index on the deformation of the cell is small compared to the biological variance of actual cell deformation (Figure 4.9 b). This means that while the refractive index does play a role in the determination of the optical forces acting on the cell, its effect is negligible as compared to the effect of the cytoskeleton on the mechanical properties.
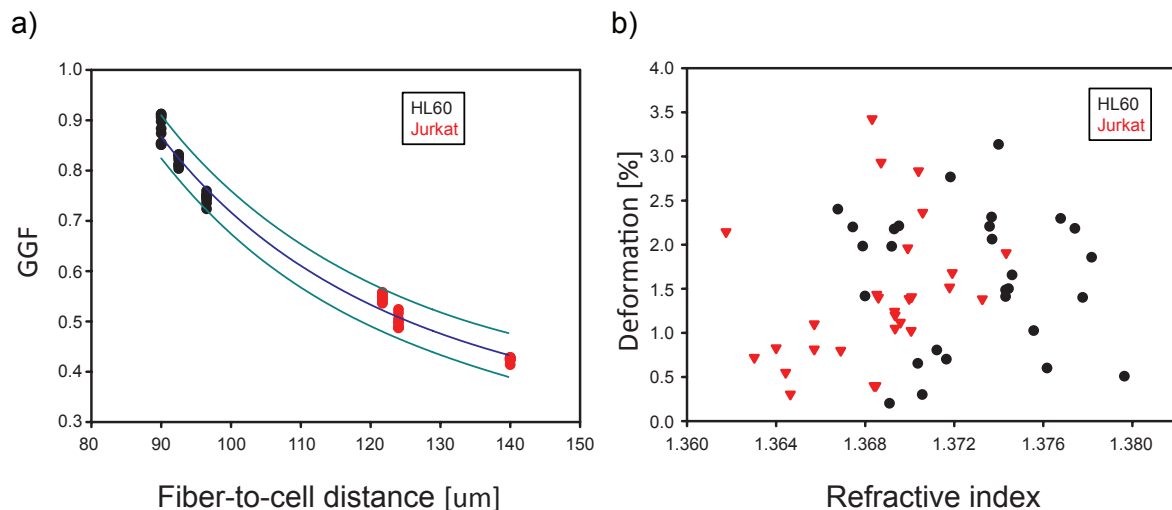
Figure 4.9: Cell deformation variation between sets is mainly dependent on fiber distance. a) The interpolated geometric factors of different cell lines (HL60 black, Jurkat red) can be fit to an inversely quadratic equation (fit blue, 95 % confidence intervals cyan) corresponding to the fiber distance. b) The refractive index of a cell does not correlate with its deformation for both cell lines.

On a side note, a longer incubation time after splitting cell populations and refreshing the cell medium seems to soften the Jurkat cells. For example, comparing the compliance of Jurkat cells shortly after splitting with the compliance after three days of incubation at 37°C shows a significant softening of the older cells (Figure 4.10). This observation is supported by independent real-time deformability cytometry [147] measurements. It is possible that the acidification over time of the growth medium leads to a change in metabolism, as cancerous cells are known to increase aerobic glycolysis in acidic environments [148] which is called *Warburg effect* [149]. As statistics are very low, further studies might prove fruitful.

4.2.3.3 Classification

The data acquired from the homogeneous populations of HL60 and Jurkat cells was used to determine if a useful classification boundary can be obtained following the multivariate procedure described in Section 2.5 and if this multivariate classification boundary can improve classification with respect to a monovariate boundary. For this, three parameters were chosen: cellular radius, refractive index and me-
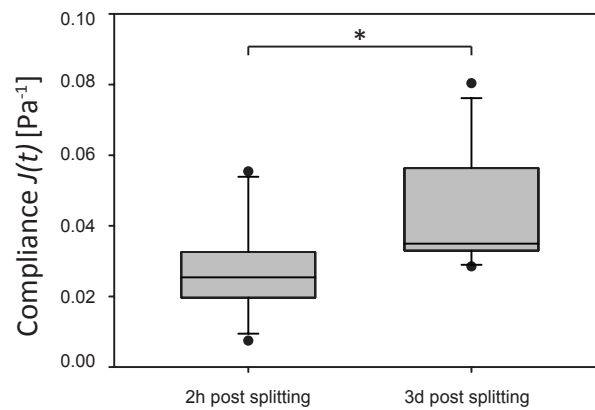
Figure 4.10: Jurkat cells soften with incubation time. Jurkat cells measured shortly after splitting
($n = 19$) are significantly ($p = 0.003$) softer than cells measured after 3 days post splitting,
leading to a more acidic environment ($n = 11$).

dian compliance. Cellular fluidity was not chosen as a parameter as the difference between experimental
sets was too great. First, the same procedure as in Chapter 3 was used to determine the boundary value
for each of the used parameters (Figure 4.11 a,b,c).

The calculated boundaries and expected errors are given below:

Monovariate classification	Radius	Refractive Index	Compliance
Significance between populations	$p < 0.001$	$p < 0.001$	$p = 0.016$
Boundary value	$7.56 \, \mu m$	1.3701	$0.182 \, \text{Pa}^{-1}$
Misclassified HL60 cells	1	8	16
Misclassified Jurkat cells	11	6	6
True HL60 rate (THR)	96.4 %	76.9 %	66.6 %
True Jurkat rate (TJR)	60.7 %	73 %	57.9 %
Accuracy	78.6 %	75 %	60.7 %

The question is whether the best result of 78.6 % accuracy can be improved by using multivariate classi-
fication in three dimensions such as proposed in Section 2.5. Solving the necessary equations leads to a
boundary function

$$-14.28 - 0.03 \times radius + 10.62 \times RI - 2.63 \times compliance = 0$$

However, using linear discriminant analysis only leads to an accuracy of 76.8 %. The linear discriminant
misclassifies 8 of the HL60 cells and 5 of the Jurkat cells. The rate of correctly detected HL60 cells THR
is 80 %, that of correctly detected Jurkat cells TJR 74.2 %.

How can this result be interpreted, as it appears, at first glance, to be worse than with monovariate
classification using the radius? The good result of the radius classifier is mainly due to the fact that the
Jurkat radii include a set of large outliers (Figure 4.7 a, Set 3) which possibly inadvertently stems from
the experimenter's choice of cells. The classification boundary thus lies at a value that is higher than
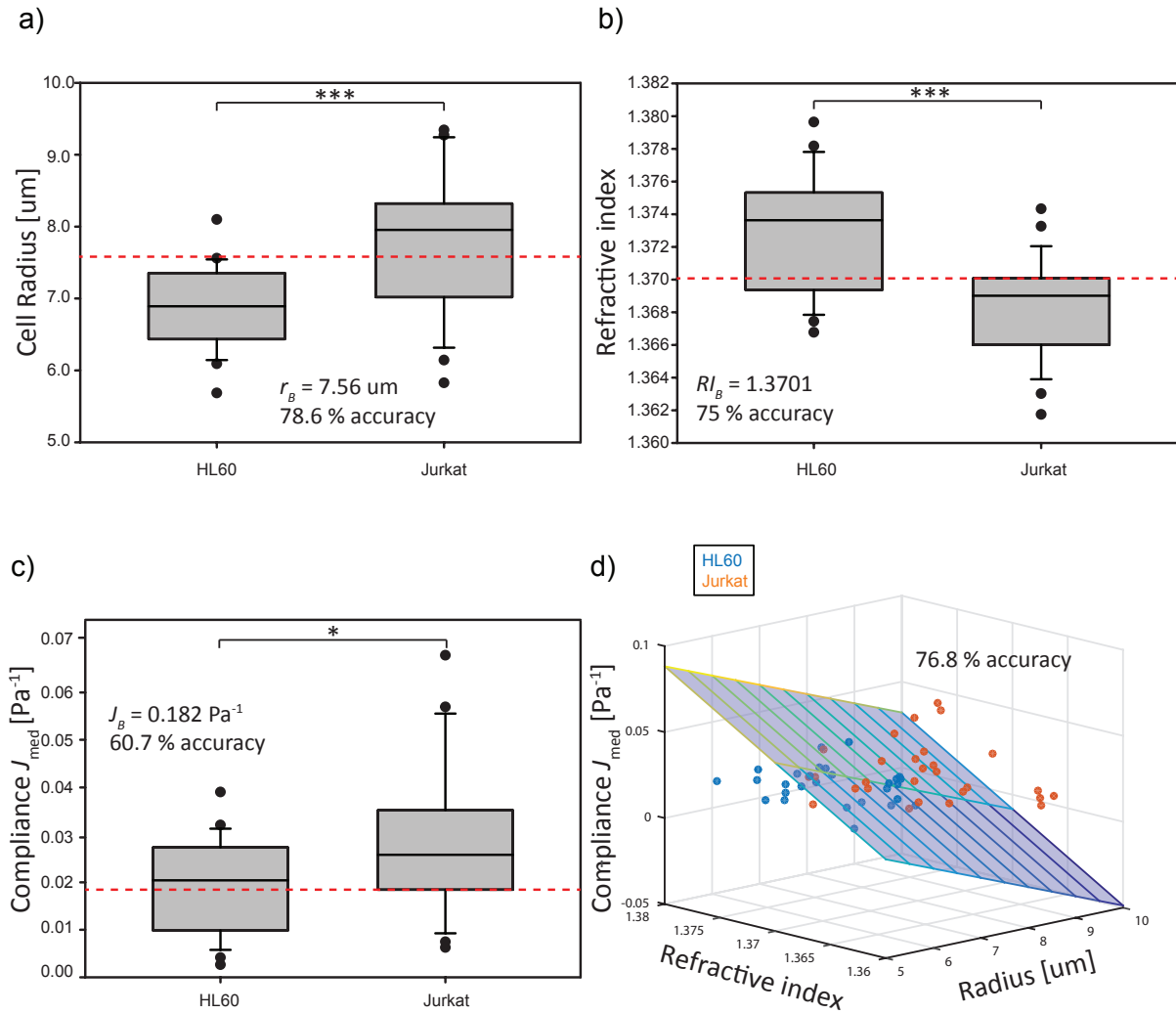
Figure 4.11: Choice of boundary values for mono- and multivariate classification. HL60 and Jurkat
cells (each $n = 28$) can be classified by one parameter each. The populations differ from
each other significantly with each parameter (radius $p < 0.001$, refractive index $p < 0.001$,
compliance $p = 0.016$). For each parameter (a) radius, b) refractive index, c) compliance)
a boundary value is determined and the misclassification error is giving. d) plots the com-
bined data points in a three-dimensional lattice. In blue, the boundary plane found by linear
discriminant analysis.

almost all HL60 cells' radii. All HL60 thus will be sorted correctly, while a large number of Jurkat cells fall below this range and are misclassified. This is reflected in the true classification rates THR and TJR which are closer to each other in the multivariate case. As is expected in this specific device, very small sample numbers distort the outcome significantly. It thus makes sense to enlarge the sample size as much as possible.

4.2.3.4 Inter-setup variance

If one desires to raise the sample size, might it be advantageous to first use a faster device such as the monolithic glass chip or the single capillary device to classify a homogeneous population, thus raising the training sample number and improving its confidence level, after which the results are used for classification of a heterogeneous population in the open device. To resolve this issue, it is necessary to see if all current Optical Stretcher devices are actually comparable. These devices are; the closed microcapillary setup [7], the monolithic glass chip presented in Section 3, and the micropipette setup presented here. In theory, although one would expect the deformations of the same cell populations to differ due to different stretching powers and changing fiber distances, normalizing by the global geometric factor should lead to similar compliance curves, since the mechanical behavior of cells should not depend on the device they are measured in.
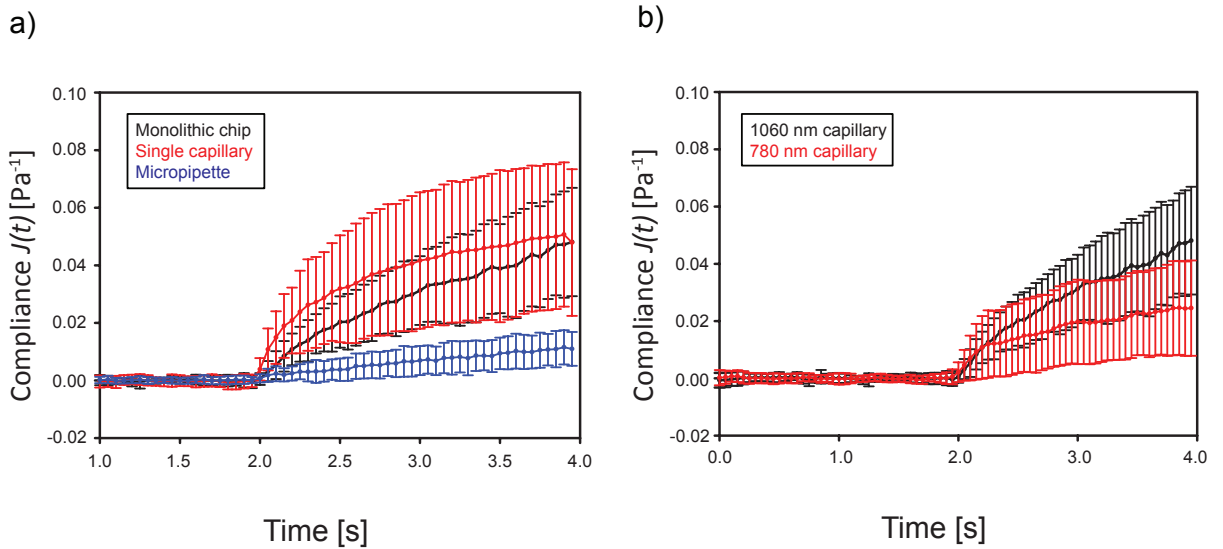
Figure 4.12: Comparison of different stretcher types a) Normalized compliance plot of HL60 cells stretched at 1064 nm in a microcapillary setup (black, laser power per fiber $P = 0.78W$, refractive index $RI = 1.38$, fiber-to-cell distance $d = 80\mu m$, number of cells $n = 22$), monolithic chip (red, $P = 1.2W$, $RI = 1.38$, $d = 188\mu m$, $n = 21$) and micropipette setup (blue, $P = 0.75W$, $n = 28$, variable fiber distances and refractive indices). b) Normalized compliance plot of HL60 cells stretched at 1064 nm (microcapillary setup, values as in a, in black) and at 780 nm (microcapillary setup, $P = 0.78W$, $n = 27$, $RI = 1.38$, $d = 80\mu m$, in red).

However, it appears as though the global geometric factor normalization does not hold for all setups

equally (Figure 4.12 a). While both the monolithic chip and the microcapillary setup have similar compliance values, the open micropipette setup has a much lower apparent compliance. A possible reason for this is differing temperature increase during the stretch. Heating, which occurs due to the absorption of energy in water at 1064 nm, seems to have significant effect on the deformability of cells [24, 27]. This is supported by comparing HL60 deformation by 1064 nm and 780 nm lasers (Figure 4.12 b). As the absorption of 780 nm light in water is much lower than in 1064, heating is less pronounced and cells deform less.

Previous temperature distribution simulations were done using the nonhomogeneous heat equation [150, 151]

$$\nabla^2 T = -\frac{1}{k}S + \frac{c_p \rho}{k}$$

where T is the temperature distribution in the system, k is the thermal conductivity of the medium, S is the heat source, c_p is the heat capacity of the medium, ρ is the density of the medium, and t is time. They showed that cells experience a temperature increase of around 10 Kelvin per Watt [150] when using a 1060 nm laser. However, this does not take the convective component of heat transfer into account, which adds another partial differential term depending on the geometric properties of the system to the equation.

Thermal conductivity k describes the material's capability to transport heat from a warm to a cold region in the material. Glass has a higher conductivity than water, so one would expect heat in a system with more glass (as in the monolithic chip and the glass capillary setup) to be transported away faster, lowering the overall temperature. However, the glass in these setups constrains the system so that a convective flow never is established. Thus, the open setup might have a lower heat conductivity, but the convective currents spread the heat faster over a larger region. The bottom glass slide might also act as a heat reser- voir and pull heat from the large area of liquid. It would be interesting to compare heating in the open setup with different slide thicknesses and see if a setup with a thinner glass slide heats up more.
These observations indicate that different setups cannot be compared and simulations might be necessary to establish the exact amount of heating in an open setup, as such simulations have only taken place in glass capillary setups [150, 27, 22].

4.3 Concluding remarks

This chapter discussed a Optical Stretcher device capable of inserting, measuring and extracting rare single cells. New classification boundaries can be found using individual cell data, such as per cell compliance and refractive index measurements. Through the use of a micropipette, all cells can be retrieved and placed wherever needed. In future, this device can be used for the measurement and classification of rare or shear-dependent cells or biological structures.

It is shown that two different cell lines, HL60 and Jurkat, can be distinguished with a certain accuracy by their radius, compliance or refractive index. The classification accuracy is high when using multivariate classification. It is likely that increasing sample sizes improves classification.

Furthermore, it is shown that the measurement of the global refractive index of a cell does not affect its compliance as much as the biological conditions within. This means that calculating a global geometric factor does not improve classification of cells other than taking into account the fiber distance for different setups. The chapter also discusses the difficulty of comparing different optical stretcher devices. Intra-device measurement variations mainly arise due to different heating conditions within the differing setups. This potentially makes previous assumptions about the open setup [22] that have been made by simulating heating in a closed setup invalid.

One major drawback is the very low throughput of around 10 cells per hour. While data acquisition of deformation and phase might be sped up slightly, the main time constraint is the transport of the cells to and from the optical trap, leaving little room for improvement.

Due to the bending of thin glass coverslips, fluorescence imaging is hampered. It is possible that a solution which tightly clamps the thin slide while leaving space for the micropipette to maneuver might solve this problem in the future.

5 Conclusion and Outlook

5.1 Conclusion

In the last fifteen years, the Optical Stretcher has evolved from a groundbreaking prototype to an established tool used on a daily basis for research on the mechanical properties of single cells. The wide range and impact of research on cells over the years show that the Optical Stretcher has proven its worth. In this thesis, the "industry standard" consisting of a single capillary setup has been extended to achieve more functionality. Several questions were identified in the introductory chapter. In this conclusion, these questions will be adressed individually once more, emphasizing capabilities and limitations.

Obtaining single cell data in real time

In order to effectively gather viscoelastic data on individual cells, both their deformation and refractive index have to be measured. An image processing algorithm based on circular shortest path calculations was designed to detect cell boundaries. In real time, its applicability was achieved through use of recursive properties of the image, leading to analysis capabilities of up to 20 frames per second. Furthermore, a camera that is able to acquire phase images was successfully used to calculate the refractive index of a cell while still in the optical trap. The deformation data obtained by image processing was combined with the refractive index in order to calculate the *geometric factor* which is needed to calculate the creep compliance of a cell. Since this calculation is time intensive, an interpolation method was presented to rapidly interpolate the geometric factor from a dataset of known values in three dimensions. Finally, statistical methods that allow classification of unknown cells into separate populations were presented. These additions allow for the analysis and classification of cells while they are still present in the optical trap, a significant acceleration of the analysis process which previously had to be done after the experiment.

Some constraints however still apply. The image processing algorithm has a minimum speed on a conventional computer of 50 ms per frame. Cameras with higher acquisition speed therefore cannot be used for analysis in real time, which limits the temporal resolution of the compliance measurement. It is likely that processing time can be lowered by effective use of multicore computing and dedicated fast-access memory. The proposed workflow for obtaining a refractive index measurement introduces some uncertainty since the acquisition of the phase and the reference image are separate by around 30 s. Movement of the microscope and artifacts within the measurement zone can undetectably skew the refractive index measurements. The interpolation data set is not large as yet. This means that values that exceed the boundaries of the current dataset are misinterpolated. To extend the range of possible measurement, laser power should be included as a variable. Finally, statistical analysis may become more involved if normality of the data and class covariance equality are not given. In biological research, non-normal data is common, especially with small sample sizes. This might lead to high misclassification.

Classifying single cells and sorting them:

Optical stretcher technology was extended by introducing a novel device capable of sorting unknown cells into distinct classes according to their mechanical properties. This device was a monolithic Optical Stretcher consisting of two glass halves chemically etched and bonded together and connected both to a microfluidic pressure pump as well as to a laser via optical fibers. The capabilities of this chip were demonstrated by measuring the compliance of two different homogeneous cell population with a throughput of 50 - 100 cells per hour. A boundary compliance value determining the minimum error discriminant between the two populations was calculated using the previously introduced methods. Then, a sample consisting of equal amounts of cells of both populations was measured. Each cell was sorted into one of the previous populations according to its individual compliance. Simultaneously, fluorescence detection was used as a control. Sorting was achieved by optical manipulation of cells while in the optical trap. Results showed that in this case, the sorting accuracy was on the order of 75 % correctly classified cells. The work presented was the first demonstration of active marker-free mechanical phenotyping of single cells.

Nonetheless there are some limitations. Due to the fabrication process, the imaging quality is somewhat reduced compared to previous devices. This can affect the cell contour detection algorithm. Due to increased fiber distance, more optical power is needed for deformation of cells. More critically, cells are not extractable from the chip. This is mainly due to the small sample size measured, since cells get lost in the connecting tubing after leaving the chip. A method to extract cells is needed so that the sorted cells may be reused.

Rare cell measurements:

In order to measure small samples of rare cells or to be able to extract all measured cells in their entirety, an Optical Stretcher device is needed that dispenses with connecting tubing. Here, a simple design based on the first Optical Stretcher, called *open setup*, was presented. An optical trap was assembled on a structured glass slide and a micropipette was used to deliver and remove cells into the optical trap. Imaging quality was maintained through use of an O-ring. The functionality of this device was demonstrated by the measurement of the deformation as well as the per cell refractive index of two distinct cell populations that were placed on a reservoir on the microscope stage holder. The setup was able to select cells from this reservoir and move them to the optical trap without losing them. The effect of the cellular refractive index on the cell's deformation is shown to be negligible compared to the biological variance of the cytoskeleton of cells. Using the data acquired from both deformation analysis as well as individual refractive index measurements, the data was classified into two groups and the potential classification was error estimated using both mono- and multivariate classification. This work showed that a simple but effective Optical Stretcher device can be fabricated in a straightforward manner and that it is useful for applications, particularly with rare or shear-sensitive cells.

The main limitation of the open setup is its low throughput. The measurement rate of 10 cells per hour leads to very low sample sizes, making statistics more unreliable since normality and covariance equality assumptions are potentially violated. In fact, classification results are significantly skewed due to outliers. Furthermore, the calculation of an individual geometric factor does not have the desired effect of reducing misclassification of cells since its value depends mainly on fiber distance. Results of measurements obtained in the open setup cannot be related to results obtained in closed setups due to

different temperature conditions which affect cellular deformability. Finally, due to bending of the chip, a thick glass slide has to be used, which can affect fluorescence measurements.

5.2 Outlook

The Optical Stretcher has evolved and with it, so has its field of applications. When work on this thesis began, the Optical Stretcher was still one of only a few tools to measure cellular deformability of suspended cells without using any sort of marker and with a reasonable throughput. This has changed rapidly within a few years. New techniques with very high throughput, both actively and passively measuring cellular deformability, have emerged [97, 100, 152, 153], most notably real-time deformability cytometry [123] which is poised to become both a commercial diagnostic as well as a fundamental research device. These techniques' throughput of thousands of cells per second stands in stark contrast to the Optical Stretcher's maximal throughput of 100 cells per hour. Does this mean that the era of the Optical Stretcher has passed and further research in this direction is no longer needed?

The answer is no. The Optical Stretcher remains a valid research tool for a variety of reasons. This dissertation is accordingly subtitled "Towards a Cell Sorter Based on High-Content Analysis", as the rapid methods mentioned above do not provide detailed information about the temporal behavior of cells that can only be obtained by the Optical Stretcher. To derive viscoelastic parameters or to measure active cellular response to external forces, the Optical Stretcher remains the non plus ultra. The extension of capabilities that are described in this thesis allows the integration of fluorescent markers as well as a determination of the cellular refractive index. In fact, as yet the Optical Stretcher is the only technique that combines temporally resolved measurement of cellular mechanics with individual measurements of the refractive index.

Where does the Optical Stretcher go from here? In its current form, the Optical Stretcher cannot be used as a diagnostic device in a clinical setting due to its low throughput. However, the sorting capability introduced in this work is a helpful addition to research. Cells with interesting mechanical properties can be retrieved directly after measurement and can be subjected to further investigation. The Optical Stretcher could then act like a FACS machine, sorting on mechanical rather than fluorescent properties. Special note must be taken of the open setup. Although its throughput is admittedly low and makes standard population measurements tedious, it has the unique capability of measuring and sorting samples with low cell numbers, which is not possible in high-throughput alternatives. Due to its simple setup, it can be readily combined with tools such as optical tweezers. The open setup can even be used to measure non-cellular or shear-sensitive objects and research concerning the forces acting on the mitotic spindle of *Xenopus* [154], as well as research on the viscoelastic properties of perinuclear P granules [155], is currently ongoing.

Technically, the future will potentially see a complete integration of all necessary components for both deformation and refractive index measurements such as the Optical Stretcher with fluorescent capabilities, the DHM (still superior to the SID4Bio camera in terms of speed and resolution) and the Optical Cell Rotator [156] into one simple setup that allows both detailed viscoelastic measurements with a good

temporal resolution as well as a detailed three-dimensional refractive index map of a cell. This technical setup could aid further investigation of the difficult problem of relating cellular behavior under stress to the cell's inner components.

This thesis has aimed to extend the scope and possibilities of research with the Optical Stretcher. While certain difficulties remain, the Optical Stretcher is now ready to answer a new range of biomechanical questions.

6 Publication list

The following publications resulted from the work presented here:

1. **C. Faigle**, F. Lautenschläger, G. Whyte, P. Homewood, E. Martin-Badosa, J. Guck. A monolithic glass chip for active single-cell sorting based on mechanical phenotyping, *Lab on a Chip* **15** 1267 (2015)

2. **C. Faigle**, J. Guck. A recursive shortest circular path algorithm for real-time cell contour detection (in preparation)

7 Appendix

7.1 Materials and Methods

7.1.1 Microfluidic fabrication

7.1.1.1 Wet etching

The process of microstructuring bulk material by chemically removing parts of it is called etching. The material to be etched is covered with a resistant masking material that protects areas that are not to be etched.
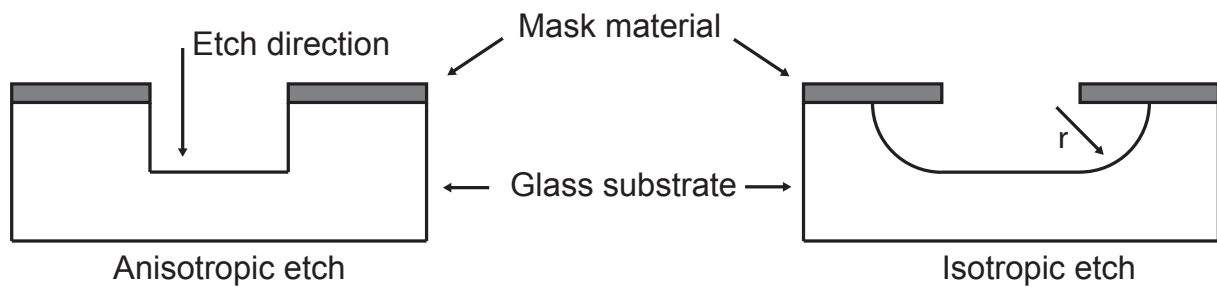
Figure 7.1: Comparison of etching types. Anisotropic etching etches in one direction, leading to straight channel walls, while isotropic etches radially with a depth r, typically undercutting the masking material.

There are two main ways of etching: dry etching, which can be divided into physical and chemical etching, and wet etching (Figure 7.1). In physical dry etching, the bulk substrate is sputtered by reactive ions or electrons, vaporizing the substrate material when it is unprotected. This allows for highly anisotropic (etching downwards much more than sideways) features, but has a low etch rate of around 500 nm/min [157] and is very energy intensive. Chemical dry etching is also known as plasma etching. It typically has a fluorine [158] or chlorine [159] gas as a source for the plasma which flows over the substrate. Free radicals produced by the plasma react with the surface of the substrate to volatile products. This process can be both isotropic and anisotropic depending on the substrate material and process parameters. Wet etching, on the other hand, is done with liquid chemicals. For etching oxides such as glass (mainly composed of silicon dioxide), hydrofluoric acid [160] is used. In contrast to dry etching, it is isotropic. This means that the resistant mask, usually made from photoresist or a deposited Cr/Au layer, will be undercut, leaving a semicircular edge in the material, which has to be accounted for when designing the mask. This is a good solution for creating semicircular grooves in the glass. The high etch rate of around 8 μm/min in glass and the straightforward process, compared to plasma etching, make this the gold standard of bulk microstructuring. Wet etching allows for the creation of as many devices at a time as the resist mask allows, which can be reused indefinitely. However, for each new design a new

mask must be made. It also does not allow simple structuring of three-dimensional devices. Because hydrofluoric acid is such a hazardous chemical, a specialized cleanroom and expertise is imperative.

The device presented in this work is microstructured from B270 Superwite soda lime glass. Two glass sheets are wet etched in a buffered oxide etch [161] solution according to the design presented in Section 3. After etching, the two glass halves are thermally bonded [162] by bringing them into contact with each other at high temperatures, through which covalent connections are formed between the two halves. Fluidic connections are achieved by tightly pressing sealed tubing connectors (8-way linear connectors) to the side of the chip. The device presented in this work was produced by Dolomite Microfluidics (Royston, United Kingdom).

7.1.1.2 Photoresist lithography

Photoresist lithography describes the structuring by light of a photoresist which has been coated on a substrate material. Typical photoresists such as SU-8 [163] (Microchem Corp.) used in this work consist of a cross-linking polymer and a solvent that allows it to spread on a surface. This photoresist is poured onto a glass slide (subsequently called chip) that has been cleaned and coated with an adhesion-improving chemical named OmniCoat (Microchem Corp.). Once the resist has been spread to a defined height using a spin coater, it is baked in a two-step process at 65 °C and 95 °C for a time which depends on the height of the photoresist, in this case either 25 or 40 μm. The solvent evaporates during the baking, solidifying the coating (Figure 7.2 a). Then, a fused silica mask with a structured chrome film is placed tightly onto the glass slide, which is irradiated with an UV lamp (350 nm wavelength) for a precise time depending on the thickness of the SU-8 layer. Since SU-8 is a negative photoresist, irradiated areas will *crosslink*, while areas blocked by the chrome film will not (Figure 7.2 b). Another *soft bake* (baking at 95 °C) for a few minutes strengthens the crosslinking. Finally, the chip is placed into a developer bath which dissolves the unlinked photoresist (Figure 7.2 c). The manufacturer recommends washing the developed slide with Isopropyl Alcohol and curing the developed photoresist in a *hard bake* (150 °C), although adverse effects can occur (see Section 4.2.2.1). Now, the glass slide has an additional SU-8 structure of constant height and straight walls.
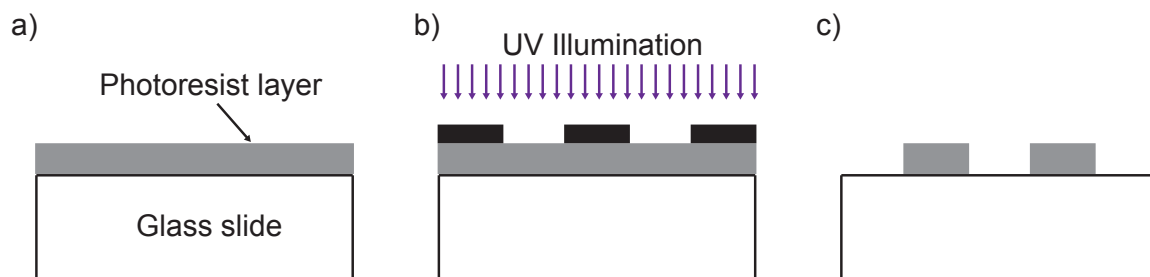
Figure 7.2: Photoresist lithography. a) SU-8 Photoresist is spin-coated onto a glass slide and baked. b) Photoresist is irradiated through a chrome mask (shown in black) . c) After development and rinsing, only irradiated areas remain.

Photoresist lithography is a simple process, so it can be done in-house with non-hazardous chemicals,

greatly reducing the production time. Through the use of large photomasks, chips can be produced in great quantities. However, as with wet etching, a new mask is required each time a new design is produced.

The chips used in this work were produced in cooperation with Saeed Ahmed and Joan Antoni Soler Blasco (both BIOTEC, TU Dresden).

7.1.2 Cell culture

7.1.2.1 HL-60

HL-60 (Human promyelocytic leukemia) cells were cultured in RPMI-1640 medium containing 10% FBS, 2 mM L-glutamine and 100 U/ml penicillin-streptomycin (all from Life Technologies). Some cells were treated with 0.2 µM cytochalasin D (Sigma-Aldrich, C2618). Cells were typically split every two or three days and measured on the day following the splitting of the cell culture. Viability test were performed using the vital stain Trypan blue (Sigma-Aldrich, T8154) and used according to manufacturer's protocol.

7.1.2.2 Jurkat

Jurkat (Human T lymphocyte leukemia) cells were cultured in RPMI-1640 medium containing 10% FBS, 2 mM L-glutamine and 100 U/ml penicillin-streptomycin (all from Life Technologies). Some cells were treated with 0.2 µM cytochalasin D (Sigma-Aldrich, C2618). Cells were typically split every two or three days and measured on the day of the splitting of the cell culture.

7.1.3 Imaging

7.1.3.1 Image acquisition

For imaging, an Axiovert Observer Z.1 (Zeiss, Germany) microscope was used. Cell deformation was measured with a 40x (EC-Plan-Neofluar Ph2) or 63x objective (LCI Plan-Neofluar Ph2) and a Ph2 condenser plate. A monochrome CMOS camera (DMK 23UM021, The Imaging Source, Germany) was used to acquire images.

7.1.3.2 Fluorescence imaging

In addition to standard phase contrast imaging, the microscope used also offers the possibility of imaging fluorescent samples. For this, a mercury vapor lamp (HXP, Zeiss) is used as an excitation source. By use of a dichroic filter, only light of a certain excitation wavelength can pass and be is focused through the objective onto the specimen. The fluorescent probes in the specimen are excited and emit light at an emission wavelength, which is again picked up by the objective. In order to avoid the emission being drowned in the excitation light, a dichroic mirror and emission filter ensure that only the emission wavelength light reaches the ocular or camera. In this work, the fluorescent probe Hoechst 33258 nuclear stain (Sigma-Aldrich, 861405) with a maximum excitation/emission wavelength of 352 and 461 nm, respectively was used according to the manufacturer's protocol.

7.1.3.3 Digital holographic microscopy

Digital holographic microscopy (DHM) is a phase imaging method that uses interference between an object and a reference laser beam to calculate the phase shift of said object [84, 164]. The off-axis setup [83] (Figure 7.3) uses a coherent light source, in this case a HeNe-laser (P=0.5 W, $\lambda = 632.8$ nm, HNL050L-EC, Thorlabs, Germany) that is split using a non-polarizing beam splitter (BS1). The object beam passes through the sample, which delays its wavefront. This delayed wavefront is collected by a standard microscope objective (MO, 40x/NA 0.75, Zeiss, Germany) and recombines with the reference laser beam at a second beam splitter cube (BS2). This creates a characteristic interference pattern: an image of the object with superposed interference lines. In this case, object beam and reference beam are not exactly aligned, hence the name "off-axis". The resulting hologram is acquired by a monochrome CMOS camera (MCE-B013-UW, Mightex Systems, USA). The used DHM was built by Mirjam Schür-mann, BIOTEC TU Dresden.
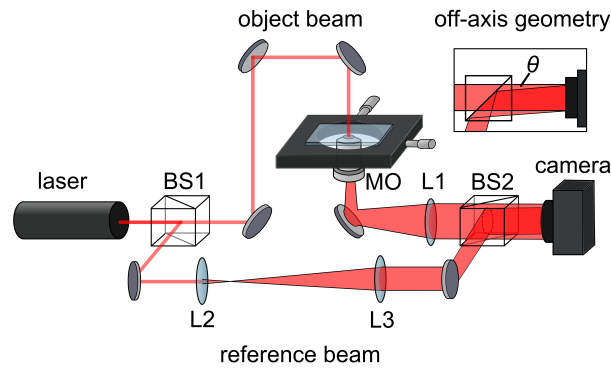
Figure 7.3: Schematic of DHM setup. The inset shows a top view of the off-axis geometry. Image courtesy of Mirjam Schürmann [59].

7.1.3.4 Quadriwave lateral shearing interferometry

The quadriwave lateral shearing interferometry used in the SID4Bio camera, also known as the modified Hartmann mask technique, is an evolution of the Hartmann-Shack [165] wavefront sensor technology. The Hartmann-Shack sensor consists of a microarray of optical lenses that focuses incoming light onto a camera in form of a spotted image. Shifts in the wavefront (i.e. by passing through an object) result in a shift of the spots on the image. With the help of a *reference* image, a phase image can be calculated from the local deviations. The modified Hartmann mask uses a diffraction grating instead of a lens array, which improves the resolution [166, 167]. Incident beams are diffracted into four directions. From their interference pattern, the intensity and phase change of the image can be reconstructed. The advantage of this sensor compared to the DHM is that it does not need a coherent light source or specialized optics and thus can easily be integrated onto any commercial camera [168] and microscope. In the case of this study, a dedicated SID4bio camera (Phasics, France) was used for the acquisition of phase images. The camera does the calculation of the phase image on-chip which severely limits the frame rate of 10 frames per second for intensity imaging to 2 frames per second for phase image calculation.

7.2 Software

Setup control programs were written with LabView (National Instruments, Austin, Texas). Image analysis algorithms were written in C++ on MS Visual Studio 2010 (Microsoft, Redmond, Washington) using the free libraries OpenCV [169] and DLIB [170]. Statistical analysis and graph visualization was done in SigmaPlot (Systat Software, San Jose, California) and MATLAB (MathWorks, Natick, Massachusetts). For single comparisons, the student's t-test was performed if data passed normality assumptions. If data did not pass normality assumptions, the Mann-Whitney U test was performed. For comparison of multiple sets of data, one-way Analysis Of Variance was used. Significance levels are visualized accordingly: 1 star corresponds to $p < 0.05$, 2 stars to $p < 0.01$, 3 stars to $p < 0.001$, while $n.s.$ (not significant) corresponds to $p > 0.05$.

7.2.1 Circular shortest path - recursive algorithm DLL

```cpp
1   // OS_CSPBruteRec_DLL.cpp : Defines the exported functions for the DLL application.
2   //
3   #include "stdafx.h"
4   #include "opencv2/imgproc/imgproc.hpp"
5   #include "opencv2/highgui/highgui.hpp"
6   #include "opencv2/opencv.hpp"
7   #include <stdlib.h>
8   #include <stdio.h>
9   #include "OS_CSPBruteRec_DLL.h"
10  #include <dlib/optimization.h>
11  #include <numeric>
12
13
14  const float INFINITY = std::numeric_limits<float>::max();
15  const float CONST_PI = 3.1415926;
16
17
18  //struct header
19  struct PathAndDistance {
20          cv::vector<cv::Point> Path;
21          float traversedDistance;
22          PathAndDistance() : Path(720) {}
23  };
24
25
26  //DLIB Levenberg−Marquadt algorithm
27
28  // −−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−
29
30  typedef dlib::matrix<double,1,1> input_vector; //input x, output y2
31  typedef dlib::matrix<double,2,1> parameter_vector; //major axis and minor axis
32
33  // −−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−
```

```
34
35    // We will use this function to generate data. It represents a function of 2 variables
36    // and 3 parameters. The least squares procedure will be used to infer the values of
37    // the 3 parameters based on a set of input/output pairs.
38    double model (
39        const input_vector& input,
40        const parameter_vector& params
41    )
42    {
43        const double p0 = params(0);
44        const double p1 = params(1);
45        //const double p2 = params(2);
46
47        const double i0 = input(0);
48        //const double i1 = input(1);
49
50        const double temp = p1*p1*(1−i0*i0/(p0*p0)); // + p1*i1 + p2;
51
52        return temp;
53    }
54
55    // ----------------------------------------------------------------------
56
57    // This function is the "residual" for a least squares problem. It takes an input/output
58    // pair and compares it to the output of our model and returns the amount of error. The idea
59    // is to find the set of parameters which makes the residual small on all the data pairs.
60    double residual (
61        const std::pair<input_vector, double>& data,
62        const parameter_vector& params
63    )
64    {
65        return model(data.first, params) − data.second;
66    }
67
68    // ----------------------------------------------------------------------
69
70    // This function is the derivative of the residual() function with respect to the parameters.
71    parameter_vector residual_derivative (
72        const std::pair<input_vector, double>& data,
73        const parameter_vector& params
74    )
75    {
76        parameter_vector der;
77
78        const double p0 = params(0);
79        const double p1 = params(1);
80        //const double p2 = params(2);
81
```

```
82        const double i0 = data.first(0);
83        //const double i1 = data.first(1);
84
85        const double temp = p1∗p1∗(1−i0∗i0/(p0∗p0)); // + p1∗i1 + p2;
86
87        der(0) = p1∗p1∗2∗i0∗i0/(p0∗p0∗p0);
88        der(1) = 2∗p1∗(1−i0∗i0/(p0∗p0));
89        //der(2) = 2∗temp;
90
91        return der;
92   }
93
94   // − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −
95
96
97   /// Function headers
98   cv::vector<float> linspace(float, float,int);
99   void bruteForcePathFinder( cv::Mat&, cv::Point, cv::Point, PathAndDistance &); //returns path coordinates
100  cv::Point PathsTouching(cv::vector<cv::Point> &, cv::vector<cv::Point> &);
101  //returns first common point (can't be zero due to constraint that calculation starts at different area
102  void newShiftedPath (const cv::Mat&, cv::Point, PathAndDistance &); //returns path through mat from common point
103
104  OS_CSPBruteRec_DLLV_API int OS_CSPBruteRec_DLL(int IMAGE_HEIGHT,
105                                                 int IMAGE_WIDTH,
106                                                 unsigned char ∗IMAGE,
107                                                 float minRadLV,
108                                                 float maxRadLV,
109                                                 double ∗majAx,
110                                                 double ∗minAx)
111  {
112
113          /// Load an image
114          cv::Mat src = cv::Mat(IMAGE_HEIGHT, IMAGE_WIDTH, CV_8UC1, IMAGE);
115
116          float steps = 400;
117          float minRad = minRadLV; //00;
118          float maxRad = maxRadLV; //200;
119
120  //get centroid of src center for remapping to polar coordinates
121          cv::Mat src_gray, thresholded;
122          //cv::cvtColor(src, src_gray,CV_RGB2GRAY);
123          src.copyTo(src_gray);
124          cv::threshold(src_gray, thresholded, 0,255,cv::THRESH_BINARY+cv::THRESH_OTSU);
125          cv::Moments mom = cv::moments(thresholded, true);
126          //cv::Moments mom = cv::moments(src_gray, false);
127
128          cv::Point CoM(mom.m10/mom.m00, mom.m01/mom.m00);
129
```

```
130
131         //convert to polar coordinates
132
133
134         cv::Mat angle;
135         cv::Mat radius;
136         cv::Mat xCoord;
137         cv::Mat yCoord;
138         cv::Mat source32;
139         src_gray.convertTo(source32, CV_32F);
140
141         std::vector<float> linearAngle = linspace(0,2*CONST_PI,720);
142         //linspace approx function a la Matlab 0,2pi,512 -> minAng, maxAng, Ang graduations
143         std::vector<float> linearRadius = linspace(minRad,maxRad,steps);
144         // 0, 400, 512 -> minRad, maxRad, number of Rad graduations
145
146         cv::repeat(cv::Mat(linearAngle).reshape(1,1).t(), 1, cv::Mat(linearRadius).total(), angle);
147         //imitates Matlab meshgrid
148         cv::repeat(cv::Mat(linearRadius).reshape(1,1), cv::Mat(linearAngle).total(),1, radius);
149
150         cv::polarToCart(radius,angle,xCoord,yCoord);
151
152
153         //remap polar coordinates
154         cv::Mat remapped;
155         remap(source32,remapped,xCoord+CoM.x,yCoord+CoM.y,
156                     CV_INTER_LINEAR,cv::BORDER_CONSTANT,255);
157         //around center point
158         cv::Mat filtered = cv::Mat(remapped.size(), remapped.type());
159
160 Scharr( remapped, filtered, CV_32F, 1, 0);
161
162 double min2, max2;
163 cv::Point minloc, maxloc;
164 cv::minMaxLoc(filtered, &min2, &max2, &minloc, &maxloc);
165 cv::add(filtered,min2*(-1),filtered);
166
167
168         //initialize
169         std::vector<PathAndDistance> vectorPathAndDistance(3);
170         cv::Point CommonPoint = cv::Point(0,0);
171         int SPvectorIndex = 0;
172
173         //first round: get boundary shortest paths.
174          bruteForcePathFinder(filtered, cv::Point(100,0), cv::Point(100,719), vectorPathAndDistance[0] );
175          bruteForcePathFinder(filtered, cv::Point(300,0), cv::Point(300,719), vectorPathAndDistance[1] );
176
177
```

```
178        //do these paths have a common point? if yes, use it to recalculate a new path.
179        //add this path to the total path list.
180
181        //check if there is a common point. If there is, this one will be the shortest path.
182        if (CommonPoint != cv::Point(0,0)){
183                //calculate new path
184                newShiftedPath(filtered, CommonPoint, vectorPathAndDistance[0]);
185                SPvectorIndex = 0;
186        }
187        else { //there is no touching point
188
189                //expand middle vector
190                bruteForcePathFinder(filtered, cv::Point(200,0), cv::Point(200,719), vectorPathAndDistance[2]);
191
192                //common point between vector 0 and 2
193                CommonPoint = PathsTouching(vectorPathAndDistance[0].Path,vectorPathAndDistance[2].Path);
194
195                //check if there is a common point
196                if (CommonPoint != cv::Point(0,0)){
197                        //calculate new path
198                         newShiftedPath(filtered, CommonPoint, vectorPathAndDistance[0]);
199                }
200                //common point between vector 1 and 2
201                CommonPoint = PathsTouching(vectorPathAndDistance[1].Path,vectorPathAndDistance[2].Path);
202                //check if there is a common point
203                if (CommonPoint != cv::Point(0,0)){
204                        //calculate new path
205                         newShiftedPath(filtered, CommonPoint, vectorPathAndDistance[1]);
206                }
207                if (vectorPathAndDistance[0].traversedDistance
208                        < vectorPathAndDistance[1].traversedDistance) {
209                        SPvectorIndex = 0;
210                } else {
211                        SPvectorIndex = 1;
212                }
213                if (vectorPathAndDistance[SPvectorIndex].traversedDistance
214                        > vectorPathAndDistance[2].traversedDistance){
215                SPvectorIndex = 2;
216                }
217
218        }
219
220
221
222        std::vector<float> linearRadiusEnd(720);
223        for (int row = 0; row < 720; row++){
224                linearRadiusEnd[row] = vectorPathAndDistance[SPvectorIndex].Path[row].x/steps
225
```

```
226              }

227

228

229

230              std::vector<float> linearPathFourier(720), linearPathX(720);
231              std::vector<float> linearPathRadiusEnd(720), linearPathY(720);
232              std::vector<cv::Point2f> linearPathXY(720);

233

234              cv::dft(linearRadiusEnd, linearPathFourier, cv::DFT_SCALE); //FFT of interpolated values
235              for (int a = 19; a < 720; a++) { //keep only first 10 fourier components
236                      linearPathFourier[a] = 0.0;
237              }

238

239              cv::dft(linearPathFourier , linearPathRadiusEnd, cv::DFT_INVERSE); ///inverse fourier transform
240              cv::polarToCart(linearPathRadiusEnd,linearAngle,linearPathX,linearPathY);

241

242              std::vector<std::pair<input_vector, double> > data_samples;
243         input_vector input;
244              double output;

245

246              for (int i = 0; i<720; i++){
247                      circle(src, cv::Point(linearPathX[i]+CoM.x,linearPathY[i]+CoM.y),0,255); //draws smoothed contour
248                      linearPathXY[i].x =linearPathX[i];
249                      linearPathXY[i].y =linearPathY[i];
250                      if (i % 4 == 0 ){
251                      input(0,0)=linearPathX[i];
252                      output = linearPathY[i]*linearPathY[i];
253                      data_samples.push_back(std::make_pair(input, output));
254                      }

255

256              }
257              /* opencv fitted ellipse
258              cv::RotatedRect ell = cv::fitEllipse(linearPathXY);
259              ell.center.x += CoM.x;
260              ell.center.y += CoM.y;
261              cv::ellipse(src, ell, 255,2); //draws fitted ellipse

262

263              *majAx = ell.size.width;
264              *minAx = ell.size.height;
265              */
266                //guessing circle parameters

267

268                      parameter_vector x;
269                    x(0,0) = abs(linearPathX[0]);
270                   x(1,0) = abs(linearPathX[128]);

271

272                      // Use the Levenberg-Marquardt method to determine the parameters which
273              // minimize the sum of all squared residuals.
```

```
274            solve_least_squares(dlib::objective_delta_stop_strategy(1e−7),
275                                      residual,
276                                      residual_derivative,
277                                      data_samples,
278                                      x);
279
280
281
282            *majAx = x(0,0);
283            *minAx = x(1,0);
284
285        cv::ellipse(src, cv::Point(CoM.x, CoM.y), cv::Size(x(0,0),x(1,0)),0,0,360,10,2);
286
287            return 0;
288    }
289
290    cv::vector<float> linspace(float a, float b, int n) {
291            cv::vector<float> array(n);
292            float step = (b−a) / (n−1);
293
294            for (int i = 0; i < n; i++){
295                    array[i] = (a+i*step);
296            }
297            return array;
298    }
299
300    void bruteForcePathFinder( cv::Mat& image, cv::Point source, cv::Point goal, PathAndDistance &PD){
301
302            cv::Mat distance(720,400,CV_32F);
303            cv::Mat predecessor(720,400,CV_16U);
304            distance.row(0) = INFINITY;
305            distance.at<float>(source) = image.at<float>(source);
306
307
308            std::vector<float> t(3);
309            float* rowMinusPoint;
310            float* rowPoint;
311            float* imagePoint;
312            ushort* predPoint;
313
314            int colstart = 0;
315            int colend = 399;
316
317                    for (int row = 1; row < 720; row++){ //first row is already clear so start at 1
318
319                            //pointers to row locations
320                            rowMinusPoint = distance.ptr<float>(row−1);
321                            rowPoint = distance.ptr<float>(row);
```

```
322                         imagePoint = image.ptr<float>(row);
323                         predPoint = predecessor.ptr<ushort>(row);
324
325
326                 //create rhomb shape
327                 if (row <= image.rows/2) {
328                         if (source.x−row > 0){
329                                 colstart = source.x−row;
330                         }
331                         if (source.x+row < image.cols){
332                                 colend = source.x+row;
333                         }
334                 } else { //exactly other way around
335                         if (source.x−(image.rows−row) > 0 ) {
336                                 colstart = source.x−(image.rows−row);
337                         }
338                         if (source.x+(image.rows−row) < image.cols){
339                                 colend = source.x+(image.rows−row);
340                         }
341                 }
342
343                 for (int col = colstart; col < colend+1; col++){
344                         if (row <= image.rows/2){
345                                 if (col == colstart){  //first col (left)
346                                         t[0] = INFINITY; //outside bounds, so inf energy
347                                         t[1] = INFINITY;
348                                         t[2] = distance.at<float>(row−1,col+1);
349                                 }
350                                 else if (col == colend){
351                                         t[0] = distance.at<float>(row−1,col−1);
352                                         //t[1] = distance.at<float>(row−1,col);
353                                         t[1] = INFINITY;
354                                         t[2] = INFINITY;
355
356                                 }
357                                 else if (col == colstart + 1) {
358                                         t[0] = INFINITY; //outside bounds, so inf energy
359                                         t[1] = distance.at<float>(row−1,col);  //energy row above
360                                         t[2] = distance.at<float>(row−1,col+1);
361
362                                 }
363                                 else if (col == colend − 1) {
364                                         t[0] = distance.at<float>(row−1,col−1);
365                                         t[1] = distance.at<float>(row−1,col);
366                                         t[2] = INFINITY;
367
368                                 }
369                                 else {
```

```
370                                        t[0] = distance.at<float>(row−1,col−1);
371                                        t[1] = distance.at<float>(row−1,col);
372                                        t[2] = distance.at<float>(row−1,col+1);
373
374                                }
375                            if (col == colstart +1 && col == colend −1) {
376                                        t[0] = INFINITY;
377                                        t[1] = distance.at<float>(row−1,col);
378                                        t[2] = INFINITY;
379
380                                }
381                    } else {
382                            if (col == colstart){  //first col (left)
383                                        t[0] = INFINITY; //outside bounds, so inf energy
384                                        t[1] = distance.at<float>(row−1,col);
385                                        t[2] = distance.at<float>(row−1,col+1);
386                            }
387                            else if (col == colend){
388                                        t[0] = distance.at<float>(row−1,col−1);
389                                        t[1] = distance.at<float>(row−1,col);
390                                        t[2] = INFINITY;
391                            }
392                            else {
393                                        t[0] = distance.at<float>(row−1,col−1);
394                                        t[1] = distance.at<float>(row−1,col);
395                                        t[2] = distance.at<float>(row−1,col+1);
396
397                            }
398
399                    }
400                    /*
401                    for (int col = 0; col < 399; col++){
402                                        t[0] = rowMinusPoint[col−1];
403                                        t[1] = rowMinusPoint[col];
404                                        t[2] = rowMinusPoint[col+1];
405                    */
406
407                    //Find minimum of t
408                    auto result = std::min_element (t.begin(),t.end());
409                    float v= *result; // minimum value of the higher three pixels
410                    int i = (result−t.begin()); // index of min value of the higher three pixels −> parent pixel
411                    //rowPoint[col] = v + imagePoint[col];
412                    //predPoint[col] = col+i−1;
413                    distance.at<float>(row,col) = v + image.at<float>(row,col);//+ cv[1];
414                    // new energy of pixel (current value + minimum energy of top pixel)
415                    predecessor.at<ushort>(row,col) = col + i −1; // father pixel
416                    }
417
```

```cpp
418          }
419
420
421          //afterwards, distance pointer is at last row
422          PD.traversedDistance = rowPoint[goal.x];
423
424          //backtrack along predecessor matrix
425          //std::vector<cv::Point> path(image.rows);
426          PD.Path[image.rows−1] = goal;
427          ushort minEnergyInd = goal.x;
428
429          //cv::imshow("pred", predecessor∗255);
430          //cv::imshow("dist", distance);
431
432          /*
433          for (int row = (image.rows−2); row >= 0; row−−){
434                  predPoint= predecessor.ptr<ushort>(row+1);
435                  minEnergyInd = predPoint[minEnergyInd];
436                  PD.Path[row] = cv::Point(minEnergyInd, row); //col is angular value
437          }
438          */
439          for (int row = (image.rows−2); row >= 0; row−−){
440                  minEnergyInd = predecessor.at<ushort>(cv::Point(minEnergyInd,row+1)); //get idx of father pixel
441                  PD.Path[row] = cv::Point(minEnergyInd, row); //col is angular value
442          }
443
444
445
446  }
447
448  cv::Point PathsTouching(cv::vector<cv::Point> &path1, cv::vector<cv::Point> &path2){
449          cv::Point ReturnPoint = cv::Point(0,0);
450          //get first common element
451          for (int i =0; i < 720; i++){
452                  if (path1[i].x == path2[i].x){
453                          ReturnPoint = path1[i];
454                          break;
455                  }
456          }
457          return ReturnPoint;
458  }
459
460  void newShiftedPath (const cv::Mat& image, cv::Point CommonPoint, PathAndDistance &pdReturn){
461
462          //shift image
463          //PathAndDistance pdShift;
464          cv::Mat newImage(720,400,CV_32F);
465          cv::Mat top = cv::Mat(image, cv::Rect(0,0,image.cols,CommonPoint.y)).clone();
```

```cpp
466            cv::Mat bottom = cv::Mat(image, cv::Rect(0,CommonPoint.y,
467                                      image.cols,image.rows−CommonPoint.y)).clone();
468            cv::vconcat(bottom,top,newImage);
469
470
471            bruteForcePathFinder(newImage, cv::Point(CommonPoint.x,0),
472                    cv::Point(CommonPoint.x,newImage.rows−1), pdReturn);
473            //pdReturn.Path = pdShift.Path;
474
475            //pdReturn.traversedDistance = pdShift.traversedDistance;
476
477            //reshift back
478            cv::Mat mpts(pdReturn.Path);
479        cv::Mat xy = mpts.reshape(1);
480        cv::Mat x = xy.col(0);
481        cv::Mat y = xy.col(1);
482            std::rotate(y.begin<cv::Point::value_type>(), y.begin<cv::Point::value_type>()
483                    + CommonPoint.y, y.end<cv::Point::value_type>());
484            std::sort(pdReturn.Path.begin(), pdReturn.Path.end(),
485              [](const cv::Point &a, const cv::Point &b)
486              {
487                  return a.y < b.y; //resort
488              });
489            /*
490            for (int i = (pdShift.Path.size() − CommonPoint.y); i < pdShift.Path.size() ; i++){
491                    pdReturn.Path[i−(pdShift.Path.size() − CommonPoint.y)].x = pdShift.Path[i].x;
492                    pdReturn.Path[i−(pdShift.Path.size() − CommonPoint.y)].y
493                        = i−(pdShift.Path.size() − CommonPoint.y);
494            }
495
496            for (int i = 0; i < (pdShift.Path.size() − CommonPoint.y) ; i++){
497                    pdReturn.Path[i+ CommonPoint.y].x = pdShift.Path[i].x;
498                    pdReturn.Path[i+ CommonPoint.y].y = i+ CommonPoint.y;
499            }*/
500            //return pdReturn;
501    }
```

Bibliography

[1] A. Ashkin, "Acceleration and trapping of particles by radiation pressure," *Physical Review Letters*, vol. 24, pp. 156–159, 1970.

[2] A. Ashkin, J. M. Dziedzic, J. E. Bjorkholm, and S. Chu, "Observation of a single-beam gradient force optical trap for dielectric particles," *Optics Letters*, vol. 11, no. 5, pp. 288–290, 1986.

[3] A. Constable, J. Kim, J. Mervis, F. Zarinetchi, and M. Prentiss, "Demonstration of a fiber-optical light-force trap," *Optics Letters*, vol. 18, pp. 1867–1869, Nov 1993.

[4] J. Guck, R. Ananthakrishnan, T. Moon, C. Cunningham, and J. Käs, "Optical deformability of soft biological dielectrics," *Physical Review Letters*, vol. 84, pp. 5451–5454, 2000.

[5] J. Guck, R. Ananthakrishnan, H. Mahmood, T. Moon, C. Cunningham, and J. Käs, "The optical stretcher: A novel laser tool to micromanipulate cells," *Biophysical Journal*, vol. 81, pp. 767–784, 2001.

[6] J. Guck, R. Ananthakrishnan, C. Cunningham, and K. J, "Stretching biological cells with light," *Journal of Physics Condensed Matter*, vol. 14, p. 4843, 2002.

[7] B. Lincoln, S. Schinkinger, K. Travis, F. Wottawah, S. Ebert, F. Sauer, and J. Guck, "Reconfigurable microfluidic integration of a dual-beam laser trap with biomedical applications," *Biomedical Microdevices*, vol. 9, pp. 703–710, 2007.

[8] B. Lincoln, F. Wottawah, S. Schinkinger, S. Ebert, and J. Guck, "High-throughput rheological measurements with an optical stretcher," in *Cell Mechanics* (Y. Wang and D. E. Discher, eds.), vol. 83 of *Methods in Cell Biology*, pp. 397 – 423, Academic Press, 2007.

[9] J. Guck, S. Schinkinger, B. Lincoln, F. Wottawah, S. Ebert, M. Romeyke, D. Lenz, H. Erickson, R. Ananthakrishnan, D. Mitchell, J. Käs, S. Ulvick, and C. Bilby, "Optical deformability as an inherent cell marker for testing malignant transformation and metastatic competence," *Biophysical Journal*, vol. 88, p. 3689–3698, 2005.

[10] T. W. Remmerbach, F. Wottawah, J. Dietrich, B. Lincoln, C. Wittekind, and J. Guck, "Oral cancer diagnosis by mechanical phenotyping," *Cancer Research*, vol. 69, no. 5, pp. 1728–1732, 2009.

[11] J. Runge, T. Reichert, A. Fritsch, J. Käs, J. Bertolini, and T. Remmerbach, "Evaluation of single-cell biomechanics as potential marker for oral squamous cell carcinomas: a pilot study," *Oral Diseases*, vol. 20, no. 3, pp. e120–e127, 2014.

[12] L. Kastl, B. Budde, M. Isbach, C. Rommel, B. Kemper, and J. Schnekenburger, "Optomechanical properties of cancer cells revealed by light-induced deformation and quantitative phase microscopy," in *Proceedings of SPIE*, vol. 9529, pp. 952908–952908–6, 2015.

[13] F. Lautenschläger, S. Paschke, S. Schinkinger, A. Bruel, M. Beil, and J. Guck, "The regulatory role of cell mechanics for migration of differentiating myeloid cells," *Proceedings of the National Academy of Sciences*, vol. 106, no. 37, pp. 15696–15701, 2009.

[14] A. Ekpenyong, G. Whyte, K. Chalut, S. Pagliara, F. Lautenschläger, C. Fiddler, S. Paschke, U. Keyser, E. Chilvers, and J. Guck, "Viscolelastic properties of differentiating blood cells are fate- and function-dependent," *PLoS One*, vol. 7, p. e45237, 2012.

[15] J. M. Maloney, D. Nikova, F. Lautenschläger, E. Clarke, R. Langer, J. Guck, and K. J. Van Vliet, "Mesenchymal stem cell mechanics from the attached to the suspended state," *Biophysical Journal*, vol. 99, no. 8, pp. 2479–2487, 2010.

[16] K. Seltmann, A. W. Fritsch, J. A. Käs, and T. M. Magin, "Keratins significantly contribute to cell stiffness and impact invasive behavior," *Proceedings of the National Academy of Sciences*, vol. 110, no. 46, pp. 18507–18512, 2013.

[17] C. J. Chan, A. E. Ekpenyong, S. Golfier, W. Li, K. J. Chalut, O. Otto, J. Elgeti, J. Guck, and F. Lautenschläger, "Myosin ii activity softens cells in suspension," *Biophysical Journal*, vol. 108, pp. 1856–1869, 2015/09/07 2015.

[18] M. Gyger, D. Rose, R. Stange, T. Kießling, M. Zink, B. Fabry, and J. A. Käs, "Calcium imaging in the optical stretcher," *Optics Express*, vol. 19, pp. 19212–19222, Sep 2011.

[19] M. Gyger, R. Stange, T. Kießling, A. Fritsch, K. Kostelnik, A. Beck-Sickinger, M. Zink, and J. Käs, "Active contractions in single suspended epithelial cells," *European Biophysics Journal*, vol. 43, no. 1, pp. 11–23, 2014.

[20] M. E. Solmaz, R. Biswas, S. Sankhagowit, J. R. Thompson, C. A. Mejia, N. Malmstadt, and M. L. Povinelli, "Optical stretching of giant unilamellar vesicles with an integrated dual-beam optical trap," *Biomedical Optics Express*, vol. 3, pp. 2419–2427, Oct 2012.

[21] T. M. Pinon, A. R. Castelli, L. S. Hirst, and J. E. Sharping, "Fiber-optic trap-on-a-chip platform for probing low refractive index contrast biomaterials," *Applied Optics*, vol. 52, pp. 2340–2345, Apr 2013.

[22] U. Delabre, K. Feld, E. Crespo, G. Whyte, C. Sykes, U. Seifert, and J. Guck, "Deformation of phospholipid vesicles in an optical stretcher," *Soft Matter*, pp. 6075–6088, 2015.

[23] T. R. Kießling, R. Stange, J. A. Käs, and A. W. Fritsch, "Thermorheology of living cells—impact of temperature variations on cell mechanics," *New Journal of Physics*, vol. 15, no. 4, p. 045026, 2013.

[24] C. J. Chan, G. Whyte, L. Boyde, G. Salbreux, and J. Guck, "Impact of heating on passive and active biomechanics of suspended cells," *Interface Focus*, vol. 4, no. 2, p. 20130069, 2014.

[25] T. Yang, G. Nava, P. Minzioni, M. Veglione, F. Bragheri, F. D. Lelii, R. M. Vazquez, R. Osellame, and I. Cristiani, "Investigation of temperature effect on cell mechanics by optofluidic microchips," *Biomedical Optics Express*, vol. 6, no. 8, pp. 2991–2996, 2015.

[26] B. U. S. Schmidt, T. R. Kießling, E. Warmt, A. W. Fritsch, R. Stange, and J. A. Käs, "Complex thermorheology of living cells," *New Journal of Physics*, vol. 17, no. 7, p. 073010, 2015.

[27] F. Wetzel, S. Rönicke, K. Müller, M. Gyger, D. Rose, M. Zink, and J. Käs, "Single cell viability and impact of heating by laser absorption," *European Biophysics Journal*, vol. 40, no. 9, pp. 1109–1114, 2011.

[28] L. Boyde, K. J. Chalut, and J. Guck, "Interaction of gaussian beam with near-spherical particle: an analytic-numerical approach for assessing scattering and stresses," *Journal of the Optical A*, vol. 26, pp. 1814–1826, Aug 2009.

[29] L. Boyde, A. Ekpenyong, G. Whyte, and J. Guck, "Comparison of stresses on homogeneous spheroids in the optical stretcher computed with geometrical optics and generalized lorenz–mie theory," *Applied Optics*, vol. 51, pp. 7934–7944, 2012.

[30] L. Boyde, A. Ekpenyong, G. Whyte, and J. Guck, "Elastic theory for the deformation of a solid or layered spheroid under axisymmetric loading," *Acta Mechanica*, vol. 224, no. 4, pp. 819–839, 2013.

[31] A. E. Ekpenyong, C. L. Posey, J. L. Chaput, A. K. Burkart, M. M. Marquardt, T. J. Smith, and M. G. Nichols, "Determination of cell elasticity through hybrid ray optics and continuum mechanics modeling of cell deformation in the optical stretcher," *Applied Optics*, vol. 48, pp. 6344–6354, Nov 2009.

[32] R. Ananthakrishnan, J. Guck, F. Wottawah, S. Schinkinger, B. Lincoln, M. Romeyke, and J. Käs, "Modelling the structural response of an eukaryotic cell in the optical stretcher," *Current Science*, vol. 88, no. 09, 2005.

[33] R. Ananthakrishnan, J. Guck, F. Wottawah, S. Schinkinger, B. Lincoln, M. Romeyke, T. Moon, and J. Käs, "Quantifying the contribution of actin networks to the elastic strength of fibroblasts," *Journal of Theoretical Biology*, vol. 242, no. 2, pp. 502 – 516, 2006.

[34] F. Wottawah, S. Schinkinger, B. Lincoln, R. Ananthakrishnan, M. Romeyke, J. Guck, and J. Käs, "Optical rheology of biological cells," *Physical Review Letters*, vol. 94, p. 098103, Mar 2005.

[35] F. Wottawah, S. Schinkinger, B. Lincoln, S. Ebert, K. Müller, F. Sauer, K. Travis, and J. Guck, "Characterizing single suspended cells by optorheology," *Acta Biomaterialia*, vol. 1, no. 3, pp. 263 – 271, 2005.

[36] P. B. Bareil, Y. Sheng, and A. Chiou, "Local scattering stress distribution on surface of a spherical cell in optical stretcher," *Optics Express*, vol. 14, pp. 12503–12509, Dec 2006.

[37] P. B. Bareil, Y. Sheng, Y.-Q. Chen, and A. Chiou, "Calculation of spherical red blood cell deformation in a dual-beam optical stretcher," *Optics Express*, vol. 15, pp. 16029–16034, Nov 2007.

[38] S.-K. Teo, A. B. Goryachev, K. H. Parker, and K.-H. Chiam, "Cellular deformation and intracellular stress propagation during optical stretching," *Physical Review E*, vol. 81, p. 051924, May 2010.

[39] F. Gast, P. Dittrich, P. Schwille, M. Weigel, M. Mertig, J. Opitz, U. Queitsch, S. Diez, B. Lincoln, F. Wottawah, S. Schinkinger, J. Guck, J. Käs, J. Smolinski, K. Salchert, C. Werner, C. Duschl, M. Jäger, K. Uhlig, P. Geggier, and S. Howitz, "The microscopy cell (miccell), a versatile modular flowthrough system for cell biology, biomaterial research, and nanotechnology," *Microfluidics and Nanofluidics*, vol. 2, pp. 21–36, 2006.

[40] R. Osellame, S. Taccheo, M. Marangoni, R. Ramponi, P. Laporta, D. Polli, S. D. Silvestri, and G. Cerullo, "Femtosecond writing of active optical waveguides with astigmatically shaped beams," *Journal of the Optical Society of America B*, vol. 20, pp. 1559–1567, Jul 2003.

[41] N. Bellini, K. C. Vishnubhatla, F. Bragheri, L. Ferrara, P. Minzioni, R. Ramponi, I. Cristiani, and R. Osellame, "Femtosecond laser fabricated monolithic chip for optical trapping and stretching of single cells," *Optics Express*, vol. 18, pp. 4679–4688, Mar 2010.

[42] F. Bragheri, L. Ferrara, N. Bellini, K. C. Vishnubhatla, P. Minzioni, R. Ramponi, R. Osellame, and I. Cristiani, "Optofluidic chip for single cell trapping and stretching fabricated by a femtosecond laser," *Journal of Biophotonics*, vol. 3, no. 4, pp. 234–243, 2010.

[43] N. Bellini, F. Bragheri, I. Cristiani, J. Guck, R. Osellame, and G. Whyte, "Validation and perspectives of a femtosecond laser fabricated monolithic optical stretcher," *Biomedical Optics Express*, vol. 3, pp. 2658–2668, 2012.

[44] F. Bragheri, P. Minzioni, R. Martinez Vazquez, N. Bellini, P. Paie, C. Mondello, R. Ramponi, I. Cristiani, and R. Osellame, "Optofluidic integrated cell sorter fabricated by femtosecond lasers," *Lab on a Chip*, vol. 12, pp. 3779–3784, 2012.

[45] M. Khoury, C. Vannahme, K. Sørensen, A. Kristensen, and K. Berg-Sørensen, "Monolithic integration of duv-induced waveguides into plastic microfluidic chip for optical manipulation," *Microelectronic Engineering*, vol. 121, pp. 5 – 9, 2014. Nano Fabrication 2013.

[46] M. Khoury, R. Barnkob, L. Laub Busk, P. Tidemand-Lichtenberg, H. Bruus, and K. Berg-Sørensen, "Optical stretching on chip with acoustophoretic prefocusing," in *Proceedings of SPIE - The International Society for Optical Engineering*, pp. 84581E–84581E–11, 2012.

[47] G. Nava, F. Bragheri, T. Yang, P. Minzioni, R. Osellame, I. Cristiani, and K. Berg-Sørensen, "All-silica microfluidic optical stretcher with acoustophoretic prefocusing," *Microfluidics and Nanofluidics*, pp. 1–8, 2015.

[48] B. Alberts, A. Johnson, J. Lewis, D. Morgan, M. Raff, K. Roberts, and P. Walter, *Molecular Biology Of The Cell*. Garland Science, 2014.

[49] F. Gittes, B. Mickey, J. Nettleton, and J. Howard, "Flexural rigidity of microtubules and actin filaments measured from thermal fluctuations in shape.," *The Journal of Cell Biology*, vol. 120, no. 4, pp. 923–934, 1993.

[50] Z. Qin and M. J. Buehler, "Flaw tolerance of nuclear intermediate filament lamina under extreme mechanical deformation," *ACS Nano*, vol. 5, no. 4, pp. 3034–3042, 2011.

[51] A. Ashkin and J. M. Dziedzic, "Radiation pressure on a free liquid surface," *Physical Review Letters*, vol. 30, pp. 139–142, 1973.

[52] G. Roosen and C. Imbert, "Optical levitation by means of two horizontal laser beams: A theoretical and experimental study," *Physics Letters A*, vol. 59, no. 1, pp. 6 – 8, 1976.

[53] G. Roosen, "A theoretical and experimental study of the stable equilibrium positions of spheres levitated by two horizontal laser beams," *Optics Communications*, vol. 21, no. 1, pp. 189 – 194, 1977.

[54] B. Fabry, G. N. Maksym, J. P. Butler, M. Glogauer, D. Navajas, and J. J. Fredberg, "Scaling the microrheology of living cells," *Physical Review Letters*, vol. 87, p. 148102, 2001.

[55] P. Kollmannsberger and B. Fabry, "Linear and nonlinear rheology of living cells," *Annual Review of Materials Research*, vol. 41, no. 1, pp. 75–97, 2011.

[56] J. Maloney, E. Lehnhardt, A. Long, and K. VanVliet, "Mechanical fluidity of fully suspended biological cells," *Biophysical Journal*, vol. 105, no. 8, pp. 1767 – 1777, 2013.

[57] M. L. Rodriguez, P. J. McGarry, and N. J. Sniadecki, "Review on cell mechanics: Experimental and modeling approaches," *Applied Mechanics Reviews*, vol. 65, no. 6, pp. 060801–060801, 2013.

[58] G. H. Mealy, "A method for synthesizing sequential circuits," *Bell Systems Technical Journal*, vol. 34, no. 5, pp. 1045–1079, 1955.

[59] M. Schürmann, J. Scholze, P. Müller, C. J. Chan, A. E. Ekpenyong, K. J. Chalut, and J. Guck, "Chapter 9 - refractive index measurements of single, spherical cells using digital holographic microscopy," in *Biophysical Methods in Cell Biology* (E. K. Paluch, ed.), vol. 125 of *Methods in Cell Biology*, pp. 143 – 159, Academic Press, 2015.

[60] K. Demarsin, D. Vanderstraeten, T. Volodine, and D. Roose, "Detection of closed sharp edges in point clouds using normal estimation and graph theory," *Computer-Aided Design*, vol. 39, no. 4, pp. 276 – 283, 2007.

[61] T. Chan and L. Vese, "Active contours without edges," *Image Processing, IEEE Transactions on*, vol. 10, no. 2, pp. 266–277, 2001.

[62] R. Schnabel, R. Wessel, R. Wahl, and R. Klein, "Shape recognition in 3d point-clouds," in *The 16-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision'2008* (V. Skala, ed.), UNION Agency-Science Press, 2008.

[63] D. Mukherjee, N. Ray, and S. Acton, "Level set analysis for leukocyte detection and tracking," *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 562–572, 2004.

[64] R. Ali, M. Gooding, T. Szilágyi, B. Vojnovic, M. Christlieb, and M. Brady, "Automatic segmentation of adherent biological cell boundaries and nuclei from brightfield microscopy images," *Machine Vision and Applications*, vol. 23, no. 4, pp. 607–621, 2012.

[65] J. Su, Z. Zhu, A. Srivastava, and F. Huffer, "Detection of shapes in 2d point clouds generated from images," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, pp. 2640–2643, 2010.

[66] J. Serra, *Image Analysis and Mathematical Morphology*. Orlando, FL, USA: Academic Press, Inc., 1983.

[67] N. Otsu, "A threshold selection method from gray-level histograms," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 9, no. 1, pp. 62–66, 1979.

[68] J. Canny, "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[69] B. Cherkassky, A. Goldberg, and T. Radzik, "Shortest paths algorithms: Theory and experimental evaluation," *Mathematical Programming*, vol. 73, no. 2, pp. 129–174, 1996.

[70] G. Gallo and S. Pallottino, "Shortest path algorithms," *Annals of Operations Research*, vol. 13, no. 1, pp. 1–79, 1988.

[71] E. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[72] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *Systems Science and Cybernetics, IEEE Transactions on*, vol. 4, no. 2, pp. 100–107, 1968.

[73] P. Bamford and B. Lovell, "Unsupervised cell nucleus segmentation with active contours," *Signal Processing*, vol. 71, no. 2, pp. 203–213, 1998.

[74] C. Sun and S. Pallottino, "Circular shortest path in images," *Pattern Recognition*, vol. 36, no. 3, pp. 709 – 719, 2003.

[75] B. Appleton and C. Sun, "Circular shortest paths by branch and bound," *Pattern Recognition*, vol. 36, no. 11, pp. 2513 – 2520, 2003.

[76] B. Jähne, H. Scharr, and S. Körkel, "Principles of filter design," in *Handbook of Computer Vision and Applications* (B. Jähne, H. Haußecker, and P. Geißler, eds.), vol. 2, pp. 125–151, Academic Press, 1999.

[77] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[78] H. Kogelnik, "On the propagation of gaussian beams of light through lenslike media including those with a loss or gain variation," *Applied Optics*, vol. 4, no. 12, pp. 1562–1569, 1965.

[79] A. E. Ekpenyong, S. M. Man, S. Achouri, C. E. Bryant, J. Guck, and K. J. Chalut, "Bacterial infection of macrophages induces decrease in refractive index," *Journal of Biophotonics*, pp. 393–397, 2012.

[80] J. S. T. Gongora and A. Fratalocchi, "Optical force on diseased blood cells: Towards the optical sorting of biological matter," *Optics and Lasers in Engineering*, no. 76, pp. 40–44, 2015.

[81] Y. Park, M. Diez-Silva, G. Popescu, G. Lykotrafitis, W. Choi, M. S. Feld, and S. Suresh, "Refractive index maps and membrane dynamics of human red blood cells parasitized by plasmodium falciparum," *Proceedings of the National Academy of Sciences*, vol. 105, no. 37, pp. 13730–13735, 2008.

[82] Z. Wang, K. Tangella, A. Balla, and G. Popescu, "Tissue refractive index as marker of disease," *Journal of Biomedical Optics*, vol. 16, no. 11, pp. 116017–116017–7, 2011.

[83] K. J. Chalut, A. E. Ekpenyong, W. L. Clegg, I. C. Melhuish, and J. Guck, "Quantifying cellular differentiation by physical phenotype using digital holographic microscopy," *Integrative Biology*, vol. 4, pp. 280–284, 2012.

[84] B. Kemper, S. Kosmeier, P. Langehanenberg, G. von Bally, I. Bredebusch, W. Domschke, and J. Schnekenburger, "Integral refractive index determination of living suspension cells by multifocus digital holographic phase contrast microscopy," *Journal of Biomedical Optics*, vol. 12, no. 5, pp. 054009–054009–5, 2007.

[85] Á. B. Peña, B. Kemper, M. Woerdemann, A. Vollmer, S. Ketelhut, G. von Bally, and C. Denz, "Optical tweezers induced photodamage in living cells quantified with digital holographic phase microscopy," in *SPIE Photonics Europe*, pp. 84270A–84270A, International Society for Optics and Photonics, 2012.

[86] M. A. Oliver and R. Webster, "Kriging: a method of interpolation for geographical information systems," *International Journal of Geographical Information Systems*, vol. 4, no. 3, pp. 313–332, 1990.

[87] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, (Berkeley, Calif.), pp. 281–297, University of California Press, 1967.

[88] E. L. Lehmann, "Some principles of the theory of testing hypotheses," *The Annals of Mathematical Statistics*, vol. 21, no. 1, pp. 1–26, 1950.

[89] C. R. Rao, "Tests of significance in multivariate analysis," *Biometrika*, vol. 35, no. 1/2, pp. 58–79, 1948.

[90] T. Li, S. Zhu, and M. Ogihara, "Using discriminant analysis for multi-class classification: an experimental investigation," *Knowledge and Information Systems*, vol. 10, no. 4, pp. 453–472, 2006.

[91] H. R. Hulett, W. A. Bonner, J. Barrett, and L. A. Herzenberg, "Cell sorting: Automated separation of mammalian cells as a function of intracellular fluorescence," *Science*, vol. 166, no. 3906, pp. 747–749, 1969.

[92] S. Miltenyi, W. Müller, W. Weichel, and A. Radbruch, "High gradient magnetic cell separation with macs," *Cytometry*, vol. 11, no. 2, pp. 231–238, 1990.

[93] M. A. Nolte, G. Kraal, and R. E. Mebius, "Effects of fluorescent and nonfluorescent tracing methods on lymphocyte migration in vivo," *Cytometry Part A*, vol. 61A, no. 1, pp. 35–44, 2004.

[94] D. L. Coutu and T. Schroeder, "Probing cellular processes by long-term live imaging–historic problems and current solutions," *Journal of Cell Science*, vol. 126, no. 17, pp. 3805–3815, 2013.

[95] W. Moller, I. Nemoto, and J. Heyder, "Effect of magnetic bead agglomeration on cytomagnetometric measurements," *NanoBioscience, IEEE Transactions on*, vol. 2, no. 4, pp. 247–254, 2003.

[96] A. Tiwari, G. Punshon, A. Kidane, G. Hamilton, and A. Seifalian, "Magnetic beads (dynabead^{TM}) toxicity to endothelial cells at high bead concentration: Implication for tissue engineering of vascular prosthesis," *Cell Biology and Toxicology*, vol. 19, no. 5, pp. 265–272, 2003.

[97] J. P. Beech, S. H. Holm, K. Adolfsson, and J. O. Tegenfeldt, "Sorting cells by size, shape and deformability," *Lab on a Chip*, vol. 12, pp. 1048–1051, 2012.

[98] D. Holmes, G. Whyte, J. Bailey, N. Vergara-Irigaray, A. Ekpenyong, J. Guck, and T. Duke, "Separation of blood cells with differing deformability using deterministic lateral displacement," *Interface Focus*, vol. 4, no. 6, 2014.

[99] S. Zheng, R. Yung, Y.-C. Tai, and H. Kasdan, "Deterministic lateral displacement mems device for continuous blood cell separation," in *Micro Electro Mechanical Systems, 2005. MEMS 2005. 18th IEEE International Conference on*, pp. 851–854, 2005.

[100] S. Hur, N. Henderson-MacLennan, E. McCabe, and D. Di Carlo, "Deformability-based cell classification and enrichment using inertial microfluidics," *Lab on a Chip*, vol. 11, p. 912, 2011.

[101] J. Zhang, S. Yan, R. Sluyter, W. Li, G. Alici, and N.-T. Nguyen, "Inertial particle separation by differential equilibrium positions in a symmetrical serpentine micro-channel," *Scientific Reports*, vol. 4, p. 4527, 2014.

[102] B. K. Lin, S. M. McFaul, C. Jin, P. C. Black, and H. Ma, "Highly selective biomechanical separation of cancer cells from leukocytes using microfluidic ratchets and hydrodynamic concentrator," *Biomicrofluidics*, vol. 7, no. 3, p. 034114, 2013.

[103] W. Zhang, K. Kai, D. S. Choi, T. Iwamoto, Y. H. Nguyen, H. Wong, M. D. Landis, N. T. Ueno, J. Chang, and L. Qin, "Microfluidics separation reveals the stem-cell–like deformability of tumor-initiating cells," *Proceedings of the National Academy of Sciences*, vol. 109, no. 46, pp. 18707–18712, 2012.

[104] S. M. McFaul, B. K. Lin, and H. Ma, "Cell separation based on size and deformability using microfluidic funnel ratchets," *Lab On A Chip*, vol. 12, pp. 2369–2376, 2012.

[105] P. Preira, V. Grandne, J.-M. Forel, S. Gabriele, M. Camara, and O. Theodoly, "Passive circulating cell sorting by deformability using a microfluidic gradual filter," *Lab On A Chip*, vol. 13, pp. 161–170, 2013.

[106] S. C. Chapin, D. C. Pregibon, and P. S. Doyle, "High-throughput flow alignment of barcoded hydrogel microparticles," *Lab On A Chip*, vol. 9, pp. 3100–3109, 2009.

[107] A. E. Kamholz, B. H. Weigl, B. A. Finlayson, and P. Yager, "Quantitative analysis of molecular interaction in a microfluidic channel:? the t-sensor," *Analytical Chemistry*, vol. 71, no. 23, pp. 5340–5347, 1999.

[108] C. Faigle, F. Lautenschlager, G. Whyte, P. Homewood, E. Martin-Badosa, and J. Guck, "A monolithic glass chip for active single-cell sorting based on mechanical phenotyping," *Lab On A Chip*, vol. 15, pp. 1267–1275, 2015.

[109] E. Snitzer, H. Po, F. Hakimi, R. Tumminelli, and B. C. McCollum, "Erbium fiber laser amplifier at 1.55m with pump at 1.49m and yb sensitized er oscillator," in *Optical Fiber Communication*, Optical Society of America, 1988.

[110] B. J. Kirby, *Micro- and Nanoscale Fluid Mechanics: Transport in Microfluidic Devices*. Cambridge University Press, 2010.

[111] W. Federspiel and I. Valenti, "On laminar flow in microfabricated channels with partial semi-circular profiles," *Open Journal of Applied Sciences*, vol. 2, no. 1, pp. 28–34, 2012.

[112] A. E. Siegman, *Lasers*. University Science Books, 1986.

[113] A. A. Tovar and L. W. Casperson, "Generalized beam matrices: Gaussian beam propagation in misaligned complex optical systems," *Journal of the Optical Society of America A*, vol. 12, no. 7, pp. 1522–1533, 1995.

[114] A. A. Tovar and L. W. Casperson, "Generalized beam matrices. iii. application to diffraction analysis," *Journal of the Optical Society of America A*, vol. 13, no. 11, pp. 2239–2246, 1996.

[115] M. Schliwa, "Action of cytochalasin d on cytoskeletal networks.," *The Journal of Cell Biology*, vol. 92, no. 1, pp. 79–91, 1982.

[116] W. Strober, *Trypan Blue Exclusion Test of Cell Viability*, ch. Appendix 3, pp. A3.B1 – A3.B2. John Wiley & Sons, Inc., 2001.

[117] J. Helenius, C.-P. Heisenberg, H. E. Gaub, and D. J. Muller, "Single-cell force spectroscopy," *Journal of Cell Science*, vol. 121, no. 11, pp. 1785–1791, 2008.

[118] A. S. Stender, K. Marchuk, C. Liu, S. Sander, M. W. Meyer, E. A. Smith, B. Neupane, G. Wang, J. Li, J.-X. Cheng, B. Huang, and N. Fang, "Single cell optical imaging and spectroscopy," *Chemical Reviews*, vol. 113, no. 4, pp. 2469–2527, 2013.

[119] M. J. Mitchell and M. R. King, "Fluid shear stress sensitizes cancer cells to receptor-mediated apoptosis via trimeric death receptors," *New Journal of Physics*, vol. 15, no. 1, p. 015008, 2013.

[120] N. Shiragami and H. Unno, "Effect of shear stress on activity of cellular enzyme in animal cell," *Bioprocess Engineering*, vol. 10, no. 1, pp. 43–45, 1994.

[121] A. Ludwig, G. Kretzmer, and K. Schügerl, "Determination of a "critical shear stress level" applied to adherent mammalian cells," *Enzyme and Microbial Technology*, vol. 14, no. 3, pp. 209 – 213, 1992.

[122] J. M. Barnes, J. T. Nauseef, and M. D. Henry, "Resistance to fluid shear stress is a conserved biophysical property of malignant cells," *PLoS ONE*, vol. 7, no. 12, p. e50973, 2012.

[123] O. Otto, P. Rosendahl, A. Mietke, S. Golfier, C. Herold, D. Klaue, S. Girardo, S. Pagliara, A. Ekpenyong, A. Jacobi, M. Wobus, N. Topfner, U. F. Keyser, J. Mansfeld, E. Fischer-Friedrich, and J. Guck, "Real-time deformability cytometry: on-the-fly cell mechanical phenotyping," *Nature Methods*, vol. 12, no. 3, pp. 199–202, 2015.

[124] R. Reneman and A. Hoeks, "Wall shear stress as measured in vivo: consequences for the design of the arterial system," *Medical & Biological Engineering & Computing*, vol. 46, no. 5, pp. 499–507, 2008.

[125] A. M. Malek, S. L. Alper, and S. Izumo, "Hemodynamic shear stress and its role in atherosclerosis," *Journal of the American Medical Association*, vol. 282, no. 21, pp. 2035–2042, 1999.

[126] H. Lorenz, M. Laudon, and P. Renaud, "Mechanical characterization of a new high-aspect-ratio near uv-photoresist," *Microelectronic Engineering*, vol. 41–42, pp. 371 – 374, 1998.

[127] R. Martinez-Duarte and M. J. Madou, *SU-8 Photolithography and Its Impact on Microfluidics*, ch. 8, pp. 231–268. CRC Press, 2011.

[128] R. Kitamura, L. Pilon, and M. Jonasz, "Optical constants of silica glass from extreme ultraviolet to far infrared at near room temperature," *Applied Optics*, vol. 46, no. 33, pp. 8118–8133, 2007.

[129] W. She, J. Yu, and R. Feng, "Observation of a push force on the end face of a nanometer silica filament exerted by outgoing light," *Physical Review Letters*, vol. 101, p. 243601, 2008.

[130] D. Wu, G. Ma, F. Niu, and D. Guo, "Temperature gradient mechanism on laser bending of borosilicate glass sheet," *Journal of manufacturing science and engineering*, vol. 132, no. 1, p. 011013, 2010.

[131] H. Exner and U. Loschner, "Contactless laser bending of silicon microstructures," in *Smart Sensors, Actuators, and MEMS*, vol. 5116, pp. 383–392, 2003.

[132] T. Kießling, M. Herrera, K. Nnetu, E. Balzer, M. Girvan, A. Fritsch, S. Martin, J. Käs, and W. Losert, "Analysis of multiple physical parameters for mechanical phenotyping of living cells," *European Biophysics Journal*, vol. 42, no. 5, pp. 383–394, 2013.

[133] M. Calin, M. Calin, and C. Munteanu, "Determination of the complex refractive index of cell cultures by reflectance spectrometry," *The European Physical Journal Plus*, vol. 129, no. 6, 2014.

[134] B. Rappaz, E. Cano, T. Colomb, J. Kühn, C. Depeursinge, V. Simanis, P. J. Magistretti, and P. Marquet, "Noninvasive characterization of the fission yeast cell cycle by monitoring dry mass with digital holographic microscopy," *Journal of Biomedical Optics*, vol. 14, no. 3, pp. 034049–034049–5, 2009.

[135] K. H. Elstein and R. M. Zucker, "Comparison of cellular and nuclear flow cytometric techniques for discriminating apoptotic subpopulations," *Experimental Cell Research*, vol. 211, no. 2, pp. 322 – 331, 1994.

[136] B. Kemper, D. Carl, J. Schnekenburger, I. Bredebusch, M. Schäfer, W. Domschke, and G. von Bally, "Investigation of living pancreas tumor cells by digital holographic microscopy," *Journal of Biomedical Optics*, vol. 11, no. 3, pp. 034005–034005–8, 2006.

[137] J. Klossa, B. Wattelier, T. Happillon, D. Toubas, L. de Laulanie, V. Untereiner, P. Bon, and M. Manfait, "Quantitative phase imaging and raman micro-spectroscopy applied to malaria," *Diagnostic Pathology*, 2013.

[138] X. Liang, A. Liu, C. Lim, T. Ayi, and P. Yap, "Determining refractive index of single living cell using an integrated microchip," *Sensors and Actuators A: Physical*, vol. 133, no. 2, pp. 349 – 354, 2007.

[139] C. D. Bortner and J. A. Cidlowski, "A necessary role for cell shrinkage in apoptosis," *Biochemical Pharmacology*, vol. 56, no. 12, pp. 1549 – 1559, 1998.

[140] R. M. Zucker, K. Whittington, and B. J. Price, "Differentiation of hl-60 cells: Cell volume and cell cycle changes," *Cytometry*, vol. 3, no. 6, pp. 414–418, 1983.

[141] Y. Gazitt, A. D. Deitch, P. A. Marks, and R. A. Rifkind, "Cell volume changes in relation to the cell cycle of differentiating erythroleukemic cells," *Experimental Cell Research*, vol. 117, no. 2, pp. 413 – 420, 1978.

[142] X. Yang and T. Xu, "Molecular mechanism of size control in development and human diseases," *Cell Research*, vol. 21, no. 5, pp. 715–729, 2011.

[143] M. Balland, N. Desprat, D. Icard, S. Féréol, A. Asnacios, J. Browaeys, S. Hénon, and F. m. c. Gallet, "Power laws in microrheology experiments on living cells: Comparative analysis and modeling," *Physical Reviews E*, vol. 74, p. 021911, 2006.

[144] R. K. Bista, S. Uttam, P. Wang, K. Staton, S. Choi, C. J. Bakkenist, D. J. Hartman, R. E. Brand, and Y. Liu, "Quantification of nanoscale nuclear refractive index changes during the cell cycle," *Journal of Biomedical Optics*, vol. 16, no. 7, pp. 070503–070503–3, 2011.

[145] H. Ding, J. Q. Lu, R. S. Brock, L. Burke, D. A. Weidner, T. J. McConnell, and X.-H. Hu, "Determination of intracellular distributions of refractive index of b-cells and hl60 cells at 442, 633 and 850nm," in *Frontiers in Optics*, Optical Society of America, 2006.

[146] J.-Y. Tinevez, U. Schulze, G. Salbreux, J. Roensch, J.-F. Joanny, and E. Paluch, "Role of cortical tension in bleb growth," *Proceedings of the National Academy of Sciences*, vol. 106, no. 44, pp. 18581–18586, 2009.

[147] O. Otto, P. Rosendahl, A. Mietke, S. Golfier, A. Jacobi, N. Töpfner, C. Herold, D. Klaue, E. Fischer-Friedrich, and J. Guck, "Real-time deformability cytometry: High-throughput mechanical phenotyping for changes in cell function," *Biophysical Journal*, vol. 108, no. 2, p. 140, 2015.

[148] W. Huang, W. Choi, Y. Chen, Q. Zhang, H. Deng, W. He, and Y. Shi, "A proposed role for glutamine in cancer cell growth through acid resistance," *Cell Research*, vol. 23, pp. 724–727, May 2013.

[149] O. Warburg, "On the origin of cancer cells," *Science*, vol. 123, no. 3191, pp. 309–314, 1956.

[150] S. Ebert, K. Travis, B. Lincoln, and J. Guck, "Fluorescence ratio thermometry in a microfluidic dual-beam laser trap," *Optics E*, vol. 15, pp. 15493–25499, 2007.

[151] A. Tikhonov and A. Samarski, *Equations of Mathematical Physics*. Dover Books on Physics, Dover Publications, 1963.

[152] J. S. Dudani, D. R. Gossett, H. T. K. Tse, and D. Di Carlo, "Pinched-flow hydrodynamic stretching of single-cells," *Lab on a Chip*, vol. 13, pp. 3728–3734, 2013.

[153] D. R. Gossett, H. T. K. Tse, S. A. Lee, Y. Ying, A. G. Lindgren, O. O. Yang, J. Rao, A. T. Clark, and D. Di Carlo, "Hydrodynamic stretching of single cells for large population mechanical phenotyping," *Proceedings of the National Academy of Sciences*, vol. 109, no. 20, pp. 7630–7635, 2012.

[154] J. Brugués, V. Nuzzo, E. Mazur, and D. Needleman, "Nucleation and transport organize microtubules in metaphase spindles," *Cell*, vol. 149, no. 3, pp. 554–564, 2012.

[155] A. A. Hyman, C. A. Weber, and F. Jülicher, "Liquid-liquid phase separation in biology," *Annual Review of Cell and Developmental Biology*, vol. 30, no. 1, pp. 39–58, 2014.

[156] M. K. Kreysing, T. Kießling, A. Fritsch, C. Dietrich, J. R. Guck, and J. A. Käs, "The optical cell rotator," *Optics Express*, vol. 16, no. 21, pp. 16984–16992, 2008.

[157] C. Iliescu, H. Taylor, M. Avram, J. Miao, and S. Franssila, "A practical guide for the fabrication of microfluidic devices using glass and silicon," *Biomicrofluidics*, vol. 6, no. 1, p. 016505, 2012.

[158] T. Ichiki, Y. Sugiyama, R. Taura, T. Koidesawa, and Y. Horiike, "Plasma applications for biochip technology," *Thin Solid Films*, vol. 435, no. 1–2, pp. 62 – 68, 2003.

[159] C. Steinbruchel, H. W. Lehmann, and K. Frick, "Mechanism of dry etching of silicon dioxide: a case of direct reactive ion etching," in *Symposium F – Plasma Synthesis and Etching of Electronic Materials*, vol. 38 of *MRS Proceedings*, 1984.

[160] D. C. S. Bien, P. V. Rainey, S. J. N. Mitchell, and H. S. Gamble, "Characterization of masking materials for deep glass micromachining," *Journal of Micromechanics and Microengineering*, vol. 13, no. 4, p. 557, 2003.

[161] C.-H. Lin, G.-B. Lee, Y.-H. Lin, and G.-L. Chang, "A fast prototyping process for fabrication of microfluidic systems on soda-lime glass," *Journal of Micromechanics and Microengineering*, vol. 11, no. 6, p. 726, 2001.

[162] J. B. Lasky, "Wafer bonding for silicononinsulator technologies," *Applied Physics Letters*, vol. 48, no. 1, pp. 78–80, 1986.

[163] H. Lorenz, M. Despont, N. Fahrni, N. LaBianca, P. Renaud, and P. Vettiger, "Su-8: a low-cost negative resist for mems," *Journal of Micromechanics and Microengineering*, vol. 7, no. 3, p. 121, 1997.

[164] B. Kemper, A. Vollmer, C. E. Rommel, J. Schnekenburger, and G. v. Bally, "Simplified approach for quantitative digital holographic phase contrast imaging of living cells," *Journal of Biomedical Optics*, vol. 16, no. 2, pp. 026014–026014–4, 2011.

[165] J. Schwiegerling, "History of the shack hartmann wavefront sensor and its impact in ophthalmic optics," in *Proceedings of SPIE*, vol. 9186, pp. 91860U–91860U–8, 2014.

[166] J.-C. Chanteloup, "Multiple-wave lateral shearing interferometry for wave-front sensing," *Applied Optics*, vol. 44, no. 9, pp. 1559–1571, 2005.

[167] S. Velghe, J. Primot, N. Guérineau, M. Cohen, and B. Wattellier, "Wave-front reconstruction from multidirectional phase derivatives generated by multilateral shearing interferometers," *Optics Letters*, vol. 30, no. 3, pp. 245–247, 2005.

[168] P. Bon, G. Maucort, B. Wattellier, and S. Monneret, "Quadriwave lateral shearing interferometry for quantitative phase microscopy of living cells," *Optics Express*, vol. 17, no. 15, pp. 13080–13094, 2009.

[169] G. Bradski, "Open cv," *Dr. Dobb's Journal of Software Tools*, 2000.

[170] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

List of Figures

Acknowledgments

There are many people I wish to thank for their support during my PhD research.

First and foremost, I would like to thank my advisor, Jochen Guck, for his generous scientific, moral and financial support during this period and for keeping the pressure up.

Thanks as well go to Prof. Dr. Roberto Osellame of the Department of Physics, Politecnico di Milano, Italy and to Prof. Dr. Thomas Bley of the Institute of Bioprocess Engineering, Technische Universität Dresden, for their valuable time and effort spent reading this dissertation.

I would also like to thank the Dresdner Optical Stretcher group: Saeed Ahmed, Joseph Chan, Geo Cojoc and Andrew Ekpenyong, as well as the Optics group: Paul Müller and Mirjam Schürmann, for their help and scientific discussions. A thanks for the fun times also goes to the rest of the Guck group. Furthermore, I am grateful to my collaborators from Roberto Osellame's group in Milan: Francesca Bragheri and Petra Paié.

This thesis would not be the literary gem it is without my parent's patient proofreading and comments. I am very grateful for your support.

Finally, my heartfelt thanks goes to my favorite Dresdner: my colleague, friend and girlfriend Janine, without whom Mensa, hiking and the winter seminars would have been only half the fun. I am looking forward to many more a vacation and dinner with you!

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Weitere Personen waren an der geistigen Herstellung der vorliegenden Arbeit nicht beteiligt. Insbesondere habe ich nicht die Hilfe eines kommerziellen Promotionsberaters in Anspruch genommen. Dritte haben von mir keine geldwerten Leistungen für Arbeiten erhalten, die in Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen.

Die Arbeit wurde bisher weder im Inland noch im Ausland in gleicher oder ähnlicher Form einer anderen Prüfungsbehörde vorgelegt und ist auch noch nicht veröffentlicht worden.

Diese Promotionsordnung wird anerkannt.

I hereby declare that the present work has been produced without undue help of third parties and without use of any other than the indicated resources. The thoughts stemming directly or indirectly from external sources have been indicated as such.

Additional persons were not involved in the intellectual production of the present work. In particular, I did not enlist the help of a commercial consultant. Third parties have not received monetary benefits from me connected with the content of the present thesis.

This work has not been submitted to another examination board domestically and abroad in equal or similar form and has not been published yet.

These doctorate regulations are respected.

Dresden, den 4.11.2015

Christoph Faigle