LUDWIG-MAXIMILIANS-UNIVERSITÄT
TECHNISCHE UNIVERSITÄT MÜNCHEN

Helmholtz Zentrum München
Institut für Bioinformatik und Systembiologie

Diplomarbeit
in Bioinformatik

Computational prediction of hematopoietic cell fates using single cell time lapse imaging

Felix Buggenthin

Aufgabensteller:	Prof. Dr. Dr. Fabian Theis
Betreuer:	Michael Schwarzfischer, Dr. Carsten Marr
Abgabedatum:	02.08.2011

Ich versichere, dass ich diese Diplomarbeit selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

02.08.2011 _____

Felix Buggenthin

Abstract

Stem cells are able to regenerate and to give rise to specialized cell types. Due to these unique properties, stem cells represent a huge opportunity in the treatment of severe diseases such as dementia or leukemia. The hematopoietic stem cells residing in the bone marrow are the origin of blood regeneration and have been studied for many years. However, the processes driving hematopoiesis on cellular as well as molecular level are not fully understood. A major question to be elucidated is, at which generation after entering differentiation a hematopoietic progenitor cell is committing to the erytroid or myeloid lineage and which cellular factors are involved in this decision. Continuous imaging of living hematopoietic stem cells and all their progeny *in vitro* allows detailed insights into differentiation. The vast amount of data and information generated in these experiments render it necessary to apply computational methods for quantification and analysis.

In this thesis, we present a method capable of predicting a hematopoietic stem cell's decision to commit to the erytroid or myeloid lineage. Based on experiments from the Institute of Stem Cell Research at the Helmholtz Zentrum Munich, an automated image processing pipeline is established. The pipeline analyses 14 morphological properties of hematopoietic stem cells and all their progeny in brightfield images. In addition, fluorescence intensities of eYFP-labeled PU.1 molecules, a transcription factor that is thought to play a key role in lineage decision, are quantified. All features, intensities and annotations per cell are then utilized to train a random forest classifier that is able to predict the fate of unlabeled cells. Evaluation by ten-fold cross-validation results in a macro-averaged f1-measure of 0.83. Based on randomly drawn differentiation trees it is shown, that fate prediction is applicable for formerly unknown trees. The methods developed in this thesis can be adapted to other cell types, which could allow detailed analysis of stem cell behavior over their entire lifetime.

Zusammenfassung

Stammzellen besitzen die Fähigkeit zur Reproduktion sowie zur Differenzierung in spezialisierte Zelltypen. Aufgrund dieser Eigenschaften gelten sie als medizinische Hoffnungsträger in der Therapie vieler Krankheiten, wie zum Beispiel Demenzerkrankungen oder Leukämien. Die im Knochenmark reifenden hämatopoetischen Stammzellen sind der Ursprung der Blutneubildung und werden bereits seit vielen Jahren untersucht. Obwohl die Hämatopoese das am besten verstandene Stammzellsystem ist, sind viele Prozesse auf zellulärer und molekularer Ebene weiterhin unklar. So ist bis heute nicht bekannt, in welcher Generation nach Eintritt in die Differenzierung hematopoetische Vorläuferzellen die Entscheidung treffen, erythroide oder myeloische Vorläuferzellen zu erzeugen und welche zellulären Programme diese Entscheidung bewirken. Seit einigen Jahren ermöglicht die Technik der kontinuierlichen Einzelzellmikroskopie die Beobachtung der *in vitro* Differenzierung von Stammzellen bis zur adulten Blutzelle. Die enorme Menge an Bilddaten, die in diesen Experimenten generiert werden, sowie die Fülle an enthaltenen Informationen legen den Einsatz computergestützter Methoden zur Quantifizierung und Analyse nahe.

In dieser Diplomarbeit wird eine Methode vorgestellt, welche die Entscheidung einer hämatopoetischen Stammzelle zur erytroiden oder myeloischen Linie vorhersagt. Anhand experimenteller Daten des Instituts für Stammzellforschung des Helmholtz Zentrums München wird eine automatisierte Pipeline zur Bildprozessierung entwickelt. Die Pipeline analysiert 14 morphologische Eigenschaften der Stammzellen und all ihrer Nachkommen in Durchlichtbildern. Zusätzlich wird die Fluoreszentintensität des mit eYFP markierten Transkriptionsfaktors PU.1 quantifiziert, einem Schlüsselmolekül in der Differenzierungsentscheidung hematopoetischer Vorläuferzellen. Die aus sämtlichen Analysen resultierenden Eigenschaften, Intensitäten und Annotationen pro Zelle werden verwendet, um mit Hilfe eines random forest Klassifizierers unterschiedlich entschiedene Zellen zu erkennen. Das macro-averaged f1-measure bei zehnfacher Kreuzvalidierung beträgt hierbei 0.83. Anhand zufällig gewählter Differenzierungsbäume wird demonstriert, dass die Entscheidungsvorhersage auch für gänzlich unbekannte Bäume funktioniert. Die hier entwickelten Methoden können an weitere Zelltypen angepasst werden und ermöglichen so eine detailierte Analyse des Verhaltens von Stammzellen über die gesamte Lebensspanne.

Acknowledgments

First of all I would like to thank Michael Schwarzfischer for great supervision and assistance during both practical and writing period of this thesis.

Another huge thank you goes to Dr. Carsten Marr, my second supervisor, for his help with the cell movement analyses and support during the writing period.

Furthermore, many thanks to our biological collaborators, Dr. Timm Schroeder and Philipp Hoppe from the Institute of Stem Cell Research for the experiments and valuable feedback.

Next, I would like to thank Ivan Kondofersky for the help in statistical questions and the whole CMB group for the nice atmosphere.

Special thanks to Stephanie May and Sebastian Pölsterl who both supported me very much during the writing period, as well as my family who greatly supported me throughout the last six years.

Finally and most importantly, I would like to thank Prof. Dr. Dr. Fabian Theis for giving me the opportunity to work on this interesting project in his research group. Thanks as well for the supervision especially in the machine learning part of my thesis.

Contents

List of Figures

1. Introduction

1.1. The importance of stem cell research

Over the last decades stem cell research has been in the focus of public attention. Starting with the cloning of Dolly the sheep and followed by several promising works in the field of regenerative medicine [1], stem cells represent an important possibility in the therapy of many life-threatening diseases such as cancer, leukemia, cardiovascular diseases or dementia [2].

Stem cells are defined as biological cells that have the capacity to self-renew as well as the ability to generate differentiated cells [3]. More explicitly, stem cells can divide into daughter cells identical to their mother (self-renewal) as well as produce progeny with more restricted potential, which are eventually ending up as mature cells of a certain tissue or organ (differentiated cells). Stem cells can be assigned to classes by the amount of different cell types they are able to produce. These classes are called totipotent, pluripotent, multipotent, oligopotent and unipotent. Totipotent cells are capable of differentiating into all cell types and can be found in fertilized eggs. Blastomeres from a five-day-old human embryo can only give rise to cells that are part of the three germ layers, that is endoderm, mesoderm or ectoderm and are thus labeled pluripotent. As development progresses, individual cells become multipotent, i.e. they can give rise to all lineages of a tissue or organ. The hierarchy is continued with oligopotent cells that are able to differentiate only to certain lineages in a specific tissue or organ and closes with mature cells which are restricted to a single lineage, called unipotent. Table 1.1 summarizes the different levels of stem cell potency [4].

All eukaryote cells have progenitors that are at some point of their existence able to divide and differentiate, whereas many mature cells have lost these capabilities [5]. Most of the human organs show active cell division during embryogenesis, but are not able to regenerate in adult state. Examples for this are spinal cord, heart, kidneys, muscles and the brain [6]. This loss of function is needed in order to maintain the structure of a cell population. Mature cells arranged in a large network to form an organ or tissue need to show predictable behavior to ensure a certain size and shape of the assembled organ, a complex process that would be hindered if cells were able to divide.

However, some tissues and organs, such as skin, liver and bone marrow maintain a pool of multipotent stem cells that are able to produce new mature cells over an organisms whole lifetime, thus allowing replacement of damaged or destroyed cells [5]. An example for these processes is the constant regeneration of cells in the blood system (hematopoiesis).

Designation	Differentiation potential implied by designation	Examples of stem/progenitors with these properties
Totipotent	All embryonic and extraembryonic tissues (i.e. yolk sac)	Zygote
Pluripotent	All embryonic tissues	ICM, ESC
Multipotent	All lineages of a tissue/organ	HSC, NSC
Oligopotent	Several but not all lineages of a tissue/organ	CMP, CLP
Unipotent	Single lineage of a tissue/organ	Monocyte

Table 1.1. – Nomenclature defining differentiation potential of stem cell types. CLP, common lymphoid progenitor; CMP, common myeloid progenitor; ESC, embryonic stem cell; HSC, hematopoietic stem cell; ICM, inner cell mass; MacP, macrophage progenitor; NSC, neural stem cell. Table was taken from Seita and Weissman [4].

Red blood cells, for instance, have a lifespan of around 120 days after which they undergo apoptosis [7]. Other mature blood cells have even shorter lifetimes. Thus, an organism produces fresh blood cells constantly, a process which originates in blood stem cells residing in the bone marrow that maintain replenishment of the hematopoietic cell pool [8]. Due to extensive research of human blood and several well established model organisms, hematopoiesis is the best understood adult stem cell system [9–12]. It is thus not unexpected that the first stem cell therapies were applied to human patients suffering leukemia, a cancer type causing abnormal development of mature leukocytes (white blood cells). Stem cells programmed to differentiate into healthy leukocytes were used to replace cancerous cells in the patients body, a technique that was a lot more effective than transplantation of mature blood cells due to their short lifetimes [13]. Furthermore, hematopoietic stem cells could possess the abilities to effectively treat cancers like chronic myelogenous leukemia (CML) or Hodgkin's disease, inherited disorders like anemias, as well as cardiovascular diseases. First advances were already achieved for some of these disorders [14].

1.2. Current understanding of the hematopoietic system

The hematopoietic system is very complex and despite a long history of research and much progress in the field, the processes driving differentiation are not fully understood. Thus, a few alternative models explaining hematopoiesis exist, which are actively discussed. Current results suggest that development of blood in vertebrates is comprised of two phases. First, embryogenic hematopoiesis generates transitory hematopoietic cell types, for example primitive erythrocytes and some myeloid cells. After maturation the adult hematopoiesis emerges, a hierarchically structured system that is eventually giving rise to all blood lineages of the adult organism [12].

Hematopoietic stem cells (HSCs) are the root of this hierarchy and the only hematopoietic cell type that is capable of both multipotency and self-renewal. Adult hematopoiesis primarily takes place in the bone marrow of pelvis, cranium, vertebrae, and sternum, as well as femur and tibia, where stem cells are residing in a microenvironment (the so-called niche), that is thought to maintain the amount of stem cells and to trigger lineage commitment [10]. Current believe is that HSCs leave their niche and dissolve in the blood stream only if commited to a differentiation lineage. However, one in 100,000 to 300,000 thousand cells in the blood is thought to be a HSC [14].

Since morphological appearance in size and shape of hematopoietic stem cells does not differ compared to ordinary white blood cells, it is very difficult to determine the amount of real stem cells in the niche. Purifying stem cells in order to cultivate them is quite challenging, as 10,000 to 15,000 bone marrow cells are thought to include one HSC [14]. Even with modern purification methods such as sorting by flow cytometry (see 3.1 for details), the purification rate is only about 50% [15].

Differentiation of a HSC to mature hematopoietic cells arises through several subsequent cell types [4]. First, one or both daughters of a HSC lose their ability to self-renew but retain multipotency, becoming multipotent progenitors (MPPs). MPPs can give rise to two oligopotent cell types, the common myeloid progenitor (CMP) and the common lymphoid progenitor (CLP). CMPs are able to differentiate into oligopotent myeloid/erythrocyte progenitors (MEPs) and granulocyte/macrophage progenitors (GMP). CLPs in turn are able to differentiate into all lymphocytes, namely dendritic cells (DCs), B-cells, T-cells, natural killer cells (NK-cells) and all their unipotent progenitors, respectively. MEPs are predecessors of red blood cells and megakaryocytes, whereas daughters of GMPs are able to differentiate into the three types of granulocytes as well as macrophages [4; 16; 17]. Figure 1.1 schematizes the most recent model of hematopoiesis.

Functions of mature cell types in the organism

Red blood cells (erythrocytes) are an organism's oxygen carriers and the most common type of blood cells, accounting for approximately one quarter of all human cells. Erythrocytes are concave shaped disks that lack most organelles and nucleus. Instead, they are rich in hemoglobin, a molecule that is able to bind oxygen in an efficient way. The lifetime of red blood cells is around 120 days [7]. Diseases that involve erythrocytes are for example anemias, the low oxygen transport capacity of the blood due to low red cell count or some abnormality of the red blood cells or the hemoglobin.

Huge bone marrow cells that do not undergo mitosis, called Megakaryocytes, dissolve into small cell fragments, the platelets (thrombocytes). Thrombocytes are responsible for formation of blood clots that are able to temporarily stop bleeding (hemostasis). In addition, cell signaling of thrombocytes promotes the invasion of fibroblasts to the wounded area in order to repair damaged blood vessels. Overproduction of platelets can cause thrombosis, which often results in stroke or myocardial infarction [14].

The remaining six cell types that are listed on the terminal level in Figure 1.1 are part of the immune system, whose function is the identification and destruction of foreign particles, such as bacterial cells or viruses.

Granulocytes are categorized in three subclasses, namely neutrophils, eosinophils and basophils. Neutrophils are phagocytes engulfing external particles, bacteria or dead cells. Constituting 50% to 60% of circulating leukocytes (white blood cells), they are the most abundant type of phagocyte. The average liftetime of these cells amounts roughly 6 hours. Eosenophils are able to release cathepsin, a toxic basic protein that is able to kill parasites attacking the organism. Basophils contain histamine and heparine, both molecules are crucial parts of immune response [8].

Another type of phagocytes are macrophages. These cells are important in removal of necrotic tissue and generally in immune response. Macrophages have a comparatively long lifetime of up to seven months.

Dendritic cells are antigen presenting cells that are crucial to the presentation of peptides and proteins to T and B lymphocytes. They have a lifetime of a few days.

T and B cells are representing the adaptive immune system. They produce antibodies to mark infected cells and expose them for destruction by macrophages or other phagocytes.

NK-cells are a special form of leukocytes, as they need no activation and can destroy infected cells without marking of antibodies.

Identification of cell types through antibody markers

To date, different cell populations of the hematopoietic system are discriminated by surface markers that are unique to each cell type [14]. HSCs are currently identified by the CD150 surface molecule, which are also expressed in progenitors commited to the erythroid lineage

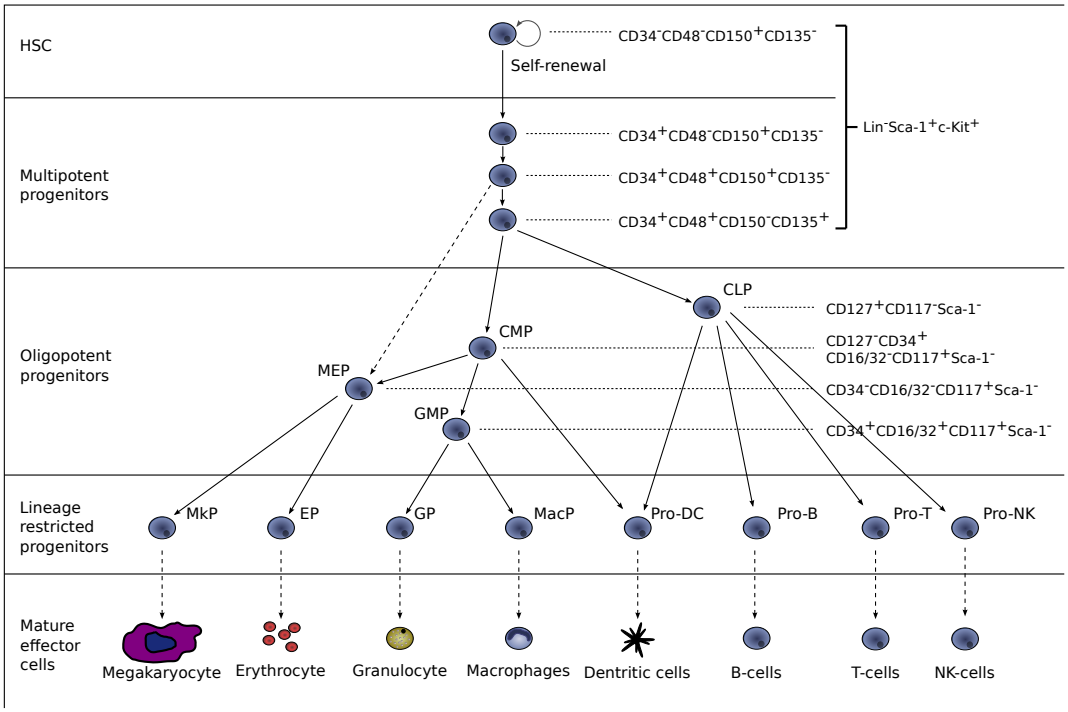


Figure 1.1. – Current understanding of the hematopoietic hierarchy and surface markers that allow discrimination of cell types. Hematopoietic stem cells (HSCs) are the only hematopoietic cells that are able to self-renew and in addition give rise to all hematopoietic cell types. HSCs differentiate first into multipotent progenitors (MPPs), i.e. cells that have lost the ability to self-renew but retain multipotency. MPPs are capable of giving rise to common myeloid progenitors (CMPs) and common lymphoid progenitors (CLPs), although recent evidence suggests that MPPs are also able to directly differentiate into megakaryocyte/erythrocyte progenitors (MEPs, dashed line) Seita and Weissman [4]. CMPs are then able to produce MEPs and granulocyte/macrophage progenitors (GMPs) which in turn give rise to unipotent progenitors of megakaryocytes (MkPs) and erythrocytes (EPs) for MEPs, as well as granulocytes (GPs) and macrophages (MacP) for GMPs, respectively. CLPs directly differentiate into unipotent progenitors of mature leukocytes. These are dendritic cells (DCs), natural killer cells (NK), B-cells and T-cells. Image adapted from Seita and Weissman [4], as well as Hermann [18].

(MEPs) but not in multipotent progenitors (MPPs) [19]. Hematopoietic progenitors that will differentiate into cells that are part of the granulocyte/macrophage lineage develop Fcγ receptors [20; 21]. The cells are then labeled by antibodies that are able to bind specifically to one of the surface markers. Fluorescence markers that are attached to the antibody allow identification of particular cell populations, for example by sorting with flow cytometry [15]. A variety of markers exist, some of the most commonly used are green fluorescent protein (GFP), yellow fluorescent protein (YFP) or mCherry (red fluorescence spectrum).

Molecular processes driving differentiation

Although the hierarchical structure of adult hematopoiesis has been deciphered to quite some detail on cellular level, the molecular processes that are responsible for the fate decision of a hematopoietic progenitor cell are mostly unknown. A variety of factors, such as cell to cell signaling or environmental changes, as well as gene regulation through transcription factors and micro-RNAs have to be taken into account. A challenge that gets even more complex if all possible interactions between these processes are considered. Successful studies in this field were conducted for example on some signaling molecules such as cytokines, which are signaling glycoproteins that are involved in cell cycle and thus cell growth and differentiation [17; 22; 23]. Furthermore it is generally accepted that transcription factors (TFs) are key intrinsic regulators of fate decisions. Two crucial factors that could be involved in decision of HSCs (or one of their multipotent descendants) to differentiate into MEPs or GMPs are PU.1 and GATA-1. Studies suggest that these proteins show a cross-antagonistic relationship, meaning that PU.1 levels in cells of the erythrocyte lineage drop significantly while expression of GATA-1 rises. The opposite is observed in GMPs, were PU.1 expression rises and GATA-1 expression declines. To date there is no evidence that the antagonistic behavior of PU.1 and GATA-1 is triggering lineage decision or if biased expression levels are a consequence thereof [24; 25].

Investigation of regulatory factors in stem cell systems in general has already achieved results with huge impact. The probably most impressive study in recent time revealed that somatic cells can be reprogrammed to a pluripotent state by controlling a small amount of transcription factors, generating so-called induced pluripotent cells (iPSCs) [26; 27]. IPSCs share many properties to embryonic stem (ES) cells, such as the expression of certain genes and proteins, chromatin methylation patterns, doubling time, embryoid body formation, teratoma formation, viable chimera formation, as well as potency and differentiability, but there is no final proof that iPSCs really have the same potential as ES cells [26]. However, the ability to reprogram cells to completely different functions exemplifies the necessity to understand regulatory networks in cell differentiation.

Necessity of Continuous observation of stem cell differentiation

Despite the advances that were made in order to understand hematopoiesis, its hierarchy and the processes that are involved are still subject of active discussion on both cellular and molecular level [11; 12]. In order to analyze the fate decision in more detail, it is necessary to connect behaviors and properties of individual HSCs to the fates of their progeny. Methods that measure cell properties such as transcription factor expression by mRNA microrrays require a group of cells to produce a reasonable measurement, thus the received signal is an average response of the used cells and analysis on single cell level is not possible. Furthermore, the cell typically is destroyed, which interrupts differentiation and makes it impossible to get subsequent measurements off the same cell. Analysis by flow cytometry is a method to measure cell properties without destroying the cells. As discussed in detail in 3.1, the method is able to obtain measurements on single-cell level and subsequent measurements of the same cell at later timepoints can be conducted. However, it is unfeasible to determine predecessors and progeny of the observed cell in order to obtain a complete picture of the differentiation process. Additionally, discontinuous measurements possess the problem that it is not known when fate decisions occur in stem cells. If the timepoint of measurement is chosen even shortly after or before an event of interest has happened, the information will be missed. Those events could be for example asymmetric cell division, uneven transcription factor expression or changes in movement of cells commited to different fates. Examples for events that are not observable by the discussed methods are shown in Figure 1.2

1.3. Continuous imaging of hematopoietic differentiation

A novel technique based on long-term microscopy of stem cells *in vitro* was published by Schroeder et al. [16; 28]. Living cells are observed continuously through brightfield microscopy for several days. The method is able to take images in very short intervals (about 30 seconds to a few minutes), while maintaining cell health. In addition to that, fluorescence images, i.e. expression levels of a labeled transcription factor can be measured, although these images are recorded at larger time intervals ($\sim$ 30 minutes). Long-term single-cell microscopy is able to simultaneously record a huge amount of cells and follow their complete differentiation process, thus covering morphological behavior of differentiating cells as well as fluorescence measurements for molecular factors of interest for hundreds of cells at single-cell resolution. This results in differentiation trees which make it possible to examine the different features of HSCs, as well as all daughter cells and connect individual signals to a more complete picture.

Due to the lack of automatic tracking software it is necessary to manually follow single HSCs and their progeny, connecting them to so-called tracking trees. For this purpose a tool that allows processing of recorded time lapse movies, namely Timms Tracking Tool (TTT), is applied for tracking and analysis [29]. Although TTT speeds up manual tracking enormously, this process is very time consuming. A single tree is comprised of thousands

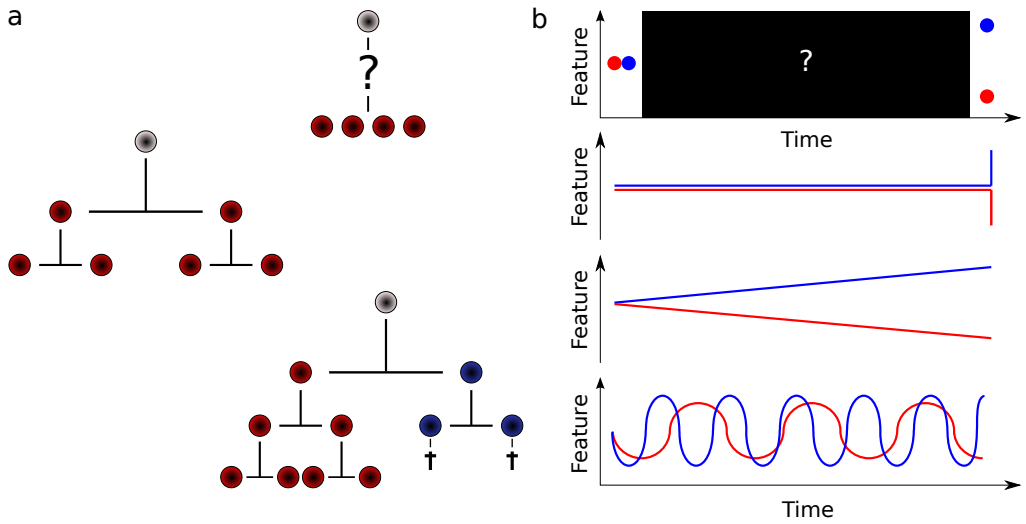


Figure 1.2. – Examples for signals and features that continuous single-cell analyses are able to observe. a) Possible topologies of differentiation trees for a HSC that is generating four daughter cells. This is not observable by flow cytometry or other methods that do not measure the cells continuously. Image adapted from [16]. b) Possible time courses of the same feature but two different cells (for example cell growth or transcription factor expression), that are at same level at the beginning and have all the same outcome. Only by continuous measurements it is possible to see different behaviors of both cells. Image adapted from [28].

of images that need to be analyzed subsequently, occupying a researcher for many days to complete a tracking tree. Computational techniques that are able to conduct this task are thus eagerly awaited, but tools that are available to date do not deliver satisfying results. Hematopoietic stem and progenitor cells are moving over the microscope's slide continuously and in later stages of the differentiation process, the slide is crowded with cells. In addition, constant imaging puts living cells under a large amount of stress, which makes it necessary to reduce image quality with regard to improved cell health, resulting in a differentiation process that is disturbed as less at possible, yet cells that are hardly identifiable in the images [28]. Software such as ImageJ [30], CellProfiler [31] or commercial solutions such as Simi BioCell [32] already offer a collection of algorithms and methods capable of cell tracking and automated image analysis, however neither of these solutions is able to complete the task on the described time-lapse movies.

1.4. Employing the powers of computational image processing and data mining

Continuous single-cell imaging is a powerful technique to decipher the mechanisms and molecular factors that drive a hematopoietic stem cell's decision to self-renew or commit to a certain lineage, providing expression levels of transcription factors (fluorescence images), as well as morphological behavior (brightfield images). Huge progress has been made in large scale analyses of transcription factor expression already, however most brightfield images that were generated during the conduction of time lapse experiments were only used for tracking purposes, a large scale analysis was not feasible due to the lack of tools that allowed analysis of the huge amounts of data in reasonable time.

The computational research fields of image processing and data mining possess these capabilities, algorithms that are dedicated to the special type of data need to be developed and comparable studies have already been published. Digital image processing denotes the automatic manipulation of image and video data by computational methods and algorithms in order to enhance informative regions in the picture. Thus, subsequent analysis by statistical or computational methods become feasible, whereas data mining is one possible application. Examples for image manipulation are transformation of colored images to black and white [33; 34], sharpening of blurry photographs [35], recognition of regions of interests in a large image or registration of sequences of pictures, i.e. normalizing an offset of a picture showing the same scene but of slightly varying angles [36].

Data mining and machine learning describe the development and validation of algorithms that are able to find patterns of relationships between different dimensions of data in large, often noisy sets of measurements. This could potentially reveal information that would not be discovered by analyzing each dimension of the data separately [37]. Furthermore, these algorithms are able to derive models that were learned on a set of measured processes, that allow prediction of future behavior. Three studies that were recently published combine image processing and machine learning on biological data and results indicate that automatic analysis of time lapse data seems possible:

A) Neumann et al. [38] applied computational image processing and data mining to conduct phenotypic profiling of the human genome. Knockout cell populations for 21,000 human protein-coding genes were generated through siRNA silencing. These populations were imaged for two days by fluorescent microscopy, chromosomes in the cells were stained with green fluorescent protein (GFP). The images were processed and quantitative features were derived. A supervised learning method was applied, classifying each sample into one of 16 morphological classes.

B) A computational method to track, process and classify different cell types based on high-throughput live cell imaging is called CellCognition [39]. A set of 186 morphological features (shape, size and textural features [40]) was derived from fluorescence images of 96 movies with imaging intervals of less than five minutes and a movie duration around 24 hours. Eight class labels (interphase, six different mitotic stages

and apoptosis) were defined. Cells were automatically tracked and segmented. A data mining approach was then used to train and predict different cell types based on the data.

C) The advantages and possibilities of computational prediction of stem cell fates have recently been shown by Cohen et al. [41]. Time-lapse movies of retinal progenitor cells (RPCs) were conducted and computationally processed. Based on this data a classification algorithm was trained, revealing that RPCs show distinctive behavior, depending on which combination of daughter cell types they were about to generate. The authors employed a set of six morphological features consisting of movement, direction, net movement, eccentricity and cell size. The method predicted fate decisions of 95% of tested progenitor cells correctly.

These studies exemplify that image processing coupled with the capabilities of data mining constitutes a powerful approach that makes fast analysis of gigabytes of image data generated by time lapse experiment feasible. The analysis of differential behavior such as movement or cell growth of hematopoietic stem and progenitor cells commited to different fates have not yet been examined at this level of detail. Combining all morphological features as well as expression profiles of transcription factors such as PU.1 could potentially reveal completely new insights into the processes underlying stem cell differentiation. Furthermore, these computational approaches drastically reduce the amount of time that is needed for postprocessing the data, allowing users to focus on analysis of interesting information instead of spending days and weeks in order to make this information visible.

1.5. Aim of this thesis

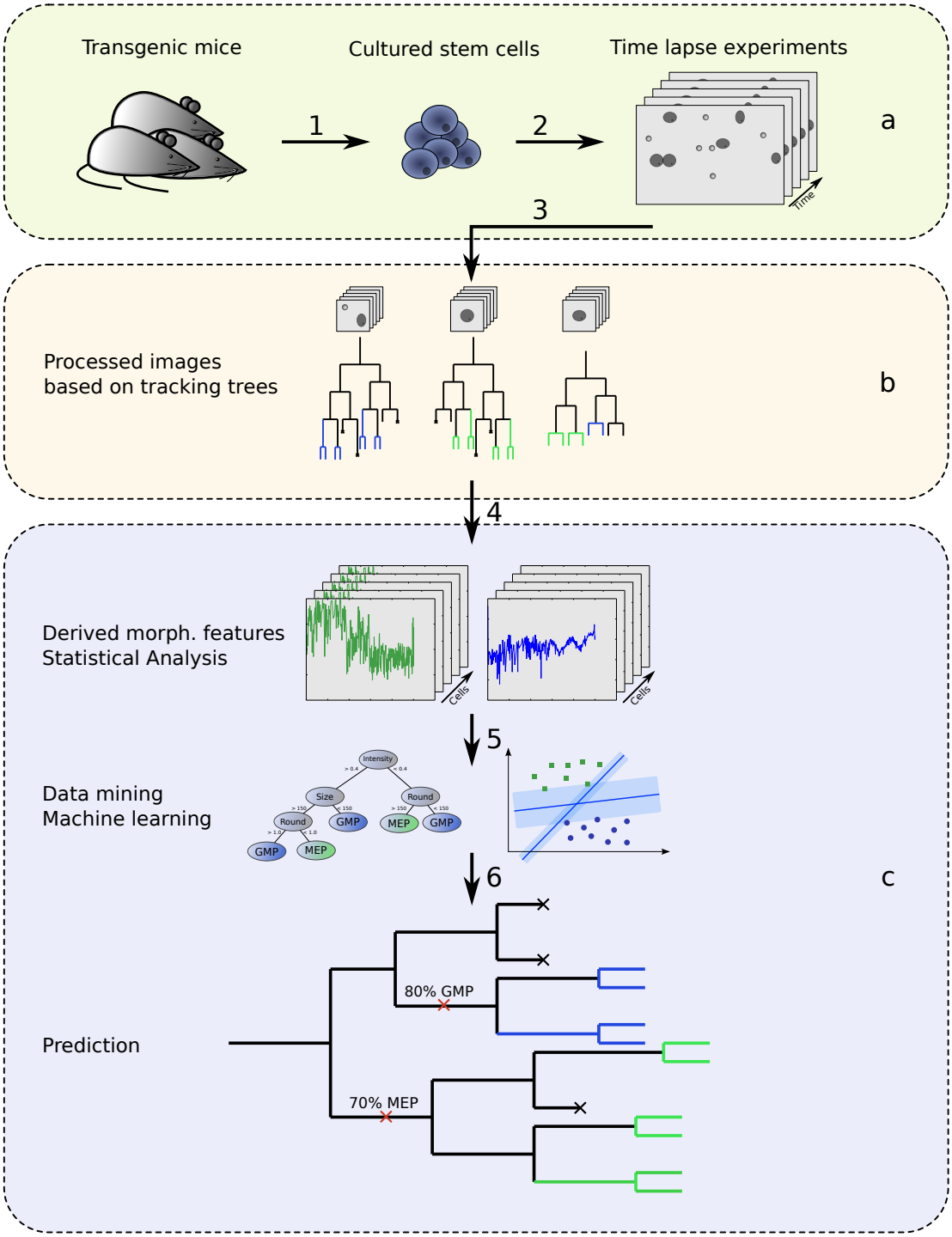
In this thesis, our aim was to predict if a hematopoietic stem cell is committing to the erytroid or myeloid lineage as early as possible in the differentiation tree. The prediction should be based on morphological cell behavior such as cell growth, cell movement, as well as expression levels of the transcription factor PU.1, which were measured in continuous time-lapse movies of HSCs that differentiate to MEPs or GMPs. The underlying data was generated at the Institut für Stammzellforschung (ISF, Helmholtz Zentrum München).

Cohen et al, already demonstrated for retinal progenitor cells that differential morphological behavior of cells commited to different lineages can be utilized to predict their fates at high accuracy [41]. However, these cells exhibit a morphology that is more diverse than that of hematopoietic stem cells. This fact as well as the technical restrictions in long term single cell imaging make automatic recognition of cell morphology and prediction of hematopoietic lineage decisions a challenging task.

The project covered the establishment of an image processing pipeline that is capable of measuring around one million images in 5 hours, generating a dataset of 14 morphological cell features and quantification of PU.1-eYFP fluorescence intensities. The obtained time courses per cell were approximated by functions in order to correct measurement errors and to allow analysis of the data by the means of functional data analysis [42]. After that, time courses of all features of MEPs and GMPs were compared, identifying signals possibly representing biological processes and behaviors that discriminate the observed cell types as best as possible. In addition, we analyzed cell movement in detail and were able to reject the assumption that hematopoietic stem cell movement is following brownian motion or a lévy walk.

The generated dataset was used to train a random forest classifier, using the functional representation to retain the factor of time in the analysis for each feature. The eventually generated model computes the probability for a given cell to either commit to erythroid or myeloid lineage. Evaluation by 10-fold cross-validation on all cells resulted in a macro-averaged f1-measure of 0.83. Additionally, we demonstrated that a feature set without PU.1-eYFP intensities resulted in a macro-averaged f1-measure of 0.77 after 10-fold cross-validation on the complete set. Thus, PU.1-eYFP intensities are an important feature for classification but morphological features alone yield satisfying performance. In a last step, we demonstrated that the prediction of cell fates of completely new trees is returning macro-averaged f1-measures of 0.92, a score that makes the possibility of fate prediction conceivable. Figure 1.3 schematizes all steps of development.

Figure 1.3. ***(following page)*** – Scheme of the whole image processing and prediction pipeline that was established in this thesis. (a) Biological Experiments. Stem cells are extracted of PU.1-eYFP transgenic mice and time lapse experiment is conducted. (b) Images are segmented by an image processing pipeline and time courses of 14 morphological features per cell are derived. In addition, PU.1-eYFP intensities are quantified aided by AMT. (c) The dataset is normalized for each feature and functional data analysis is applied to correct mis-segmented timepoints, reducing the number of predictor variables and allowing statistical analysis. The morphological behavior of cells commited to different fates is elucidated, finding out which feature could be the most important. A random forest classifier and a support vector machine are evaluated, the better is used to process the huge amount of data and to generate a model that should be able to predict a cell's most likely fate. In addition, the classifier should be able to give information about the most important features (e.g. movement) and a probability score for each cell and cell type.


Transgenic mice
Cultured stem cells
Time lapse experiments
1
2
3
a
Time
Processed images
based on tracking trees
b
4
Derived morph. features
Statistical Analysis
Cells
Cells
5
Data mining
Machine learning
Intensity
Size
Round
Round
GMP
MEP
GMP
GMP
MEP
6
c
Prediction
80% GMP
70% MEP

2. Methodological background

In this chapter we describe all methods that have been used in this thesis. At first we will discuss the algorithms and tools that were used for image processing. This is followed by detailed information about methods used to analyze the generated data. Finally we will provide definitions and explanations concerning the data mining and machine learning approaches. The commercial numerical computing environment Matlab was used to implement or develop most of the algorithms applied in this thesis.

2.1. Image processing

Digital image processing is the computational analysis of image data, such as photographs or videos. Images contain specific features that are derived by signal processing algorithms. Like most data sources, images or videos are often noisy and overloaded with information, therefore it is necessary to enhance or suppress certain features to receive a signal that provides as much useful information as possible.

In order to measure the morphological properties, the actual shape and area of tracked cells in all images were determined. For this purpose thresholding, a process separating the foreground of an image (i.e. a cell) from its background by a given intensity threshold was performed. Thresholding produces a binary image, where pixels below the threshold are set to zero (black) and pixels above the threshold are set to one (white). The resulting image should have a white region which covers the cell's area (the so-called cell mask) and a uniformly black background. Two thresholding methods were tested and evaluated.

2.1.1. Otsu thresholding

The distribution of pixel intensities in an image can be represented as a histogram. Histograms describe the distribution of data by grouping it into a fixed number of intervals. An interval is commonly referred to as a bin. A bin covering the interval $]a; b]$ gives information about how many pixels in the image are greater than a and smaller than or equal to b. For example, an 8bit grayscale image produces a histogram with 255 bins. Brighter foreground pixels and darker background pixels would thus result in a bimodal histogram, were each mode represents one of the two classes. Foreground and background can also be regarded as clusters, i.e. subsets of pixels with similar intensities. It is now necessary to define a threshold that optimally discriminates both clusters, which is typically an intensity bin in the histogram lying between both modes. Otsu's method solves this task by

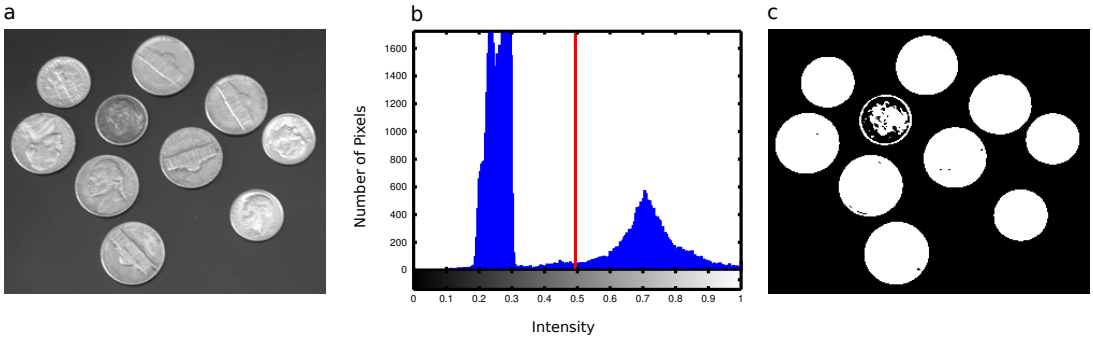


Figure 2.1. – Example for thresholding by Otsu's method. a) Original 8bit grayscale image. b) Histogram of Pixel intensities, showing two clearly separated peaks in pixel intensities, representing foreground (right) and background (left) pixels. The red line is indicating the threshold computed by the Otsu's method. c) Resulting binary image after the method was applied. White: foreground, black: background. Image taken from [43].

iterating over all possible thresholds t, minimizing the within-class variance [33]. This is denoted as

$$\sigma^2_{Within}(t) = n_B(t)\sigma^2_B(t) + n_F(t)\sigma^2_F(t), \tag{2.1}$$

where $n_B(t)$ and $n_F(t)$ denote the percentage of pixels in background B and foreground F, for a given threshold t. Computing the within-class variance of the two classes and all possible thresholds is computationally expensive, but Otsu showed that minimizing the within-class variance is the same as maximizing the between-class variance:

$$\sigma^2_{Between}(t) = n_B(t)n_F(t)[\mu_B(t) - \mu_F(t)]^2, \tag{2.2}$$

where $\mu_B(t)$ and $\mu_F(t)$ represent the mean intensity of all pixels in each class. For every potential threshold t, Otsu's method performs the following steps:

1. Assign pixels to foreground and background clusters, according to t.
2. Find the mean of each cluster.
3. Square the difference between the means.
4. Multiply by the number of pixels in one cluster time the number in the other

Otsu returns the value of t that maximizes σ^2 which represents the optimal threshold. Figure 2.1 shows an example image and the found optimal threshold that was computed by Otsu's method.

2.1.2. MSER thresholding

The MSER (Maximally Stable Extremal Regions) algorithm is a feature detector, originally designed to find informative regions (descriptors) in two images of the same scene which were taken under different conditions or arbitrary viewpoints [34].

The algorithm regards an image as a The algorithm takes an image I as input and outputs a list of nested extremal regions. A region R in an image is a contiguous subset of pixels which are 4-neighborhood connected. Two pixels are 4-neighborhood connected iff both pixels share one edge. A region $R^{extremal}$ is an extremal region iff the intensities of the pixels belonging to the boundary of the region are lower than any intensity inside of the region. The algorithm outputs those extremal regions that are maximally stable (MSERs). The term maximally stable indicates that a region $R_i^{extremal}$ is satisfying a stability criterion $q(i)$, defined as

$$q(i) = \frac{|R_{i-\Delta}^{extremal} \; R_{i+\Delta}^{extremal}|}{|R_i^{extremal}|}. \tag{2.3}$$

$R_i^{extremal}$ is maximally stable iff $q(i)$ has a local minimum at i. The stability of an extremal region R is the relative area variation of the region R when the intensity level is increased by Δ, where Δ is a parameter of the method. More concrete an extremal region is maximally stable if the area of the region varies only little when the intensity threshold is increased or decreased by Δ, respectively. Thus, Δ determines how big the contrast between foreground objects and background is.

The algorithm finds MSERs in three steps: First, pixels of the image are sorted by intensity in descending order, where pixels with the same intensity values are grouped in bins. In the case of 8bit images, this results in 255 bins. After sorting, pixels of a single bin are placed in an empty image of same size as the original according to the order described above. After adding a bin of pixels to the empty image, all connected regions are saved into a list. If a region grew after adding another bin, the list is updated. In the final step the algorithm evaluates all connected regions to check if they satisfy the stability criterion $q(i)$. The remaining regions form the output.

The implementation used here was taken from the VLFeat toolbox [44], a set of computer vision methods that are implemented in C with interfaces for Matlab. This implementation has the advantage that MSER is applicable on both the raw and the inverted image in the same run, thus identifying the cell's bright halo as well as its dark boundary as one region. The VLFeat implementation requires the following parameters:

- MinArea: Minimum Area of a MSER relative to image size.
- MaxArea: Maximum Area of a MSER relative to image size.
- MinDiversity: Absolute stability score of regions.
- MaxVariation: When the relative area variation of two nested regions is below this threshold, then only the most stable one is selected.

- Delta: Stability threshold for surviving MSER. Larger means higher stability is needed.

2.1.3. Watershedding

Undersegmented objects (that is, an area considered as foreground covering two or more objects) were split up using the watershedding algorithm published by Vincent and Soille [45]. An image can be interpreted as a landscape, where pixels with high intensity comprise hills and pixels with low complexity are valleys. This landscape is then inverted and sources of water are placed in each valley (which is now a region of maximum intensity in the original image). The level is raised, until water of two different sources touches each other or the boundaries of the image are reached. At these watersheds a line is drawn which separates one valley from another. To assign a minimum, two different methods were used. The first approach uses distance transformation [46], where the euclidean distance to the nearest pixel whose intensity value equals zero is assigned to each nonzero valued pixel of a binary image. The result is inverted and watershedding is applied. A second approach is called marker based watershedding. This method enhances the fraction of pixels in the original image with maximum intensity to islands, building so called markers. If the image is inverted the valleys were water sources are put originate in these markers.

2.2. Quantification of PU.1 levels

Schwarzfischer [47] and Krumsiek [48] developed AMT (Aided Manual Tracking) at our group. AMT is a Matlab based tool that allows semi-automatic quantification of PU.1 expression levels of single cells based on fluorescence images of the time lapse experiments. AMT is able to correct varying image illumination levels, as well as other preprocessing steps such as background subtraction and flat-field correction to allow a quantification as good as possible. AMT was used to quantify eYFP fluorescence of each movie, based on cell masks that were derived from brightfield images by our processing pipeline.

2.3. Data Analysis

The following methods were used to either normalize the data in order to correct mismeasurements or to examine the time courses per cell and feature for interesting behavior that could give insights in the lineage decision.

2.3.1. Cross correlation

Cross correlation is a measure of similarity of two measurement series (here: time courses) [49]. For two time courses $x(t)$ and $y(t)$, cross correlation is denoted as

$$r(d) = \frac{\sum_{t=0}^{N-1} ((x(t) - \mu_x) * (y(t-d) - \mu_y))}{\sqrt{\sum_{t=0}^{N-1} (x(t) - \mu_x)^2} \sqrt{\sum_{t=0}^{N-1} (y(t-d) - \mu_y)^2}}, \tag{2.4}$$

where μ_x and μ_y are means of the corresponding series, N is the length of the longer series and $d = 0, 1, 2, ...N-1$ is the delay the two series are shifted against. Thus, for each delay the score $r(d)$ is calculated and the resulting vector can be plotted. $\mathbf{r}$ has a length of $2N-1$. The higher $r(d)$, the higher is the cross correlation between both processes. Cross correlation of the same process (called auto correlation) would show a steady increase to $r(0) = 1$. After that, The curve would continuously decrease.

We used this method to examine unexpected periodic signals in cell growth, as discussed in section 5.1.

2.3.2. Interpolation

As some processing steps such as functional data analysis and random forest classification were not able to deal with missing data and time courses of different lengths, we applied linear interpolation [50]. The method takes two datapoints given by the coordinates (x_a, y_a) and (x_b, y_b) and computes the linear interpolant, i.e. a straight line between both points, denoted as

$$y(x) = y_a + (y_b - y_a)\frac{(x - x_a)}{x_b - x_a)} \text{at the point} (x, y). \tag{2.5}$$

On a time course with measurements $x_0, y_0), (x_1, y_1), ..., (x_n, y_n)$, linear interpolation is applied to each pair of adjacent datapoints, resulting in a continuous curve. It is then possible to represent the time course with a set of datapoints that were computed by this function.

2.3.3. Functional data analysis

In functional data analysis (FDA), it is assumed that all data points of a time course $x_1, ..., x_n$ are generated by a function $x_t = f(t)$ where t is a point in time. It is thus possible to approximate the underlying function using the data. FDA was proposed by Ramsay et al. in 1997. The following definitions are adapted from the second edition of Ramsay's book [42].

Each observation of time series data is regarded as a curve originating from an unknown function. These functions are assumed to be smooth, allowing to fit a continuous curve using available data points. Ramsay uses a computationally efficient way to describe an approximated function, namely a system of basis functions defined as

$$x(t) = \sum_{k=1}^{K} c_k \phi_k(t), \tag{2.6}$$

where K is the number of basis functions ϕ_k and c_k is a coefficient for the $k-$th function. Each basis function represents data of one interval $\tau_\ell, \ell = 1, ..., L-1$. The chosen basis system should have features that match those that are assumed for the unknown functions. The basis system used in this thesis are b-splines, as described below.

B-spline basis systems

Splines in general are sets of polynomials with specified order m, that are defined for a set of disjoint and compact intervals L. The intervals are limited by so called breakpoints, denoted as

$$\tau_\ell, \ell = 1, ..., L-1. \tag{2.7}$$

At each breakpoint ending and starting polynomials are restricted to the same function value, resulting in a smooth and continuous function, if the order of the spline is equal or greater to three. The order of a polynomial is the number of constants required to define it, and is one more than its degree, its highest power. The number of continuous derivatives of a spline with order m is $m-2$, whereas an order four spline results in a satisfying approximation of most data. A spline of order four has two continuous derivatives and thus is completely smooth for the human eye. In addition, derivatives are useful in order to obtain more information such as acceleration of the curve. Splines can be easily manipulated by changing the number of breakpoints and using flexible instead of equally distributed intervals. The latter can be reasonable if the curve that is to be represented exhibits high variation only in a small interval.

A popular way to represent splines is the definition of a system of basis functions $\phi_k(t)$. These functions have the following essential properties [42]:

- Each basis function $\phi_k(t)$ is itself a spline function as defined by an order m and a breakpoint sequence $\boldsymbol{\tau}$.
- Since a multiple of a spline function is still a spline function, and since sums and differences of splines are also splines, any linear combination of these basis functions is a spline function.
- Any spline function defined by m and $\boldsymbol{\tau}$ can be expressed as a linear combination of these basis functions.

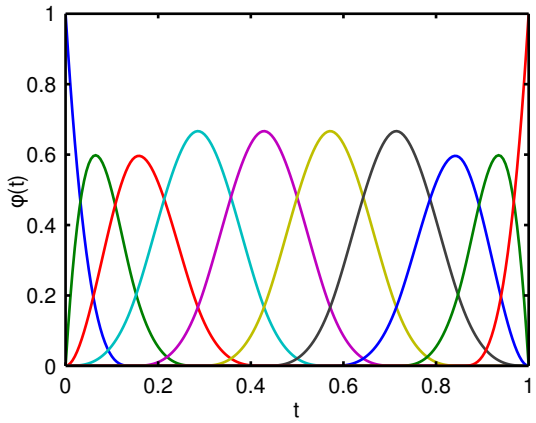


Figure 2.2. – B-spline Basis System of order 4 with 10 basis functions in interval $[0, 1]$. Basis functions have an equal shape because breakpoints are set equally distributed.

A disadvantage of b-splines, however, is a lack of approximation accuracy at both ends of the curve, because the first (last) polynomials lack the preceding (descending) neighbor and are thus badly smoothed. Figure 2.2 visualizes a basis system with 10 basis functions.

Smoothing

The main challenge in function approximation on the basis of our measurements and yet for data in general, was the demand to achieve an approximation as accurate as possible but without overfitting, i.e. representing erroneous measurements in the curve. The bias how well each measurement is represented by the fit is denoted as

$$Bias\left(\hat{x}(t)\right) = x(t) - E\left(\hat{x}(t)\right), \tag{2.8}$$

where $x(t)$ is the observed value in the time course at timepoint t, $\hat{x}(t)$ is the approximated function and $E\left(\hat{x}(t)\right)$ is the average. A b-spline with as much basis functions as measurements would result in a bias equal to zero, noise and measurement errors would be represented by the function. It is thus necessary to compute the variance of a fit

$$Var\left(\hat{x}(t)\right) = E\left(\hat{x}(t) - E\left(\hat{x}(t)\right)^2\right), \tag{2.9}$$

which is anticorrelated to bias. The mean squared error of a fit is often used to denote its quality:

$$MSE\left(\hat{x}(t)\right) = E\left(\hat{x}(t)^2\right). \tag{2.10}$$

MSE can also be calculated by

$$MSE\left(\hat{x}(t)\right) = Bias^2\left(\hat{x}(t)\right) + Var\left(\hat{x}(t)\right), \tag{2.11}$$

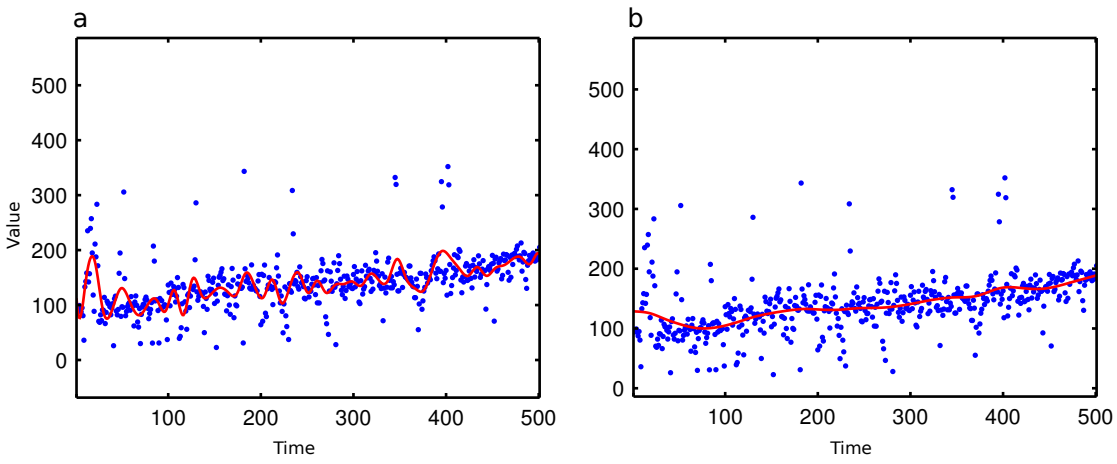


Figure 2.3. – Effect of smoothing on a b-spline function. a) The function represents most data points very well, resulting in high curvature. With respect to distribution of measurements, this does not seem reasonable. b) Smoothed curve after λ was fitted with generalized cross validation. The curve misses some data points but represents real signals much better.

implying that increase in bias can be tolerated if in turn variance is reduced [42]. This tradeoff can be achieved by computing the penalized residual sum of squares as

$$PENSSE_\lambda(x|\mathbf{y}) = (\mathbf{y} - x(t))'(\mathbf{y} - x(t))^2 + \lambda PEN_2(x), \tag{2.12}$$

where λ is a smoothing parameter, $\mathbf{y}$ is the vector of measurements. PEN_2 is the roughness of the curve, calculated by the integrated squared second derivative

$$PEN_2(x) = \int \left(D^2 x(s)\right)^2 ds, \tag{2.13}$$

This derivative describes the amount of local variance of a curve. The smoothing parameter λ is then optimized to get the best tradeoff, resulting in a curve that fits the data without accounting too much for noise.

Optimization of λ is achieved here by generalized cross-validation (GCV). A set of time series is split in combinations of training and test samples, so that each sample is used as a test exactly one time. Curves are then fitted to time courses in the training set and evaluated on the test set. The optimal λ is chosen as the value that minimizes PENSSE on all test sets. Detailed information about GCV can be found in chapter 5 of [42]. An example for this method is shown in Figure 2.3.

2.3.4. Statistical analysis of cell movement

Another step in elucidating interesting behavior of the cells was to examine the processes that are involved in cell movement. As described in 5.5, the following distributions were plotted against the data.

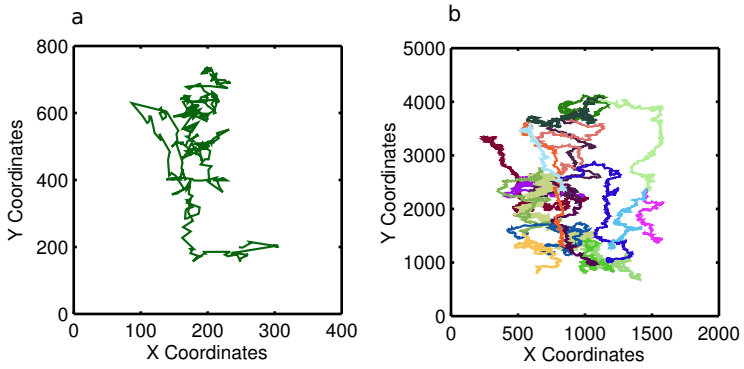


Figure 2.4. – Example plots for two-dimensional displacement of hematopoietic stem cells *in vitro*. a) Movement of a single cell on the ibidi slide. b) Movement pattern of a whole differentiation tree. Each color denotes the path that was taken by one cell. Higher resolutions and three-dimensional visualization (third dimension is the time) allow interesting insights into cell movement.

Calculation of pairwise displacement

The manually tracked differentiation trees covered annotation of two-dimensional coordinates for each cell. We used this information to calculate pairwise displacements for all cells. It is important to note here that these calculations were conducted on the original coordinates, that means no normalization steps were performed. Originally the coordinates were annotated relative to the slide position a cell was residing on at each timepoint. As cells able to move across positions, we needed to transform each value in order to derive correct displacements.

A microscopic culture slide was comprised of 39 positions arranged as shown in Figure 3.1, each position (e.g. one image) spanned 1388 by 1040 pixels. The positions were overlapping, however this was not taken into account in this analysis. We iterated over all timepoints of each tracking tree, recalculating x- and y-coordinates by following equation:

$$\begin{aligned} x_{new}^{(t)} &= (x_{old}^{(t)} - (x_{seg}^{(t)} - 25)) + (i-1) * 1388, \\ y_{new}^{(t)} &= (y_{old}^{(t)} - (y_{seg}^{(t)} - 25)) + (j-1) * 1040, \end{aligned} \tag{2.14}$$

where x and y are the coordinates of timepoint t; i, j are indices (e.g. column and row) of a matrix $\mathbf{P}$, representing the arrangement of position on the slide. Furthermore, coordinates of the segmentation results ($x_{seg}^{(t)}$ and $y_{seg}^{(t)}$) were integrated in the equation to improve the precision of a cell's position calculation on the slide. The subtraction of 25 was necessary, since the annotated x and y coordinates represented always the center of the 50 by 50 pixels subimage and our segmentation returned values relative to the subimage. A plot of exemplified cell movement is shown in Figure 2.4. The set of movement values was then compared with the following distributions.

Normal distribution

Normal distributions are one of the most important classes of distributions. Its density function is symmetric, unimodal and "bell"-shaped [51]. The function denoting the probability density function for all $x \in \mathbb{R}$ is written as

$$f(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right), \tag{2.15}$$

where μ is the mean and σ is the standard deviation of x. If $\mu = 0$ and $\sigma^2 = 1$, x is called standard normal.

Laplace and exponential distribution

A continuous random variable X with non-negative values is called exponentially distributed with $\lambda > 0$, if the density function is denoted as follows [51]:

$$f(x|\lambda) = \begin{cases} \lambda e^{-\lambda x} & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases} \tag{2.16}$$

Here, the parameter λ regulates how fast the function is reaching 0 for $x \to \infty$.

A probability distribution that shows the same behavior as an exponential distribution but is defined for negative values is called laplace distribution [52]. The density function is written as

$$f(x|\mu,b) = \frac{1}{2b}\exp\left(-\frac{|x-\mu|}{b}\right) = \frac{1}{2b}\begin{cases} \exp\left(-\frac{\mu-x}{b}\right) & \text{if } x < \mu \\ \exp\left(-\frac{x-\mu}{b}\right) & \text{else} \end{cases} \tag{2.17}$$

where μ is a location parameter and $b > 0$ is a scale parameter. If $\mu = 0$ and $b = 1$, the positive half line is exactly an exponential distribution scaled by $1/2$

Q-Q plot

The described distributions were compared to our data vector using a quantile-quantile plot (q-q plot) [53]. A q-q plot is a visual tool which plots quantiles of two distributions against each other, where each distribution is plotted on one axis. Two equally distributed sets of data should show a diagonal line which is nearly perfectly straight. Figure 2.5 exemplifies the shape of the line in the plot for two identical and two different distributions.

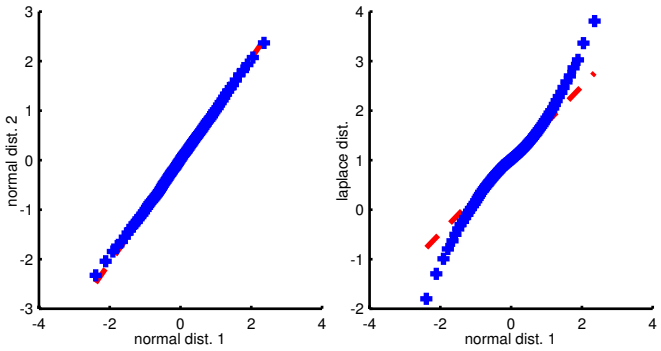


Figure 2.5. – Example of Q-Q plots. Quantiles of two datasets that are to be compared are plotted against each other. A diagonal blue line would indicate that both datasets follow the same distribution. a) Two independently sampled normally distributed sets of real values. b) Normally distributed set against laplace distributed set.

2.4. Supervised machine learning

In supervised machine learning, a computational method analyses a set of observations with two or more output classes. Based on a function that is inferred from variables that are provided for each sample the data set, a new sample with unknown class can be assigned to one of the classes that were trained in the method. This procedure is also called classification and the algorithm that performs these analyses is called classifier. Subsequently, the set of observations with known output value is called training set and the set of samples with unknown output values is called test set. Figure 2.6 schematizes the steps that are generally conducted in supervised machine learning.

Training an algorithm for data analysis makes it possible to find correlations and signals in masses of data that were not analyzable manually. In the case of single cell time lapse movies, we performed machine learning to find signals in the time series for each of the features described in 4.2.

A huge amount of different supervised classification methods exist and all have particular strengths and downsides. We chose two approaches that are well described in literature, namely Random Forest and Support Vector Machines, optimized them to fit best on our data and compared results afterward.

2.4.1. Random forest

To understand the functionality of random forests, it is necessary to introduce decision tree classifiers [54]. This method uses a tree structure were each interior node represents a variable of the input set and a leave corresponds to one of the classes. The tree is constructed top down (e.g. from root to leaves), at each branch the variable that best splits the data set based on a score is used. A popular score function is Gini's diversity

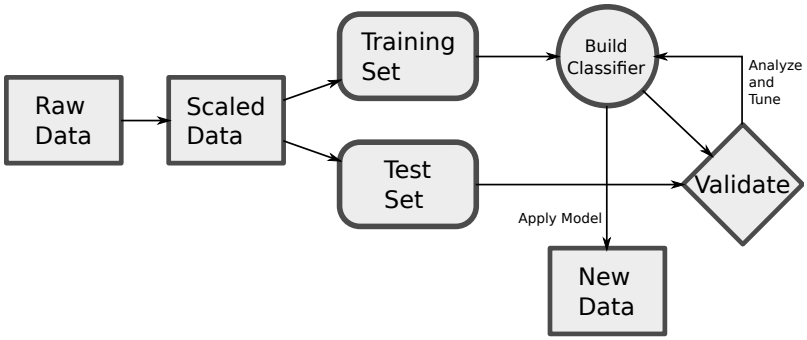


Figure 2.6. – Scheme of a supervised learning pipeline. Raw data is normalized to equal scales to make it comparable. The set is split in training and test samples. A model is build based on the training cases and evaluated with test cases. As most datasets are only a snapshot of a more complex model were not all variance is exposed, train and test cases are a commonly used method to check if the trained model is overfitted, e.g. samples that show slightly different behavior are not correctly represented by the model. The model is then refined iteratively and applied to new data if the model performs good enough.

index [55]. Let A be the variable that for a set of samples T is generating a partitioning $T_1, T_2, ..., T_m$. The Gini index is then denoted as

$$gini_A(T) = 1 - \sum_{i=1}^{m} p_i^2, \tag{2.18}$$

where p_i is the relative abundance of labels $i = 1, ..., m$ in the set. Thus, for each generated partition T_i, Gini's index computes the relative abundance of class labels. A low Gini index is indicating high variable importance and vice versa. An example of a decision tree is shown in Figure 2.7.

Random forest classification [56] is a subclass of ensemble learning methods. In ensemble learning, many classification algorithms are applied to the same data set and predictions are aggregated, which often results in improved predictive power. Two well-known methods for ensemble learning are on the one hand boosting [57], where successive trees are given an extra weight to points incorrectly predicted by earlier predictors and in the end a weighted vote is taken for prediction. On the other hand bagging [58] constructs each predictor independently using a bootstrap sample of the data set and a majority vote over all trees is taken for the final prediction. Random forests are a variant of bagging. A - ideally much smaller - subset of features is randomly drawn and a decision tree is grown based on this subset. This is repeated many times, resulting in a "forest" of decision trees, each based on a subset of the original features. Each sample in the training set is then classified by every decision tree and the most probable class is chosen by majority vote. In his paper Breiman stated the following advantages of random forests:

– It is unexcelled in accuracy among current algorithms, at least for many popular benchmark data sets.

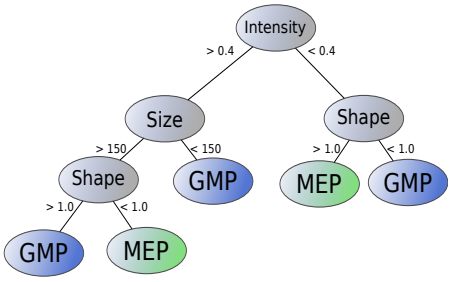


Figure 2.7. – Example structure of a decision tree. Trees are constructed top-down, starting with the feature that has the most predictive power, all preceding nodes are labeled with other features in ascending order according to Gini's index [55]. A split-criterion is defined to get a threshold that indicates which branch should be followed for a sample. After some steps a leaf is reached and the most likely class label is returned. Techniques like pruning can be applied to prevent overfitting the model to the training data.

- It runs efficiently on large data bases.
- It can handle thousands of input variables without variable deletion.
- It gives estimates of what variables are important in the classification.
- It generates an internal unbiased estimate of the generalization error as the forest building progresses.
- It has an effective method for estimating missing data and maintains accuracy when a large proportion of the data are missing.
- It has methods for balancing error in class population unbalanced data sets.
- Generated forests can be saved for future use on other data.
- Prototypes are computed that give information about the relation between the variables and the classification.
- It computes proximities between pairs of cases that can be used in clustering, locating outliers, or (by scaling) give interesting views of the data.
- The capabilities of the above can be extended to unlabeled data, leading to unsupervised clustering, data views and outlier detection.
- It offers an experimental method for detecting variable interactions.

Knowledge of the degree of influence that each feature has on the final prediction is of great biological interest in our case. It could help to develop new or specify existing theories on when and how lineage decision takes place during differentiation of a hematopoietic stem cell.

Feature importance is computed by averaging the changes in split criterion (see Gini's index) over the entire ensemble of grown trees. Change in split criterion is computed by

estimates of input feature importance for every decision tree by summing changes in the risk due to splits on every feature. At each node, the risk is estimated as node impurity. This risk is weighted by the node probability. Variable importance associated with this split is computed as the difference between the risk for the parent node and the total risk for the two children.

Subsequently, the given probability to which a sample is assigned a class can be used to examine borderline cases and to improve the training procedure. In contrast to other classification methods, decision trees deliver an output which can be intuitively interpreted. This simplifies presentation of results to collaborators without strong theoretical background and allows an interdisciplinary analysis of the results.

We applied the method to all train and test combinations defined in section 6. To compare classifier performance, we also trained a support vector machine and compared results.

2.4.2. Support vector machine

Support vector machines are supervised learning methods to discriminate data linearly [59]. A simple SVM is a binary classifier, where two classes are separated by a hyperplane, which maximizes the distance of the data points between the classes. The distance depends only on data points that are closest to the hyperplane (called the support vector), this is also described as maximizing the margin of support vectors and hyperplane.

For data where linear separation is not possible, the so-called kernel-trick allows transformation of the data to a feature-space were linear separation can be achieved. It has been shown that the kernel-trick can be applied to every type of data.

Principal component analysis

The computational effort that is needed to train a SVM rises exponentially with the number of features. It is thus necessary to reduce the featurespace, which was achieved here by Principal Component Analysis.

PCA is a statistical method to structure, reduce and analyze huge data sets [60]. A number of possibly correlated variables is transformed into a - mostly smaller - number of uncorrelated variables called principal components. These principal components have the following properties:

- The first principal component is a vector that explains most of the variability in the data.
- Principal components are ranked by the amount of variability in the data that they explain.
- Each principal component is orthogonal to its preceding component.

– Principal components are guaranteed to be independent only if the data set is jointly normally distributed.

PCA performs best if all features are equally scaled. This can be achieved by applying the zscore to the data matrix. The formula is denoted as

$$z = \frac{x - \mu}{\sigma}, \tag{2.19}$$

where μ is the mean and σ is the standard deviation of the data set. We applied this method to create a dataset that was processable by a support vector machine.

2.4.3. Evaluation

The methods described below were used to analyze or visualize the performance of the classification methods.

Cross-validation

Cross-validation denotes an evaluation method that is very popular in the field of machine learning to evaluate generalization abilities of a classifier given a set of samples [61]. A dataset is split into x folds of training and test set pairs, where each sample is at least once in the test set, but never in test and training set of the same fold. x is a scalar specifying the number of folds, a value of 10 has proven to be a reasonable choice for most applications. For each fold the classifier then learns the rules based on the training set and is evaluated on the test set. Each fold is scored by one of the methods described above or another measurement that allows to estimate the classifiers performance. The scores of all folds are then averaged, resulting in a final estimation on how good the classifier will perform on unknown cases.

Confusion matrix

A confusion matrix contains information about known and predicted class labels in a dataset. There is no agreement in the literature if predicted classes are written horizontally or vertically at the matrix, thus we are using the following definition as described in Matlab's documentation for the respective method: Let $\mathbf{C} \in \mathbb{R}^{2\times 2}$ be the confusion matrix, than is $C_{i,j}$ the count of observations known to be in class i but predicted in group j. In typical machine learning problems were a positive class is predicted against a negative class, for example if a patient is affected by a disease or not, one can apply definitions for each cell of the matrix:

– True positive (TP): Samples of the positive class that were correctly predicted.

– True negative (NP): Samples of the negative class that were correctly predicted

– False negative (FN): Samples of the positive class that were predicted as negative

– False positive (FP): Samples of the negative class that were predicted as positive.

Table 2.1 visualizes a confusion matrix for two arbitrary classes, and the definition of its cells. The following scores were used to evaluate classification performance. All calcula-

	$Class_{pos}$ (pred.)	$Class_{neg}$ (pred.)
$Class_{pos}$	TP	FN
$Class_{neg}$	FP	TN

Table 2.1. – Example of a confusion matrix. True positive (TP): Samples of the positive class that were correctly predicted; True negative (NP): Samples of the negative class that were correctly predicted; False negative (FN): Samples of the positive class that were predicted as negative; False positive (FP): Samples of the negative class that were predicted as positive.

tions are done on the basis of a confusion matrix.

Precision

The precision measures the portion of the assigned classes that were correct. It falls in the range from $[0; 1]$, with 1 being the best score. Precision is denoted as

$$prec = \frac{TP}{TP + FP} \tag{2.20}$$

Recall

Recall is a measure to determine the portion of the correct classes that were assigned. It falls in the range $[0; 1]$, with 1 being the best score. Recall is denoted as

$$rec = \frac{TP}{TP + FN} \tag{2.21}$$

F1-measure

The f1-measure combines precision and recall in a single score. Its values are also within the interval $[0; 1]$, where 1 denotes the best score. The F1-measure is denoted as

$$f_1 = 2 \cdot \frac{prec \cdot rec}{prec + rec} \tag{2.22}$$

Macro- and microstatistics

In multi-class classification tasks, the measures described above only give information about performance on one class. In order to obtain a single measure that accounts for performance on both classes, it is necessary to apply macro and micro statistics. For example, the f1-measure can be calculated for each class and then averaged, which is called macro-averaging. In contrast, the true positives, false positives and false negatives can be summed up for all classes and precision, recall and f1-measure are calculated afterward. The two procedures bias the results differently - micro-averaging tends to over-emphasize the performance on the class with the most samples, while macro-averaging over-emphasizes the performance on the class with the fewest samples. For a set of class labels $C = c_1, ..., c_n$, the micro-averaged f1 measure is defined as:

$$TP' = \sum_{i=1}^{|C|} TP(c_i), \tag{2.23}$$

$$FP' = \sum_{i=1}^{|C|} FP(c_i), \tag{2.24}$$

$$FN' = \sum_{i=1}^{|C|} FN(c_i), \tag{2.25}$$

$$prec' = \frac{TP'}{TP'+FP'}, \tag{2.26}$$

$$rec' = \frac{TP'}{TP'+FN'}, \tag{2.27}$$

$$micro_{f_1} = 2 \cdot \frac{prec' \cdot rec'}{prec'+rec'}, \tag{2.28}$$

and the macro-averaged f1-measure is defined as the mean of all class labels C_i

$$macro_{f_1} = \frac{1}{|C|} \sum_{i=1}^{|C|} F_1(c_i). \tag{2.29}$$

3. Biological data used in this thesis

In this chapter, we describe the first steps in Figure 1.3 that were done to generate the time lapse movies that were the basis of this thesis. All experiments were conducted at the Institut für Stammzellforschung (ISF, Helmholtz Zentrum München), in particular by Philipp Hoppe and Dr. Timm Schroeder. Parts of this section were adapted from Hoppe [23].

3.1. Sample preparation

The model organism used here was a healthy mouse strain with no phenotypical difference to wildtype mice. To quantify expression levels of PU.1, the coding sequence for eYFP was knocked-in into the endogenous locus of this transcription factor. Femur and tibia were removed from 12 to 16 weeks old mice and bone marrow was extracted. Hematopoietic stem cells (HSCs) were isolated and sorted by antibodies binding to the CD150 surface protein that identifies HSCs to receive a high concentration of this cell type in the medium. This step was conducted by flow cytometry. This processes are referring to step 1 in Figure 1.3.

Flow cytometry is a mechanical method to sort, count and analyze a mixture of biological cells with a throughput rate of ∼10000 cells per second [62]. Antibody labeled cells (see 1.2) are transferred to a tube with medium. The cells are passing three lasers with wavelengths of 405 nm, 488 nm and 633 nm, that are measuring size and granularity of each cell and activating fluorescent markers. Subsequently, a photomultiplier detects the emitted light spectrum and identifies the marker. All measured properties are then used to classify the cell to one of three classes. Cells are eventually sorted according to the desired marker expression. In addition, this method is capable of measuring expression of labeled transcription factors on single cell level.

3.1.1. Conduction of time lapse movies

After sorting, HSCs were pipetted on a ibidi μ-slide $VI_{0.4}$ (Integrated BioDiagnostics GmbH, Munich, Germany). A μ-slide consists of a plastic carrier with a channel and a thin bottom foil. The chip combines a cell culture dish with an optical observation chamber. If these slides are placed in an incubator, gas exchange ($5\%CO_2, 5\%O_2, 90\%N_2$) allows cell growth inside the channel over long time periods.

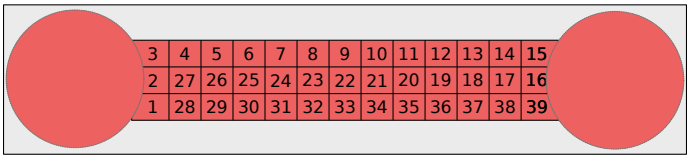


Figure 3.1. – Example of an ibidi slide with 39 positions. Round channels allow gas exchange to maintain a comfortable environment for the cells. Positions are labeled according to the moving pattern of the imaging robot.

Since the microscope's camera was not able to cover the whole slide, a grid of overlapping subimages (so-called positions) between two channels was recorded, resulting in 39 up to 45 positions per pair of channels [63]. See 3.1 for a sample of one channel pair.

Plastic beads (BD Calibrite APC Beads) were added to the medium to allow normalization of illumination differences, in order to make different movies and positions comparable. Furthermore, the beads supported the autofocussing procedure, resulting in better image quality.

The prepared slide was then installed in an inverse fluorescence microscope AxioVert 200 (Zeiss). An acrylic glass case was used to maintain a constant temperature of 37°C. Images were obtained by an AxioCAM HRM (Zeiss). A TV-Adapter 0.63 (Zeiss) was attached to achieve a coverage of one position per image. The camera produced 14-bit gray-scale images with a size of 1388 by 1040 pixels. To enable automatic recording of all positions, the slide was attached to a motorized platform that was moving the slide in a programmable pattern.

Commands necessary to perform the automated time lapse experiment were set up using TAT (Timm's Acquisition Tool, developed by Dr. Timm Schroeder), a script which is based on the microscopes original software AxioVision. TAT allows to use freely selectable time intervals for imaging as well as usage of slides with any number of positions. A time interval of 90 seconds for brightfield and 22.5 minutes for fluorescence images was used, respectively. Experiments were run up to 6 days, resulting in a sequence of tens of thousands of images per time lapse movie. The methods described above refer to step 2 in Figure 1.3.

3.1.2. Manual tracking

Differentiation trees were recorded with TTT (Timm's Tracking Tool, developed by Dr. Timm Schroeder), a software suite consisting of algorithms allowing fast and interactive manual tracking of single cells. This enabled researchers to track hematopoietic stem cells as well as their progeny, annotating events like cell division, apoptosis or cell motility. The complete data structure is called tracking tree. A visualization of such a tree is shown in Figure 3.2. The cell number is a unique scalar. The root of a tree (e.g. the HSC) is labeled with 1. Numbers of all progeny are then computed by

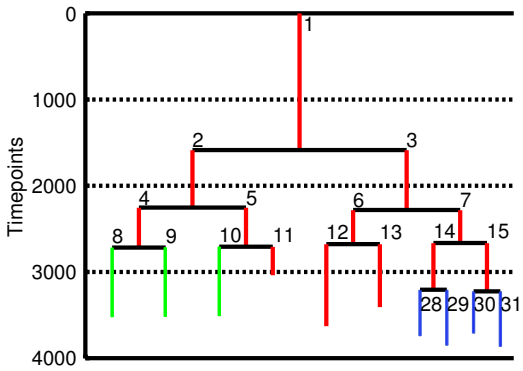


Figure 3.2. – Visualization of a tracked differentiation tree. Lines shown in red represent HSCs which undergo cell division and eventually differentiate to a GMP (blue) or MEP (green). The y-axis denotes a cell's lifetime.

$$c_{ld} = c_{mother} * 2 \tag{3.1}$$

$$c_{rd} = c_{mother} * 2 + 1, \tag{3.2}$$

where c_{ld} is the left daughter and c_{rd} is the right daughter of a cell c_{mother} in the tracking tree. It is thus possible to reconstruct the position of a cell in the tree by its annotated number. Additionally, the plot covers the lifespan of each cell by the length of its line. Different colors of these lines indicate annotated cell types, were red lines are representing MPPs (starting cells could also be HSCs), blue is representing GMP and green MEP, respectively.

3.2. Overview of the used time lapse movies

A total of 142 differentiation trees (step 3 in Figure 1.3) based on five movies was available for this thesis, details are shown in Table 3.1. Movie names are consisting of the date of recording, denoted in the scheme YYMMDD, as well as two characters identifying the annotator (i.e. PH), and a single number (1-5) according to the microscope that was used to conduct the experiment. The movies shown here can be subdivided into three classes:

1. Experiment 100922PH3 was conducted originally to measure the protein decay of different cell types. Translation was inhibited by adding cycloheximide. Populations of HSC, MPP and GMP cells purified by flow cytometry sorting and pipetted to separate channels on the μ-slide, preventing cells of different types to cross positions. In addition to positions treated by cycloheximide, an equal amount of control positions was loaded with untreated cells. Continuous imaging was performed for 24 hours, resulting in trackings that cover the life cycle of only one cell.

Name	Trees	HSCs	MEPs	GMPs	Pos.	Days	Size(gb)
100201PH2	79	628	165	95	39	6	134
100301PH4	22	679	44	197	39	5	78
100922PH3	11	20	30	33	36	0.5	11
101125PH2	21	278	39	46	39	6	27
110322PH5	9	77	0	0	45	4	119
Sum	142	1682	278	371	198	21.5	369

Table 3.1. – Overview of all movies that were used in this study. Column two shows the amount of trees that were available for the respective movie, columns three to five show the total amount of cell types that were annotated in these trackings. Experiments highlighted in gray were mainly used in this thesis.

2. Experiments 100201PH2, 100301PH4 and 101125PH2 represent long term time lapse movies, consisting of series of brightfield images, as well as three fluorescence channels. Channel one emits the eYFP response that is tagged to PU.1 proteins, channels two and three show response of antibody markers FcγR-Alexa647 (GMP) and CD150-Alexa555 (MEP), respectively. Although image quality clearly differs, all experiments were performed under comparable conditions and technical equipment.

3. The most recent experiment 110322PH5 allowed emission of 4 fluorescence channels, namely PU.1-eYFP, GATA-Cherry, FCγ and CD150 for channels one to four, respectively. Additionally, a laser was used for autofocusing that reduced illumination times and replaced the addition of beads.

Since types for all cells in movie 100922PH3 were known, we used trackings in the control positions to conduct a proof-of-concept by applying an early prototype of the image processing pipeline (see section A). Pipeline development was conducted based on experiment 100201PH2. All other standard movies were added to the data set after the full pipeline was established.

The tree files were preformatted by AMT. For each timepoint, the dataset contained the relative path to the image on which the annotated cell was found, the position on the slide, the cell number in the tree, as well as two-dimensional coordinates indicating the center of a tracked cell on the image. The tree topology is completely stored in data structure. This enabled us to preserve all information of a cell during the processing steps

and allowed to map each time course of a morphological property in the time lapse movie on the differentiation tree.

4. An image processing pipeline to quantify stem cell morphology

In this chapter, we describe the development of an image processing pipeline and its application on tracked differentiation trees of time lapse movies in order to derive time courses of the morphological behavior of hematopoietic stem cells and their progeny (step three in Figure 1.3). We discuss each morphological feature in detail, as well as the performance of the final pipeline and the dataset of time courses that was generated.

4.1. Development of the pipeline

Time-lapse experiments are conducted by continuously imaging all positions of an ibidi slide, producing thousands of images. For example, one of the experiments used here was comprised of 39 positions times 6173 timepoints, thus 240747 images with a size of 1388×1040 pixel, respectively. On each image there were several HSCs at the beginning, ending up in hundreds of different cells after several days. To date, there are no working automatic tracking methods that are able to process this amount of data and to reconstruct differentiation trees without manual supervision. To identify the cells, we obtained tracked differentiation trees provided by the ISF, as described in section 3.1.

In a first step, we established a method capable of identifying the outline of each cell per image in order to measure its morphological properties. Therefore, it was necessary to first discriminate the uninformative regions of a picture from those showing regions of interest (ROI). In this early stage a ROI could be a cell as well as a plastic bead or other regions that were not considered as background. The process to achieve this discrimination is called segmentation. Previous approaches that were applied to segment fluorescence images of time lapse experiments, such as Schwarzfischer [47] or Krumsiek [48], used a semi-automatic approach since automatic processing of continuous time lapse data did not produce satisfying results. However, a manually aided procedure seemed not feasible, because fluorescence recordings were conducted in intervals of 30 minutes but brightfield images were taken every 90 seconds. Considering that, our approach needed to be capable of processing 15 times as many images. Nevertheless an advantage of the huge amount of timepoints was that erroneous values could be corrected more easily by taking into account adjacent timepoints where segmentation was successful.

In order to identify the cells in the image we first converted the grayscale brightfield images to binary images, a process that is called thresholding. In general, thresholding

algorithms try to assign each pixel of an image to one of two classes, resulting in white pixels representing foreground and black pixels that are background. As the tracking provided the approximate center of each cell, the underlying image was read and a section of $w \times w$ pixels surrounding the annotated centroid was cut out, where w was set to 50. The size of this subimage was large enough to cover the cell even if the annotated centroid was not accurately set. In addition, this method reduced the complexity of identifying the correct cell and allowed faster computational processing.

We applied two different thresholding algorithms on a set of representative images showing problematic cases for segmentation. As exemplified in Figure 4.1, most images did not show a single, well contrasted cell. In addition, we tested the performance of the methods in large scale by applying them to the complete dataset.

4.1.1. Otsu thresholding

Since Otsu thresholding is a method well known and often used in image processing, our first approach was based on this algorithm. Furthermore, Schwarzfischer [47] showed that the method produces reasonable results on fluorescence images of the experiments used here. Details of the algorithm are discussed in section 2.1.1.

If an image was well focused as well as evenly illuminated and all objects in the image were well contrasted, the method was able to correctly identify them, yet segmentation mostly covered only the bright halo of a cell. Computation times of the subimages were below one second on a standard computer. However, the algorithm exhibited huge problems thresholding images that differed slightly from optimal conditions. If a cell was not clearly visible, single background pixels were considered as foreground, leading to a noisy segmentation and identification of the original cell mask was impossible. Otsu's method is a global thresholding algorithm, meaning that all pixels in an image are thresholded at the same time and with same parameters. This caused problems if objects in the image were better contrasted or illuminated than the cell. This was the case, for example, under the presence of beads. In these cases the cell was classified as background and measurements were not possible. Reasonable results were obtained for $\sim 25\%$ of all images from experiment 100201PH2. Segmentation of all trees of other experiments failed.

The experience with Otsu's method indicated that an optimal thresholding algorithm should possess the following features:

1. Sensitivity for size of a foreground object. The method should take into account the size of all thresholded regions and ideally provide a parameter to customize this cutoff. This would help to prevent regions which are too small to represent a cell or single pixels to be classified as foreground. In addition, areas that are much too large would be discarded.

2. Local instead of global thresholding. The algorithm needs to consider different parts of the image separately instead of regarding all pixels of the complete image at once

and thus assigning the much brighter beads to foreground, while missing real cells. In addition this would help to reduce mis-segmentations of uneven illuminated images

3. A more sophisticated method to discriminate pixels. The method should not just discriminate foreground regions from background by the histogram distribution of brightness values alone. This is needed to recognize the dark outline and bright halo (e.g. a cell's core) a typical cell is exhibiting in brightfield images as one region.

4.1.2. MSER thresholding

MSER (Maximally Stable Extremal Regions) is a method well known in the field of computer vision and is most often used for image registration. As described in section 2.1.2, MSER allows to set limits for a region's size as well as parameters defining the degree of change that is needed to consider a region as foreground or background. Furthermore, the algorithm computes binary images based on the original and inverted image, allowing to account for both very dark and very bright regions in an image.

MSER thresholding produced well segmented binary images on long periods of most tracking trees of movie 100201PH2, showing high robustness even if cells were contrasted to a level that made them barely identifiable by eye. On other experiments MSER found a satisfying amount of time points, allowing to use the measured tracking trees in further analyses. A drawback could be the computational effort of MSER. However, due to the subimage size of 50×50 used here thresholding took less than 0.5 seconds on a standard computer, which is fast enough for large scale application. The parameters of the algorithm were empirically optimized and set as follows:

- MinArea: 0.08
- MaxArea: 0.3
- MinDiversity: 0.2 (default)
- MaxVariation: 1
- Delta: 1

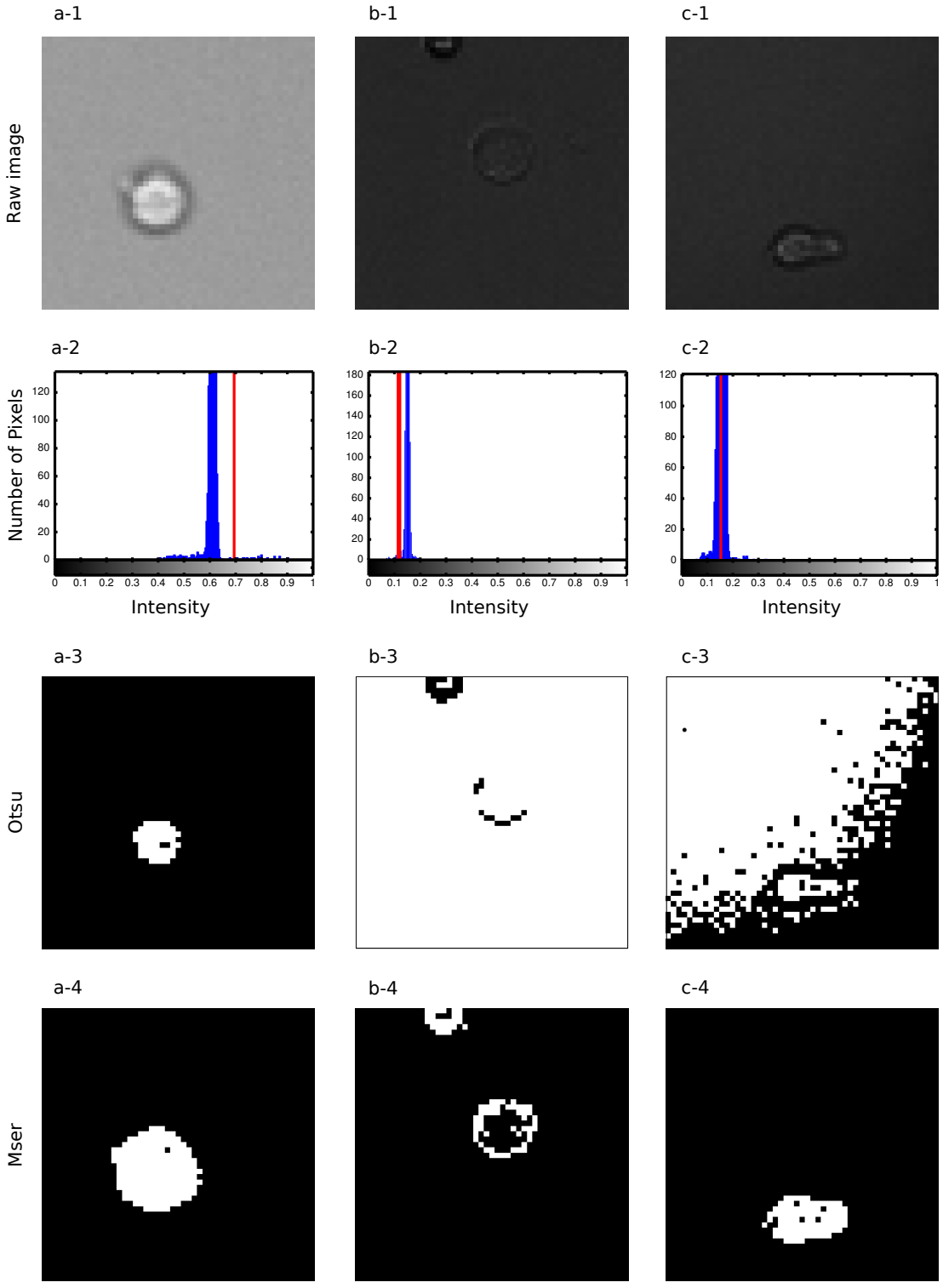
Aside from implementing MSER into our image processing pipeline, the method was used on brightfield images of embryonic stem cells, which are to date not processable automatically. The possibilities and preliminary results of the application of MSER on these kind of movies are discussed in appendix B.

4.1.3. Watershedding

Under ideal conditions, i.e. if a single cell was shown in the 50×50 subimage that was evenly illuminated, the cell masks computed by MSER thresholding were accurate. However, a large amount of images exhibited more than one cell or other unwanted objects, such as beads or contamination on the slide. This was especially a problem if the cell lay adjacent to other objects which were considered as foreground. In the binary image after thresholding this resulted in a large white area covering all objects that touch each other, rendering reconstruction of the original cell outline impossible. This would produce erroneous measurements. The technical term used here is undersegmentation, e.g. a group of distinct foreground objects is regarded as one. A typical case of undersegmentation is represented by images that show a cell that is undergoing mitosis, e.g. cell division. This event results in $\sim$20 timepoints, where daughter cells are residing adjacent to each other. In order to to split undersegmented objects, we employed a method called watershedding, as described in section 2.1.3. At default, we used the distance transformation approach, yet marker based watershedding was applied if the first method was not able to produce a reasonable split, based on the criteria discussed in the following section. The outcome was a grayscale image (see Figure 4.2), were black lines indicate the objects boundaries.

If the two objects exhibited clearly separable halos and had a nearly round shape, watershedding by distance transformation worked quite well. This was for example the case in late mitosis, where daughter cells were clearly identifiable, yet lying adjacent in the image. In addition, pairs of cells and beads showed satisfying results, if the cell was well contrasted

Figure 4.1. ***(following page)*** – Comparison of thresholding results on three sample images. Red lines in the histogram plots are indicating the threshold selected by Otsu's method. a) Well illuminated image with only one region of interest that represents the cell. Intensity histogram exhibits a unimodal distribution, leading to Otsu's method being not able to threshold the complete cell outline. Due to thresholding both raw and inverted image, the binary image produced by MSER covers the complete region representing the cell. b) Very dark image with a bead in the upper left that is much better contrasted. Otsu's method is not able to identify the cell outline, the bead is detected correctly. As the histogram does not show a clear bimodal distribution, the binary image is inverted, i.e. foreground pixels were classified as background and vice versa. MSER uses an energy function that is not dependent on histogram shape, thus cell outline as well as the bead are correctly identified. c) Uneven illuminated image. Otsu's method splits the histogram peak, resulting in half the pixels of the image being considered as foreground. MSER does only take pixels in a neighborhood into account for segmentation and produces correct results here, too.


a-1
b-1
c-1
Raw image
a-2
b-2
c-2
Number of Pixels
Intensity
Intensity
Intensity
a-3
b-3
c-3
Otsu
a-4
b-4
c-4
Mser

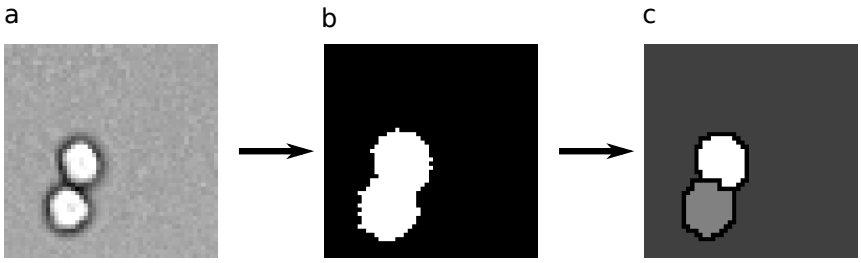


Figure 4.2. – Watershedding applied on thresholded images. a) Original image. b) Binary image after thresholding. c) Watershedding based on distance transformation. Cells are well separated and two different regions can be identified.

and the halo was clear. However, in the case of deformed cells (often observed in early mitosis), poor contrast or uneven illumination, marker based watershedding sometimes achieved better results.

Compared to thresholded binary images, the sizes of watershedded regions turned out to be slightly smaller, since the watershedded boundaries lay within the thresholded area. To retain the scale of measurements across all images we applied watershedding to all timepoints, regardless whether undersegmentation was detected or not. A big drawback of this strategy was that now single objects that were correctly thresholded could get split up by watershedding, a process that is called oversegmentation. This was the case if a single cell exhibited uneven illumination patterns or shapes that are similar to those of clumped objects (see Figure 4.5). However, this tradeoff was reasonable, since loss of information due to oversegmentation is negligible if in turn segmentation accuracy of cells undergoing mitosis is increased. Most of the cells observed in the trees were undergoing mitosis sooner or later, and this event always represents the beginning and end of a cells life and so the measured time course, were it is crucial to get correct measurements as a later correction is much harder. The combination of MSER thresholding and distance transformed watershedding represented the segmentation step of our pipeline. It was now necessary to identify the correct foreground object in a binary image.

4.2. Morphological features and additional information

A binary image that was returned by our pipeline was comprised of regions of interest and background, thus we were now able to measure the geometrical properties of each region, as well as its brightness intensities. To understand how each feature was computed based on the raw and binary image, the following basic terms are necessary:

- **Region**: A segmented foreground object. Can be a cell, bead, or contamination in the image.
- **Bounding Box**: A rectangle that exactly surrounds the region.

- **Ellipse**: An ellipse that has the same second moments as the region.
- **Convex Hull**: Smallest convex polygon that can contain a region.

4.2.1. Features

The segmented image was used as a mask for the raw image. Every feature was computed for each timepoint in a tracking tree, i.e. for every processed subimage. Figure 4.3 illustrates the calculation of some features.

- **Area**: Number of pixels in the region. This feature denotes the two-dimensional size of a cell.
- **Convex area**: Number of pixels in the convex hull.
- **Roundness**: Roundness of the ellipse. Continuous value of the interval $[0; 1]$, where 0 represents a circle and 1 represents a rectangle. The technical term for this feature is eccentricity.
- **Equiv diameter**: Diameter of a circle with the same amount of pixels as the region.
- **Extent**: Ratio of pixels in the region to pixels in the bounding box.
- **Major axis length**: Length (in pixels) of the major axis of the ellipse.
- **Minor axis length**: Length (in pixels) of the minor axis of the ellipse.
- **Max intensity**: Value of the pixel with the greatest intensity in the region.
- **Mean intensity**: Mean intensity of pixels in the region.
- **Min intensity**: Value of the pixel with the lowest intensity in the region.
- **Perimeter**: Distance around the boundary of the region in pixels.
- **Solidity**: Proportion of the pixels in the convex hull that are also in the region.
- **Orientation**: The angle (in degrees ranging from -90 to 90 degrees) between the x-axis and the major axis of the ellipse.
- **Movement**: Root Squared pairwise displacement from one timepoint to another. Calculated by $d_i = \sqrt{(x^{(t)} - x^{(t+1)})^2 + (y^{(t)} - y^{(t+1)})^2)}$, where x_i is the $i - th$ value of Position X and y_i is the $i - th$ value of Position Y.
- **PU.1-eYFP intensity**: Scalar value of eYFP intensity, indicating PU.1 expression. Derived from fluorescence images after normalization by AMT.
- **Lifetime**: Amount of time points the cell lived. One value per cell.

4.2.2. Additional information

The information here was necessary to retain all annotated information that was provided by the manually tracked differentiation trees. We used this information in the later prediction step to evaluate classifier performance, for example for different generations.

- **Tree Number**: Number of tree in the results structure of AMT and TTT.
- **Cell Number**: Number of a cell, beginning with 1 for the tree root. The number of two daughter cells is calculated by $c_{mother} * 2$ and $c_{mother} * 2 + 1$, where c_{mother} is the cell preceding the two daughter cells in the tree.
- **Marker**: Denotes if an antibody marker is switched on at this time point. Discrete value out of $[0, 1, 2, 3, 4]$. 0 = no marker is on; 1 = Fcγ (GMP) switches on during cell cycle; 2 Fcγ is on over whole cell cycle; 3 CD150 switches on during cell cycle 4 CD150 is on over whole cell cycle.
- **Generation**: Distance to tree root calculated by $\lfloor log_2(cellnumber) \rfloor$
- **Stopreason**: Reason why tracking of the cell was stopped. Causes are division (value = 1), apoptosis (value = 2), lost while tracked (value = 3) or no reason (value = 0).
- **Inverted Generation**: Amount of divisions that are lying between the particular cell and its first descendant with a marker that is switched on. Detailed definition can be found in section 6.1.1
- **Position X**: Coordinate of the cell's centroid on the x-axis. If no segmentation was possible for a time point, the annotated coordinate was taken.
- **Position Y**: Coordinate of the cell's centroid on the y-axis. If no segmentation was possible for a time point, the annotated coordinate was taken
- **Centroid**: The center coordinates of a region.

4.3. Quantification of PU.1 expression levels

The transcription factor PU.1 is thought to play a major role in lineage commitment of MEPs and GMPs and could serve as an interesting feature in the later fate prediction. In the movies provided by the ISF, information of PU.1 expression was recorded on fluorescence images and already analyzed in detail by Schwarzfischer [47] and Krumsiek [48]. Quantification was done by AMT, as described in section 2.2. Since AMT works semi-automatically, only a 20 out of the 74 tracking trees used in this project provided PU.1 expression levels. It was thus necessary to think of a method that increased the available annotations.

Quantification of PU.1 expression by AMT can be subdivided into two parts. First, uneven illumination of the fluorescence image is corrected to ensure that every pixel has the same

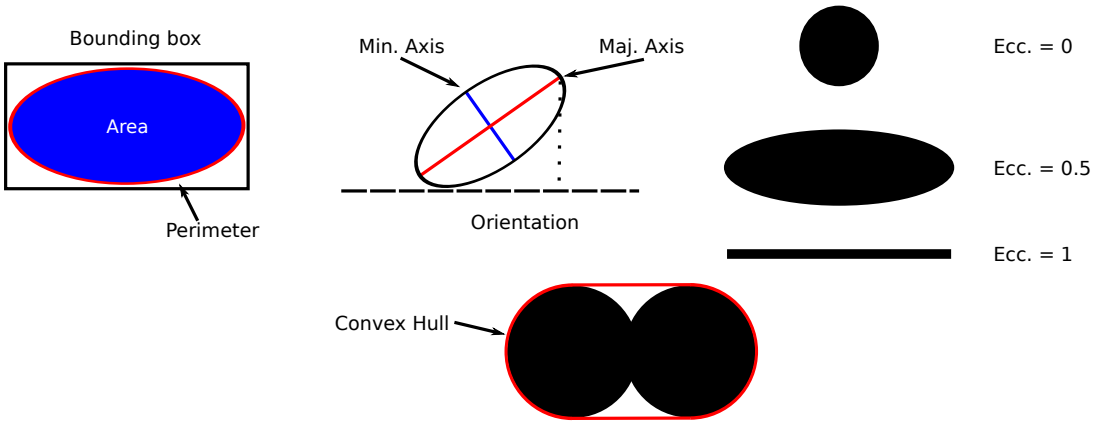


Figure 4.3. – Scheme of features that were measured based on segmented images. A bounding box is a square exactly fitting the region. The area is the amount of pixels in the region, whereas the perimeter is the amount of border pixels of the region. Minor and major axis are computed based on the ellipse fitting the region. Orientation is the angle between the x-axis of the image and the major axis. The eccentricity gives information about a cell's roundness. The smaller eccentricity, the rounder the cell. The convex hull is the smallest convex polygon that can fit the region. This is important especially if regions are undersegmented.

ratio of intensity. This is conducted in completely a automated fashion. Second, the cells have to be segmented as accurate as possible in order to achieve correct measurements. This step needs manual supervision and was replaced by our segmentation method. The cell masks that were identified on brightfield images of the same timepoint as a fluorescence measurement were used to identify the cell in AMT. This approach was not as accurate as the semi-automatic method, nevertheless it allowed quantification of PU.1 expression at large scale and enabled us to use this feature for the prediction of cell fates.

4.4. Identification of cells in segmented subimages

As segmentation was not able to decide which of the identified object represented the tracked cell, feature measurements of all regions were recorded and compared. A control step was established to define the correct region (i.e. the cell). A typical incorrect region are for example beads, which are smaller than cells. Thus, an object was considered a candidate if its area (i.e. size) was larger than the mean plus the standard deviation of a set of 500 manually measured bead sizes, denoted as

$$L_{bead} = \mu_{bead} + 2\sigma_{bead}, \tag{4.1}$$

where $\mu_{bead} = 16.27$ and $\sigma_{bead} = 4.80$ pixels. In addition to that we filtered out regions having a size greater than half the subimage (1250 pixels), since regions of these size are much too large to represent a cell.

Many subimages exhibited two or more cells, thus we selected the object closest to the center of the cell identified in the preceding timepoint. For example, if an image showed two cells shortly after mitosis, the method returned two possible cell candidates, were it was necessary to always choose the cell at roughly the same position as the object that was chosen at the previous timepoint. A formula combining these control steps is written as

$$\{r_i | min(dist(c_i^{(t)}, c_i^{(t-1)})) \wedge c_i^{(t)} > L_{bead}\}, \tag{4.2}$$

where $c_i^{(t)}$ is an object's centroid, $a_i^{(t)}$ represents its area at timepoint t and r_i is one of i objects that were regarded as foreground. In the case that no region of the thresholded image passed the quality control, measurements were repeated based on the binary image without watershedding if oversegmentation was detected. In case of undersegmentation, marker based watershedding was applied.

We combined all steps described above to the final image processing pipeline. Pseudocode notation is shown in algorithm 1. Figure 4.4 illustrates the complete workflow, as well as two examples of problematic segmentation and the approaches to solve or correct them.

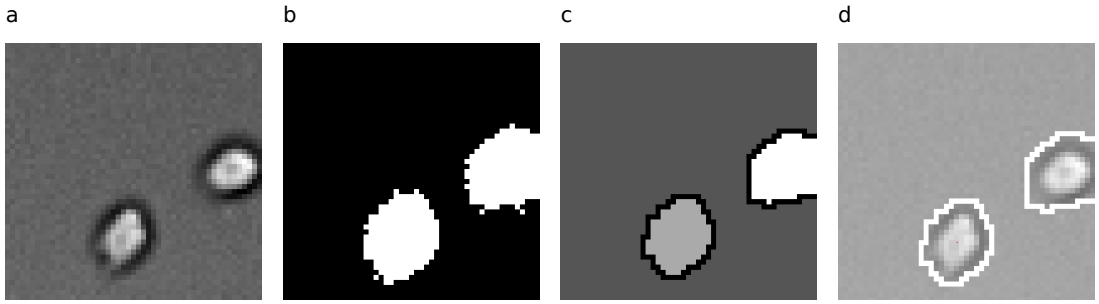


Figure 4.4. – General workflow of image processing algorithms on gray scale brightfield images. a) Raw image of one tracked timepoint. b) Binary image showing the regions of recognized cells with some impurities. c) The binary image is watershedded to separate possibly clumped objects and produces a grayscale image. d) The raw image is superimposed with the outlines of watershedded regions and measurements can be conducted.

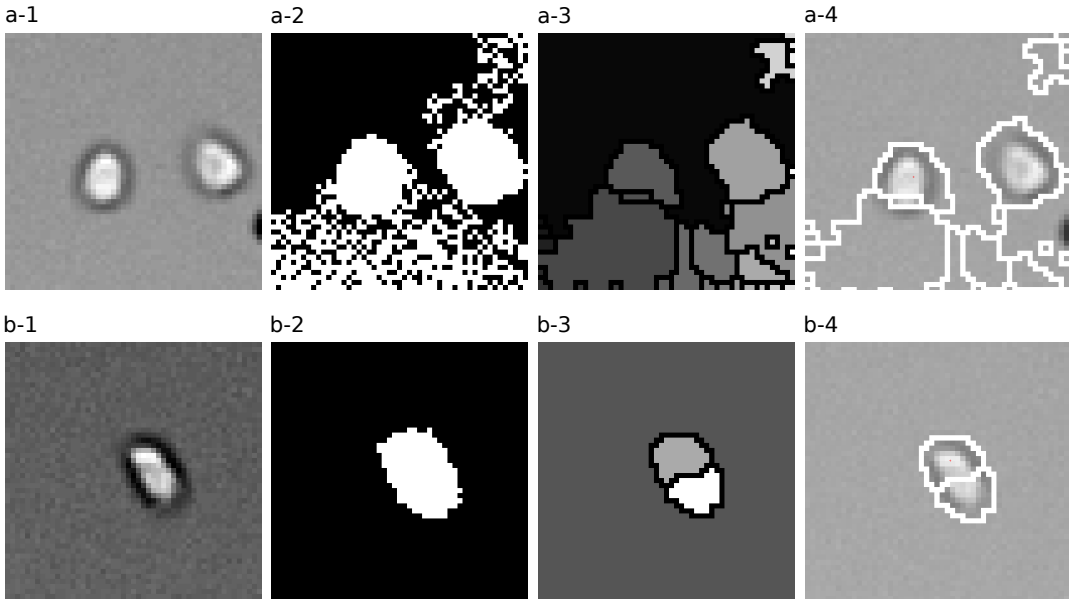


Figure 4.5. – Oversegmentation due to failed distance transformation. Steps 1-4 are in accordance to figure 4.4. a) Mis-segmentation as a result of failed thresholding. In this case it was possible to recover a reasonable cell shape by marker based watershedding. b) Oversegmentation due to two halos. This problem could not be resolved and was later filtered out by comparing the complete time series of measurements.

Algorithm 1: MSER based image processing pipeline.

input : A time lapse movie saved in AMT format
output: Structure θ with features annotated to each timepoint

$T \leftarrow$ all tracking trees
$\Delta \leftarrow$ threshold parameter for MSER
$\theta \leftarrow \emptyset$
$cellFound \leftarrow False$
$status \leftarrow normal$

foreach $t \in T$ **do**
  $I \leftarrow$ all annotated timepoints in t
  foreach $i \in I$ **do**
    $[x, y] = getCoordinates(i)$
    $i_c \leftarrow cropImage(i, x, y, 50, 50)$
    $i_{bw} \leftarrow MSER(i_c, \Delta)$
    while *!cellFound* **do**
      if $status = normal$ **then**
        $i_{ws} \leftarrow watershed(i_c, i_{bw}, 'distance')$
      else if $status = bead$ **then**
        $i_c \leftarrow maskBeads(i_c, i_{bw})$
      else if $status = undersegmented$ **then**
        $i_{ws} \leftarrow watershed(i_c, i_{bw}, 'marker')$ `/* Marker based watershedding */`
      else if $status = oversegmented$ **then**
        $i_{ws} \leftarrow binary(i_{ws})$
      else
        $cellFound \leftarrow True$
      end
      $features \leftarrow regionProps(i_{ws}, i_c)$
      $status \leftarrow checkRegions(features)$ `/* Select region which is most likely a cell or do additional processing */`
    end
    $\theta \leftarrow \theta \cup features$
  end
end

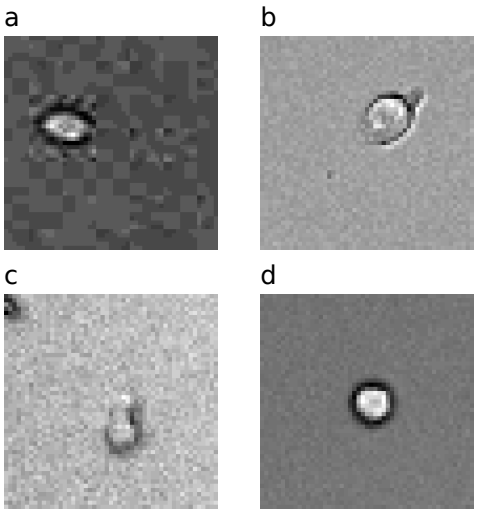


Figure 4.6. – Comparison of image quality. Shown are 50x50 images with one cell (cut-out from large images). a) Sample from experiment 101125PH2. High compression results in huge artifacts. b) Sample from experiment 090901PH2. Cell is uneven illuminated. c) Sample from experiment 110322PH5. Autofocusing failed, resulting in a fuzzy cell that is hard to recognize computationally. d) Sample from experiment 100201PH2. Although jpg compressed, this experiment exhibited the best quality of all movies available. The Cell is clearly separable from the background by MSER thresholding.

4.5. Variation in image quality across experiments

An analysis of the movies after a first run of our pipeline in order to determine segmentation quality revealed great differences. In experiment 101125PH2 many images turned out to be too noisy to identify a cell. Illumination of experiment 100301PH4 was very dark, resulting in many mis- segmentations. Movie 100201PH2 exhibited the best average image quality, showing clearly contrasted cells which were evenly illuminated at most timepoints. Examples for these differences are provided in Figure 4.6. There are two possible explanations for these discrepancies. First, live cell imaging always deals with the problem of best technical results versus best biological results. The continuous illumination of living cells puts them under a large amount of stress, leading to early cell death or a change in behavior. Thus, differentiation events could be biased or completely made unusable if a cell dies early. Therefore, a tradeoff had to be found that produced images good enough to identify single cells in order to track them, yet harming the cells as little as possible. This resulted in brightfield images that were qualitatively rather bad compared to theoretical possible quality. A recently published review by Schroeder concerning long term single cell imaging discusses most of the problems that arise in long term single cell imaging [28]. Since computational analysis of brightfield images was not considered at the time most of the movies were generated, focus was put on best quality of fluorescence images. Addi-

tionally, all brightfield images were saved in compressed jpg format so save storage space. This lead to the great variability in image quality, depending on the degree of compression and experimental setup. Another problem was the tracking process. As cells were moving across the slide during the complete recording time, several problematic events arose:

- Uneven illumination resulted in some positions of the slide that were too dark to identify cells residing in this area. Even if tracking was started on a well illuminated position, some of the daughter cells could move to dark areas and the segmentation of a full tree would become impossible.
- Impurities on the slide grow to large areas if observed under the microscope. If a cell crossed such areas segmentation was not possible for many timepoints, thus rendering the time course unusable.
- Cells could clump together to huge colonies. Even if we were provided by the approximate coordinates of a cell, segmentation was very difficult in these instances.

All of these events had been difficult to predict before manual tracking was started at a certain point, thus our method needed to correct the problems as well as possible.

4.6. Generating the dataset for analysis and computational prediction

We were now able to perform large scale measurements on complete tracking trees. To reduce computation time, tracking trees with less than three annotated cells or less than 100 timepoints were excluded from the dataset, as these trees covered no useful information. The completed image processing pipeline was applied to all images of 74 trees out of the three movies mentioned in section 3.2. Table 4.1 covers detailed information about the segmented dataset. Processing of the movies took around 5 hours, if 20 trees were

Movie	**Trees**	**HSCs**	**MEPs**	**GMPs**	**Images**
100201PH2	34	628	165	95	433386
100301PH4	20	679	44	197	424144
101125PH2	20	278	39	46	213640
Sum	74	1585	248	338	1071170

Table 4.1. – Overview of experiments, tracked differentiation trees, as well as cell types and amount of images that were processed by our pipeline.

computed in parallel on a high performance computing cluster. On a Laptop (dual-core,

2.80 Ghz, 4gb Ram, 100mbit lan connection) calculations ran about 40 minutes for a tree with 6000 timepoints. The limiting factor in this approach is to load images from a networks storage space at every iteration, due to the high load more than 20 trees were not processable in parallel. This bottle-neck can be improved in future versions.

4.6.1. Data storage

We split up the continuous measurements of a complete tree into time courses of single cells, yet conserving all information that was needed to reconstruct the tree topology, resulting in a matrix $\mathbf{X}$, denoted as

$$\mathbf{X} \in \mathbb{R}^{C \times F \times L}, \tag{4.3}$$

that is $C = 1970$ the number of cells, $F = 15$ the amount of features and $L = 4886$ the lifetime of the longest time course. Supporting information was stored in additional structures. Based on this dataset we assessed the average segmentation quality of our method.

4.6.2. Manual quality control

First, we checked if cells were measured with a precision that made subsequent analysis possible. We conducted this step by examining time courses of the cells area (i.e. size), as measurements of this feature are intuitively understandable and comparable across cells. All hematopoietic stem cells and their progenitors are underlying a biological cycle, controlling cell growth and other behavior, until undergoing mitosis. As intervals of 90 seconds between each measurement gave a continuous picture of the process, the resulting time-course should lie in a certain confidence interval. A biological cell is, for example, not able to grow to 200 percent of its size in two adjacent timepoints, thus an erroneous measurement is likely at these points. Figure 4.7 shows samples of three cells that were segmented in different qualities and the respective PU.1 quantification.

The high proportion of completely mis-measured cells made it necessary to introduce a manual filtering step. We examined every cell and decided if precision of the measurements was high enough to take this cell into account in further studies. The resulting dataset of 1217 cells represented the basis of our subsequent work. Figure 4.8 shows the amount of segmented cells per movie and cell type. $\mathbf{X}$ was now of size $1217 \times 15 \times 4886$. We were now able to conduct first analyses in order to find differential behavior of individual cells during the differentiation process.

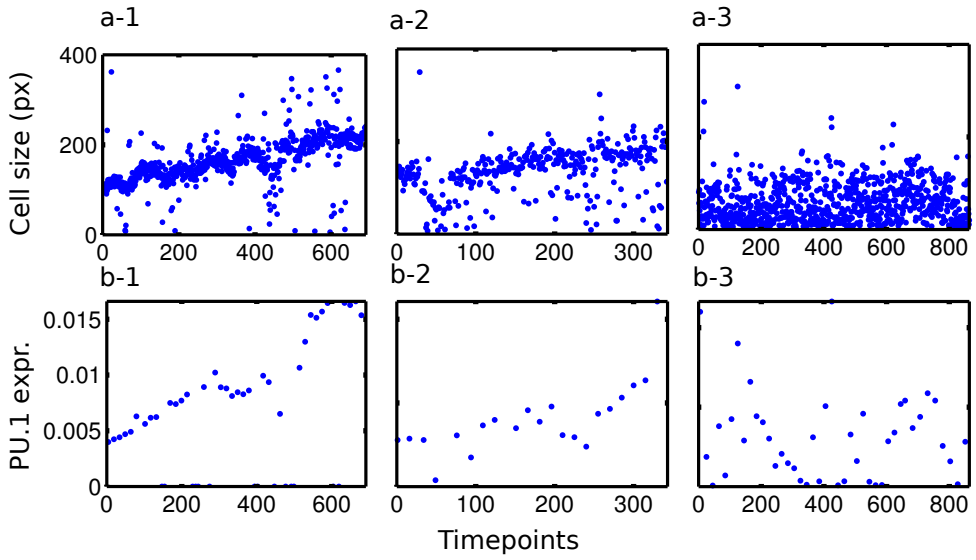


Figure 4.7. – Comparison of segmentation of different movies. First row: Measurements of cell size (in pixels) for three different cells. a-1) Measurements were correct in most parts of the time course, resulting in a very accurate representation of the growth rate during the cell's life cycle. a-2) More erroneous measurements are observed, yet cell growth is still restorable. a-3) Mis-segmentations of nearly all timepoints. It can be assumed that the selected areas do not represent the cell and are thus not usable. Second row: PU.1 expression levels for same cells. b-1,b-2) Measurements resulted in reasonable PU.1 expression. b-3) Information is not usable.

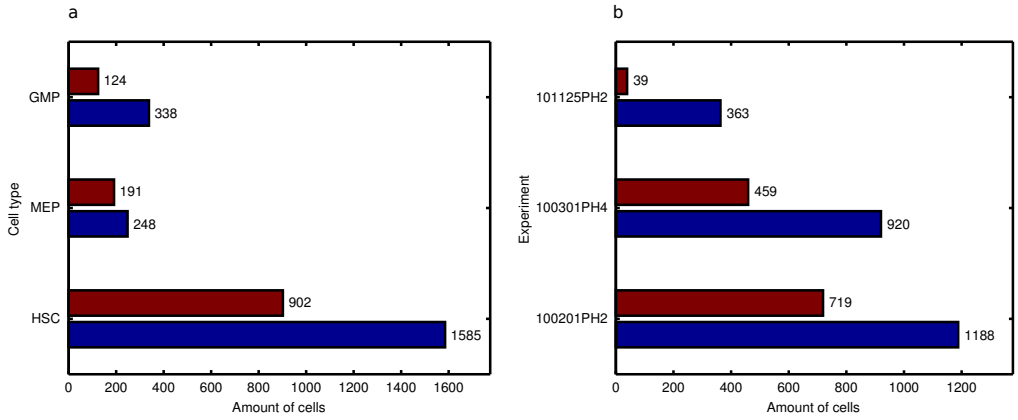


Figure 4.8. – Dataset of cells that were extracted out of three movies. a) Amount of cells per annotated cell type before (blue) and after (red) manual quality control of all movies combined. b) Amount of cells per movie before (blue) and after (red) manual quality control relabeling HSCs as described in inverted generation. Differences in image quality are huge, as 87% of all cells in experiment 100201PH2 were kept after manual selection, whereas in experiment 101125PH2 only 11% of the cells remained.

4.7. Conclusion

Measuring the morphological behavior of cells based on single-cell time lapse movies turned out to be a challenging but feasible task. In this chapter, we documented the establishment of a novel image processing pipeline that is capable of automatically processing thousands of images in a few hours, generating time courses for 14 morphological features and PU.1 expression of differentiating hematopoietic stem cells. Due to the short imaging intervals, a very detailed view of a cell's life cycle can be achieved, including interesting events that could occur and possibly provide information that will enhance our understanding of factors driving an HSC's lineage decision.

The bad performance of Otsu's method showed that this algorithm is not applicable for brightfield images. Most of the subimages exhibited a unimodal histogram, were it was not possible to separate foreground from background pixels only by taking this information into account. In contrast, the different methodology of MSER turned out to perform well on the unimodal distributed images.

The method is able to process manually tracked differentiation trees at parallel and the used file formats can easily be used in tools like AMT or TTT for further analysis. Slight modifications could also enable the processing of continuous time lapse movies that were conducted under different experimental conditions or with cells exhibiting a more complex morphology than hematopoietic stem cells. The algorithm is able to correct for many errors that arise in life cell imaging, such as clumped cells, contamination in the medium or poor contrasted cells.

However, our analyses suggested that achievable results could be improved, if the quality of recorded brightfield images could be increased. This method was developed based on images that were produced under the specification that cells only needed to be roughly identifiable in order to allow manual tracking. The resulting time courses of single cells showed many erroneous measurements which lead to imprecise representations of a cell's behavior. As discussed earlier, image quality comes always at the cost of cell health and one should not anticipate perfect images to work with. Nevertheless, time lapse experiments that were conducted under the considerations that brightfield images are optimized for computational processing could improve the output of this method tremendously.

In future versions of this method, an algorithm should be developed that is able to decide if the the object of a binary image that is regarded as the cell is a true positive or not. This would render the manual analysis of time courses that was conducted here unnecessary. A first approach compared the mean of 10 preceding measurements to the actual object properties and allowed a new object only if the value resided between two standard deviations between or above the mean value. This method was too restrictive and was thus not used in the final algorithm. Another improvement would be the application of several different segmentation approaches to find a cells outline. In image processing, a huge amount of possible events could lead to errors in cell recognition and it will not be possible to develop a pipeline that works best on all images. Thus, the result of the method

that performed best on a particular image could be chosen, further improving quality of measurements.

Nevertheless, our method produced a dataset with highly detailed time courses showing morphological features of differentiating cells, which we now analyzed in order to find interesting behaviors of cells committed to different lineages.

5. Postprocessing and analysis of cell behavior

In this chapter, we evaluate the quality of the measured time courses generated by image processing and the level of detail that was achieved. After that, we compare sets of cells commited to different lineages and discuss different behavior over their lifetime. In a last step, we analyze cell movement in detail. This chapter represents step four in Figure 1.3.

5.1. Oscillating cell growth

After manual assessment of the generated time courses as described in section 4.6, we started to analyze the data. Our first finding were periods of growth and decline of cell size in all 719 time courses out of experiment 100201PH2. The periods showed intervals of roughly 1.6 hours and were not expected, since a differentiating cell should grow nearly constantly, or at least retain its size for a certain time span during its life cycle. In order to find out whether this behavior could be of biological interest or an experimental artifact, we superimposed the growth over time of two sister cells. In addition, we chose a cell of a different tracking tree that was recorded in the same time interval, but exhibited a later birth time. The time courses were then analyzed by cross-correlation as described in section 2.3.1, as well as visual examination by plotting.

As depicted in Figure 5.1, the periods showed high correlation, indicating that regardless of a cell's birth time, oscillations were synchronous over the measurement. This was evidence that the behavior was caused by experimental conditions, as there is no reasonable biological explanation why cells of different age at completely different positions should show the same behavior at the same timepoints. So far, these findings could not be verified on time courses of cells that were recorded in other experiments. However, as the qualities of obtained measurements heavily decreased in experiments 100301PH4 and 101125PH2, it could be possible that this behavior was not identified due to too much noise in the data.

Oscillating cell sizes are important for the normalization calculations that were done on fluorescence images in order to measure the PU.1 expression level (see section 2.2). Due to larger time spans between the recording of an image, the periodic growth could be missed and drops in intensity would be regarded as biological relevant. Thus, if curves of PU.1 expression and cell growth would correlate, it would be crucial to correct this effect in order to achieve correct PU.1 quantification. To date we were not able to draw this

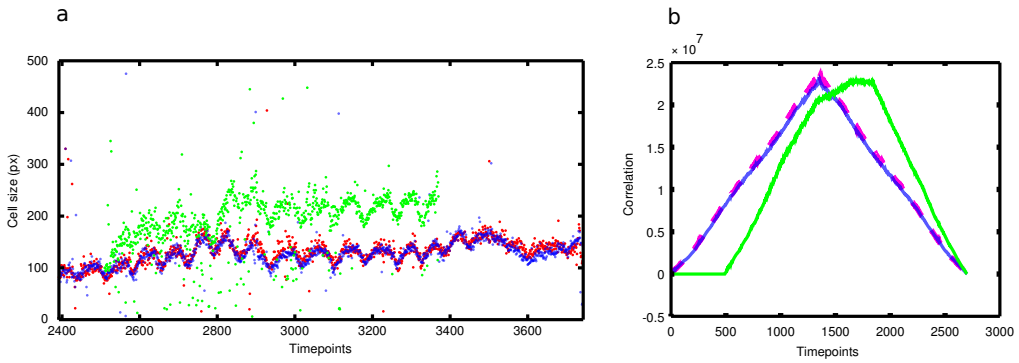


Figure 5.1. – Oszillation in cell growth over time. a) Time courses of three cells recorded in experiment 100201PH2. Red and Blue: Sister cells in the same differentiation tree. Those cells resided at the same position of the ibidi slide. Green: Cell of a completely different position that lived around the same time. Clear periods of cell growth are occurring at intervals of ∼1.6 hours. b) Cross-correlation of 3 cells. Red: Autocorrelation of one cell. Blue: Cross-correlation of sister-cells. The red and blue curve are completely overlapping, thus cell growth oscillated at the same frequency. Green: Cell 1 cross-correlated with cell 3. Especially the second half of the cell cycle shows close correlation.

conclusion. First tests have shown that oscillating cell growth does not have a huge effect on the fluorescence intensity, yet further investigation of oscillating cell growth is necessary to solve this experimental issue. A possible explanation for the observed behavior would be fluctuating temperature levels in the microscope's incubator case or phases were cells were under-supplied with the gasses that should circulate through the slide continuously.

In this work, the oscillating signal was regarded as technical noise and thus corrected in the further postprocessing steps, as described in section 5.4.

5.2. Normalization

Due to varying image quality across experiments, measurements of the same feature spanned a different scale for each movie. These differences could be caused by different settings of camera zoom, illumination or focussation, to just name a few of the possible influences. For example, the average cell size of all measured timepoints of experiment 100301PH4 was around 88 pixels, whereas experiment 100201PH2 yielded a mean cell size of around 149 pixels. In addition, different illumination levels were observed by comparing averaged values of mean brightfield pixel intensities of experiments 100201PH2 ($\mu = 177$) and 100301PH4 ($\mu = 37$). The range of brightfield pixel intensities is 0 to 255. We corrected these influences by following procedures.

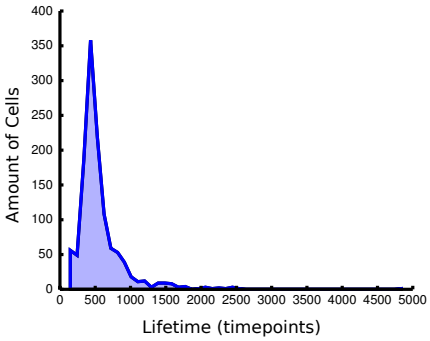


Figure 5.2. – Histogram of cell lifetimes of 1217 cells in the dataset. Mean was 552 timepoints. A few cells exhibited much longer life times, however these cells were mostly starting cells hat were removed from the dataset in a later step.

5.2.1. Normalizing lifetimes

The recorded hematopoietic stem cells exhibited different lifetimes with a mean of 552 timepoints, resulting in a varying amount of points of measurements for each cell and feature. In addition to that most time courses contained missing values, a result of failed segmentation on single image patches. As supervised classification as well as methods for functional representation of the time courses needed equally spaced intervals of measurements without missing points, we normalized each time course to a length of 500 points of measurement and linearly interpolated missing values, as described in section 2.3.2. Dimension L, representing the timepoints in data matrix $\mathbf{X}$ as defined in section 4.6.1 was now of size $L = 500$. The lifetime for each cell was saved separately to retain this information for machine learning.

We are aware that this solution is a very naive approach and alternative methods to achieve comparability of cells in a more sophisticated way will be discussed in section 5.6. Nevertheless, hematopoietic stem and progenitor cells should follow roughly the same cell cycle during their lifetime, which allows this approach. The method has also been used by Schwarzfischer [47] as well as Krumsiek [48].

5.2.2. Different scales across movies

In order to make cells comparable across movies, we normalized all time courses for each feature to the same scale. This step was skipped for movement and orientation, since these features were not affected by different sizes of segmented areas. In addition, PU.1-eYFP intensity was excluded since normalization had been applied by AMT already. Normal-

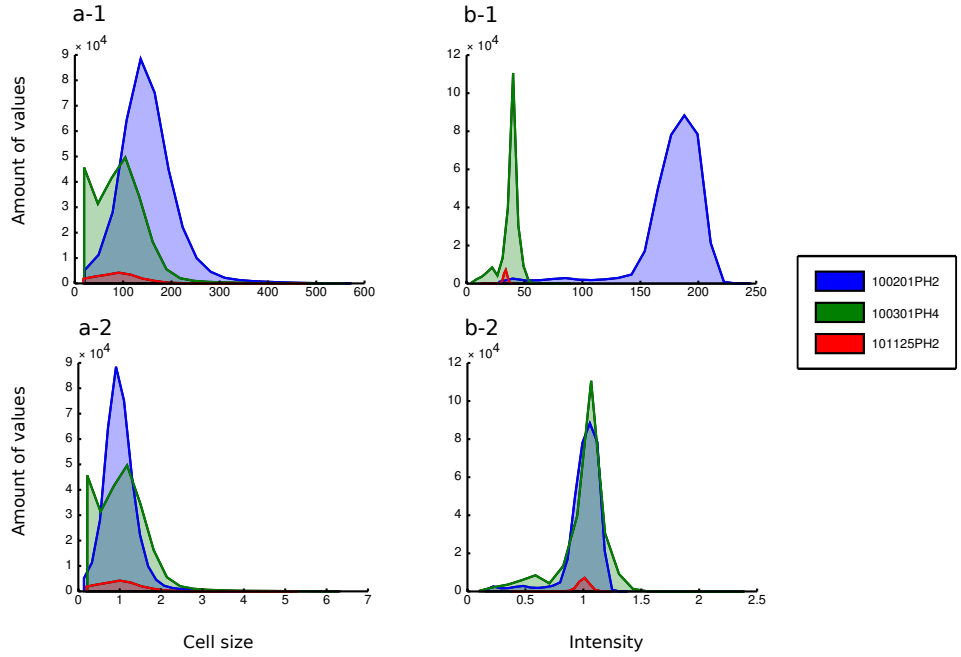


Figure 5.3. – Histogram plots of cell size and mean brightfield pixel intensities per movie. a-1) Cell sizes before normalization. b-1) Mean intensities per cell before normalizaton. a-2) Cells sizes after normalization. b-2) Mean pixel intensities for all cells in brightfield images after normalization.

ization was performed by dividing all measured timepoints of a particular feature by their overall mean of each movie, denoted as

$$\mu_f = C^{-1} \sum_{c=1}^{C} (L^{-1} \sum_{l=1}^{L} x_{c,f,l}), x_{c,f,l} \in \mathbf{X} \in \mathbb{R}^{C \times F \times L}, \tag{5.1}$$

that is μ_f the overall mean of all time courses of the f-th feature in $\mathbf{X}$. The normalized time course was then calculated by

$$\forall x \in \mathbf{X} : x'_{c,f,l} = \frac{x_{c,f,l}}{\mu_f}. \tag{5.2}$$

Histograms visualizing distributions of cell sizes and mean pixel intensities before and after normalization are shown in Figure 5.3. The last step before representing all cells functionally was to equalize length of each time course.

5.3. Comparing the growth ratio to previous findings

One possibility to evaluate the quality of our measurements was to compare the mean growth ratio of the cells with the one that was stated by Schwarzfischer [47]. In his

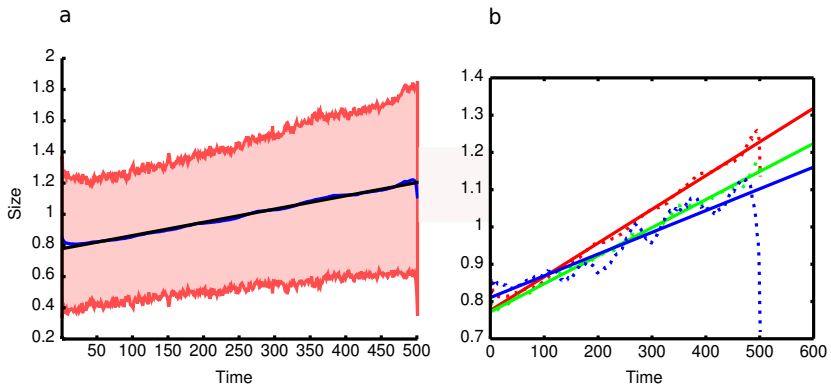


Figure 5.4. – Cell growth for all measured time courses. a) Average cell size over normalized lifetime with standard deviation. Red lines: standard deviation; Blue line: mean curve; Black line: linear fit, with a growth ratio of 1.55. b) Growth rates of different cell types. Dotted lines represent moving average curves, continuous lines represent linear fits. Red: All cells with no marker onset (Ratio: 1.59). Blue: Cells with annotated cell type GMP (Ratio: 1.34). Green: Cells with annotated cell type MEP (Ratio: 1.48).

thesis, time courses of manually segmented fluorescence images were normalized to same length and then superimposed. A window of 15 timepoints was then moved across the measurements, generating a mean value of each window. This method has the advantage that it does not overestimate single outliers in the mean curve. Then, the linear growth ratio gr was approximated by linear fitting. The function value of the latest timepoint was divided by the earliest timepoint, denoted as

$$gr = \frac{f(x_{end})}{f(x_1)}, \tag{5.3}$$

where $f(x)$ is the fitted linear function and x are values of the time course. This resulted in an averaged growth ratio of ∼1.59. As shown in Figure 5.4, we were able to roughly confirm this ratio by applying the method with parameters equally set on our data, calculating a growth ratio of ∼1.55. In addition, we analyzed the growth rates of undecided cells (1.59), as well as cells annotated as MEPs (1.48) and GMPs (1.34).

This comparison indicated that the data reflected a true behavior of biological cells, as the moving average shows constant growth and the resulting ratio roughly agrees with manually measured cell sizes. However, the automatic pipeline is not as accurate as the semi-automatic approach used by Schwarzfischer and the quality of measurements will be improved in future versions.

5.4. Functional data analysis of cell properties

To account for mis-segmented images and observed oscillation in cell growth it was necessary to find a method that was able to correct the time courses. For example, if segmentation of a single cell failed in more than 10 adjacent timepoints, a correct representation of its behavior in this timespan was not achievable without approximation. To cover these problems, we decided to apply functional data analysis, an approach that approximates each time course as a function, which is represented by a system of basis functions. Detailed descriptions about FDA can be found in section 2.3.3.

Here, we used a b-spline system of order 4 consisting of 50 basis functions. As time courses were normalized to a length of 500 timepoints, each basis function represented ∼10% of the timepoints, thus allowing sufficient flexibility of the fit. A possible value for the smoothing parameter λ was computed by generalized cross-validation (gcv), which resulted in a value of 10^3. The fits where then smoothed in accordance to λ. It turned out that this did not achieve the desired results, as oscillations were still represented by the approximation. We thus empirically optimized λ based on gcv results and a set of sample time courses, according to following considerations:

1. The fit must not represent oscillations in cell growth.
2. Outliers should change a curves shape as little as possible.
3. Larger intervals of mis-measurements should be approximated without changing the curve's shape locally
4. The fit should still retain global variances in the time series (i.e. it should not approximate the time course linearly)

A λ that was set to 10^7 produced best approximation results. FDA was applied to the data for each feature separately. The functional representation of time series and the effect of different smoothing factors is exemplified in Figure 5.5.

After the dataset was normalized and errors were corrected, we were able to examine the behavior of cells over time for each feature. Since we were going to use all measured features in machine learning, we did a first check here if cells commited to either the erytroid or myeloid lineage were exhibiting different behavior. Time courses of 124 GMPs and 189 MEPs (i.e. cells with annotated marker onset event) were plotted per feature, as shown in Figure 5.7. In addition, mean curves and curves for positive and negative standard deviation were plotted for both types.

Both mean curves per feature followed the same global trend, for example cell size rises constantly and movement decreases in later timepoints for both cell types. However, we found that roundness, mean brightfield intensity and cell movement exhibited differences throughout the whole time course, indicating that MEPs seem to be rounder, less bright and are moving not as much as GMPs. Movement seemed to get equal in later phases of the time course. These findings demonstrated that GMPs and MEPs differed at least in these features in a way that should made it possible to apply machine learning methods.

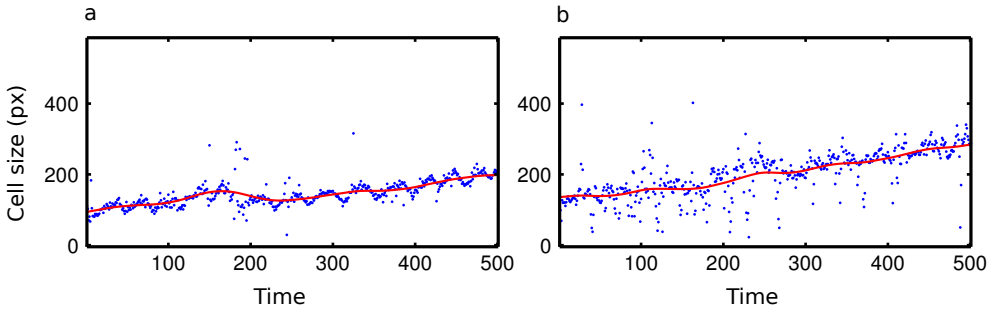


Figure 5.5. – Example of time series represented by a b-spline with 50 basis functions and λ set to $1e^{-5}$ (optimized by general cross validation). a) Oscillating cell growth is smoothed. b) Erroneous measurements are corrected.

In addition, all time courses reflected true biological behavior, such as lower eccentricity (i.e. higher roundness) and decrease of movement at beginning and ends of time courses. The findings were confirmed by our experimental collaborators which examined single cells in the movies.

In some features such as PU.1-eYFP intensity and solidity, there were clear outlier curves identifiable. Detailed analysis revealed that problematic linear interpolation caused this behavior. If a time course exhibited missing values at its beginning or end, the interpolation produced values that were out of range. The same was true for long periods of missing values in PU.1-eYFP intensity.

Detailed analysis of orientation revealed that functional representation of this feature was not applicable. Time courses of this feature followed no clear trend and examination of the distribution of values across all measurements revealed a mean of 0 degrees and a standard deviation of 45 degrees. A histogram of the distribution as well as a q-q plot comparing orientation values with a vector of randomly drawn values is shown in Figure 5.6. We thus decided to remove this feature from the set of candidates for predictor variables.

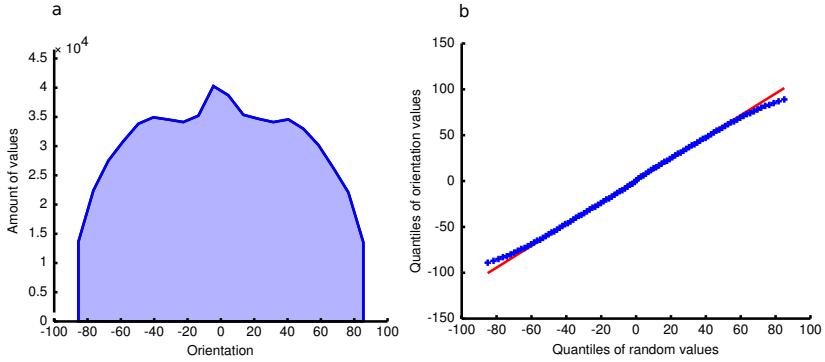
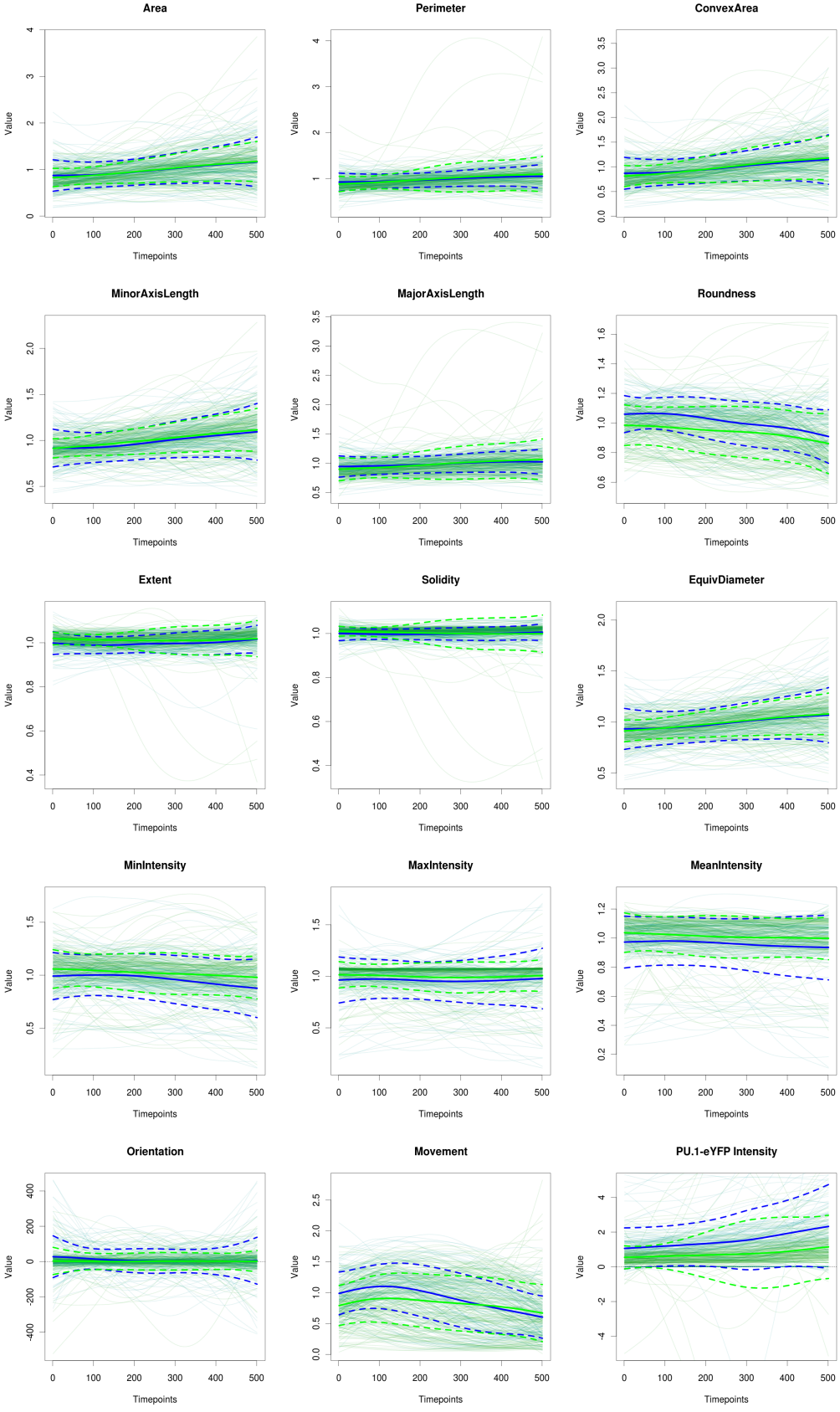


Figure 5.6. – Analysis of cell orientation. a) Histogram of all values of orientation. There is no clear peak identifiable, standard deviations are very high. b) Q-Q plot of quantiles of orientation values versus a random vector in the same interval $[-90; 90]$. Distributions are nearly equal, thus a random process for orientation can be assumed.

Figure 5.7. ***(following page)*** – Functional representation of normalized behavior of 124 GMPs and 189 MEPs. Light lines in the background represent different cell types, solid lines are mean curves and dashed lines show standard deviations. Green: All cells in the set that were annotated as MEP. Blue: All cells in the set that were annotated as GMP. Trends in the curves is in accordance to biological assumptions. Some features, such as roundness, movement and mean brightfield intensity exhibited differences in the cells. Outlier curves, for example in PU.1-eYFP intensity, arose due to errors in linear interpolation.


Area
Perimeter
ConvexArea
MinorAxisLength
MajorAxisLength
Roundness
Extent
Solidity
EquivDiameter
MinIntensity
MaxIntensity
MeanIntensity
Orientation
Movement
PU.1-eYFP Intensity
Value
Timepoints

5.5. Elucidating the factors driving cell movement

The differences in cell movement of hematopoietic stem cells commited to the erytroid or myeloid lineage that were discovered after applying FDA brought up the question, whether we could propose a model representing the factors driving cell motility. As discussed before, the exact timepoint and the mechanisms playing a role in lineage commitment are not known to date [15]. A progenitor cell must be able to produce two daughter cells that adopt different fates, a process that is also called asymmetric cell division. Heterogeneous daughter cells can theoretically arise by uneven distribution of determinants upon cell division, i.e. due to intrinsic factors, or become different upon subsequent exposure to environmental signals, i.e. due to extrinsic factors. In the extrinsic case, direct contact with cellular determinants in the niche would be necessary and has already been shown to play an essential role in fate decision [64]. Furthermore, it was stated that direct communication between hematopoietic progenitor cells and osteoblasts provides essential cues for their proliferation and survival [65]. In addition, it is still not completely understood how HSCs that are implanted in a patients body are able to find the way back to their niche in the bone marrow, a process that is called cell homing [66]. All these examples motivated studies of hematopoietic progenitor cell movement, as it could provide interesting information to understand at least a part of the complex network of factors that have an influence on cell heterogeneity.

We computed the pairwise displacement for all cells as described in section 2.3.4. The resulting dataset was comprised of 290000 values per coordinate vector and it is important to note here that the calculations that were conducted in this section were done before the dataset was normalized in order to retain correct values for cell displacement per timepoint. We were now able to assign the most likely model for the movement behavior we observed in our data.

5.5.1. Brownian motion

The perhaps most simplified assumption for displacement of an object is brownian motion, that is, the random movement of particles suspended in fluid [67]. If the model would fit to our data, this would imply that the movement of hematopoietic stem cells *in vitro* is following a random process and is not driven by either intrinsic or extrinsic factors.

In order to simulate brownian motion and compare the behavior of cells in our dataset with the model, we used a biophysical lab protocol created at the university of Berkeley [68]. Here, it was stated that one-dimensional brownian motion is composed of a sequence of normally distributed random displacements. In addition, probabilities for a particle to move in a certain direction are equal, even if the model is extended to two dimensions. This

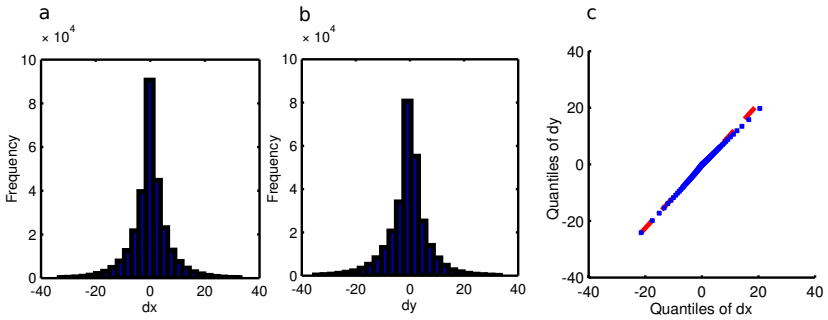


Figure 5.8. – Analysis of distributions for one-dimensional pairwise displacements of all cells in experiment 100201PH2. a) Histogram of dx. b) Histogram of dy. c) Q-Q plot of dx against dy. Slight deviations at high valued displacements are possibly due to measurement errors.

implied that the values of displacement of both coordinates should be equally distributed. Displacement is denoted as

$$\begin{aligned} d_x^{(t)} &= x^{(t)} - x^{(t+1)}, \\ d_y^{(t)} &= y^{(t)} - y^{(t+1)}, \end{aligned} \tag{5.4}$$

that is $d_x^{(0)}, d_y^{(0)}$ were set to 0. Histograms and a q-q plot (see 2.3.4 for an explanation) of both distributions confirmed this hypothesis, as shown in Figure 5.8. Deviations that are observed in the upper right quarter of the q-q plot (e.g. regions of high movement values) are most likely due to measurement errors.

As both coordinates were equally distributed, it was sufficient to compare only d_x with a vector of randomly sampled normally distributed values with the same length as d_x. Figure 5.9 exemplifies the movement of two-dimensional brownian motion, as well as the histogram of a one-dimensional vector and a q-q plot comparing it with d_x. There is clearly no diagonal line observable in the plot, thus the hypothesis that movement of hematopoietic stem cells is following brownian motion was rejected.

5.5.2. Lévy flight

Literature research revealed a modification of brownian motion, were a particle is moving randomly, but intercepted by directed flights. The lévy flight is a random walk in which the step-lengths are distributed according to a power law of the form $y = x^{-\alpha}$ where $1 < \alpha < 3$ [69]. This process is already widely accepted as a model for food search behavior of large animals such as sharks and other ocean predators as well as monkeys [70; 71]. We hypothesized that cells could show a similar behavior during their search for nutrients in the medium or while trying to get contact with other cells. Potdar et al. [72] and Reynolds [73] already suggested that cell motility could follow lévy flights, however

Potdar eventually published a more complex model called bimodal correlated random walk (BCRW), which will not be discussed in detail here. As lévy flights are power-law distributed, it was, it was adequate to compare our data with a so-called Pareto q-q plot [74]. The log-scaled absolute valued dataset is plotted against a vector of exponential distributed values with mean equal to one. After analyzing the plot we also could not confirm that cell movement follows a lévy-distribution. Figure 5.10 shows an example lévy walk in two-dimensional space, the histogram of the power-law distributed vector and the q-q plot against our data.

5.5.3. Laplace distribution

As we were not able to link the distribution of hematopoietic stem cell displacements to an existing model, we sampled vectors of random values according to several distributions and compared them with our data. A laplace distributed (see section 2.3.4 for details) vector returned best results, as shown in Figure 5.11. The laplace distribution is also sometimes called the double exponential distribution, because it can be thought of as two exponential distributions (with an additional location parameter) spliced together back-to-back [75]. Thus, using absolute values of d_x the q-q plot against an exponential distributed vector yielded satisfying results (data not shown). In the given time to write this thesis we were not able to assign our findings to an existing model or could explain the factors that are leading to this behavior of movement. Further analysis of the data will be necessary, especially other models like BCRW should be tested.

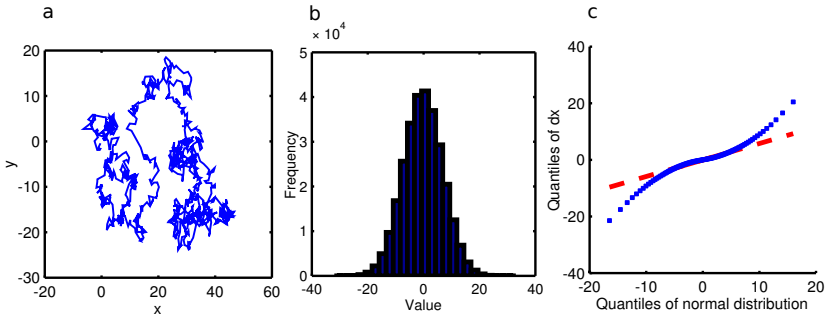


Figure 5.9. – Comparison of movement of dx against brownian motion. a) sample path of a simulated particle following brownian motion in two dimensions. b) Histogram of a normal distributed random process simulating one-dimensional brownian motion. c) Q-Q plot of the normal distributed vector against dx. Deviations indicate that cell movement is not following brownian motion.

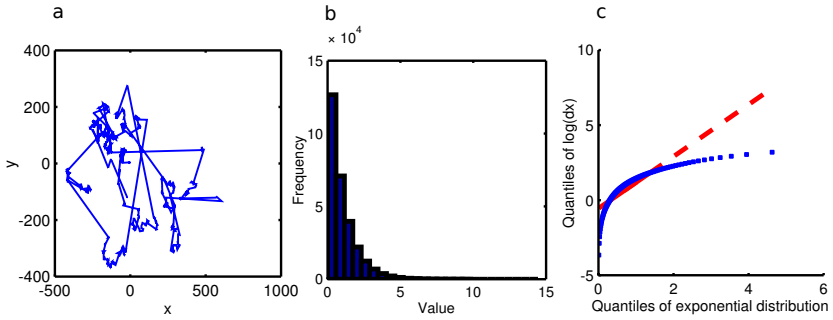


Figure 5.10. – Comparison of movement of dx against a simulated lévy walk. a) Example for two-dimensional lévy walk. b) Histogram of the power-law distributed vector. c) Q-Q plot of $log(abs(d_x))$ against exponential distributed vector with mean equal to 1. The distributions clearly differ, thus dx is not behaving according to a lévy walk.

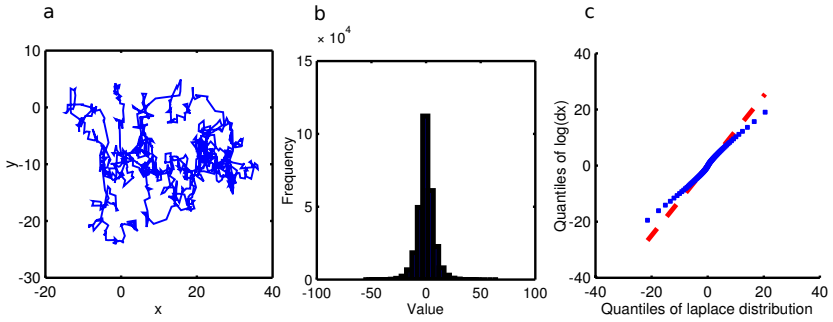


Figure 5.11. – Comparison of movement of dx against movement were both vectors are laplace distributed. a) Example for two-dimensional displacement. b) Histogram of the laplace-distributed vector. c) Q-Q plot of d_x against laplacian distributed vector. Distributions showed the most accordance, thus cell movement could be laplacian (and thus exponential) distributed.

5.6. Conclusion

Analysis of measured time courses revealed that image processing produced results that reflected true cell behavior. The image processing produces measurements with a level of detail that allowed us to see unexpected oscillations in cell growth, which are not observable by eye. This finding and differing image quality across the experiments made it necessary to normalize differences in the experiments as well as segmentation errors of our pipeline. The use of linear interpolation to resolve missing values and to normalize cell lifetimes sometimes led to huge errors, resulting in time courses that were clear outliers if compared to the remaining dataset. In addition, the normalization method used here is probably too restrictive and could destroy variances between the cell types.

Functional representation of the time courses and thus correction of errors was successful. Yet, this step had to be optimized manually, since automatic methods like generalized cross validation to determine the smoothing factor λ are approximating oscillations in the measurements which are technical noise. If experiments with better image quality and thus less erroneous segmentation become available, image processing will produce less errors and thus functional approximation should be achieved automatically and with better results. In addition, it should be possible to replace normalization and linear approximation completely by functional representation. This would remove two possible error sources and will be conducted in subsequent steps after this thesis.

The comparison of behavior of different cell types revealed that there are differences in some morphological features and that it should be possible to achieve reasonable classification results. A plot showing separate generations of cells commited to different lineages could reveal even more information and will be done in subsequent steps after this thesis. Literature implicates that even if cells are not differing in their final state (i.e. GMP or MEP), it could be possible that this is the case in earlier generations, for example exactly before lineage decision [41].

In a last step in this chapter, we demonstrated that cell movement is not following brownian motion or a lévy walk, however it was not possible to propose a possible model for the found laplace distribution in the given time. Since elucidating cell movement is a field of rapidly growing interest, we will put more effort in this task.

All together, analysis of time courses revealed interesting insights in the behavior of GMPs and MEP. Even in single features we could demonstrate that there are differences in the cells. Thus, the efforts in visualizing cell behavior should be intensified and the results should be discussed with our experimental collaborators. This could allow us to identify little changes in cell behavior that indicate the timepoint of lineage decision.

6. Prediction of hematopoietic cell fates

The last part in this project was covered by the establishment of a model able to predict a cell's lineage decision as early as possible in the differentiation tree, using the time courses generated in section 4. Postprocessing in section 5 resulted in a dataset of 1217 cells with time courses of 13 features (orientation was discarded in the preceding section), each with a normalized length of 500 timepoints. This chapter covers steps five and six of the pipeline shown in Figure 1.3.

6.1. Definition of class labels and variables

The huge amount of data and the high number of possible relationships across features yielded a high dimensional space that was not analyzable without strong computational help. Let $R = 10$ be the number of samples, $S = 5$ the number of predictor variables in a dataset. The possible amount of combinations of samples is then $\sum_{i=1}^{R} \binom{10}{i}$, as well as $\sum_{i=1}^{S} \binom{5}{i}$ predictor variables, respectively. This results in 31713 possible combinations of samples and features, a number that is exponentially growing.

In order to manage this massive computational task, we utilized the methodology of machine learning and data mining, whose capabilities have already been exemplified in section 1.4. However, many parameters and possibilities on how to train the model were left to set and will be discussed in detail in the following sections. All methods that were used to evaluate the classifier performance are defined in section 2.4.3.

Before we were able to start with classifier training, it was necessary to build a data structure that was compatible with classification methods. In general, these datasets are arranged in a two-dimensional matrix $\mathbf{M} \in \mathbb{R}^{S \times F}$, where S denotes the amount of samples in the dataset and F represents the number of variables that are utilized by the classifier. An additional vector $\mathbf{c}$ of length S covers the class labels for each sample, which were in our case GMP and MEP. The labeling of all samples is crucial for supervised classification, as a classifier is trained and evaluated to discriminate the provided classes by learning rules on the predictor variables.

6.1.1. Inverted generation

The tracked differentiation trees provided two reliable class labels, represented by the annotated fluorescence markers that indicated if a cell was a MEP or GMP. 25% of all

cells in the dataset were labeled in this way, thus it was necessary to establish a method that allowed to label all cells with unknown type. Most of the tracking trees exhibited either MEP or GMP marker onsets in later generations, but not both. This was only the case in three tracking trees (i.e. four percent of all trees). In addition, antibody markers are not switching on immediately after lineage decision, since the surface proteins that are needed for antibodies to bind are emerging in later generations. Thus, a predecessor of cells which are reliably annotated as either MEP or GMP could already be commited to the lineage. We used this assumption to define the inverted generation.

For each cell, we examined its progeny from top to bottom. The first onset event of one of the lineage markers, giving clear evidence that this daughter was commited to MEP or GMP lineage was then annotated to the originating cell. In addition, the number of division events between the examined cell and the progeny with a marker onset event was calculated and saved as inverted generation. The more division events occurred between a cell and its labeled progenitor, the lower its inverted generation. A cell with annotated marker onset was labeled with inverted generation 0, its predecessor with inverted generation -1, and so on. If a cell produced no progeny with an annotated marker onset, its inverted generation was left blank and its type was set to MPP. A special case was defined for all root cells of a differentiation tree. These cells were identified as HSCs by flow cytometry sorting and were thus labeled accordingly. The inverted generation of these cells was also not annotated.

The trackings allowed the annotation of six inverted generations and three cell types. An example for this method is shown in Figure 6.1 a).

6.1.2. Selection of samples and correlation of features

We removed all cells labeled as HSC or MPP from the set, since we were only interested in cells that could possibly be commited to either the erytroid or myeloid lineage. The dataset was now comprised of 576 cells out of 74 trees and 3 movies, subdivided into the inverted generations and classes shown in Figure 6.1 b).

To get an impression of correlation between features we generated a scattermatrix, shown in Figure 6.2. Histograms of all features exhibited unimodal distributions, rendering good classification performance by using only one feature impossible. In addition features such as area, convex area, equivalence diameter and perimeter showed high correlation coefficients of over 0.90. This was expected, since all of these features were dependent on a regions' size.

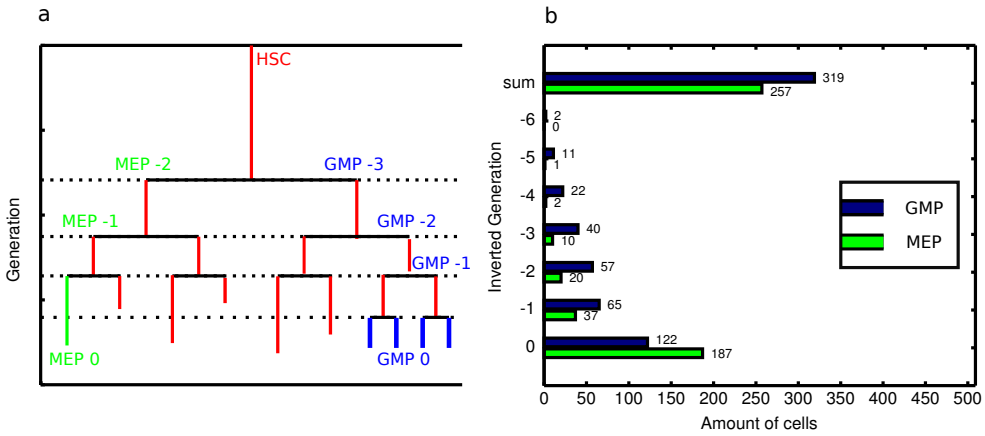
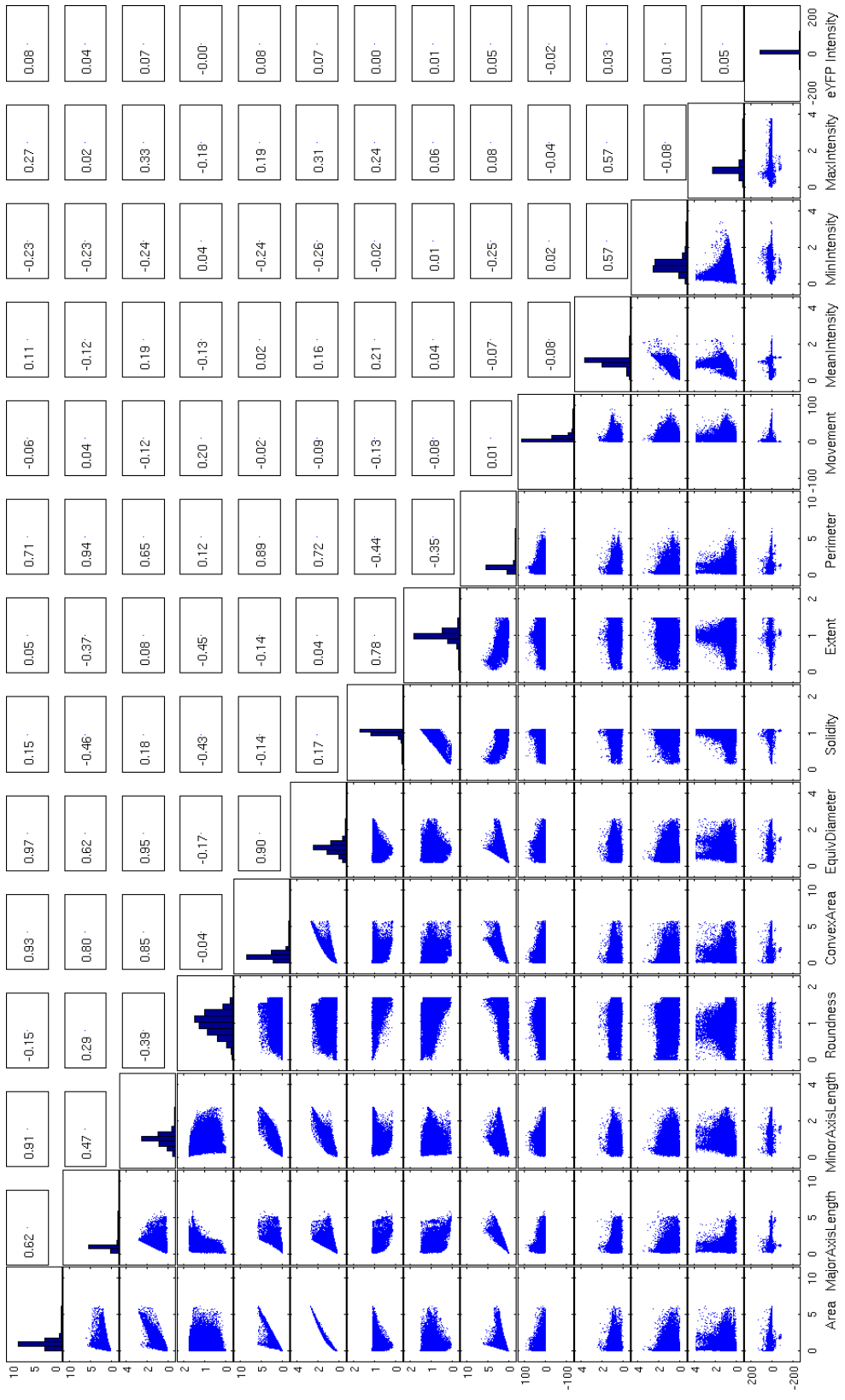


Figure 6.1. – Relabeling of cell types. a) Scheme how inverted generation and inverted type were calculated in a differentiation tree. Cells with the annotated cell types GMP or MEP were labeled with generation 0. Then, all predecessors of this cell were labeled with negative generations according to the number of division events necessary to reach the cell with a marker set on. If two markers were found in the tree, the type with smallest inverted generation was taken as inverted type (here MEP). First cells of all trees were labeled as HSC since this cell type was sorted by flow cytometry. b) Amount of relabeled cell types per inverted generation. 0 denotes cells with annotated marker onset.

6.2. Functional feature representation and overall classification performance

With a defined set of of 319 GMPs and 257 MEPs that were obtained after applying inverted generation, it was now necessary to determine the best parameter settings for functional data analysis. As already discussed in section 5.4, the functional representation of each time course resulted in a number of basis functions with coefficients, which we now used as single predictor variables for classification. This approach made it necessary to think about the number of basis functions that should be used. Application of FDA with, for example, five basis functions would result in rough, nearly linear approximation of a time course, whereas too many basis functions would produce approximations with very

Figure 6.2. ***(following page)*** – Scattermatrix of 13 morphological features and PU.1-eYFP intensity. The Diagonal shows histograms of each feature, the upper triangular matrix shows correlation coefficients for each combination of features, the lower triangular matrix shows scatterplots of two features. EquivDiameter exhibits high correlation with Area, as well as MajorAxisLength and Perimeter.


Area
MajorAxisLength
MinorAxisLength
Roundness
ConvexArea
EquivDiameter
Solidity
Extent
Perimeter
Movement
MeanIntensity
MinIntensity
MaxIntensity
eYFP Intensity

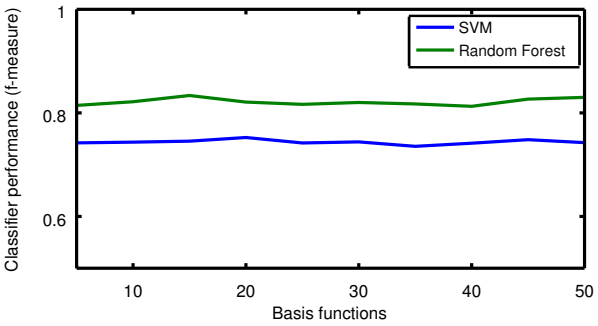


Figure 6.3. – Classification results of datasets build with 5 to 50 basis functions. Ranking is conducted by examining the mean macro-averaged f1-measure of 10-fold cross-validation. A score of 0.5 denotes the worst, 1.0 the best value. Green: Random forest classifier, Number of trees = 100, split-criterion = Gini's index; Blue: Support vector machine (SVM) with gaussian kernel. For the SVM, dataset was reduced by principal component analysis to reduce computational effort to a manageable level. Random forest outperforms the SVM at all numbers of basis functions. The highest value here is a mean macro-averaged f1-measure of 0.83 with random forest and 15 basis functions.

high curvature that is reflecting errors. In addition, different parameters for functional representation could have varying effects on different classification methods.

Classification accuracy of different methods that are applied on the same data can vary tremendously. Every algorithm has its strengths and drawbacks, depending on the methodology that it is based on and the dataset that is to be classified. This motivates the application of several different methods in order to select the classifier that performed best at the given problem. Here, we applied random forests and support vector machines (see section 2.4). Support vector machines are already widely used in machine learning tasks that are based on biological data and have led to impressive findings, as discussed in section 1.4. To our knowledge random forests were to date applied much less in comparable approaches, nevertheless the intuitively understandable methodology of decision trees, as well as studies stating that random forests outperform other classifiers in many applications legitimate the usage here [56].

We trained classifiers on an interval of five to fifty basis functions. Random forest was executed with 100 randomly sampled trees and Gini's index as the split criterion. Variable selection was not necessary, as the classifier is performing an intrinsic ranking procedure (see section 2.4.1). The SVM was set up with gaussian kernel. To improve classification performance of the SVM, the dataset was preprocessed by principal component analysis as described in section 2.4.2 to remove correlation in the data and to reduce the number of variables. All combinations of parameters and classifiers were then evaluated by 10-fold cross-validation on the complete dataset.

It turned out that in an interval of 5 to 50 basis functions, classification performance varied only with a standard deviation of 0.02, where an macro averaged f-measure of 0.81

with 40 basis functions turned out to be the worst and of 0.83 for 15 basis functions the best settings for classification. Figure 6.3 visualizes the performance for different amounts of basis functions. The smoothing factor λ was set to 10^7, as determined in section 5.4.

Random forest classification outperformed SVM at all parameter settings, reaching a macro-averaged f1-measure of 0.83 (micro-averaged f1-measure of 0.85), where the highest score achieved by the SVM was 0.75. The averaged confusion matrix over all cross-validation folds is shown in table 6.1. Here it can be seen that GMPs were predicted with higher accuracy than MEPs.

	MEP (pred.)	**GMP (pred.)**
MEP	21.6	5.6
GMP	4.5	29.4

Table 6.1. – Confusion matrix showing mean classification results after 10-fold cross-validation and 15 basis functions per feature. The used classification was random forest with 100 randomly sampled trees. Macro-averaged f1-measure: 0.83; micro-averaged f1-measure: 0.85.

6.3. Determining classification performance on different inverted generations

Machine learning derives rules from a set of samples of an unknown process to build a model that performs best on samples that were unknown during training [76]. Thus, the training set on which the rules are learned should reflect as much variability of the complete process as possible. In the case of hematopoietic stem cells, the samples (i.e. cells) were not simply drawn from a random process, but were arranged in tree structures (the differentiation trees). This structure was important, as one of the desired results of this study was to denote the most likely generation were lineage decision occurs. Under biological considerations, it would thus be likely that cells with a low inverted generation exhibit bad classification rates, whereas the higher the inverted generation, the better the classification should be. In contrast, it could also be possible that cells were not classifiable in most generations apart from a single one, were both types change their behavior in a way that makes classification possible.

We tested these hypotheses by using only cells of specified inverted generations. First, we evaluated all cells with inverted generation 0 (i.e. cells reliably annotated as GMP or MEP) as a reference value. 10-fold cross-validation on 187 MEPs and 122 GMPs (57 test and 519 train samples per fold) resulted in a macro-averaged f1-measure of 0.84 and a micro-averaged f1-measure of 0.86.

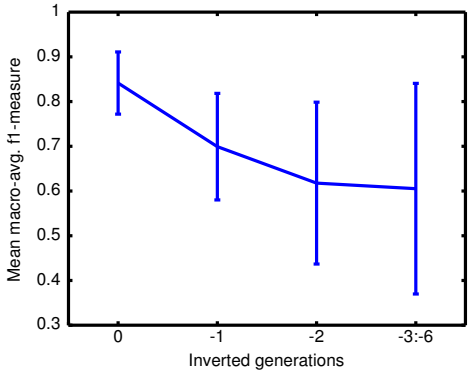


Figure 6.4. – Classification performance denoted as mean macro averaged f1-measure after 10-fold cross-validation on different inverted generations, Error bars denote standard deviation over all folds. Performance is good at inverted generation 0 but drops rapidly in lower inverted generations, while standard deviation is rising. The results are decreasing in such a high rate mostly because the amount of MEPs in in inverted generations is too low to train the classifier correctly.

Next, we used only cells in inverted generation -1, achieving a macro-averaged f1-measure of 0.70 and a micro-averaged f1-measure of 0.75. In this generation the prediction performance of GMPs and MEPs clearly differed. GMPs achieved a f1-measure of 0.83, whereas MEPs exhibited a f1-measure of only 0.51. Comparable results were obtained by classifying all other inverted generations (-2 to -6) together. These results showed that GMPs were classified correctly, but MEPs were misclassified more often than correctly classified. This is most likely due to the lack of MEPs in lower inverted generations. Figure 6.4 displays all results.

6.4. Most important features in classification

In the preceding sections we used all morphological features and PU.1-eYFP intensity to train the classifier. However, not every feature is as important as another in order to produce good classification results. For example, cells commited to different lineages could exhibit high diversity in size, but would show equal movement patterns. Thus, it was of interest which of the features had the most influence on classification results, since our collaborators could use this information to identify cells by examining this behavior on living cells. Information about feature importance in random forests can be drawn directly from the method by averaging the importance of each feature per grown tree. In our dataset, especially the geometrical features showed high correlation. This could lead to different importance measures of features in different runs of random forest. Since the method randomly samples a set of variables and for each tree ranks the subset, high correlated variables could lead to equal classification performance, but in one run perimeter

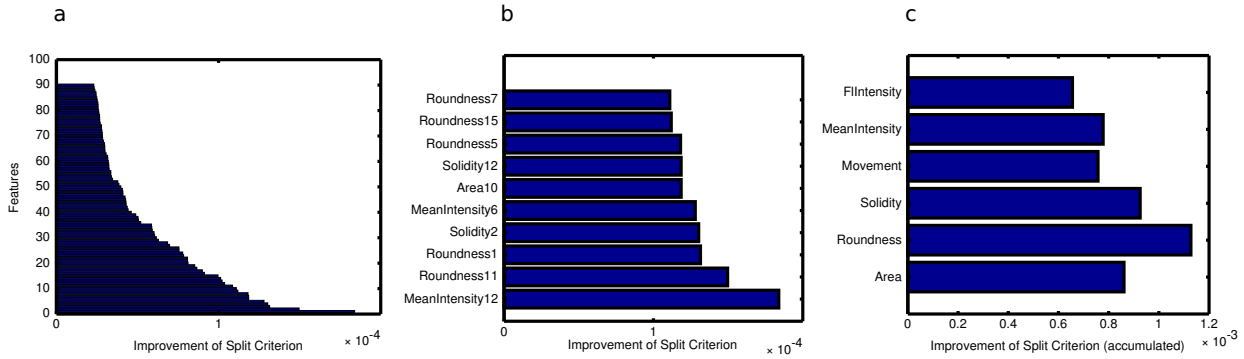


Figure 6.5. – Feature importance of six uncorrelated features. a) Features represented by 15 basis functions after functional approximation (called here predictor variables). b) 10 predictor variables that had most influence on classification. Scalars behind the feature's name are indicating the number of the basis function. Variables of roundness are representing 50% of the best variables. c) Accumulated importance for all basis functions per feature. Roundness exhibits the most importance, but all other features are important for classification too. PU.1-eYFP intensity is least important, indicating that classification is possible based on brightfield features alone.

would be the best feature and in another cell size. Therefore, we used the correlation coefficients of all features that were computed in section 6.1.2 and removed all features correlated to an earlier one in the list with a correlation coefficient not falling in the interval of $[-0.5; 0.5]$. 10-fold cross-validation on all 576 cells (57 test and 519 train samples per fold) using the remaining features area, roundness, solidity, movement, mean brightfield intensity and PU.1-eYFP intensity, each represented by the 15 basis function coefficients, resulted in a macro-averaged f1-measure of 0.83 and a micro-averaged f1-measure of 0.85. In addition, 10-fold cross-validation on a dataset comprised of cells with inverted generation 0 (i.e. reliably annotated MEPs and GMPs, 30 test samples and 279 training samples per fold) yielded a macro-averaged f1-measure of 0.85 and a micro-averaged f1-measure of 0.86. These results indicated that the minimal set of six features was sufficient to achieve comparable classification results as a set with all available features.

We then evaluated if one of these features had significantly more influence on classification results as the others. We computed a ranking of all 90 variables. As shown in Figure 6.5 a), some variables exhibited a higher increase of the split criterion than others. In the list of best scoring features shown in Figure 6.5 b), roundness appeared quite often. We accumulated all 15 variables per feature, resulting in a ranking of the six features as shown in Figure 6.5 c). Here, roundness was slightly more important as other features. Interestingly, PU.1-eYFP exhibited the least importance.

We confirmed this finding by removing PU.1-eYFP from the set of features, thus five morphological features, i.e. 75 variables remained in the dataset for classification. The

resulting macro-averaged f1-measure was at 0.77 und the micro-averaged f1-measure was at 0.80. Thus, PU.1-eYFP intensity is a useful feature in classification, but not the determining one.

6.5. Fate prediction on differentiation trees

In a last step, we checked whether it was possible to predict cell fates of completely unknown trees. In the cross-validation approaches that were conducted in the preceding sections we did not take into account, whether cells originating in the same tree were in the test-set as well as in the training set of a single fold. This could lead to a bias in classification performance. For example, sister cells could exhibit nearly the same behavior, thus it would be an unwanted advantage if the classifier is trained on one sister cell and evaluated on the other.

Five tracking trees were randomly drawn and all cells originating in these trees were excluded from the dataset, resulting in a testset of 34 MEPs and 43 GMPs. 244 MEPs and 244 GMPs were left for training. Figure 6.6 exemplifies this procedure. 10-fold cross-validation resulted in a macro-averaged f1-measure of 0.92 and a micro-averaged f1-measure of 0.92, based on Table 6.2. The very high performance on unknown trees indicates that the method could be applicable in fate prediction and that our method is not dependent on equal behavior of sister cells or other dependencies that could be in cells residing in the same differentiation tree. However, we did not perform an analysis on all trees in the set (leave-one-tree-out), which would be necessary in order to verify this theory. Figure 6.7 visualizes classification results for two tracked differentiation trees.

	MEP (pred.)	**GMP (pred.)**
MEP	31	3
GMP	3	40

Table 6.2. – Confusion matrix showing classification results of cells originating in 5 trees that were randomly chosen out of the dataset. Values on the diagonal denote true positives. Micro-averaged f1-measure was 0.92, macro-averaged f1-measure was 0.92 as well.

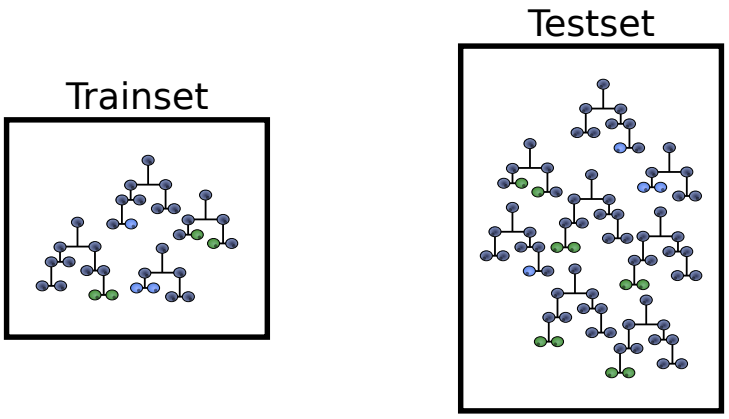
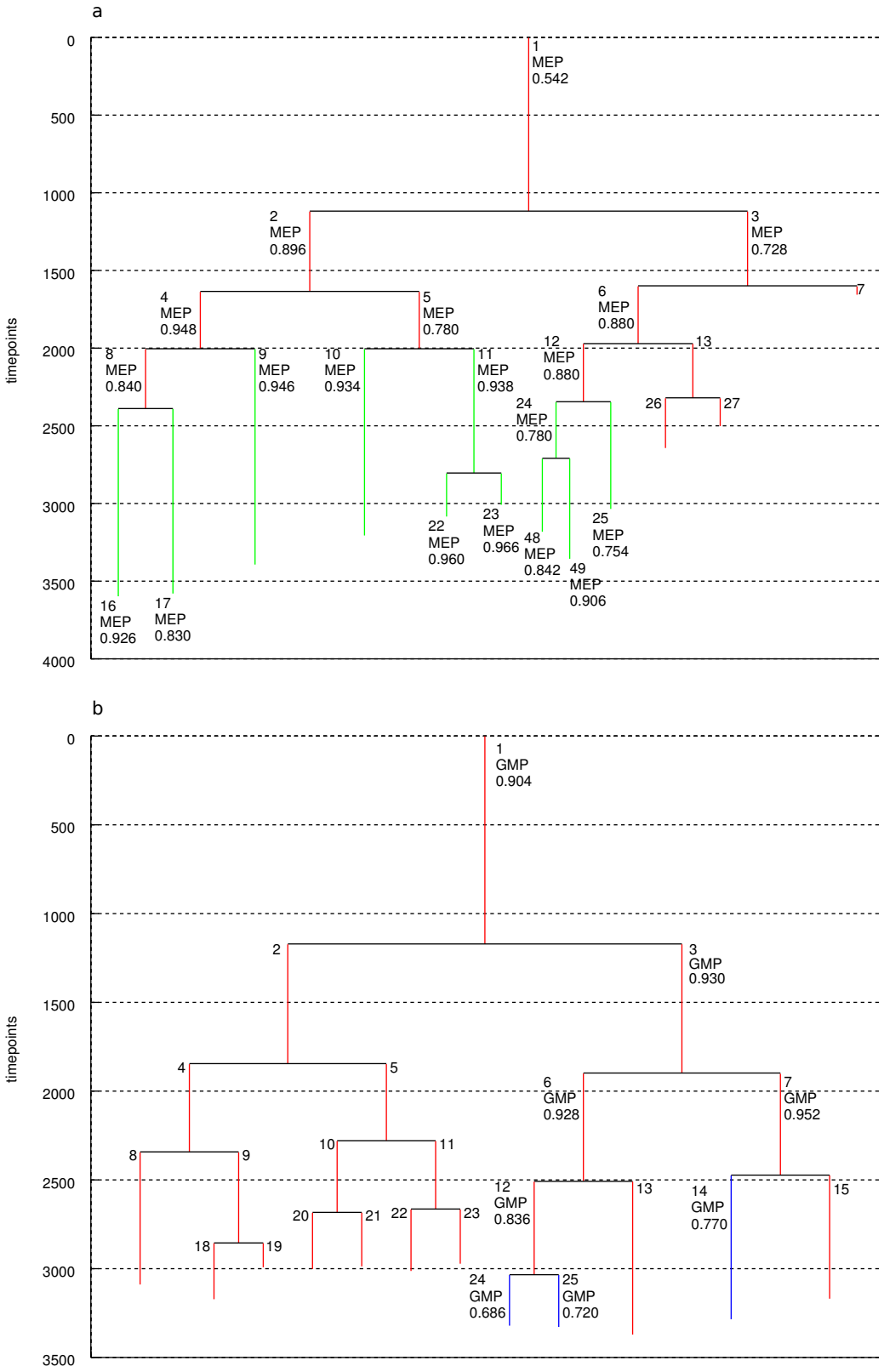


Figure 6.6. – Visualization of prediction on complete differentiation trees. A small amount of trees (e.g 5) are drawn from the set at random and all cells originating in these trees are removed from the set. The remaining cells are then used for training and the removed cells are used for evaluation. This allows to analyze whether the trained model is able to predict cells without knowledge of the behavior of predecessors or sister cells that could have been used in training and could exhibit the same behavior as the evaluated cell.

Figure 6.7. ***(following page)*** – Two differentiation trees of which all cells were used as test samples for classification evaluation, training was conducted of all remaining cells in the dataset. Red line: No antibody marker has switched on. Green line: CD150 (MEP) is switched on. Blue line: Antibody marker for FCγ is switched on. Scalar numbers denote the cell number, the cell type below (MEP or GMP) was predicted by the model with a probability that is written below the type. Cells with no predicted cell type were either discarded from the set during manual assessment or were not part of a branch that eventually generated a progenitor with an annotated marker onset.


a
timepoints
0
500
1000
1500
2000
2500
3000
3500
4000
1 MEP 0.542
2 MEP 0.896
3 MEP 0.728
4 MEP 0.948
5 MEP 0.780
6 MEP 0.880
7
8 MEP 0.840
9 MEP 0.946
10 MEP 0.934
11 MEP 0.938
12 MEP 0.880
13
24 MEP 0.780
26
27
22 MEP 0.960
23 MEP 0.966
48 MEP 0.842
25 MEP 0.754
49 MEP 0.906
16 MEP 0.926
17 MEP 0.830
b
timepoints
0
500
1000
1500
2000
2500
3000
3500
1 GMP 0.904
2
3 GMP 0.930
4
5
6 GMP 0.928
7 GMP 0.952
8
9
10
11
12 GMP 0.836
13
14 GMP 0.770
15
18
19
20
21
22
23
24 GMP 0.686
25 GMP 0.720

6.6. Conclusion

In this chapter, we utilized the functional representation of time courses to derive 15 predictor variables per feature and used this information to train the classification methods. Furthermore, we defined an approach how to relabel cells that were not annotated as GMPs or MEPs by inverted generations.

Analysis of all features revealed huge correlations, especially between geometrical features. Although not unexpected, in future versions of the machine learning approach these features could be discarded from the set or not being measured by image processing. This would increase the performance of the complete pipeline. Nevertheless, we chose a set of features that describes a cell's behavior in many different aspects, taking into account a cell's size and shape, its movement behavior, as well as its intensity in brightfield images and the quantified intensities of PU.1-eYFP, which reflect expression levels of the transcription factor.

Classification performance on the complete set as well as on cells annotated as GMP or MEP resulted in high micro-averaged f1-measures (0.83, 0.85), indicating that discriminating different cell types with the chosen features and classification method is possible. Under biological considerations, our results suggest that hematopoietic stem cells exhibit a behavior that is, even if not recognizable by eye, different enough to decide to which lineage a cell has commited.

Classification of GMPs performed better than that of MEPs, which could be due to a technical restriction in the time lapse experiments. It is known that the fluorescence antibody marker binding to CD150 that identifies MEPs switches on much earlier than the respective marker binding to Fcγ for GMPs. This is also the explanation why there are more annotated MEPs in the dataset than GMPs and in inverted generations the amount of MEPs declines. This early onset could lead to annotation of cells which are commited, but their morphological behavior is still similar to GMPs and thus, mis-classification is likely. A more technical explanation that is supporting this hypothesis is the fact random forest tends to classify uncertain samples to the class that was larger in training, which was here GMPs. The hypothesis above would also explain bad classification rates in higher inverted generations. However, the lack of data in inverted generations of -3 to -6 did not allow to make final assumptions.

The analysis of most important features implied that it is possible to only use a subset of uncorrelated features. We were able to show that PU.1-eYFP intensity is not the determining factor leading to good classification results. Moreover, classification performance decreases just by 0.05 (macro-averaged f1-measure), if the feature is excluded from the dataset. Thus it could be possible that the expression of PU.1-eYFP is not driving lineage decision but is a consequence thereof. However, the automatic measurements of PU.1-eYFP intensities as well as the interpolation of this feature could have led to errors in the data which are masking the true effect of this feature. Further research is necessary in order to elucidate the importance of PU.1 as a feature in classification.

The prediction of cell fates as early as possible in a differentiation tree was achieved in part. Prediction on differentiation trees were no cell was used for training the classifier produced very good results, indicating that this approach is possible. However, evaluation on more than the five randomly chosen trees that were used here is necessary to confirm these results. A possible approach would be leave-one-tree-out cross-validation, in accordance to 10-fold cross validation. This will be done as a next step in this project.

7. Summary and Outlook

In this thesis, we focused on time lapse movies of hematopoietic stem cells commited to the erytroid or myeloid lineage. Our aim was to extract as much information as possible from continuous time-lapse experiments and to combine all signals to predict a cells fate as early as possible in the differentiation tree. The methods were established in close collaboration with researchers from the Institute of Stem Cell Research at the Helmholtz Zentrum München.

First, we developed an image processing pipeline based on manually tracked differentiation trees that is able to measure morphological behavior of single cells. Patches of brightfield images depicting annotated cells were extracted, enhanced and segmented by a combined thresholding and watershedding approach, returning a cell mask for each timepoint. We demonstrated that the MSER algorithm is producing sound segmentation results even if the cell is hardly differentiable from background by eye. Based on the cell masks, we measured 14 morphological features for each timepoint, resulting in time courses that represent cell behavior over their full lifetime at intervals of 90 seconds. The final pipeline is able to process 20 tracking trees at the same time on a computation grid. The high level of detail and the processing speed allow a completely new view on hematopoietic differentiation events and possible identification of interesting signals that can lead to new biological insights. Results of the pipeline were also used to identify cell masks in fluorescence images, a task that is to date conducted semi-automatically and is thus quite time consuming.

In a subsequent step, we analyzed cell behavior based on the measured morphological features. All time courses were normalized and corrected by functional approximation. We validated the accuracy of our method by comparing the mean growth ratio of all cells with a manually generated dataset and were able to confirm the results. In addition, we showed that the time courses are reflecting expected biological behavior, such as constant growth or a decrease in movement before and after mitosis. Furthermore, we analyzed the processes underlying cell movement in detail and were able to proof that hematopoietic cell movement is neither following brownian motion nor a lévy walk.

The last part of this thesis comprises the computational prediction of cell fates. We used 13 morphological features and eYFP intensity to train a random forest classifier to discriminate cells commited to the erytroid or myeloid lineage, as well as their predecessors. The feature orientation was discarded since we could show that measurements of this feature are random and thus not providing useful information about cell behavior. Cross validation of 576 cells returned a macro-averaged f1-measure of 0.83, indicating that the

set of features and the differential behavior of cells commited to different lineages is sufficient for prediction. Classification using all features except PU.1-eYFP intensity yielded a macro-averaged f1-measure of 0.77. Thus, in this data set classification performance is decreasing only slightly if PU.1-eYFP intensity is not used, nevertheless it represents a good predictor variable. A clear identification of most influential features was not possible, yet we identified six features that performed as well as the complete set of features, namely area, roundness, movement, mean brightness intensity, solidity, and PU.1-eYFP intensity. At last, we demonstrated that a test set of cells that completely covered five randomly chosen differentiation trees achieved a macro-averaged f-measure of 0.92. The high performance score is indicating that it should be possible to use this method, for example, to reduce the amount of manually tracked timepoints in order to define if a cell is commited to the GMP or MEP lineage.

In future versions of our image processing pipeline, there will be several technical improvements. First, a method able to recognize under- and oversegmentation at runtime should be developed. This could be used to evaluate different segmentation approaches and to choose the method that performs best on the given image. We already implemented a basic approach that applied different watershedding methods by evaluating sizes of segmented objects, yet additional information like intensity histograms could achieve better results here. Furthermore, it is crucial to develop an automated method that is able to decide if a measured time course is accurate enough to be used in further analysis, or if it should be removed from the set. In this project quality control was done manually, but with more data being processed this is not feasible anymore. A possible approach would be fitting the time courses by functions and removing outliers, for example by cooks distance [77]. This would be in accordance to the methods of functional data analysis, which were already applied in this project. An optimization of the segmentation in order to process other cell types in an automated fashion is another possible improvement. We already demonstrated this for some samples of embryonic stem cells, but an automated method will require more time for development, since these cells are exhibiting a more diverse morphology and are thus harder to segment. At last, an increase of processing speed by a more effective method to load the large brightfield images from the network would allow parallel processing of more than 20 tracking trees at the same time. If automated tracking methods become available this improvement will become necessary to manage the huge amount of data.

Possible improvements in fate prediction are mostly dependent on quality of the computed data set and thus on the quality of image processing and error correction. A normalization method should be used that is optimized for the given task and it could be beneficial to apply different smoothing values for each feature when applying representation by b-splines. The abilities of functional data analysis were used in a limited fashion here, and the information provided by derivatives or methods that are more suited to functional representation, such as functional generalized linear models could improve prediction performance [78]. In addition, other classification methods should be tested. We applied support vector machines and random forests, but there are many more algorithms such

as artificial neural networks, ensemble learning methods or adaboost that could increase classification performance [79–81]. In addition, utilization of additional features, for example Haralick texture features as well as Zernicke moments [82; 83] that are derived from the images could reveal more interesting aspects of morphological cell behavior.

At last, the completed image processing pipeline, as well as the tools for analysis and fate prediction will be integrated into AMT. All methods described here were already developed under this consideration, thus input and output structures are fully compatible. The integration will enable users to apply our method by simply providing a tree structure and evaluating the results, without starting any external programs or changing configuration files by hand. Cell fate prediction could also render the use of antibody markers for cell type identification unnecessary. Spare channels therefore can be used to investigate other interesting elements, such as transcription factors or cytokines by fluorescence labeling.

Our method is not able to directly identify one of the processes underlying a cells decision to become a MEP or GMP, but it could be possible to narrow down a small timespan were morphological behavior indicates a likely fate decision. It is conceivable to implement the method to TTT or another imaging software. If automatic tracking methods become available, this would allow on-line segmentation and classification of a imaged cells and could identify cells that are in the event of lineage decision. These cells could then be extracted and analyzed in detail, for example by high throughput methods such as mRNA microarrays. This could be helpful in elucidating the molecular factors driving lineage commitment.

The experiments used in this project were conducted without the consideration that brightfield images should be computational processable. It will be interesting to see how much information could be gathered out of movies that were conducted especially for this task. Furthermore, it will be necessary to track more differentiation trees in order to obtain sufficient cell amounts, especially in higher inverted generations. Without this data, a reliable prediction when a cell is committing to a fate is not possible. We are aware that due to the lack of automatic tracking procedures this is still a time consuming task. Nevertheless the results we achieved based on the movies provided for this thesis are promising and efforts in brightfield segmentation and analysis of morphological cell behavior should be intensified. The development of these methods will only be successful if both biological and theoretical scientists work in close collaboration as it is done by our group and the ISF for a couple of years now.

A. Cells are small and round - a preliminary analysis

At the beginning of this project, it was not clear if hematopoietic stem cells differ in their morphological behavior enough to apply a computational classification approach. To get a first impression of the data, we used an early version of our image processing algorithm to check if it performs good enough to actually find differences in the cells' behavior. We used trackings of experiment 100922PH2 (see 3.2), as cell populations were divided into different positions and thus assigning a type to each cell was feasible without the complete image processing pipeline which was not established to this date. Cell types in this experiment were HSCs, MPPs, as well as GMPs. Table A.1 shows the amount of well segmented time points per cell type. Around 50% of the images that were segmented by Otsu's method without a working watershedding approach were found to be usable for further analysis after manual inspection. Table A.1 gives an overview of the available cell types and their abundance.

To compare the different cell populations, the mean value for all measurements per cell and feature was calculated and saved in a vector. Results of a histogram analysis as illustrated in Figure A.1 of the averaged time courses revealed differences between cell types in most features. Remarkable is the bimodal distribution of HSCs in feature major axis length. One explanation for this could be that HSCs tend to stick on the bottom of the microscopes glass slide and crawl over the plate while contracting and relaxing. Nevertheless, other features such as extent exhibited promising behavior too. This early analysis represents a proof-of-concept of the complete project that was done in this thesis, as it demonstrated that - even if not observable by watching differentiation of two HSCs directly in the movie - there are differences in behavior that should be usable to apply machine learning methods. However, the set used here was not suitable for further analysis, since it lacked MEPs and all trackings that were available covered only one cell, not a whole tree.

Cell Type	Abundance	Correct segmented
HSC	20	0.54
MPP	30	0.62
GMP	33	0.45

Table A.1. – Segmentation results and amount of cell types in experiment 100922PH2

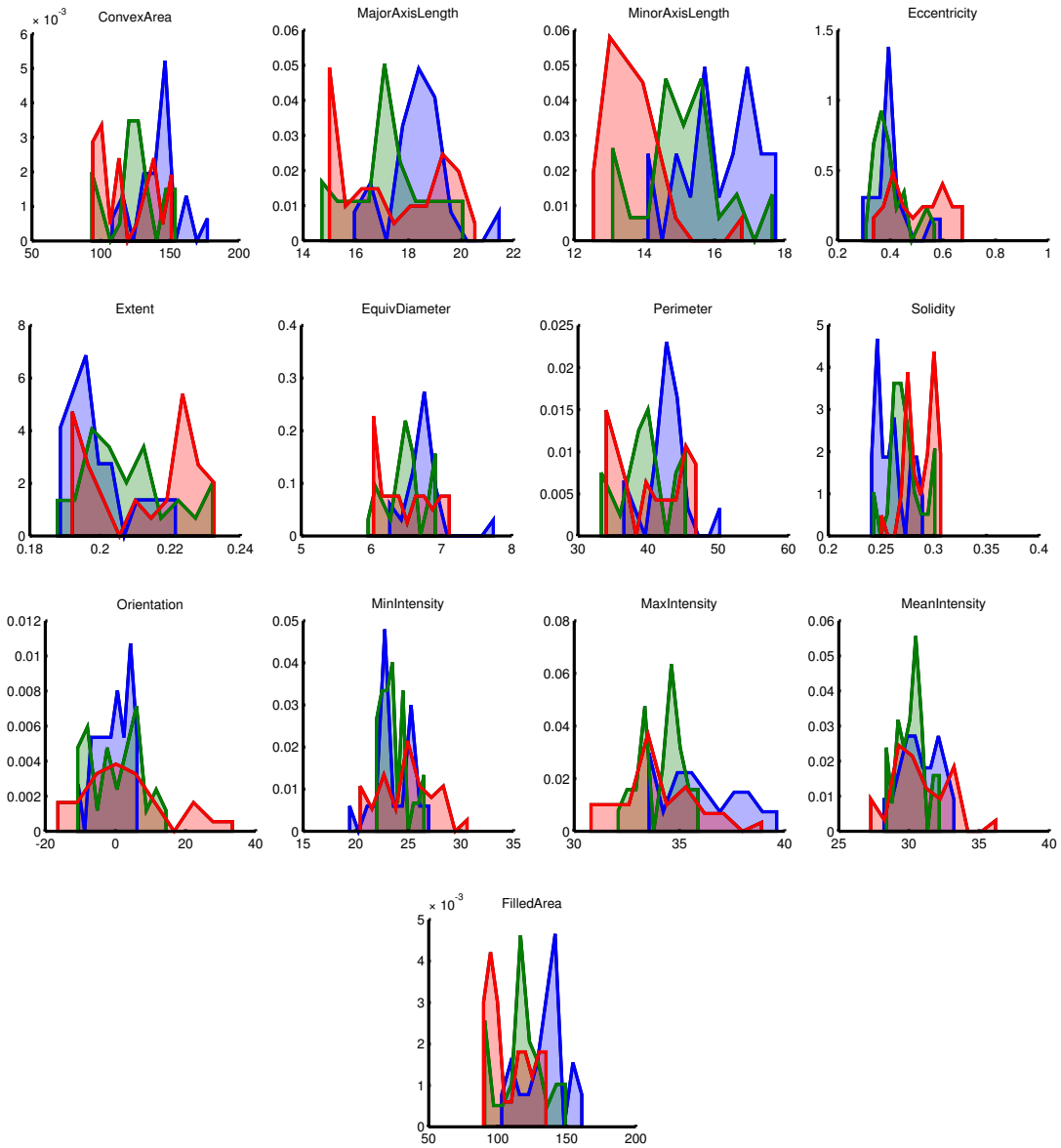


Figure A.1. – Histogram plots of all morphological features for each cell type in the experiment. Red: HSC; Green: MPP; Blue: GMP. A detailed description of the used features can be found at section 4.2. Most features exhibit differences in their peaks, for example filled area indicates that HSCs are the smallest cell type, followed by MPPs and then GMPs, which is in accordance to their expected behavior.

B. Application to other cell types

We applied the image processing steps discussed in section 4 to samples from a single cell time lapse movie that shows differentiation of embryonic stem cells (ESCs), conducted by Adam Filipczyk from the ISF. Due to diverse morphology and different experimental conditions, segmentation was a greater challenge in comparison to hematopoietic stem cells. First attempts indicated that reasonable results were possible which is shown in Figure B.1. To fully automate the segmentation on these types of movies additional preprocessing steps and parameter tuning would be necessary. However, we demonstrated that our approach is useful in other types of single cell time lapse experiments and further development of this method seems reasonable.

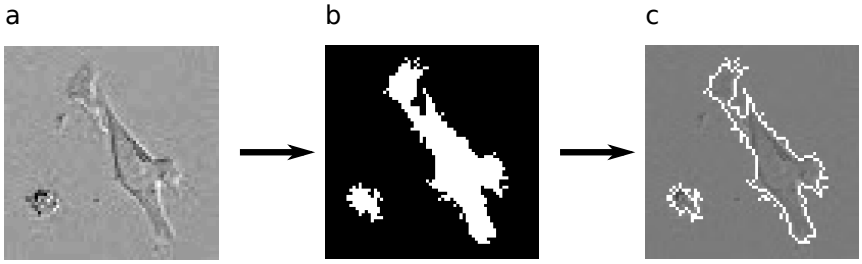


Figure B.1. – a) 70 x 70 cutout from a brightfield image. b) binary image after MSER thresholding and morphological closing. c) Overlay of segmented cell mask.

Bibliography

[1] D E Cohen and D Melton. Turning straw into gold: directing cell fate for regenerative medicine. *Nat Rev Genet*, 12(4):243–252, April 2011. doi: 10.1038/nrg2938. URL http://dx.doi.org/10.1038/nrg2938.

[2] K H Campbell, J McWhir, W A Ritchie, and I Wilmut. Sheep cloned by nuclear transfer from a cultured cell line. *Nature*, 380(6569):64–66, 1996. doi: 10.1038/380064a0. URL http://dx.doi.org/10.1038/380064a0.

[3] A G Smith. Embryo-derived stem cells: of mice and men. *Annu Rev Cell Dev Biol*, 17:435–462, 2001. doi: 10.1146/annurev.cellbio.17.1.435. URL http://dx.doi.org/10.1146/annurev.cellbio.17.1.435.

[4] J Seita and I L Weissman. Hematopoietic stem cell: self-renewal versus differentiation. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 2010. doi: 10.1002/wsbm.86. URL http://onlinelibrary.wiley.com/doi/10.1002/wsbm.86/full.

[5] B M Carlson. *Principles of regenerative biology*. Academic Press, 2007. URL http://books.google.de/books?hl=de&lr=&id=f_Epd5lHTNkC&oi=fnd&pg=PP1&dq=+Principles+of+Regenerative+Biology+&ots=1hkyfs3ZZu&sig=_4CW57ym8mfnULwUIH_DusPGaTw.

[6] E Y Snyder, D L Deitcher, C Walsh, S Arnold-Aldea, E A Hartwieg, and C L Cepko. Multipotent neural cell lines can engraft and participate in development of mouse cerebellum. *Cell*, 68(1):33–51, January 1992.

[7] A J Erslev, E Beutler, M A Lichtman, B S Coller, T J Kipps, and U Seligsohn. Williams Hematology. *Williams Hematology*, 1995.

[8] G Thews, E Mutschler, P Vaupel, and Others. Anatomie, Physiologie, Pathophysiologie des Menschen. 1999.

[9] J E Till and E A McCulloch. A direct measurement of the radiation sensitivity of normal mouse bone marrow cells. *Radiat Res*, 14:213–222, February 1961.

[10] Stuart H Orkin and Leonard I Zon. Hematopoiesis: an evolving paradigm for stem cell biology. *Cell*, 132(4):631–644, February 2008. doi: 10.1016/j.cell.2008.01.025. URL http://dx.doi.org/10.1016/j.cell.2008.01.025.

[11] B D MacArthur, A Ma'ayan, and I R Lemischka. Systems biology of stem cell fate and cellular reprogramming. *Nature Reviews Molecular Cell Biology*, 10(10):672–681, 2009. doi: 10.1038/nrm2766. URL http://www.nature.com/nrm/journal/v10/n10/abs/nrm2766.html.

[12] A Medvinsky, S Rybtsov, and S Taoudi. Embryonic origin of the adult hematopoietic system: advances and questions. *Development*, 138(6):1017, 2011. URL http://dev.biologists.org/content/138/6/1017.short.

[13] S A Giralt, M Horowitz, D Weisdorf, and C Cutler. Review of stem-cell transplantation for myelodysplastic syndromes in older patients in the context of the Decision Memo for Allogeneic Hematopoietic Stem Cell Transplantation for Myelodysplastic Syndrome emanating from the Centers for Medicare and Medicai. *J Clin Oncol*, 29(5):566–572, February 2011. doi: 10.1200/JCO.2010.32.1919. URL http://dx.doi.org/10.1200/JCO.2010.32.1919.

[14] NIH. Regenerative Medicine. 2006. URL http://stemcells.nih.gov/info/2006report/.

[15] T Schroeder. Hematopoietic stem cell heterogeneity: subtypes, not unpredictable behavior. *Cell stem cell*, 6 (3):203–207, 2010. URL http://www.sciencedirect.com/science/article/pii/S1934590910000500.

[16] T Schroeder. Imaging stem-cell-driven regeneration in mammals. *Nature*, 453(7193):345–351, 2008. doi: 10.1038/nature07043. URL http://dx.doi.org/10.1038/nature07043.

[17] J Krumsiek, C Marr, T Schroeder, and F J Theis. Hierarchical differentiation of myeloid progenitors is encoded in the transcription factor network. *Differentiation*, 49(0), 2010.

[18] A Hermann. *Analysis of asymmetric division of hematopoietic stem cells by continuous single cell observation*. PhD thesis, Ludwig-Maximilians-Universitaet Muenchen, 2009. URL http://edoc.ub.uni-muenchen.de/12130/1/Hermann_Andrea.pdf.

[19] C J H Pronk, D J Rossi, R Må nsson, J L Attema, G L Norddahl, C K F Chan, M Sigvardsson, I L Weissman, and D Bryder. Elucidation of the phenotypic, functional, and molecular topography of a myeloerythroid progenitor cell hierarchy. *Cell Stem Cell*, 1(4):428–442, 2007. doi: 10.1016/j.stem.2007.07.005. URL http://dx.doi.org/10.1016/j.stem.2007.07.005.

[20] W H Fridman. Fc receptors and immunoglobulin binding factors. *The FASEB journal*, 5(12):2684, 1991.

[21] A Shibuya and S Honda. Molecular and functional characteristics of the Fcalpha/muR, a novel Fc receptor for IgM and IgA. *Springer Semin Immunopathol*, 28(4):377–382, 2006. doi: 10.1007/s00281-006-0050-3. URL http://dx.doi.org/10.1007/s00281-006-0050-3.

[22] J Zhu and S G Emerson. Hematopoietic cytokines, transcription factors and lineage commitment. *Oncogene*, 21(21):3295–3313, 2002. doi: 10.1038/sj.onc.1205318. URL http://dx.doi.org/10.1038/sj.onc.1205318.

[23] P Hoppe. Der Einfluss von Zytokinen auf die Linienentscheidung myeloider Vorläuferzellen – Erkenntnisse aus der Einzelzellanalyse. Master's thesis, Ludwig-Maximilians-Universität München Fakultät für Biologie, 2008.

[24] T Ito, C Nishiyama, N Nakano, M Nishiyama, Y Usui, K Takeda, S Kanada, K Fukuyama, H Akiba, T Tokura, M Hara, R Tsuboi, H Ogawa, and K Okumura. Roles of PU.1 in monocyte- and mast cell-specific gene regulation: PU.1 transactivates CIITA pIV in cooperation with IFN-gamma. *Int Immunol*, 21(7):803–816, July 2009. doi: 10.1093/intimm/dxp048. URL http://dx.doi.org/10.1093/intimm/dxp048.

[25] P Burda, P Laslo, and T Stopka. The role of PU.1 and GATA-1 transcription factors during normal and leukemogenic hematopoiesis. *Leukemia*, 24(7):1249–1257, July 2010. doi: 10.1038/leu.2010.104. URL http://dx.doi.org/10.1038/leu.2010.104.

[26] K Takahashi and S Yamanaka. Induction of pluripotent stem cells from mouse embryonic and adult fibroblast cultures by defined factors. *Cell*, 126(4):663–676, 2006. doi: 10.1016/j.cell.2006.07.024. URL http://www.sciencedirect.com/science/article/pii/S0092867406009767.

[27] S Yamanaka and H M Blau. Nuclear reprogramming to a pluripotent state by three approaches. *Nature*, 465 (7299):704–712, June 2010. doi: 10.1038/nature09229. URL http://dx.doi.org/10.1038/nature09229.

[28] T Schroeder. Long-term single-cell imaging of mammalian stem cells. *Nat Methods*, 8(4 Suppl):S30—-S35, April 2011. doi: 10.1038/nmeth.1577. URL http://dx.doi.org/10.1038/nmeth.1577.

[29] M A Rieger, P S Hoppe, B M Smejkal, A C Eitelhuber, and T Schroeder. Hematopoietic cytokines can instruct lineage choice. *Science*, 325(5937):217–218, July 2009. doi: 10.1126/science.1171461. URL http://dx.doi.org/10.1126/science.1171461.

[30] M D Abramoff, P J Magalhaes, and S J Ram. Image processing with ImageJ. *Biophotonics international*, 11 (7):36–43, 2004.

[31] T R Jones, I H Kang, D B Wheeler, R A Lindquist, A Papallo, D M Sabatini, P Golland, and A E Carpenter. CellProfiler Analyst: data exploration and analysis software for complex image-based screens. *BMC Bioinformatics*, 9:482, 2008. doi: 10.1186/1471-2105-9-482. URL http://dx.doi.org/10.1186/1471-2105-9-482.

[32] R Schnabel, H Hutter, D Moerman, and H Schnabel. Assessing normal embryogenesis in Caenorhabditis elegans using a 4D microscope: variability of development and regional specification. *Dev Biol*, 184(2):234–265, April 1997. doi: 10.1006/dbio.1997.8509. URL http://dx.doi.org/10.1006/dbio.1997.8509.

[33] N Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11:285–296, 1975.

[34] J Matas, O Chum, M Urban, and T Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004. ISSN 0262-8856. doi: 10.1016/j.imavis.2004.02.006. URL http://www.sciencedirect.com/science?_ob=ArticleURL&_udi=B6V09-4CPM632-1&_user=10&_coverDate=09/01/2004&_rdoc=1&_fmt=high&_orig=gateway%&_origin=gateway&_sort=d&_docanchor=&view=c&_searchStrId=1730298633&%_rerunOrigin=scholar.google&_acct=C000050221&_version=1&_urlVersion=0&%_userid=10&md5=4619ed984d0997fee7038cf13d1a9b7b&searchtype=a.

[35] L G Shapiro and G C Stockman. *Computer Vision*. Prentice Hall, 2001. ISBN 0130307963. URL http://www.amazon.com/Computer-Vision-Linda-G-Shapiro/dp/0130307963.

[36] L G Brown. A survey of image registration techniques. *ACM Computing Surveys*, 24(4):325–376, December 1992. ISSN 03600300. doi: 10.1145/146370.146374. URL http://portal.acm.org/citation.cfm?id=146370.146374.

[37] R Nisbet, J Elder, J F Elder, and G Miner. *Handbook of statistical analysis and data mining applications*. Academic Press, 2009.

[38] B Neumann, T Walter, J K Hériché, J Bulkescher, H Erfle, C Conrad, P Rogers, I Poser, M Held, U Liebel, C Cetin, F Sieckmann, G Pau, R Kabbe, A Wünsche, V Satagopam, M H A Schmitz, C Chapuis, D W Gerlich, R Schneider, R Eils, W Huber, J Peters, A A Hyman, R Durbin, R Pepperkok, and J Ellenberg. Phenotypic profiling of the human genome by time-lapse microscopy reveals cell division genes. *Nature*, 464(7289):721–727, April 2010. doi: 10.1038/nature08869. URL http://dx.doi.org/10.1038/nature08869.

[39] M Held, M H Schmitz, B Fischer, T Walter, B Neumann, M H Olma, M Peter, J Ellenberg, and D W Gerlich. CellCognition: time-resolved phenotype annotation in high-throughput live cell imaging. *Nat Methods*, 7(9):747–754, September 2010. doi: 10.1038/nmeth.1486. URL http://dx.doi.org/10.1038/nmeth.1486.

[40] R M Haralick, K Shanmugam, and I Dinstein. Texture parameters for image classification. *IEEE Transactions on systems*, 3:610–621, 1973.

[41] A R Cohen, F L A F Gomes, B Roysam, and M Cayouette. Computational prediction of neural progenitor cell fates. *Nat Methods*, 7(3):213–218, 2010. doi: 10.1038/nmeth.1424. URL http://dx.doi.org/10.1038/nmeth.1424.

[42] J O Ramsay. Functional data analysis. *Encyclopedia of Statistical Sciences\cite{ramsay1997functional}*, 1997. URL http://onlinelibrary.wiley.com/doi/10.1002/0471667196.ess3138/full.

[43] Matlab - The language of technical computing, 2010. URL http://www.mathworks.com/.

[44] A Vedaldi and B Fulkerson. VLFeat: An Open and Portable Library of Computer Vision Algorithms. http://www.vlfeat.org, 2008. URL http://www.vlfeat.org/.

[45] L Vincent and P Soille. Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(6):583–598, 1991. ISSN 0162-8828. doi: 10.1109/34.87344. URL http://portal.acm.org/citation.cfm?id=116700.116707.

[46] C R Maurer Jr, R Qi, and V Raghavan. A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 265–270, 2003.

[47] M Schwarzfischer. Single-cell analysis of multipotent hematopoietic progenitor cells. Master's thesis, Luwig-Maximilians-Universität Technische Universität München, 2009.

[48] J Krumsiek. Computational modeling of regulatory networks in hematopoietic differentiation. Master's thesis, Ludwig-Maximilans-Universität, Technische Universität München, 2009.

[49] S Orfanidis. *Optimum signal processing : an introduction*. McGraww-Hill, New York, 2nd ed. edition, 1988. ISBN 9780023893803. URL http://www.worldcat.org/title/optimum-signal-processing-an-introduction/oclc/17732057.

[50] C De Boor. *A Practical Guide to Splines (Applied Mathematical Sciences)*. Springer, 1994. ISBN 0387903569. URL http://www.amazon.com/Practical-Splines-Applied-Mathematical-Sciences/dp/0387903569.

[51] L Fahrmeir, R Künstler, I Pigeot, and G Tutz. *Statistik. Der Weg zur Datenanalyse*. Springer, Berlin, 2002. ISBN 3540440003. URL http://www.amazon.com/Statistik-Weg-Datenanalyse-Ludwig-Fahrmeir/dp/3540440003.

[52] R M Norton. The Double Exponential Distribution: Using Calculus to Find a Maximum Likelihood Estimator. December 2007. URL http://www.jstor.org/pss/2683252.

[53] M B Wilk and R Gnanadesikan. Probability plotting methods for the analysis of data. *Biometrika*, 55(1):1–17, March 1968. ISSN 0006-3444. URL http://www.ncbi.nlm.nih.gov/pubmed/5661047.

[54] Leo Breiman, Jerome Friedman, Charles J. Stone, and R.A. Olshen. *Classification and Regression Trees*. Chapman and Hall/CRC, 1984. ISBN 0412048418. URL http://www.amazon.com/Classification-Regression-Trees-Leo-Breiman/dp/0412048418.

[55] C Gini. Measurement of inequality of incomes. *The Economic Journal*, 31(121):124–126, 1921.

[56] L Breiman. Random Forests. *Machine Learning*, pages 5–32, 2001.

[57] R E Schapire, Y Freund, P Bartlett, and W S Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *The annals of statistics*, 26(5):1651–1686, 1998.

[58] L Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[59] C Cortes and V Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995. ISSN 0885-6125. URL http://dx.doi.org/10.1007/BF00994018.

[60] K Pearson. LIII. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 6*, 2(11):559–572, 1901.

[61] P A Devijver and J Kittler. *Pattern Recognition: A Statistical Approach*. Prentice Hall, 1982. ISBN 0136542360. URL http://www.amazon.com/Pattern-Recognition-Statistical-Pierre-Devijver/dp/0136542360.

[62] L A Herzenberg and R G Sweet. Fluorescence-activated cell sorting. *Scientific American*, 234(3):108–17, March 1976. ISSN 0036-8733. URL http://www.ncbi.nlm.nih.gov/pubmed/1251180.

[63] U Radler. A Disposable Cell Culture Chip for Live Cell Imaging. *AMERICAN BIOTECHNOLOGY LABORATORY*, 22:10–13, 2004.

[64] D V Surbek, C Steinmann, M Bürk, S Hahn, A Tichelli, and W Holzgreve. Developmental changes in adhesion molecule expressions in umbilical cord blood CD34 hematopoietic progenitor and stem cells. *American journal of obstetrics and gynecology*, 183(5):1152–7, November 2000. ISSN 0002-9378. doi: 10.1067/mob.2000.109052. URL http://www.ncbi.nlm.nih.gov/pubmed/11084557.

[65] J M Gillette and J Lippincott-Schwartz. Hematopoietic progenitor cells regulate their niche microenvironment through a novel mechanism of cell-cell communication. *Communicative & integrative biology*, 2(4):305–7, July 2009. ISSN 1942-0889. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2734029&tool=pmcentrez&rendertype=abstract.

[66] A E Frimberger, C L McAuliffe, K A Werme, R A Tuft, K E Fogarty, B O Benoit, M S Dooner, and P J Quesenberry. The fleet feet of haematopoietic stem cells: rapid motility, interaction and proteopodia. *British journal of haematology*, 112(3):644–54, March 2001. ISSN 0007-1048. URL http://www.ncbi.nlm.nih.gov/pubmed/11260067.

[67] R Brown. A brief description of microscopical observations made in the months of June, July and August 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies. *Ann. Phys*, 14:294–313, 1828.

[68] Simulating Brownian Motion - Physics 111-Lab Wiki. URL http://www.advancedlab.org/mediawiki/index.php/Simulating_Brownian_Motion.

[69] G M Viswanathan, S V Buldyrev, S Havlin, M G da Luz, E P Raposo, and H E Stanley. Optimizing the success of random searches. *Nature*, 401(6756):911–4, October 1999. ISSN 0028-0836. doi: 10.1038/44831. URL http://dx.doi.org/10.1038/44831.

[70] S Benhamou. How many animals really do the Levy walk? *Ecology*, 88(8):1962–1969, 2007. URL http://www.esajournals.org/doi/abs/10.1890/06-1769.1?journalCode=ecol.

[71] F Bartumeus, J Catalan, U L Fulco, M L Lyra, and G M Viswanathan. Optimizing the encounter rate in biological interactions: L{é}vy versus Brownian strategies. *Physical review letters*, 88(9):97901, 2002.

[72] A A Potdar, J Jeon, A M Weaver, V Quaranta, and P T Cummings. Human mammary epithelial cells exhibit a bimodal correlated random walk pattern. *PLoS One*, 5(3):e9636, 2010. doi: 10.1371/journal.pone.0009636. URL `http://dx.doi.org/10.1371/journal.pone.0009636`.

[73] M Reynolds. Can spontaneous cell movements be modelled as Lévy walks? *Physica A: Statistical Mechanics and its Applications*, 389(2):273–277, January 2010. ISSN 03784371. doi: 10.1016/j.physa.2009.09.027. URL `http://linkinghub.elsevier.com/retrieve/pii/S0378437109007808`.

[74] F J Diaz. Identifying Tail Behavior by Means of Residual Quantile Functions. November 2007. URL `http://www.jstor.org/pss/1390871`.

[75] The Laplace distribution and generalizations: a revisit with applications to communications, economics, engineering, and finance, 2001.

[76] C M Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2007. ISBN 0387310738. URL `http://www.amazon.com/Pattern-Recognition-Learning-Information-Statistics/dp/0387310738`.

[77] R D Cook. Detection of Influential Observation in Linear Regression. December 2007. URL `http://www.jstor.org/pss/1268249`.

[78] G M James. Generalized linear models with functional predictors, 2002. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.165.1333`.

[79] H T Siegelmann and E D Sontag. Analog Computation Via Neural Networks, 1994. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.53.7608`.

[80] D Opitz and R Maclin. Popular Ensemble Methods: An Empirical Study, 1999. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.36.7341`.

[81] Y Freund and R E Schapire. Experiments with a New Boosting Algorithm, 1996. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.51.6252`.

[82] R M Haralick, Dinstein, and K Shanmugam. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3:610 – 621, 1973. URL `http://www.citeulike.org/user/PeterRabbit/article/80546`.

[83] F Zernike. Diffraction theory of the cut procedure and its improved form, the phase contrast method. *Physica*, 1:56, 1934.