Towards Clinical Translation of Raman Spectroscopy for Tumor Cell Identification

Dissertation

zur Erlangung des akademischen Grades *doctor rerum naturalium (Dr. rer. nat.)*

FRIEDRICH-SCHILLER-
UNIVERSITÄT
JENA

Vorgelegt dem Rat der Chemisch-Geowissenschaftlichen Fakultät der
Friedrich-Schiller-Universität Jena

von M. Sc. Roman Kiselev

Geboren am 17. Juni 1989 in Leningrad

Jena April 2019

1. Gutachter: Prof. Dr. Rainer Heintzmann, Leibniz-IPHT, Jena

2. Gutachter: Prof. Dr. Jürgen Popp, Leibniz-IPHT, Jena

Tag der öffentlichen Verteidigung: 23.01.2019

Summary

Early cancer diagnostics is a major challenge for modern medicine. Tumor cells are malfunctioning tissue cells that feature an increased metabolism. They grow, divide and disseminate throughout the body without being recognized as a danger by the immune system. The spread of cancer cells takes place via the bloodstream in the form of circulating tumor cells (CTC). CTC detection in blood samples is of great medical importance, since the identification of such CTCs can also reveal the type of cancer. However, the concentration of circulating tumor cells in the blood remains very low, even at later stages of cancer, making their detection very challenging.

Detection of the molecular changes in CTC genome and proteome (the so-called *omics* data) makes cancer diagnosis more reliable and provides important data for cancer research. These data can be extracted from individual cancer cells using modern techniques of DNA sequencing, but these specific methods are slow and require undamaged, living cancer cells. Therefore, many highly sensitive marker-based detection methods such as for example immunohistochemistry (antibody staining), or magnetic cell separation using antibody-coated magnetic particles, are unsuitable for enrichment of CTCs for subsequent molecular analysis.

One solution is offered by new microfluidic enrichment methods. These label-free procedures allows separation of cells into fractions within specially-shaped microfluidic channels. The principle is based on the fact that the cancer cells show slightly different physical properties, such as size, density or elasticity, than healthy blood cells. However, the specificity of these techniques is not sufficient for the subsequent molecular analysis: although the concentration of CTC can be increased by several orders of magnitude, the cell suspension is not yet pure enough.

A promising intermediate step is *Raman spectroscopy*: a label-free and non-destructive technique that uses a focused laser beam to probe the chemical composition of individual cells. Using Raman spectra, it is possible to detect molecular changes in cells with chemometric data analysis methods, and thus to identify circulating tumor cells. A major challenge, however, is the multivariate structure and complexity of the data. Automated cell recognition requires machine learning techniques, which for their high performance need a large amount of annotated samples.

Many bio-spectroscopic studies suffer from insufficient amount of statistically independent samples. Data complexity makes it difficult to store all relevant experimental data in a form that correctly represents the hierarchical data structure (e. g. patients, blood samples, individual cells, preparation methods, etc.).

This PhD thesis covers technological developments around Raman-based CTC identification. First, I present a design of custom application-specific microspectroscopic instruments that offer great flexibility, and allow different experimental parameters to be varied. Furthermore, I developed a special device control software to automate the experimental setups. It provides an interface to enter the details of the experiment, and allows to preselect cells for the measurement, after which the Raman spectra of the cells are automatically recorded and stored. As a result, the throughput of the Raman instrument could be significantly increased, and the subjective influence of the instrument operator could be minimized.

Additionally, I developed an SQL database for organized storage of experimental data. The core database structure reflects the hierarchy of the bio-spectroscopic experiment, so that the spectra can be searched using any piece of information in their metadata, divided into groups and compared with each other.

With the automated measurement systems and database, my colleagues and I were able to record about 40.000 Raman spectra from more than 15 incubated cancer cell lines, from healthy donated leukocytes, as well as from samples originating from 48 individual patients. Although the classification models trained with the measured data could not differentiate between individual cancer cell lines, they were able to recognize tumor cells among healthy leukocytes with prediction accuracy of more than 95%. Using the trained classifiers we were able to detect CTCs in patient samples. Finally, we demonstrated the entire workflow of cancer cell detection and isolation for a subsequent DNA sequencing.

Moreover, I present new algorithms for calibration of the wavenumber axis in dispersive spectrometers, as well as for detection of outliers in Raman spectra. The results of this work provide a basis for the clinical translation of Raman spectroscopic detection of circulating tumor cells.

Zusammenfassung

Auf dem Weg zu klinischen Anwendungen der Raman-Spektroskopie für die Tumorzellerkennung

Frühzeitige Krebsdiagnostik ist eine große Herausforderung für die moderne Medizin. Tumorzellen sind fehlfunktionierende Gewebezellen, die einen hohen Metabolismus aufweisen und unkontrolliert im Körper wachsen, teilen und ausbreiten, ohne als Gefahr vom Immunsystem erkannt zu werden. Die Verbreitung der Krebszellen findet über den Blutkreislauf im Form von *zirkulierenden Tumorzellen*, auch CTC genannt (engl. *circulating tumor cells*), statt. Die CTC-Erkennung in Blutproben hat eine große medizinische Bedeutung, da die Identifikation solcher CTC neben der Detektion von Krebs auch Rückschlüsse auf die Tumorart geben kann. Allerdings bleibt die Konzentration der zirkulierenden Tumorzellen im Blut auch bei späteren Krebsphasen sehr niedrig, was deren Erkennung stark erschwert.

Die Kenntnisse über die molekularen Änderungen an Genom und Proteom (sog. *omics*-Daten) der CTC machen eine Krebsdiagnose zuverlässiger und sind auch für die Krebsforschung wichtig. Diese Daten können aus einzelnen Krebszellen mittels modernen Techniken der DNA-Sequenzierung extrahiert werden, allerdings sind diese spezifischen Methoden langsam und setzen unbeschädigte, lebendige Krebszellen voraus. Daher sind viele hoch sensitive Marker-basierte Detektionsmethoden, wie z.B. Immunhistochemie (Antikörperfärbung), oder magnetische Zellseparation mittels Antikörper-beschichteten Magnetpartikeln, ungeeignet, um CTC für molekulare Analyse anzureichen.

Eine Lösung bieten neuartige mikrofluidische Anreicherungsmethoden an. Diese markierungsfreie Verfahren ermöglichen eine Trennung der Zellen in Fraktionen innerhalb speziellgeformten Mikrofluidikkanals. Das Prinzip basiert auf der Tatsache, dass die Krebszellen etwas andere physikalische Eigenschaften, wie Größe, Dichte oder Elastizität zeigen, als gesunde Blutzellen. Allerdings ist die Spezifizität dieser Methoden für die darauffolgende molekulare Analyse nicht ausreichend. Obwohl die Konzentration der CTC um einige Großenordnungen steigt, ist die Zellsuspension noch nicht rein genug.

Als ein vielversprechender Zwischenschritt eignet sich die *Raman-Spektroskopie*. Diese Methode ist eine Markierungsfreie und nicht zerstörende Anwendung, die mit einem fokussierten Laserstrahl die chemische Zusammensetzung einzelner Zelle bestimmen kann. Anhand der Raman-Spektren ist es möglich, mittels chemometrischen Methoden molekulare Änderungen in den Zellen nachzuweisen, und so die zirkulierenden Tumorzellen zu erkennen. Eine große Herausforderung stellt allerdings die multivariable Struktur und Komplexität der Daten dar. Für eine automatische Zellerkennung sind Techniken des maschinellen Lernens notwendig, die für eine gute Performance wiederum eine große Menge der annotierten Proben voraussetzen.

Viele biospektroskopische Studien leiden an der unzureichenden Menge der statistisch-unabhängigen Proben. Die Datenkomplexität macht es schwierig, alle relevanten experimentellen Daten in eine Form zu speichern, die die hierarchische Datenstruktur (z.B. Patienten, Blutproben, individuelle Zellen, Vorbereitungsmethoden, usw.) korrekt abbildet.

Diese Arbeit umfasst technologische Entwicklungen rund um die Raman-basierte Erkennung der CTC. Zunächst wird das Design der anwendungsspezifischen mikrospektroskopi-

schen Geräte beschrieben, welche eine höhere Flexibilität anbieten und es ermöglichen, unterschiedliche experimentelle Parameter zu variieren.

Außerdem automatisierte ich die Aufbauten, in dem ich im Rahmen dieser Dissertation eine spezielle Steuerungssoftware entwickelte. Sie macht es möglich, die Einzelheiten zum Experiment anzugeben und die Zellen für die Messung vorauszuwählen. Danach werden die Spektren der Zellen automatisch aufgenommen und gespeichert. Dadurch konnte der Durchsatz an gemessenen Proben deutlich erhöht werden, und der subjektive Einfluss der messenden Person vermieden werden.

Weiterhin entwickelte ich eine SQL-Datenbank für eine organisierte Aufbewahrung der experimentellen Daten. Die grundlegende Datenbankstruktur spiegelt die Hierarchie des biospektroskopischen Experiments wieder, so dass die Spektren nach allen relevantenn Metadaten durchsucht, in Gruppen unterteilt und miteinander verglichen werden können.

Mit den automatisierten Messsystemen und der Datenbank waren meine Kolleginnen und ich in der Lage, ungefähr 40'000 Raman-Spektren von mehr als 15 kultivierten Krebszelllinien, von gesunden gespendeten Leukozyten, sowie von Patientenproben, die von 48 individuellen Patienten stammen, aufzunehmen. Die mit den gemessenen Daten trainierten Klassifikationsmodellen konnten zwar nicht zwischen individuellen Krebszelllinien unterscheiden, zeigten aber eine durchaus höhere Genauigkeit von über 95% bei der Detektion der Tumorzellen unter gesunden Leukozyten. Ferner wurde auch die CTC-Erkennung in Patientenproben gezeigt und das gesamte Verfahren von Krebszelldetektion und Isolierung für die darauffolgende Sequenzierung demonstriert.

Außerdem präsentiere ich neue Algorithmen für die Kalibrierung der Wellenzahlachse in dispersiven Spektrometern, sowie für die Detektion von Ausreißern in Raman-Spektren. Die Ergebnisse dieser Arbeit sind eine Grundlage für die klinische Translation der Ramanspektroskopischen Erkennung der zirkulierender Krebszellen.

Автореферат Диссертации

На пути к клиническому применению Рамановской спектроскопии для распознавания раковых клеток

Ранняя диагностика рака является сложной задачей для современной медицины. Опухолевые клетки возникают из повреждённых клеток ткани, которые имеют повышенный метаболизм и беспрепятственно растут, делятся и распространяются в организме, не будучи распознанными иммунной системой как опасность. Они распространяются через кровоток в виде *циркулирующих раковых клеток*, также известных как CTC (англ. *circulating tumor cells*). Обнаружение CTC в образцах крови имеет большое медицинское значение, поскольку идентификация таких CTC также может помочь проводить мониторинг рака после его обнаружения. Однако концентрация CTC в крови остается очень низкой даже на более поздних стадиях рака, что серьёзно затрудняет их обнаружение.

Знание молекулярных изменений генома и протеома CTC (так называемые *omics* данные) делает диагностику рака более надежной и предоставляет важную информацию для онкологических исследований. Эти данные могут быть получены из отдельных раковых клеток с использованием современных методов секвенирования ДНК. К сожалению, эти точные и специфичные методы являются медленными и должны применяться к неповрежденным, живым раковым клетками. Поэтому многие высокочувствительные методы обогащения, которые основаны на иммуногистохимии (окрашивание с помощью специфичных антител) или на разделении клеток с использованием магнитных частиц, покрытых антителами, непригодны для этой цели.

Одно из решений предлагают новые методы микрофлюидики, которые позволяют разделять клетки на фракции без использования каких-либо маркеров при протекании суспензии с клетками через каналы особой формы. Принцип основан на том факте, что раковые клетки имеют несколько отличные от здоровых клеток физические свойства, такие как размер, плотность или эластичность. Однако специфичность этого метода недостаточна для последующего биохимического анализа. Хотя концентрация CTC увеличивается на несколько порядков, клеточная суспензия всё еще содержит малое количество CTC.

Перспективным промежуточным этапом является *Рамановская спектроскопия*, также известная под названием *спектроскопия комбинационного рассеяния*. Этот метод позволяет без использования маркеров при помощи сфокусированного лазерного луча проводить неразрущающий анализ отдельных клеток. Спектры содержат информацию о биохимическом составе клеток, которая может быть извлечена с помощью хемометрических методов. Одной из проблем, связанных с этим методом, является мультивариатная структура и сложность данных. Автоматическое распознавание клеток основывается на методах машинного обучения, которые, в свою очередь, для надёжных предсказаний требуют большое количество аннотированных обучающих данных.

Многие биоспектроскопические исследования страдают от недостаточного количества статистически независимых образцов. Сложная структура экспериментальных метаданных затрудняет их хранение и организацию в форме, которая правильно отображает иерархическую структуру эксперимента (индивидуальные пациенты, независи-

мые образцы крови, отдельные клетки, разные методы подготовки и т. д.).

Эта диссертация охватывает новые технологии для обнаружения циркулирующих раковых клеток при помощи Рамановской спектроскопии. Во-первых, описывается разработка новых клинических микроскопов для Рамановской спектроскопии, которые обеспечивают большую гибкость и позволяют варьировать различные экспериментальные параметры.

Во-вторых, в рамках диссертации я разработал специальное программное обеспечение для управления лабораторным оборудованием. Оно позволяет пользователю указать параметры, с которыми проводится эксперимент, а также выбрать клетки, спектры которых должны быть измерены. После этого прибор автоматически записывает спектры всех выбранных клеток и сохраняет их в базу данных. Эти разработки позволили значительно упростить и ускорить измерение большого количества образцов, а также позволили минимизировать субъективное влияние человека на результаты измерений.

Также я разработал SQL базу данных для организованного хранения экспериментальных данных. Основная структура базы данных отражает иерархическую структуру эксперимента, так что по любым из сохранённых метаданных можно проводить поиск спектров, их можно легко разбивать на группы и сравнивать друг с другом.

Используя автоматизированные измерительные системы и базы данных, я и мои коллеги смогли зарегистрировать спектры комбинационного рассеяния нескольких десятков тысяч разнообразных клеток. Конкретно речь идёт о более чем пятнадцати культивируемых линий раковых клеток, лейкоцитах от здоровых доноров, а также об образцах крови от 48 отдельных пациентов. Хотя классификационные модели, обученные на наших экспериментальных данных, не были способны различать отдельные линии раковых клеток, они показали высокую точность (более 95%) при обнаружении циркулирующих раковых клеток среди здоровых лейкоцитов. Кроме того, мы показали идентификацию CTC в образцах крови от пациентов, а также успешно продемонстрировали всю процедуру обнаружения и экстрагирования раковых клеток из смешанного образца для последующего секвенирования.

Кроме того, в рамках данной работы я разработал новый алгоритм калибровки оси абсцисс для дисперсионных спектрометров, а также алгоритм для обнаружения выбросов в спектрах комбинационного рассеяния. Результаты этой работы формируют основу для будущих клинических применений Рамановской спектроскопии для диагностики рака.

Contents

Glossary

AES	*Atomic Emission Spectra*
ADM	*Automated Digital Microscopy*
API	*Application Programming Interface*
CAD	*Computer-Aided Design software*
CanDo	*Cancer Development Monitor* is a collaborative research project with the "aim to develop a small technical lab-on-a-chip device that isolates and analyses circulating tumour cells (CTCs) from peripheral blood" [1]
CD44	*CD44 antigen* is a cell-surface glycoprotein involved in intercellular interactions, cell adhesion and migration
CD45	*CD45 antigen* also known as *Protein tyrosine phospatase* or *leukocyte common antigen*
CI	*Confidence interval*
CTC	*Circulating Tumor Cell*
DBMS	*Database Management System*
DPSS	*Diode-Pumped Solid-State laser*
EMSC	*Extended Multiplicative Signal Correction* is a method of baseline correction for spectral data, originally presented by Martens *et al.* [2]
EpCAM	*Epithelial cell adhesion molecule*
FAST	*Fiber-optic Array Scanning Technology* [3]
FDA	*U.S. Food and Drug Administration*
FWHM	*Full Width at Half Maximum*
GUI	*Graphical User Interface* of a computer program
MMP	*Matrix metalloproteinases*
MOSFET	*Metal-oxide-semiconductor field-effect transistor*
LSM	*Laser Scanning Microscope*
NGS	*Next Generation Sequencing*
NIST	*National Institute for Standartization*
mRNA	*Messenger RNA*
ML	*Machine Learning*

PBS *Physiological Buffer Saline*, a 0.9% NaCl solution

PCR *Polymerase Chain Reaction* – a technique to amplify a number of DNA molecules or their fragments by several orders of magnitude

PEEK *Polyether ether ketone*

PMT *Photomultiplier tube* is a sensitive detector often used for single photon detection

PSF *Point Spread Function* is the image of the point light source formed by the optics of a microscope

PTFE *Polytetrafluoroethylene*, also known as *Teflon*

RACS *Raman-activated cell sorting*

RamanCTC *RamanCTC* is a collaborative research project for Raman-based CTC detection, isolation and characterization using microfluidics and microarrays.

qPCR *Quantitative real-time PCR* – a common technique used to detect gene expression

RNA *Ribonucleic acid*

RS660 *Raman System from company* Till ID *with 660 nm excitation laser, also known as "Raman Reader" (see Section 3.2.2)*

RS785 *Raman System with 785 nm excitation laser and a custom microscope, also known as "Roman Raman" (see Section 3.2.3)*

SNR *Signal-to-noise ratio*. Typically the SNR in spectroscopy is defined as the signal at a given wavelength λ divided by its variance across several independent measurements

UUID *Universally unique identifier*, like `42d4d72e-2316-409f-bde1-4744880dbdd6`. It is a 128-bit number used to identify information in computer systems

1. Introduction

In the modern world, cancer is one of the leading natural causes of death [4, 5]. Besides genetic predisposition, there are several external factors, such as alcohol, smoking, ionizing radiation, unhealthy diet, obesity and lack of physical activities, which strongly correlate with the occurrence of many cancer types [6–12]. A healthy lifestyle and a balanced diet can greatly reduce the risk of cancer and thus contribute to the average life expectancy in the society. Another way towards a better public health is an early diagnosis of cancer. Early diagnosis allows to massively reduce costs and complexity of the tumor treatment, as well as directly influences the patient survival prognosis. Unfortunately, it remains one of the big challenges in the medicine [13, 14].

An oncological disease starts as a malfunction of individual cells in the organism and often slowly develops over a long period of time without showing any symptoms. In many cases, the first symptoms of cancer appear when the tumor has already progressed into a highly malignant form, and it is too late for an effective treatment. Because of this, modern early diagnostic techniques have to deal with a single cell analysis [15, 16]. A number of methods are available in this field, each with its own advantages and limitations, and sometimes applicable to specific cancer types only, as discussed in the review of Galler *et al.* [17].

1.1. Circulating Tumor Cells

Extraction of the cells or tissues from an organism for a medical test is called *biopsy*. It is an important cancer diagnostic tool, because extracted tissues can be studied with any available histopathologic methods. Unfortunately, biopsy is a highly invasive procedure with complication risks; it should preferably not be used repeatedly for tracking of the disease progression or the treatment efficiency. However, under certain circumstances the tumor cells can also be present in a vascular or lymphatic system. The first observation of CTCs[1] is attributed to Ashworth [18], who observed cells in blood of cancer patients that are morphologically similar to those seen in tumors. Thus, CTCs can be obtained from patient's blood. The corresponding procedure, often referred to as "liquid biopsy", holds promise for the future diagnostic applications and paves the way to a targeted personalized therapy. If it were possible to analyze the tumor at the molecular level and monitor the disease progression using a simple blood sample, then the treatment of cancer patients would be substantially improved.

Studies confirmed, that CTCs indeed originate from the primary tumor sites and not only look similar to the primary tumor cells, but also have the same phenotypic malignant properties as their parent cells [19, 20]. Upon invading into the lymph or blood vessels, cancer cells begin to circulate around the body, acting as seeds for new metastatic sites (Figure 1.1).

[1] *Circulating Tumor Cells*

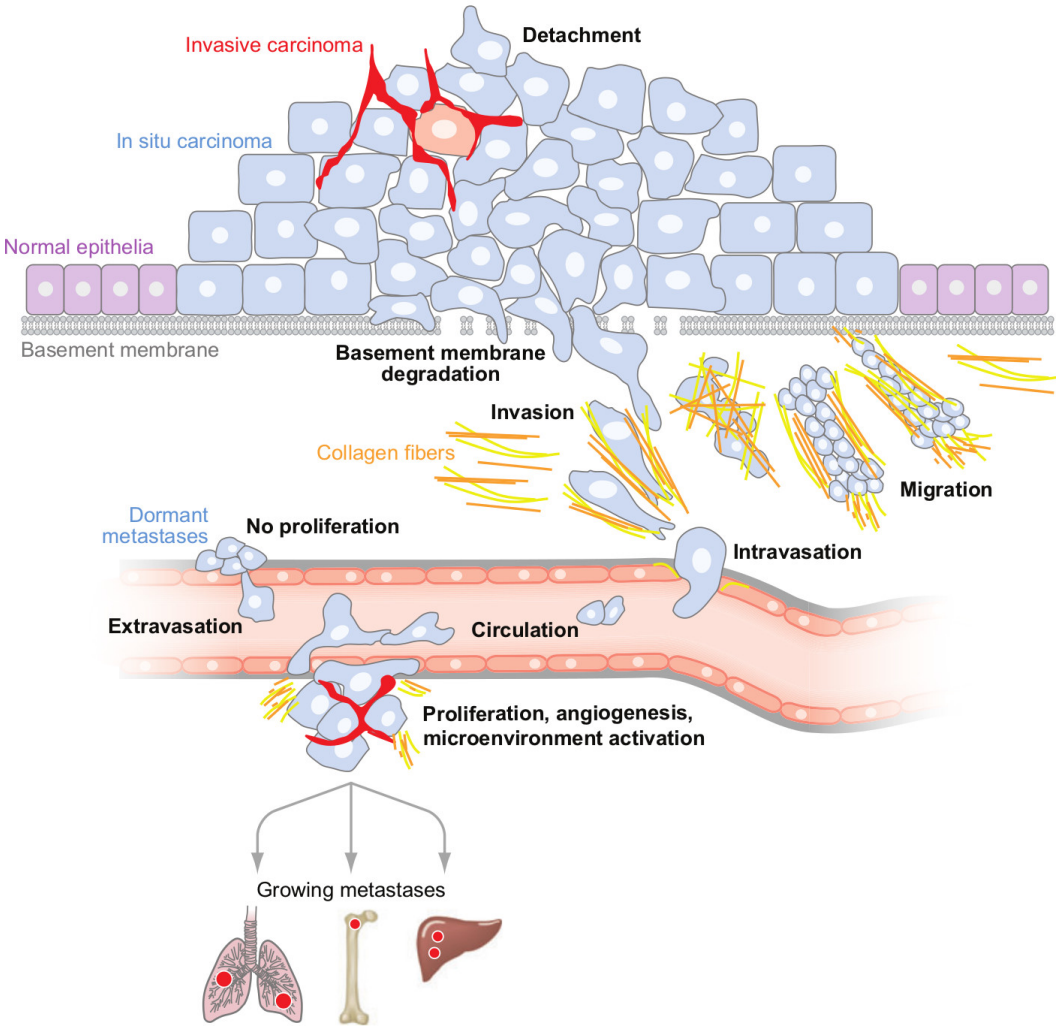


Figure 1.1.: "Principal steps in metastasis. Transformation of normal epithelial cells leads to carcinoma *in situ*, which, as a result of loss of adherent junctions, evolves toward the invasive carcinoma stage. Following basement membrane degradation, tumor cells invade the surrounding stroma, migrate and intravasate into blood or lymph vessels, and become transported until they arrest in the capillaries of a distant organ." Reprinted with permission from Bacac & Stamenkovic [16].

A comprehensive review about tumor cells, in particular CTCs, is provided by Bacac & Stamenkovic [16]. From the cellular point of view, the metastasis is a very inefficient process. Fidler [21] suggests that the absolute majority (about 99%) of tumor cells die within the first 24 hours, and among the cells that have detached from the primary tumor, only about 0.01% can form metastases. A metastatic cell requires special complex capabilities that allow it to disseminate. Most important of them are (1) the ability to disrupt the barriers that keep normal cells enclosed in a defined tissue compartment, and (2) the ability to alter the host micro-environment in such a way, that the metastatic cell is perceived as a normal one and is

supplied with required resources. The host can even be compelled to create new blood vessels in the tumor tissues, which supply the tumor with extra oxygen, hormones and glucose for a more progressive growth [22].

Bio-molecular changes play a key role in the abilities of cancer cells to metastasize. For example, experimental evidence suggests that the CD44[2] molecule enhances cell migration and survival. It also orchestrates the assembly of various classes of molecules, such as MMP[3], which mediates degradation of the basement membrane – one of the layers keeping cells confined. EpCAM[4] is another example of a molecule that is normally not present in the blood, but is typically found within CTCs. Therefore, many of the detection methods for CTCs target specific bio-molecules, called cancer diagnostic markers, that are associated with metastatic cells.

1.2. Detection Methods of Circulating Tumor Cells

CTCs are very rare cells and fall into a group of so-called *low-abundance cells*, i. e. cells whose concentration is below 1000 mL^{-1}. It is estimated that in a single milliliter of whole blood from patients suffering of a metastatic cancer, which contains about 5×10^9 erythrocytes and 10^7 leukocytes, **less than ten** CTCs can be found [23–25]. Therefore, several enrichment methods have emerged, which allow to increase the concentration of CTCs in a blood sample. Some of them are based either on physical properties of the CTCs, such as lower density (gradient centrifugation method [26]) or bigger diameter (filtration of blood through a pored membrane [27]). Others rely on binding of antibodies to epithelial-specific agent (such as EpCAM) or nucleic acid-based methods that detect genes of interest (for example the **RT-PCR!**[5] that target rapidly-degrading mRNA[6] molecules [28]).

A variety of research methods developed to isolate and enumerate CTCs are discussed in more detail elsewhere [3, 29–32]. Table 1.1 summarizes several common CTC enrichment and separation methods. As of 2017, the only methodology for enumeration of CTC in whole blood cleared by the FDA[7] (however, only for breast, colorectal and prostate cancer) is the *CellSearch* system. It is based on a selection of CTCs with ferrofluid-coupled epithelial markers (which are then manipulated by magnetic field) combined with a depletion of leukocytes by targeting the CD45[8] enzyme [24, 29]. The major limitations of *CellSearch* are relatively low sensitivity and cell recovery rates. This issue was addressed in the *IsoFlux* system [33] that combines (a) high-sensitivity cell isolation using functionalized magnetic beads with (b) a novel cell retrieval method: external magnetic field attracts the targeted cells to a polymer disk placed into a fluid channel, the disk with attached cells is transferred into another test tube, and the cells get released once the magnet is removed.

Another reliable method for CTC detection is ADM[9], which recognizes labeled cells in micro-

[2] *CD44 antigen* is a cell-surface glycoprotein involved in intercellular interactions, cell adhesion and migration
[3] *Matrix metalloproteinases*
[4] *Epithelial cell adhesion molecule*
[5] **RT-PCR!**
[6] *Messenger RNA*
[7] *U.S. Food and Drug Administration*
[8] *CD45 antigen* also known as *Protein tyrosine phospatase* or *leukocyte common antigen*
[9] *Automated Digital Microscopy*

scope images [34]; an example of a commercial ADM system is *Ariol* by *Leica Biosystems*. ADM, whose speed is constrained by the movement of the sample due to a limited field of view, is capable of recognizing of about 800 cells per second. An even faster technique developed by Hsieh *et al.* [3], incidentally called FAST[10], offers an astonishing detection rate of up to 3×10^5 cells per second directly on slide without need for any enrichment steps. This method is similar to ADM in that the fluorescent-labeled images are analyzed, but the images are collected with a large fiber bundle that is linearly scanned over the slide at the rate of 3 mm/s.

In contrast to *in vitro* methods, which enumerate CTCs in a blood sample taken from the patient, some researchers focus on CTC capture *in vivo* – directly in a blood vessel. An example of such workflow is the *CellCollector* by Gilupi – a stainless steel wire functionalized with anti-EpCAM antibodies that is temporarily injected into a blood vessel [35–37]. CTCs get captured by the 2.7-cm long functionalized part of the wire if they come into contact with it. This method was approved by The China Food and Drug Administration in 2017. *In vivo* methods, being used inside the body, allow to analyze much larger volumes of blood. However, there are several issues associated with the method, such as avoiding contamination of the wire with epithelial cells during the injection, and a gentle release of the collected cells from the wire upon its retrieval. Optical interrogation of the cells attached to the wire surface is very challenging from the technical point of view, because of the cylindrical shape of the wire and its bending. Finally, *CellCollector* captures high amount of cell debris alongside with CTCs.

Recently several new microfluidic cell separation methods emerged, which avoid the use of biochemical labels and utilize intrinsic physical cell properties for the cell sorting. A detailed overview of these techniques can be found in the review of Gossett *et al.* [38], who discusses methods based on dielectrophoresis [39], gravity-driven cell fractionation [40], acoustic separation, use of micro-pored membranes or micro-post arrays, etc.

Particularly interesting are high-throughput passive membrane-free microfluidic cell sorting techniques based solely on the hydrodynamic phenomena produced by the channel geometry. One of them uses a specially-designed herringbone structure in a micro-channel that creates a lateral pressure gradient separating particles by size [41]. Another powerful method uses centrifugal forces and a transverse *Dean vortex* acting on the cells that move in a spiral micro-channel to separate them into fractions of different sizes [42, 43]. These hydrodynamic techniques eliminate clogging issues associated with filtering membranes, require only a minimal sample preparation, and do not rely on any labels or external forces, which allows them to reach unprecedented throughput.

An efficient strategy to detect and enumerate CTCs for diagnostic purposes would involve an appropriate combination of methods that target physical properties of the cells with techniques that rely on their biochemical markers. Although methods from the first group are label-free and sensitive, they lack specificity. The second group offers more specific methods, but they typically rely on functionalized labeling molecules. However, usually it is desirable to release and recover captured viable CTCs for a downstream molecular analysis, such as single-cell **RT-PCR!** or NGS[11], which makes the majority of the more specific biochemical strategies inappropriate, as the labels can influence the molecular analysis.

[10] *Fiber-optic Array Scanning Technology* [3]
[11] *Next Generation Sequencing*

Table 1.1.: Methods for CTC enrichment and separation.

Method & Reference	Principle for enrichment	Principle for detection	Limitations	Vendor
CellSearch [24, 29]	Ferrofluids containing EpCAM antibodies	Immunodetection of *cytokeratin 8, 18, 19*	Dependent on EpCAM	Veridex
ISET	Size of CTC	Immunodetection of EpCAM	Size is highly variable	RareCells
Adna test	Immunomagnetic detection of EpCAM	RT-PCR of tumor related transcripts	Dependent on EpCAM, many false positives	Adnagen
Ariol	EpCAM or cytokeratins	Immunodetection of *cytokeratin 8, 18, 19*	Cells are no longer viable	Leica Micro systems
FAST [3]	– *no enrichment* –	Immunofluorescent labeling	No enrichment, only detection. Dependent on labeling	
EPISPOT	Antigen expression	Enzymatic activity	Enzymatic activity varies	
MACS [32]	Magnetic cell sorting using super paramagnetic beads	Specific surface markers	Dependent on surface markers	Miltenyi Biotec
CTChip FR [43]	Size-based cell separation in a spiral micro-channel	Centrifugal forces and splitting of a lateral flow	Minimum CTC size is about 10 μm. Low specificity	Clearbridge BioMedics
CellCollector [35–37]	*In-vivo* cell capture on a functionalized wire	Immunodetection of EpCAM	Dependent on EpCAM, captured cells have to be released	Gilupi
DepArray [39]	Dielectrophoretic force	Polarizability and size of cells	Oxidation of electrodes, free radical generation	Silicon Biosystems
IsoFlux [33]	Anti-EpCAM pre-conjugated magnetic beads	Immuno-magnetic enrichment	Method is not label-free	Fluxion Biosciences

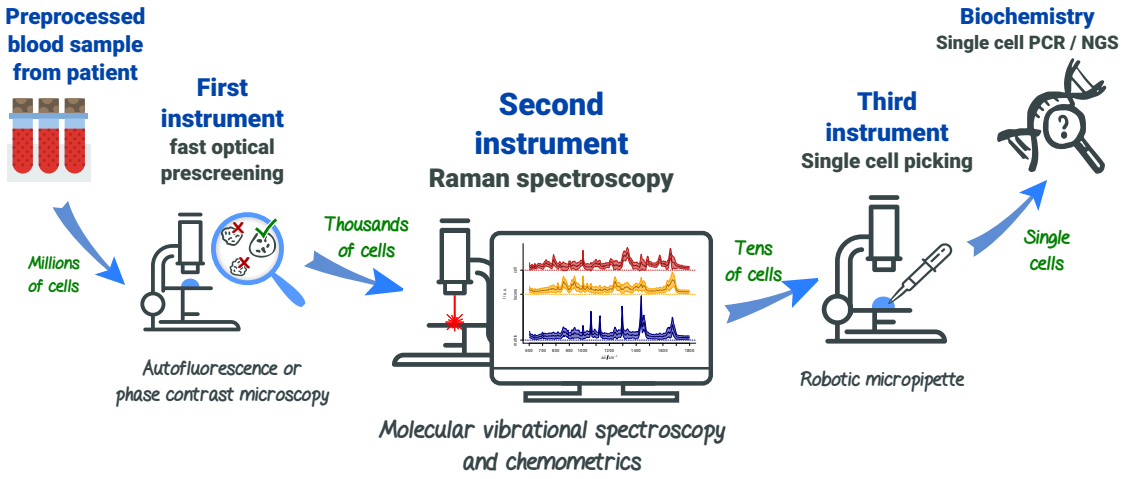


Figure 1.2.: Overview of the RamanCTC project strategy to isolate individual CTCs from a blood sample.

Because of these limitations, there is a need for a fully label-free cancer diagnostic strategy, that is both sensitive and specific. Raman spectroscopy, described below, is a promising candidate for this application, as it can optically and non-destructively detect variations in the biochemical composition between individual cells. It has, however, a limited throughput and thus cannot be applied as a clinical tool for the whole blood analysis. This issue can be circumvented by preceding the spectroscopic interrogation with a fast and sensitive label-free screening technique that results in at least 100-fold enrichment. 100-fold enrichment would bring low-abundant CTCs from the initial concentration of 10^{-6} to 10^{-4}, which is a manageable number for a Raman-based cell identification.

These considerations led to the RamanCTC[12] project with the goal to develop a fully label-free CTCs detection and characterization workflow. The framework is based on a sequential processing of the sample with several instruments. Each subsequent instrument has lower throughput than the previous one, but produces much more specific results.

Figure 1.2 gives an overview of this approach. Detection of the CTCs is like searching for a needle in a haystack: among millions of healthy cells one has to correctly identify a single malignant one [23]. The tool chain starts with a fast digital microscope that can quickly collect images of tens or hundreds thousands of cells and, using automatic image processing, select the suspicious cells among them. This pre-selection is based on the fact, that typically CTCs are much bigger than healthy cells (see, for example, Image 5.22 on page 115).

Next, thousands of suspicious cells can be analyzed with Raman-spectroscopic techniques. The cell identification is based on the differences in biochemical composition, and offers much higher specificity in comparison to the image analysis methods. This task is accomplished by the second instrument of the tool chain. On this stage the number of the suspicious cells is further reduced by several orders of magnitude.

The identified tumor cells can be extracted from the sample with a robotic micro-pipette

[12] *RamanCTC* is a collaborative research project for Raman-based CTC detection, isolation and characterization using microfluidics and microarrays.

(*CellCelector* from company *ALS*). This instrument is capable of picking an individual cell from the substrate using a glass micro-capillary attached to a microfluidic syringe. This way the suspicious cell gets separated from all other cells in a the sample. This enables biochemical interrogation of individual cells using techniques such as single-cell RT-PCR! for analysis of gene expression patterns in a given cell.

An alternative proposal for a Raman-based CTC isolation method uses a special quartz microfluidic chip named "*RoC3*". The chip contains microfluidic circuits for cell injection, a fiber optic interface for microscope-free cell trapping and spectroscopic interrogation, as well as two outputs for cell sorting [44]. The details about this chip are discussed in Section 3.4.3 on page 47.

The following section provides a physical background of the Raman effect, discusses capabilities and limitations of this technique, as well as presents a short review of its biomedical applications.

1.3. Motivation and Goals of the Work

The aim of this work was the translation of Raman spectroscopy towards clinical applications, in particular Raman-based identification and sorting of CTCs. In this scope, the goal was a development of a prototype of a clinical Raman instrument – a platform that combines efficient cell handling, label-free optical interrogation using Raman spectroscopy, and machine learning algorithms for an automated identification of the tumor cells followed by their physical separation from the other cells in a blood probe. Further, the performance of the developed instrument, its usability and applicability for studies of clinical samples had to be demonstrated.

The thesis is organized in a way that reflects the aforementioned goals and their realization from many aspects, including optical systems and spectroscopic instruments, data management techniques, and methods of data processing and analysis. All these methods, integrated together in a single platform, should result in a faster and easier identification of the tumor cells.

Chapter 3 is dedicated to the experimental methods. It starts with the description of the spectroscopic instrumentation, including general design considerations for Raman systems, overview of laboratory setups that we built, and discussion of their characteristics. It continues with the topic of microfluidic technologies. Microfluidic methods enable convenient and reliable handling of thousands of cells for their successive analysis in a timely fashion. Several different microfluidic chips were investigated in the scope of this work.

Chapter 4 focuses on efficient data management techniques combined with an automated computerized control of the hardware. These technologies drastically improve the throughput of a biospectroscopic experiment. The chapter discusses the use of databases to store experimental results, different representation of data, algorithms for automatic spectrometer calibration, as well as a software architecture that enables an efficient data management.

Overview of the data set collected in the scope of this work is given in Chapter 5. It also discusses the analysis methods of the spectroscopic data, such as algorithms for the detection of outliers, preprocessing of spectra, dimensionality reduction, training and validation of

chemometric models for cell identification, and automatic image processing for cell segmentation and enumeration. Moreover, the results of automated CTC identification in patient samples are presented and discussed. The chapter finishes with a demonstration of the CTC detection workflow that was introduced before (Figure 1.2).

2. The Raman Effect

2.1. Short Theory of Raman Scattering

Raman effect was discovered by the Indian scientist C. V. Raman in 1928 [45]. This is a phenomenon of inelastic photon scattering on the modes of molecular vibrations. Although most of the photons interact elastically with the matter (Rayleigh scattering), there is a small probability of energy exchange between them. As a result, the photon is re-emitted at another wavelength. The energy transfer from the photon to the molecules, which leads to the excitation of vibrational states and red-shifting of the photon, is called *Stokes* process. The energy transfer from the excited molecule to a scattering photon is called anti-Stokes process. These three cases, and the corresponding Jablonski energy diagrams, are shown in the Figure 2.1.

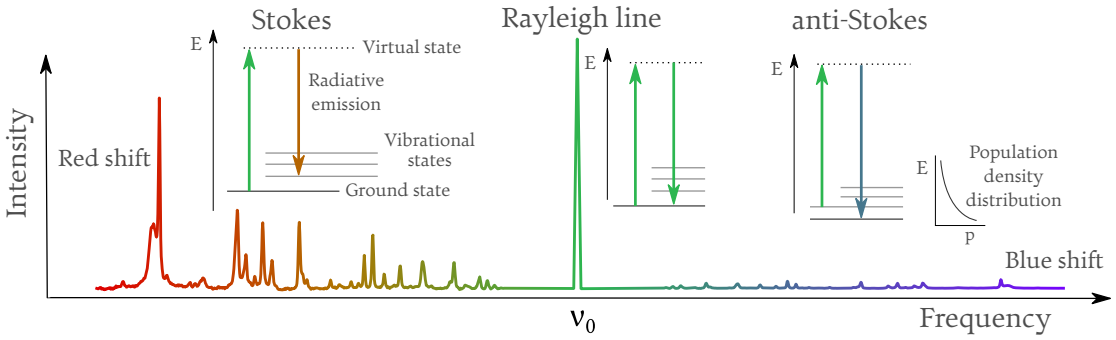


Figure 2.1.: Band diagrams of Raman effect. Stokes, anti-Stokes and Rayleigh signals are results of interactions between photons and modes of molecular vibrations. Shown here is the Raman spectrum of aspirin. The curve relating energy E and population density p is the Boltzmann energy distribution, which makes anti-Stokes process less probable than the Stokes scattering.

The basic aspects of the Raman effect can be described with the classical laws of electrodynamics and and a model of quantum harmonic oscillator [46]. The molecule interacting with an electromagnetic wave becomes an induced electric dipole with the vector of the linear dipole moment given by

$$\vec{p}^{(1)} = \hat{\alpha} \cdot \vec{E}, \tag{2.1}$$

where $\vec{E}$ is the electric field of the incident plane monochromatic wave oscillating with frequency ω_1, and $\hat{\alpha}$ is the polarizability[1] tensor of the molecule. It is convenient to consider a diatomic molecule as a simple example.

[1]Polarizability is the ability of the molecule to form instantaneous dipole in response to the external electric field. The related term describing the property of a bulk material to polarize is called *electric susceptibility* χ_e.

Only the electrons are responsible for the molecular polarizability, but their distribution in the molecule depends on the location of the atomic nuclei. The Born–Oppenheimer approximation allows us to treat the motion of nuclei and electrons independently, as the electrons are much faster and always adopt to the locations of the charged nuclei. To describe the linear Raman effect, the polarizability tensor $\hat{\alpha}$, which is in general non-linear, can be expressed as the sum of a constant and a linear term with respect to the normal coordinate[2] Q_k of molecular vibration:

$$\hat{\alpha} = \hat{\alpha}_0 + \hat{\alpha}'_k Q_k \qquad \text{with} \quad \hat{\alpha}'_k = \frac{\partial \hat{\alpha}}{\partial Q_k} \tag{2.2}$$

where k is the index of one of many possible normal modes (in general, a molecule with N atoms has $3N - 6$ normal modes, some of which can degenerate due to the symmetry).

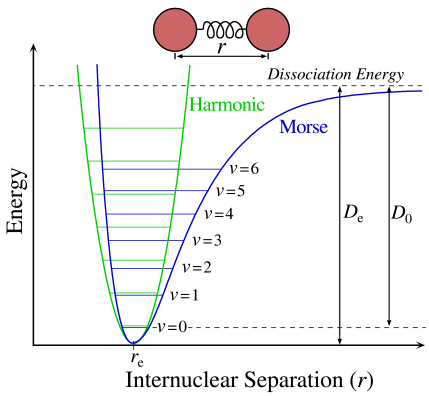


Figure 2.2.: Comparison of Morse potential and harmonic oscillator potential. The results for the fundamental vibration are almost the same.[3]

The diatomic molecule can be modeled as a *quantum harmonic oscillator*, where the potential is proportional to the deviation of the normal coordinate Q from its equilibrium position. From this model, it follows that the motion of the nuclei can be described as a harmonic oscillation with a discrete spectrum of equispaced allowed energies. A more correct model involves the use of the *Morse potential* that can also account for a molecule dissociation (see Figure 2.2), from which a similar solution follows, but the energy spacing decreases with increasing energy of the vibration. The molecule spends most of the time on the ground vibrational level ($\nu = 0$), but it can be excited to upper vibrational levels. Usually only the first excited state ($\nu = 1$), called *fundamental vibration*, is considered for the Raman effect, as the probability of transitions to $\nu > 1$ levels, called overtones, is very low.

The fundamental vibration of the normal mode k is a harmonic oscillation at the frequency ω_k, which has the form

$$Q_k(t) = Q_{k0} \cos(\omega_k t + \delta_k), \tag{2.3}$$

where Q_{k0} is the amplitude and δ_k is a phase. This means, that the polarizability $\hat{\alpha}$ also oscillates with the frequency ω_k. We combine (2.3) with (2.2) and employ a plane wave ansatz, describing an external excitation in a form of a plane wave $\vec{E} = \vec{E}_0 \cos \omega_1 t$ into (2.1), which yields

$$\vec{p}^{\,(1)} = \underbrace{\vec{p}^{\,(1)}(\omega_1)}_{\text{Rayleigh}} + \underbrace{\vec{p}^{\,(1)}(\omega_1 - \omega_k)}_{\text{Stokes}} + \underbrace{\vec{p}^{\,(1)}(\omega_1 + \omega_k)}_{\text{anti-Stokes}}, \tag{2.4}$$

[2]The normal coordinate refers to the positions of atoms away from their equilibrium positions, with respect to a normal mode of vibration.

[3]Image courtesy: Mark Somoza at English Wikipedia [public domain], via Wikimedia Commons.

Here, we get a Rayleigh term at the excitation frequency ω_1

$$\vec{p}^{(1)}(\omega_1) = \hat{\alpha}_0 \vec{E}_0 \cos\omega_1 t, \tag{2.5}$$

and two terms at the difference and sum frequencies (called Stokes and anti-Stokes, correspondingly):

$$\vec{p}^{(1)}(\omega_1 \pm \omega_k) = \underbrace{\frac{1}{2}\hat{\alpha}'_k Q_k}_{\hat{\alpha}_k^{\text{Ram}}} \cdot \vec{E}_0 \cdot \cos(\omega_1 \pm \omega_k \pm \delta_k)t. \tag{2.6}$$

We see, that the interaction of the plane electromagnetic wave with the molecule gives rise to an induced dipole oscillating at three different frequency components: ω_1, $\omega_1 - \omega_k$ and $\omega_1 + \omega_k$. The first term describes the Rayleigh scattering, when the induced dipole oscillates with the same frequency ω_1 as the driving electric field. The $\omega_1 \pm \omega_k$ terms are responsible for the Raman scattering – the electric dipole oscillating at ω_1 is additionally modulated by the oscillation of the molecule at frequency ω_k. The slow nuclear motion causes fast rearrangement of the electron density and thus leads to a harmonic change of the whole polarizability tensor $\hat{\alpha}$.

In case of a polyatomic molecule, the dipole moment is influenced by a high number of normal modes of individual bonds. This leads to many discrete vibrational frequencies in complex substances, but there are selection rules that can render some of them inactive. In accordance with (2.2), the condition $\hat{\alpha}'_k \neq 0$ is required for the vibrational mode to be Raman-active[4]. This is typically not the case for asymmetrical stretch or bending modes in linear molecules, such as CO_2, which are thus prohibited. The intensity of the observed band for a normal mode k is given by the derived polarizability tensor $\hat{\alpha}_k^{\text{Ram}}$ from (2.6).

Complex substances feature high number of Raman-active normal modes, but because many vibrational frequencies are so close in values and individual bands are quite broad, most of the normal modes cannot be resolved. Thus, even very big molecules, such as proteins, yield spectra with relatively low number of features. Still, the Raman spectrum is characteristic for each given substance – therefore it is often called "molecular fingerprint".

Although both Stokes and anti-Stokes frequency terms in Equations (2.4) and (2.6) are symmetric, their relative intensities are *not* equal. The Stokes lines are way more intense than the corresponding anti-Stokes ones. This is caused by the Boltzmann distribution of molecules over the energy states, which lowers the probability of the photon to meet an already excited molecule. Therefore, detection of anti-Stokes lines is not always practical, so the term "Raman spectroscopy" usually refers to the acquisition of Stokes signal.

[4] Another complementary technique, the *infrared spectroscopy* (IR), also probes for the molecular information by excitation of vibrational modes. Note however, that for a molecule to be IR- active its net **dipole moment** has to change as it vibrates or rotates, while the intensity of Raman bands is determined by the **polarizability** [47]. This leads to the fact, that often particular normal modes are observable in Raman spectra but inactive in the infrared, and vice versa. A classical example is a diatomic gas molecule, such as N_2, which is Raman-active because the electron cloud responses to the external field and drives the motion of the nuclei. On the other hand, the permanent molecular dipole moment of N_2 is zero and it does not change as the molecule vibrates, thus N_2 cannot interact with the infrared light.

In addition to a linear Raman spectroscopy, there is a wide variety of related techniques, such as *stimulated Raman scattering*, *surface enhanced Raman scattering*, *coherent anti-Stokes Raman scattering*, *Raman gain/loss spectroscopy*, etc. Each of these methods offers some benefits and is useful for specific applications. Among all of them, the linear Raman scattering is the most straightforward and instrumentally the simplest technique, and it is the only spectroscopic method utilized in this work.

2.2. Raman Spectroscopy for Clinical Diagnostics

Raman spectroscopy is an attractive tool for use in clinical diagnostic. The advantages and limitations of this technique are listed in Table 2.1. There is a large number of studies that demonstrate the proof-of-principle of this method for various biomedical [48–50]. In particular, it was used to detect and identify pathogenic bacteria in food samples [51], tissues [52] and body fluids [53–55]. Spectroscopic techniques were successfully applied to study the drug delivery [56], effects of drug treatment [57] and even the structure of folded bio-molecules using polarization-sensitive methods [58]. Vibrational spectroscopy was used to probe the mineral composition of the bones [59, 60]. Identification of cancer cells is another promising application that is directly related to this work [61, 62].

Table 2.1.: Raman spectroscopy for clinical applications – advantages and limitations.

Advantages	**Limitations**
Chemically-specific	Low scattering cross-section
Non-destructive	Complicated data analysis
Label-free	Special substrates are necessary
Sample can contain water	Low SNR[5]
High spatial resolution	Autofluorescence of the sample
Simple sample preparation	Hard to observe specific molecules in a mixture
Can be combined with a fiber probe	

As already noted in the introduction, tumor cells feature a number of biochemical properties, which they need to survive and thrive in the host organism. This makes Raman spectroscopy, as a chemically-sensitive method, particularly useful for cancer diagnostic, because it has a potential to probe molecular composition in a point of interest, even on a microscopic scale. Researchers applied it to delineate tumor boundaries in the tissue samples, a procedure called spectral histopathology [63–65], and to identify individual cells [66]. Integration with fiber optic probes paves a way toward endoscopic instruments that use vibrational spectroscopy for chemically-specific *in-vivo* tissue analysis [67]. Last but not least, Raman spectroscopy, being label-free and non-destructive, can be combined with a host of other diagnostic techniques.

[5] *Signal-to-noise ratio*

3. Instrumentation

This chapter describes the development of the hardware for Raman-based tumor cell identification. In the scope of this work I tightly worked with several Raman instruments for the investigation of biological samples. In particular, I designed and built custom microscopes for excitation and collection of Raman signal, which are described in Sections 3.2.1, 3.2.3 and 3.4.2. Additionally, I characterized a Raman system developed by our project partner (see Section 3.3.2), and integrated it into our automated data collection and management workflow, presented in Chapter 4.1.

The following section gives a general overview of issues that have to be considered during the design of a Raman spectroscopic instrument. Then, a detailed description of the used instruments and the results of their characterization are presented. The last section of the chapter describes the microfluidic chips used for cell handling, and results of their testing.

3.1. Design Considerations for Raman Microscopes

I designed and built several custom Raman setups for studies of biological samples. The design was based on the results of the previous works [68]. In all cases, commercially available spectrographs with multichannel detectors were used, but the microscope attachment had to be designed for a particular application and a specific excitation wavelength. Additionally, I integrated these new Raman instruments into an automated data management workflow that allowed us to efficiently handle data from a large number of spectroscopic experiments.

Custom Raman microscopes were built, because they offer very high flexibility. Such a system can be tuned and adjusted for a specific application. One of the most prominent features is the possibility to convert the microscope from the upright configuration, which we used to study the cell samples on a CaF_2 substrate, to the inverted one, which is required for the work with some microfluidic chips. Moreover, additional modalities or scientific instruments can be integrated with a custom microscope, which is not always possible with commercial systems.

The key components of a Raman instrument is a wide-field optical microscope supplemented by an excitation laser to illuminate the point of interest in the sample, and a spectroscopic detector that acquires the Raman spectrum from the very same point. Many technical details that have to be taken into account during the design of a Raman instrument are discussed in [69], where a flexible Raman instrument for biomedical applications is presented (see Figure 3.6). Here I will briefly summarize some of these details and also describe concepts that were not mentioned in the original paper.

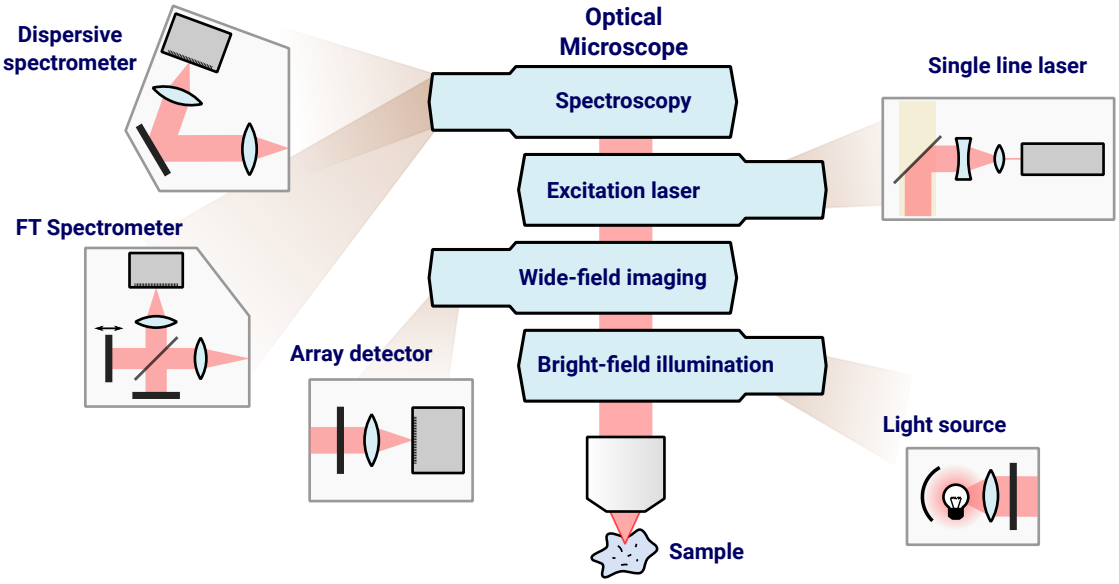


Figure 3.1.: Diagram of a generalized micro-spectroscopic instrument. The system is composed of a number of blocks, which are placed into the main parallel beam path. The blocks can be added or removed almost independently from each other, but the wavelength regions in which they operate must be taken into account.

3.1.1. Generalized Optical Microscope

We can consider a conventional far-field optical microscope as a modular system where the only mandatory element is the objective lens that focuses a parallel beam of light onto the sample. A number of different detectors or light sources can be placed into the parallel beam behind the objective lens, which would define the functionality of the microscope (see Figure 3.1). A bright-field microscope would require a light source (ideally a Köhler illumination, see page 39 of book [70]), a tube lens and a pair of oculars or a high- resolution camera (an array detector). A fluorescent microscope would require several additional filters inserted in front of the light source and the detector, as well as into the main beam path. For spectroscopic instrument, a dispersive or a Fourier-transform spectrometer is used as a detector. A laser-scanning microscope can be implemented by connecting an attachment containing a pair of galvo-mirrors, dichroic filters, photomultiplier tubes and relay lenses [71].

All these "building blocks" can be combined in a single instrument. A removable mirror placed into the main beam path reflects the light towards a given unit and activates it. With beam-splitters, ideally dichroics, it is even possible to use several units simultaneously. It is necessary, however, to ensure that the modules work in different wavelength ranges. For example, the Raman modality working in the near-infrared region could be combined with a wide-field fluorescent imaging in the visible range. The wavelength range of each individual module, however, strongly depends on the given application, as outlined in the following section.

3.1.2. Selection of the Excitation Wavelength for Raman Spectroscopy

One of the first design choices for a Raman system is the wavelength of the excitation laser, which depends on a number of factors. On one hand, although the intensity of the Raman scattering is linearly proportional to the laser power, it depends on the fourth power of the excitation light frequency. Dipole emission strength, as discussed in [72], is given by:

$$P_S \propto P_0 \nu_S^4 \left(\frac{\partial \alpha}{\partial Q} \right)^2_{Q=Q_0}, \tag{3.1}$$

where P_0 is the power of excitation wave and ν_S is its frequency, $\frac{\partial \alpha}{\partial Q}$ is the change of the polarizability α caused by the oscillation of the charge around the mean value of the coordinate Q_0. The fourth-power of the ν_S term means that shorter wavelength yields a much higher Raman signal, but on the other hand, it can cause some undesired effects. Photons of ultraviolet or blue-colored visible light have enough energy to cause electronic excitation within molecules, which under many circumstances leads to photo-degradation of a biological sample.

Additional phenomenon, which depends on the excitation wavelength, is the resonance Raman effect. When the energy of the incoming light is tuned near an electronic transition, which usually implies a rather short excitation wavelength, the vibrational modes associated with that transition exhibit increased Raman scattering intensity. This can overpower Raman signals from all of the other transitions, and thus has a high influence on the observed spectrum. Depending on the particular application this effect can be desired, or not. We did not use the resonance Raman effect in this work.

Autofluorescence Issues

Autofluorescence is another factor that has to be considered during the choice of the excitation wavelength for Raman instruments. Any fluorescent emission seriously deteriorates quality of Raman spectra of biological substances.

The fluorescent emission spectrum, in contrast to a Raman spectrum, is often very intense, but at the same time smooth and broad. The intensity of the fluorescent emission can be many orders of magnitude stronger than that one of the Raman spectrum. One can naively assume, that a Raman spectrum affected by the fluorescence can be easily recovered: the fluorescent background can be approximated with a smooth polynomial and subtracted from the spectrum. Although this works, the corrected spectrum suffers from the shot noise caused by the strong fluorescence signal, as shown in Figure 3.2.

Biological tissues produce autofluorescence due to the presence of several endogenous fluorophores, such as collagen, keratin, elastin, and NADH. In an extensive clinical study, de Veld *et al.* [73] collected autofluorescence spectra at six excitation wavelengths ranging from 365–450 nm from 172 benign, dysplastic, and cancerous lesions, and from 97 healthy volunteers. Their results clearly indicate, that high autofluorescence emission is expected in the region of 460–720 nm, whereas almost nothing is emitted at wavelengths above 800 nm.

The Raman fingerprint region starts at approximately 800 cm^{-1}. If this is the point where the autofluorescence should stop, i. e. at ~720 nm, then the reasonable excitation wavelength

should be above 681 nm. On the other hand, Aubin [74] showed that individual viable mammalian cells, in contrast to tissues, exhibit autofluorescence in the 450–600 nm range. Table 3.1 lists the most common excitation wavelengths of Raman instruments for biological applications.

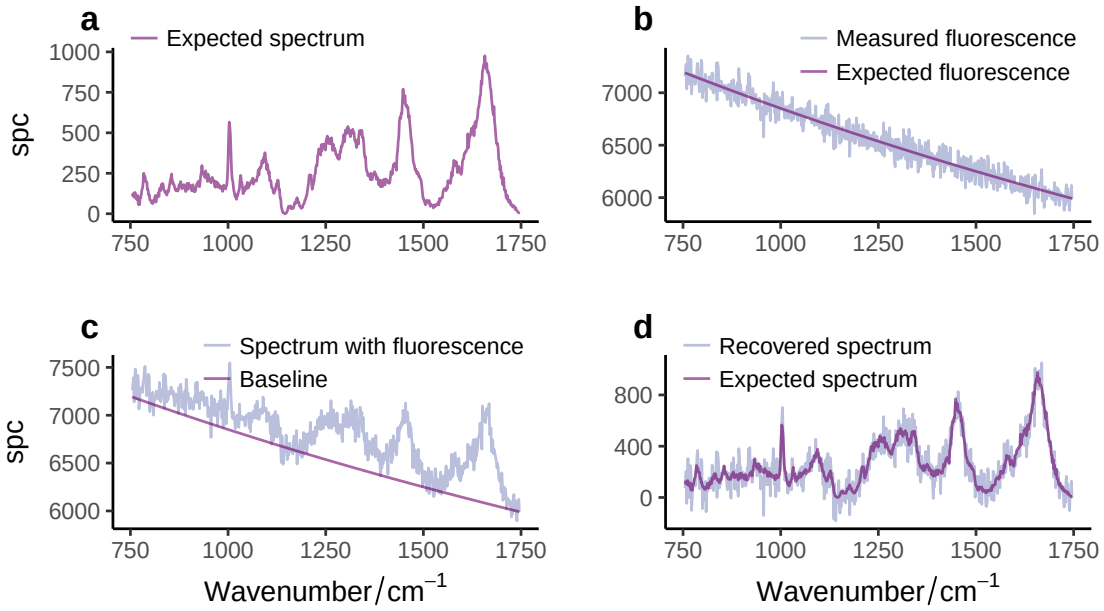


Figure 3.2.: Influence of fluorescence on the quality of spectra (simulation). The mean intensity of the fluorescence is 25 times higher than that of the Raman spectrum. **a** – expected Raman spectrum, without fluorescence, shot-noise limited. **b** – simulated smooth fluorescence background containing only shot noise. **c** – polynomial baseline subtraction to correct for fluorescence. **d** – direct comparison between the expected spectrum and the recovered one.

Thus, the choice of the laser wavelength is determined by the particular application and the sample properties in the first place. 785 nm is a suitable excitation wavelength for a majority of biological samples. 660 nm excites autofluorescence in tissues, but works pretty well for single cells.

Photodetector Limitations

Detectors create upper limits for the excitation wavelength. In addition to the aforementioned ν_S^4 dependency of the Raman signal on the excitation wavelength (Equation 3.1), there is an issue of low quantum efficiency of conventional Si-based detectors in the near-infrared region. The sensor should have enough sensitivity to detect Stokes photons corresponding to the spectral features of the interest. Typical spectroscopic systems utilize CCD or EMCCD sensors for the recording of spectra, but some designs use CMOS detectors [75, 76], like in the case of RS660 instrument, described below (Section 3.3.2). With higher wavelength, the quantum efficiency of the sensors drops down rapidly, almost linearly from ~80% at 800 nm to ~10% at

1000 nm (the quantum efficiency profiles of used detectors are shown in Appendix on page 123). At 1100 nm the sensitivity equals to zero, because photons do not have enough energy to create an electron-pair hole in the silicon. This means, that a 830 nm Raman instrument can not detect CH-stretching region unless a special detector sensitive to the IR light is used.

Table 3.1.: Detection windows of typical Raman systems for biological samples. All values are in nm, except for the "Rel. intensity" column, which shows the dipole emission strength calculated using (3.1).

Laser wavelength	Fingerprint/low	Fingerprint/high	CH-str/low	CH-str/high	Rel. intensity
458	475.4	497.9	525.4	533.8	1.00
488	507.8	533.6	565.2	575.0	0.78
514	536.0	564.8	600.4	611.4	0.63
532	555.6	586.6	625.1	637.1	0.55
635	669.0	714.4	772.3	790.6	0.27
660	696.8	746.2	809.6	829.8	0.23
785	837.6	910.0	1006.2	1037.5	0.12
830	889.0	971.0	1081.3	1117.5	0.09

In addition to high sensitivity, detectors are also a subject to additional strict requirements, such as low levels of thermal noise and read-out noise [77]. The first one describes the effect of charge accumulation in the absence of optical signal, which depends on the temperature and the exposure time. The read-out noise describes the random signal fluctuation that appears during the analog-to-digital conversion of the acquired signal.

Additional issue that has to be considered for the sensors with the back-illuminated geometry, is the effect of optical etaloning – an interference pattern that appears inside of a thin semiconductor detector. Books of Zimmermann [78], Saleh *et al.* [79] (Chapter 17), Kubitscheck [70] (Chapter 3, pages 112–127) and Gåsvik [80] (Chapter 5) provide more details on selection of optimal photodetectors for Raman spectroscopy. Finally, a comprehensive description of the imaging sensors is provided by the *EMVA Standard 1288* that is freely available online [77].

3.1.3. Confocality and Signal Collection Efficiency

Confocality describes the ability of the microscope to reject the light that does not originate from the plane on which the microscope is focused. Typically confocality is an inherent property of LSM[1] systems and is achieved by a pinhole conjugated with the focal point of the microscope [71]. The pinhole prevents the out-of-focus light from reaching the detector. A drawback of such a design is that the image has to be acquired point by point, direct imaging of the whole field of view is not possible with confocal detection.

Typical LSMs use a single detector, usually a PMT[2], to measure the whole spectral intensity of the signal that passes the pinhole. By replacing this simple detector with a spectrometer

[1] *Laser Scanning Microscope*
[2] *Photomultiplier tube*

one can analyze the spectral composition of the light that comes from the sample plane. Supplemented by a single frequency laser and a set of filters, such an instrument can illuminate the focal point and confocally collect the Raman signal from it.

Often the spectrometer is connected to the microscope via an optical fiber. The core of the fiber itself acts as a pinhole, because only the light entering the core is guided through the optical fiber. This makes extra pinhole or the input slit of the spectrograph redundant.

For an efficient signal collection in a Raman instrument several properties of the focal spots have to be optimized and matched to each other:

1. The focal point of the excitation laser and the point of the confocal detection have to **(a)** have an approximately equal sizes, and **(b)** to overlap spatially[3]. A care has to be taken to ensure that not only lateral, but also the axial positions of the focal spots overlap.

2. The axial position of the focal spot has to be in the focal plane of the bright-field attachment of the microscope. This ensures that the spectroscopic measurements are carried out in the same focal plane that is visible to the user through the camera. Improper alignment would result in the spectrum acquisition from points above or below of the currently visible sample.

3. The volume of the focal spot (i.e. the size of the PSF[4] of the imaging path) should not exceed the volume of the sample to be measured, otherwise the signal from the sample gets accompanied by the one of the surrounding medium. A prober PSF size can be achieved by selecting an optimal combination of the optical fibers and the microscope magnification. Naturally, however, the PSF size cannot go below the Abbe diffraction limit given by Formula 3.3.

Lateral Size of the Excitation and Detection Spots

Ray optics states, that microscope lenses create a de-magnified image of the optical fiber core end face in the sample plane. Thus, the focal spot size d is given by $d = \frac{D}{m}$, where D is the core diameter of the optical fiber, and $m = \frac{f_T}{f_o}$ is the actual magnification of the microscope (f_o and f_T being the focal lengths of the objective and tube lens, correspondingly). Typical microscope objectives are designed to work with standard tube lenses with f'_T = 200 mm or 180 mm, so this formula can be rewritten as

$$d = \frac{D f'_T}{M f_T} \tag{3.2}$$

where M is the nominal magnification of the objective lens, engraved on its barrel. Thus, a Raman instrument with a $60\times$ objective, a 30 mm tube lens placed into the spectroscopy

[3] While the focal spot of the excitation laser is often visible, the position of the detection spot is more challenging to estimate. A convenient way to visualize it is to temporarily replace the spectrograph with a light source, which emits in the detection range of the spectrograph, so that the light can pass through the filters and be focused on the sample. When both laser points are visible, their positions can be adjusted to make them overlap both laterally and axially (see Figure 3.4).

[4] *Point Spread Function*

beam path and 62.5 μm core diameter of the multi-mode detection fiber has a detection spot of ~7 μm size.

On the other hand, there is the Abbe/Rayleigh resolution limit, which states that the size of the focal spot given by Formula (3.2) is limited by the diffraction. An ideal microscope objective lens focuses a parallel beam of light with the wavelength λ into the Airy pattern shown in Figure 3.3, which has the diameter of the central bright disc given by

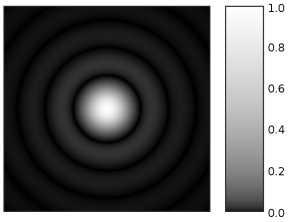


Figure 3.3.: Airy pattern[5].

$$d = 1.22\frac{\lambda}{\mathrm{NA}}, \tag{3.3}$$

where NA is the numerical aperture of the objective lens [81]. The central disc of the Airy pattern contains 84% of the total power of the focused light beam. The Airy pattern actually describes the PSF of an ideal diffraction-limited optical system.

To sum up, the size of the focused laser beam should be calculated using methods of geometrical optics (3.2) *and* the Abbe resolution limit (3.3). From these two values *the biggest one will correspond to the lateral size of the excitation spot*. For example, a 60× NA 1.0 objective combined with a 100 mm tube lens and a single-mode fiber (mode field diameter about 5 μm) will focus a 785 nm laser beam into an Airy disc of **958 nm** diameter given by (3.3), because the focal point size calculated using basic ray optics (3.2) results in the value of 167 nm, which is way under the diffraction limit.

The same considerations are valid for the size of the detection spot – the volume from which the Raman signal is collected.

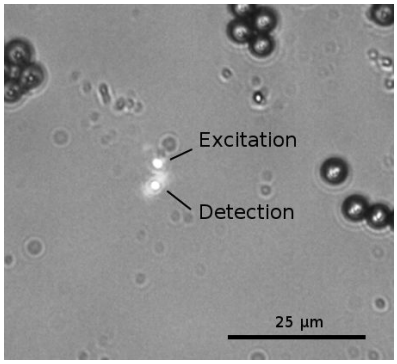


Figure 3.4.: An overlap of excitation and detection spots is mandatory for the collection of the Raman signal. Here, the both spots are visualized using laser beams during the alignment process.

Axial Resolution of the Confocal Raman System

According to Sibarita [81] and the Section 2 of the book [70], the axial resolution of a confocal microscope in a backscattering mode is typically 3-4 times worse than the lateral one. In a diffraction-limited case, the following formula gives the axial length of the PSF:

$$z_0 = \frac{2\lambda n}{\mathrm{NA}^2}, \tag{3.4}$$

where n is the index of refraction of the medium around the objective lens. Being applied to RS660 instrument that has a 40× NA 0.8 objective lens, it gives 2.7 μm. It is important to note, that these considerations are valid in the ideal case only, when the pinhole size (or the input slit width of the spectrometer) is infinitesimal, and the entire NA of the objective lens is used (i. e. the light beam fills the whole aperture of the objective lens).

[5]Image courtesy: Sakurambo at English Wikipedia 桜ん坊 [public domain], via Wikimedia Commons

3.1.4. Raman Microscope Alignment

As mentioned before, the excitation, spectroscopy, illumination and wide-field detection units in the microscope are more or less independent from one another. Each of them is inserted into the parallel optical beam behind the objective. This means, that one can independently align each module and then combine them together.

Ideally, the wide-field imaging path is aligned in such a way, that the bright-field camera is focused at infinity. This guarantees, that the sample will be sharply visible as soon as it comes to the focal point of the microscope objective.

Once both the detection module and the laser excitation module are aligned exactly along the optical axis and focused to infinity (a *shearing interferometer* is particularly useful for this procedure), they can be inserted into the microscope. If everything is correct, the excitation and detection points should be in the center of the wide field image and close to the focal plane. Then, one has to achieve perfect overlap of these two spots (see Figure 3.4 and the footnote on Page 30).

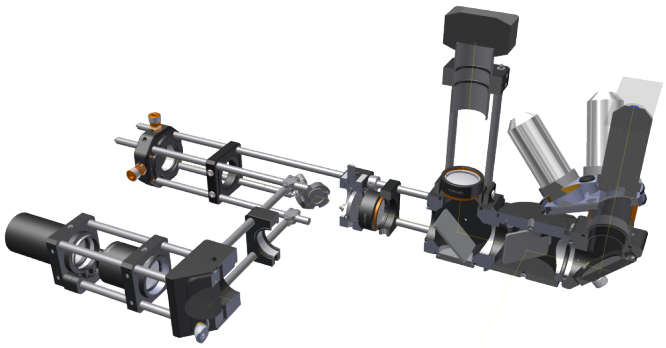


Figure 3.5.: 3D model of the microscope from the flexible Raman instrument. Optical elements are visible in the cross-section view, and distances between them can be easily measured.

It is convenient to build each module of the microscope using the so-called "cage system" (Thorlabs). These mechanical components offer great stability and rigidity; the optical axis goes parallel to the rods through the centers of the cage plates – this makes the alignment convenient. Kinematic lens mounts or cage plates allow to make fine adjustments. Ideally, each module should have three kinematic components that allow to *independently* adjust the focusing (beam divergence), the lateral beam position, and the beam direction (angle).

Further, one has to ensure that the internal optics of the spectrograph creates a sharp image of the input slit on the detector, i. e. the optics has to be focused properly. This can be checked with a gas discharge lamp – sharp lines, representing the image of the input slit of the spectrograph, should be visible. If the spectrometer is fiber-coupled, one has to adjust the focus until sharp images of the round fiber core(s) can be observed.

An excellent guide of general methods for the microscope alignment can be found in Appendix A of the book "Fluorescence microscopy" [70], pages 393–400.

3.2. Raman Instruments Used in this Work

3.2.1. Flexible Raman Instrument

Commercial Raman instruments are hard to modify and often it is hard to adapt them to a particular research task due to the lack of flexibility. I designed and implemented a modular Raman instrument that features several spectral collection modes: acquisition of single spectra, point-by-point hyper-spectral imaging, and hyper-spectral line imaging.

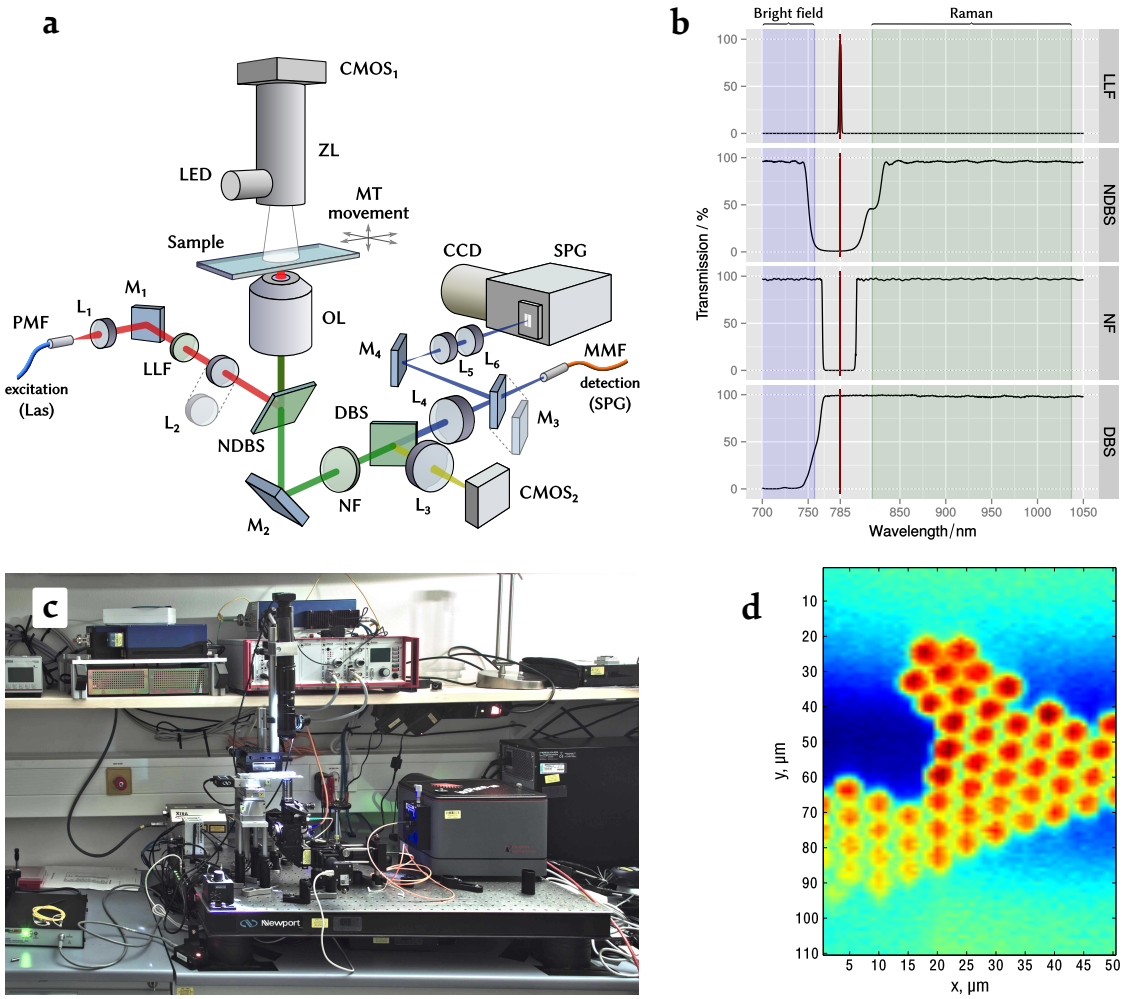
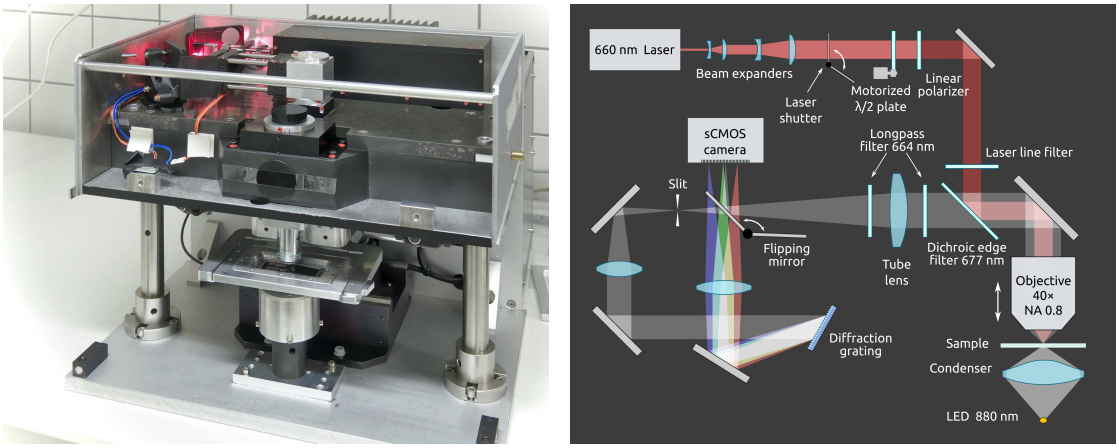


Figure 3.6.: Flexible Raman instrument, which features confocal Raman point imaging or linear scanning mode. **a** – Optical layout. `LLF` – Laser clean-up filter, *LL01-785*, Semrock; `NDBS` – Notch dichroic beam splitter, *NFD01785*, Semrock; `NF` – Notch filter, *LC-785NF-25*, Laser Components; `DBS` – Dichroic long-pass beam splitter, *FF757-Di01*, Semrock; `MMF` – Multi-mode optical fiber. **b** – Transmission profiles of optical filters[a]. **c** – Photo of the setup. **d** – A Raman image of polystyrene beads (ø5 µm) in the line-scanning mode showing the relative intensity of the 1001.4 cm^{-1} peak. 785 nm excitation wavelength. Reprinted from [69], with permission from *IOS Press*.

The Raman system, consisting mainly of commercially available individual optical and mechanical components, was prototyped in a CAD[6] program (Autodesk Inventor) using 3D mechanical drawings provided by the suppliers of these components. The CAD software allowed me to try and optimize several different designs, check mechanical compatibility of different parts and precisely measure relative positions and distances between optical components (see Figure 3.5). Furthermore, the CAD software generates and updates a list of components, which allows to easily track the costs of the system.

The layout of this instrument, shown in Figure 3.6, and an example data set collected with it, as well as general guidelines for the design of micro-spectroscopic Raman systems were published in the paper of Kiselev *et al.* [69].

Flexibility of the Raman instrument allows to include additional modules. For example, a wide-field fluorescence imaging can be added, so that the Raman- spectroscopic predictions can be validated by colocalization with the fluorescent labels [82]. This would require an additional light source and a filter cube, as shown in Appendix (Figure B.1 on the page 124). There are multi-band fluorescence filters available, so a single filter cube can be used with several different dyes, and only the light source (LED) alongside with the emission filter have to be exchanged to select another dye.



(a) Photo of the instrument with front panel removed. **(b)** Diagram of the optical layout.

Figure 3.7.: RS660 Raman instrument from *Till ID GmbH*.

3.2.2. RS660 Instrument

RS660 is a Raman instrument designed in the scope of the RamanCTC project by the company *Till ID GmbH*. The instrument combines all optical parts in a single case and offers two modalities, which are (a) bright-field or dark-field imaging, depending on the condenser configuration, and (b) single-point acquisition of Raman spectra. The modalities can be used only

[6] *Computer-Aided Design software*

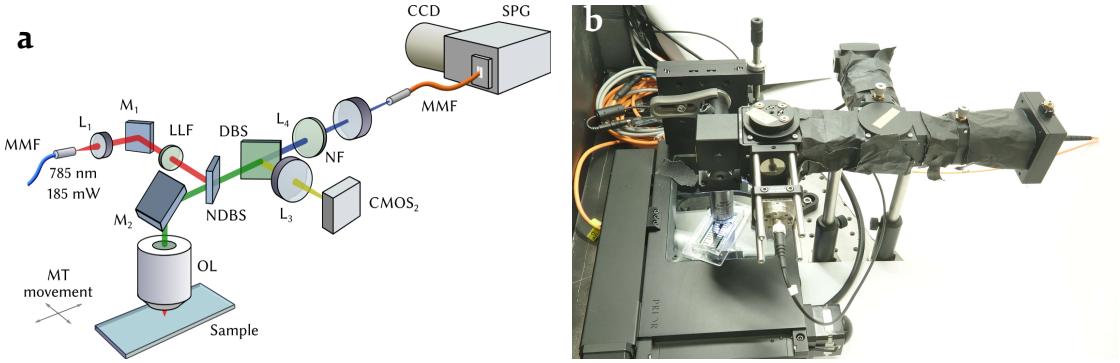


Figure 3.8.: RS785 clinical Raman instrument for cancer cell research. **a** – Optical layout. **b** – Photo.

interchangeably, as the instrument has only a single detector, which is a sCMOS camera. The modality is selected by a flipping mirror (see Figure 3.7b).

The microscope uses a $40\times$ NA 0.8 water immersion objective lens (Nikon), a 660 nm DPSS[7] laser (up to 280 mW at the sample), a lens-based spectrograph with 50 µm input slit and a NIR blazed 1800 grooves/mm grating, a sCMOS camera (Andor Zyla 5.5, see Appendix at page 123 for the quantum efficiency profile), a 880 nm LED source for imaging, a laterally-motorized stage, and a voice coil focusing unit. Several servo motors control the laser power, the flipping mirror position, and the laser shutter.

The RS660 instrument is controlled by an embedded system running Linux that communicates over a proprietary network protocol with the user software called `SIAM` that runs on a separate computer.

3.2.3. RS785 Instrument

The RS785 instrument was built in the scope of this thesis using optical and mechanical Thorlabs components (microscope assembly), and the LS785 spectrometer with the PIXIS 256E CCD camera[8] (both from Princeton Instruments). Its layout, shown in the Diagram 3.8, is based on a simplified concept of the flexible Raman instrument, discussed above in the Section 3.2.1. Since the RS785 was designed to be operated at a clinical laboratory for routine studies of cancer cells, it had to be stable and rigid, as well as compact. For this reason, I simplified the optical layout of the microscope tpo the bare minimum: the excitation, detection and the bright-field imaging optical paths. The number of movable components for the alignment was kept minimal as well. For a reasonable room light protection, all beam paths and optical fibers were covered with the appropriate black-out materials, and the microscope itself was placed into a metal enclosure.

The RS785 instrument features a bright-field imaging mode and a single-point Raman acquisition, which can be used simultaneously. The sample positioning is achieved with a laterally motorized microscopy table and a manual translational stage for the focusing. Table 3.2 con-

[7] *Diode-Pumped Solid-State laser*

[8] The quantum efficiency profile is shown in Appendix on the page 123.

tains a list of the parts of the RS785 instrument.

The used LS785 spectrometer features a manually turnable diffraction grating. In the optimal grating position the spectral range is 80–3400 cm^{-1}. The global mechanical shutter, located behind the input slit, can be controlled by a trigger signal from the CCD camera. This allows to automatically acquire dark frames and eliminates undesirable effects associated with electronic shutters [83, 84], such as the charge collection and the associated sensor heating between the actual spectrum acquisitions.

The CCD camera automatically triggers the LED light source over a BUZ 73L MOSFET and switches off the light source during the acquisition of a spectrum, so that the illumination light cannot deteriorate the measurement.

The instrument has relatively large excitation and detection spots (about 12–15 μm in diameter). This was done intentionally to acquire Raman signal from a large area of a single cell – this helps to aggregate spectral information from the whole biochemical mixture contained within a cell and allows to reduce the number of spectra required to train a good classifier [85].

Table 3.2.: Components of the RS785 Raman instrument.

Component	Name and model	Comments
Motorized table	Prior OptiScan II	x, y-stage with joystick
Manual microscope focusing	Thorlabs SM1Z	1 μm resolution knob
Objective lens	Nikon 60x 1.0 WI	Water-immersion objective
Spectrograph	PI[9] LS785	Lens-based, $f/2$
Adjustable slit	PI	10 μm (calibration), 70 μm (cells)
Optical shutter	PI	
Spectroscopic camera	PI PIXIS 256E	1024x256 CCD, USB 2.0
Wide-field camera	Thorlabs DCC1645C	1280x1024 CMOS, USB 2.0
Notch filter	Semrock NF03-785E	
Dichroic notch beam splitter	Semrock NFD01-785	See Figure 3.6b
Laser clean-up filter	Semrock LL01-785	See Figure 3.6b
Dichroic beam splitter	Semrock FF757-Di01	See Figure 3.6b
Semiconductor laser	IPS[10] I0785ML0350MF	175 mW at sample, 785 nm
Optical fiber from the laser		Multi-mode, 100 μm core
Optical fiber to the detector		Multi-mode, 300 μm core

[9] Princeton Instruments
[10] Innovative Photonic Solutions

3.3. Characterization of Raman Systems

3.3.1. RS785 instrument

This section presents characterization results the RS785 instrument, introduced earlier (see Section 3.2.3).

Spectral Resolution. To characterize the spectral resolution of the RS785 instrument, we acquired a spectrum of a Ne-Ar gas discharge lamp. Such lamps emit a set of very narrow spectral lines whose wavelength and relative intensity are determined by the electronic structure of the atoms of the substance within the lamp. The observed spectrum is a convolution of the instrument transfer function with the corresponding emission lines, as discussed in the Section 4.4.3 on page 71. The microscope attachment is connected to the spectrometer via an optical fiber that has 300 μm core diameter. The spectrograph optics images the end face of the optical fiber onto the CCD detector, and there is an *adjustable slit* that can partially obscure the core of the optical fiber. The slit almost touches the end face of the fiber, thus it always stays in focus.

Due to this design, the spectral resolution and the signal intensity strongly depend on the input slit width. To characterize this behavior, the spectra of the Ne-Ar gas discharge lamp were acquired at different slit widths ranging from 0 (closed slit) to 300 μm in 10 μm steps (see Figure 3.9). The experiment has been repeated three times to get more reliable results. All spectra were acquired with 50 ms exposure time; 50× averaging was used to compensate for the random errors such as cosmic spikes. It was necessary to remove the constant offset of 588 counts (ADC bias) from the spectra for the proper calculation of the intensity. The intensity was calculated as the area under the curve in the region between 833 and 872 nm, because a high number of bands is observed here. The obtained values were normalized to the $0 \ldots 1$ range (see Figure 3.9b).

Table 3.3.: Spectral resolution of the RS785 instrument. The table shows FWHM values of two peaks: **(A)** at 795 nm (corresponds to 160 cm^{-1} at 785 nm excitation), and **(B)** at 966 nm (2387 cm^{-1}). See Figure 3.9 for more details.

Slit/μm	Relative intensity	Peak **A**, in nm	Peak **A**, in cm^{-1}	Peak **B**, in nm	Peak **B**, in cm^{-1}
0	0.00	0.69	10.9	0.72	7.7
10	0.04	0.67	10.6	0.64	6.8
70	0.30	1.10	17.3	1.07	11.4
100	0.42	1.52	24.0	1.47	15.7
150	0.62	2.21	35.0	2.21	23.7
300	1.00	3.70	58.5	3.63	38.9

We see, that the relative signal intensity grows linearly with the slit width, but, due to the circular shape of the fiber core, starting from ~150 μm the intensity growth saturates. The spectral resolution of the instrument, which also depends on the slit width, is summarized in

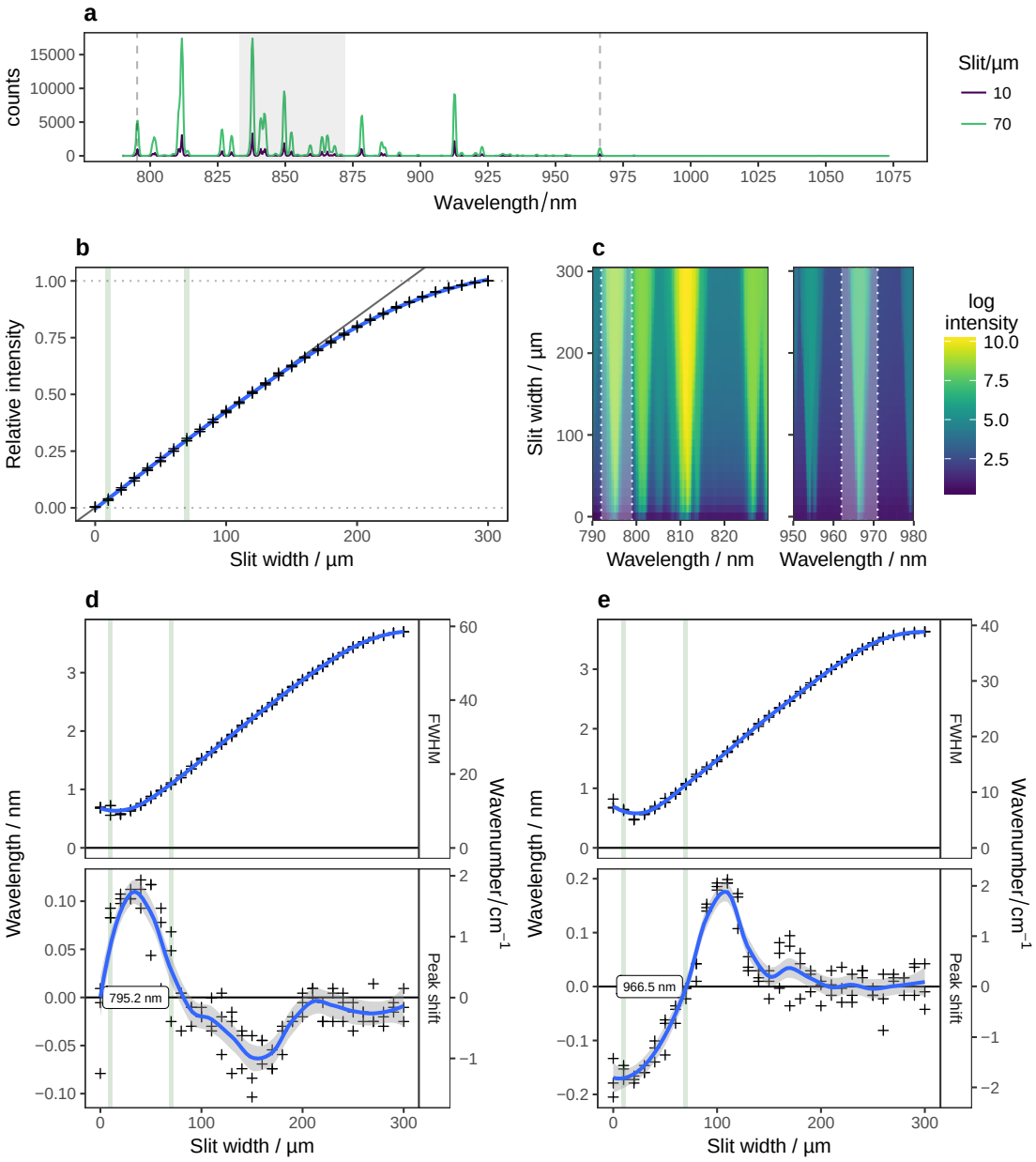


Figure 3.9.: Characterization of the spectral resolution and signal intensity of the RS785 instrument as a function of the slit width. 10 µm slit width was used during the calibration (Section 4.4), whereas the spectra of cells were collected at 70 µm. These two values are highlighted in panels **b**, **d** and **e**. **a** – Spectrum of a Ne-Ar gas discharge lamp, shown here for slit widths of 10 and 70 µm. The highlighted region was used for the calculation of the relative intensity (panel **b**). Dashed lines indicate peaks used for the estimation of spectral resolution (panels **c**–**e**). **b** – Relative signal intensity. **c** – False-color image showing logarithmic intensity as a function of the slit width. Two prominent individual peaks at ~795 nm and ~966 nm were used for the calculation of the spectral resolution. **d** – FWHM and the relative position of the peak at 795 nm. **e** – FWHM and the relative position of the peak at 966 nm.

the Table 3.3. The resolution is always slightly better in the high-wavenumber region of the spectrum, which is due to the inverse-proportional dependency of the wavenumber on the wavelength as per Equation (4.1).

From Figure 3.9 it is visible, that as the slit opens, the position of the peak maximum experiences a slight shift. This means, that the slit blades do not move fully symmetrically. Since we calibrated the instrument at 10 μm slit and acquired spectra at 70 μm slit, this could lead to small systematic errors in the measurements. This effect remained unnoticed for a long time because it is rather weak. The motivation for the calibration with the narrow input slit was almost two-fold better spectral resolution, which allows to detect more peaks by the automatic wavelength calibration algorithm (see Section 4.4).

Although the effect of peak shifting is highly undesirable, it is reproducible, as illustrated by three experiment repetitions. The magnitude of the shift is rather low (maximum 2 cm^{-1}) compared to the actual spectral resolution value of the instrument. Thus, we conclude that this effect should not deteriorate our measurements.

Spatial Resolution. The RS785 instrument was not intended for Raman mapping. On the contrary, we intentionally used multi-modal fibers for the light in- and out-coupling, which resulted in a big excitation and detection spots, about 12–15 μm in diameter. The spot size was estimated using geometrical optics principles as per Equation (3.2). Big laser spot integrates spectral information from the whole cell and helps to overcome the intracellular variance, which results in a faster classifier training [86].

3.3.2. RS660 Instrument

We evaluated the performance and usability of the RS660 instrument, which was designed and built by our collaborator *Till ID GmbH* during the RamanCTC project.

Spectral Resolution. The spectral resolution of the system was characterized using a Ne gas discharge lamp – in a similar way as it was done for the RS785 instrument. We calculated the FWHM[11] of six observed peaks in the spectrum; the results are shown in Table 3.4.

Table 3.4.: Spectral resolution of the RS660 instrument, estimated as FWHM of the observed peaks of a narrow band light source (a Ne gas discharge lamp).

	Best	Average	Worst
Spectral resolution/nm	0.25	0.29	0.34
Spectral resolution/ cm^{-1}	4.54	5.59	6.96

Spatial Resolution. To characterize the spatial resolution of the instrument, it was necessary to collect spectra from different points with a small step size. I implemented a `Python`

[11] *Full Width at Half Maximum*

script named `RamanReader_automation`[12] that repeatedly triggers acquisition of spectra on a rectangular raster grid, i. e. it allows to make Raman mapping. Figure 3.10a shows a Raman map of a thin metal foil on a glass substrate. By investigating the signal intensity across the metal/glass interface, we determined the lateral resolution to be about 3.3 μm by the 10/90% criterion.

Similarly, the axial resolution was found by performing a vertical scan through a 100 μm-thick polystyrene foil with 1 μm steps. The estimated axial resolution is about 27 μm.

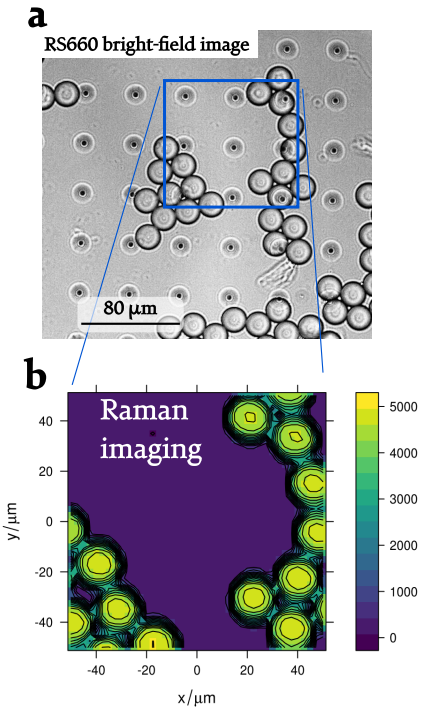


Figure 3.12.: **a** – Bright-field image of ø20 μm polystyrene beads. **b** – Raman image of the highlighted area, the color encodes the intensity of two bands between 966 and 1064 cm^{-1}.

Figure 3.11a demonstrates imaging capabilities of the RS660 instrument for biological samples. Here, we acquired Raman maps of `BT-20` cells on a CaF_2 substrate. N-FINDR algorithm [87, 88] was used to extract the most different spectra, that correspond to the cell and the background in this case. The spectra of these groups, as well as the difference spectra, are shown in Figure 3.11b.

Moreover, we developed and implemented a special imaging approach that we call *"hexspot"*. In this mode, the Raman map is acquired point-by-point on a hexagonal grid, as shown in Figure 3.13. In contrast to a conventional rectangular grid, hexagonal grid allows a more dense packing of round objects; in our case a higher proportion of the sample area can be scanned without overlapping of the detection spots [89]. The hexagonal pattern also better fits to the round cell than a rectangular one.

Cells are heterogeneous objects that contain different organelles. This fact influences the training of classification models – they require much more data to become stable [85]. We use the hexspot imaging to collect several spectra (typically 7 or 19) from each individual cell that is being interrogated by Raman spectroscopy, as illustrated in the Figure 3.13a–b. This makes it possible to efficiently account for the internal heterogeneity of the cell even with a small detection spot. Using the hexspot imaging one can also collect big Raman maps, as illustrated in Figure 3.13e. In total, our Raman spectra database contains 5345 spectra acquired in this mode.

[12] It is a command line tool that can trigger single-point acquisition, as well as an automated hyper-spectral imaging, i. e. lateral scans, *z*-stacks, time series, experiments with different exposure times, or any combination of these. The collected spectra are saved in the `spc` format, the experimental parameters are stored at the fixed positions of the file name.

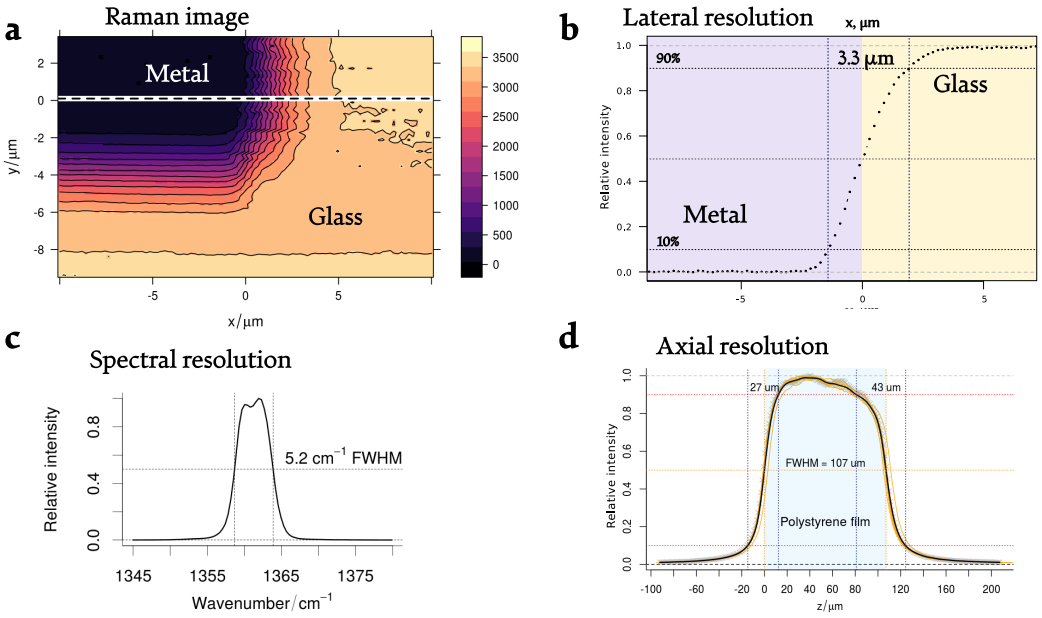


Figure 3.10.: Characterization of lateral, axial and spectral resolution of the RS660 instrument and demonstration of Raman mapping. We used an USAF resolving power test target for microscopes, from which a small area encircling a corner of a rectangle element was imaged. The target consists of a thin metal test pattern deposited on a glass substrate. Due to the fluorescence, the uncovered glass substrate delivers a strong signal, while no signal is observed from the metal. **a** – Raman map, color codes the whole spectral intensity. The imprinted pattern is visible in the upper left part of the image. **b** – Cross-section of the image **a** along the dashed line. **c** – Spectral resolution is about 5.2 cm^{-1} (estimated from a spectrum of a Ne lamp, emission line at 724.51 nm is shown here). **d** – Characterization of the axial resolution. Plotted is the intensity of Raman signal from 100 µm thick polystyrene film vs. the axial position of the focused laser spot.

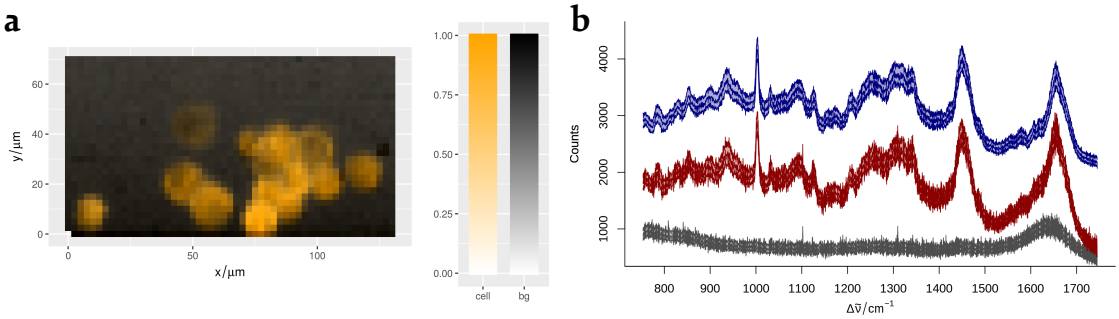


Figure 3.11.: Raman mapping of a biological sample with the RS660 instrument. **a** – Raman map of `BT-20` cells on CaF_2, the color codes the concentration of endmembers found by the N-FINDR algorithm. The image contains 1537 individual spectra (53×29 with 2.5 µm step size), acquired with the maximum power of 273 mW and exposure time of 2.5 seconds. **b** – Mean and standard deviation spectra of cells (*blue*) and background (*gray*). *Red lines* – cell spectra after subtraction of the mean background spectrum.

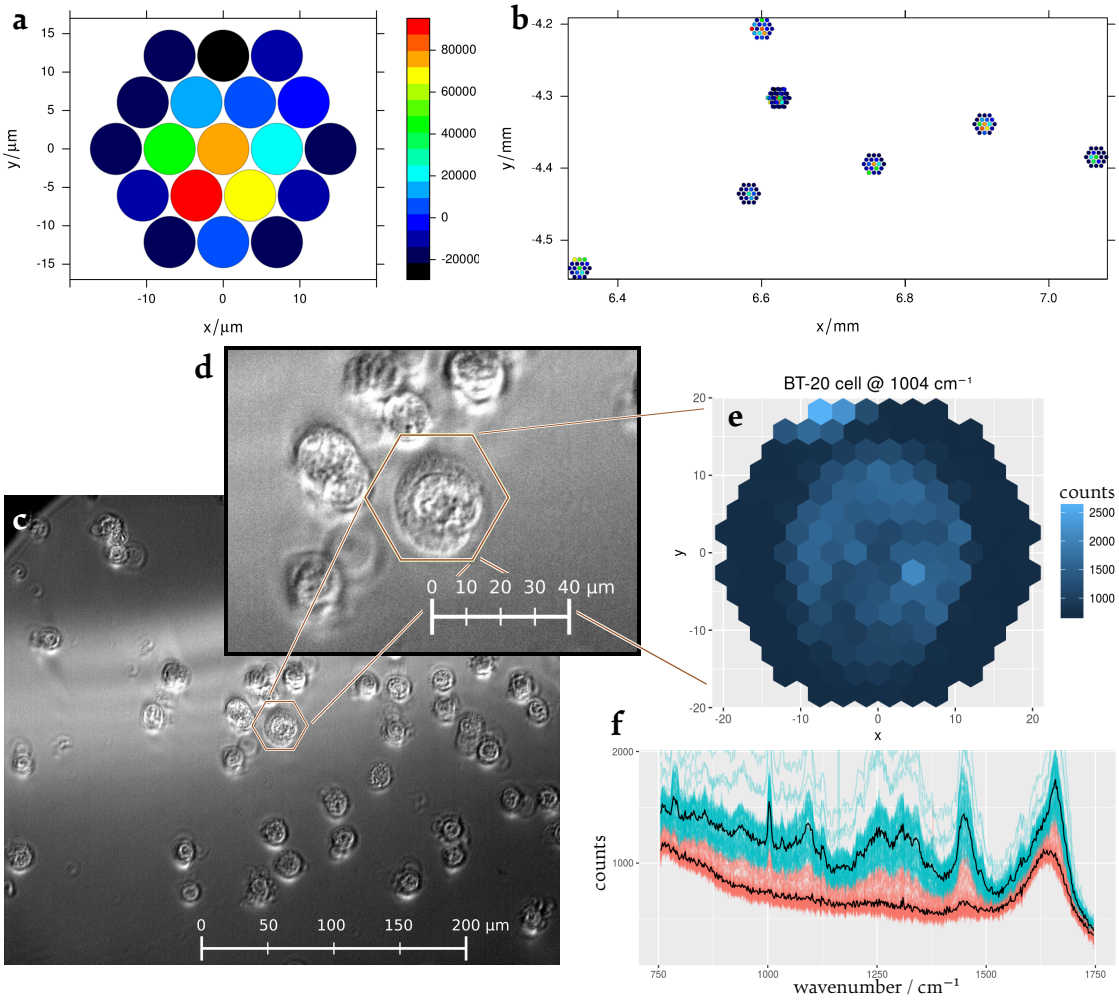


Figure 3.13.: Demonstration of *hexspot* Raman imaging with the RS660 instrument. **a** – Measurement of a single `HL60` cells which contains 19 spectra, the color shows average spectral intensity after the normalization and subtraction of background. **b** – Seven `HL60` cells measured in the *hexspot* mode, 19 spectra per cell. **c** – Bright-field image of `BT-20` cells on CaF_2 substrate. **d** – Magnified region of a bright field-image **c**. **e** – Raman map with 163 spectra acquired in the *hexspot* mode from the area highlighted in panel **d**, collected with 2.8 µm step size. The color map codes the intensity of the phenylalanine band at 1004 cm^{-1}. **f** – Individual spectra of the cell (cyan) and background (red), as well as the mean spectra (black lines) from the hexagonal Raman map shown in panel **e**.

3.4. Cell Handling with Microfluidics

Cell handling is often a major problem: they are small, look similar to each other, and there are millions of them even in a small blood sample. Moreover, cells can move, which makes it challenging to find the very same cell when using several imaging modalities. Microfluidics offers a way to control cell movement and positioning by precisely controlling the flow of liquids containing the cells.

3.4.1. Microhole Array Microfluidic Chip

The work of Neugebauer *et al.* [90] presents a wonderful idea of cell handling – a very thin membrane with a microhole array that acts as a sieve for the cells. When the under-pressure is applied, the liquid starts to flow through the microholes, and the cells stick on them due to their bigger size, and stay firmly positioned. This results in a nice arrangement of the cells on a rectangular grid, where each cell gets a unique integer coordinate and can be easily located.

The membrane is manufactured out of a Si wafer, on which a thin Si_3N_4 layer is formed using chemical vapor deposition and a subsequent anisotropic etching. The regular grid of 5 μm microholes with a 40 μm pitch is formed using photolithography and reactive ion etching [90]. The chips used in this work were designed and manufactured by "Biomedical Microsystems" group at Fraunhofer IBMT[13] in St. Ingbert, Germany.

The microfluidic chip features 34 individual fields, and each of them can accommodate up to 5600 individual cells. This gives in total about 190000 cells. Once the cells are attached to the membrane, they firmly retain their positions, so that the sample can be moved and even gently shaken without displacing the cells. This methodology allows to locate the same cells after the sample has been moved between different instruments.

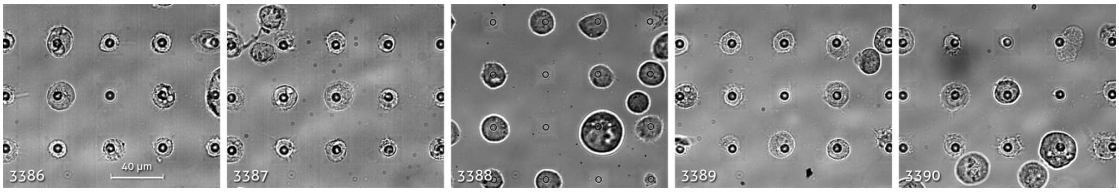
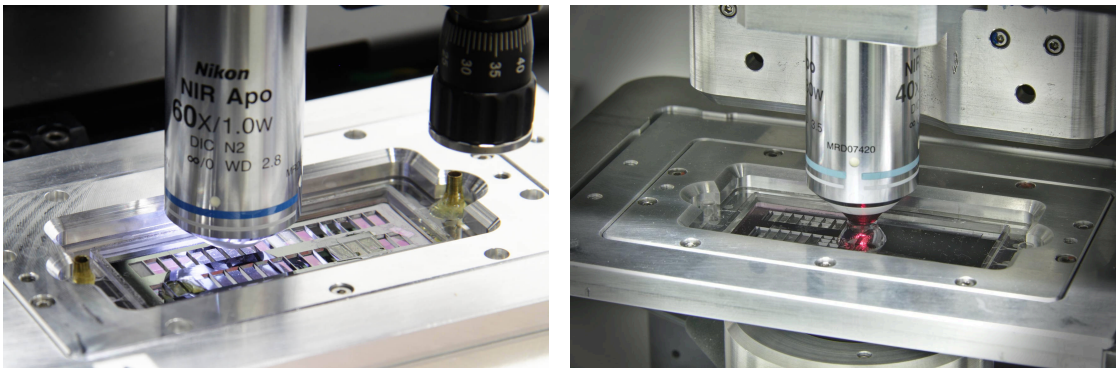


Figure 3.14.: Five bright field images of *BT-20* cells arranged on a microhole array chip made of polystyrene. The number in the corner is the measurement id in the database. All images have the same scale.

Sedimentation of Cells onto the Membrane

The chip is filled with PBS[14] from below using a syringe until the lower chamber is full, and the water comes through the membrane. Then, PBS is carefully poured from above until the top chamber is full. Next, the syringe is disconnected and the open end of the tubing is placed into a flask at a level approximately 10 cm below the microhole array chip. The gravity forces water to flow from the upper chamber of the chip through the membrane into the lower chamber, and, finally, through the tubing into the flask. Once the stable water flow is achieved, the cells can be carefully pipetted onto the water surface in the upper chamber. They slowly sink towards the membrane, and the flow of liquid forces them to move towards the open microholes, at which they stick. Thus, a nice arrangement of individual cells can be achieved, as shown in Figure 3.14. The water outlet is blocked before the upper chamber is emptied to prevent cells from drying out. The cells are studied using an upright microscope with a water-immersion objective lens.

[13] Fraunhofer Institute for Biomedical Engineering
[14] *Physiological Buffer Saline*



(a) RS785 instrument. **(b)** RS660 instrument.

Figure 3.15.: A Si_3N_4 microhole array chip in a plastic cartridge is clamped into the corresponding sample holder and placed under a Raman microscope.

Limitations and Possibilities for Improvement

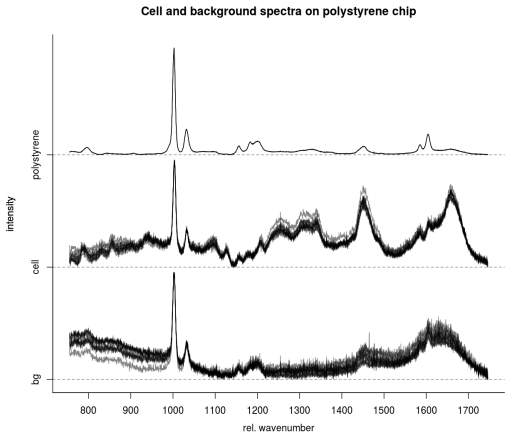
The microhole array chips are relatively easy to use, and a grid arrangement of individual cells can be readily achieved. It offers a very convenient way for micro-manipulation of cells, because the regular arrangement eliminates a need to manually identify locations of the cells. In contrast to a conventional substrate, lower number of cell clusters are formed, because liquid flow pulls the cells apart from each other; i. e. cells are stronger attracted to microholes than to each other. In the future, one can consider integrating some sort of pressure-controlling micro-machines on the chip that would allow to individually address and release a cell on a given microhole. Figure 3.15 shows a microhole array chip placed under a Raman microscope.

Despite the undoubtful benefits of the microhole array chip, there are several issues that could be improved in the future. Although each chip contains 34 individual membranes, they cannot be used for different samples, because the cells pipetted above one membrane also flow to the adjacent membranes and sediment on them. This is a design issue that may be addressed in a new specification.

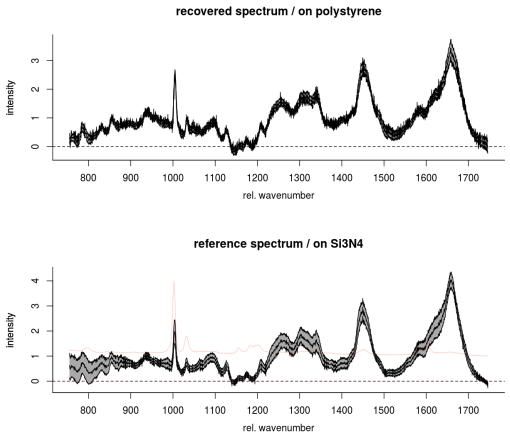
Another problem associated with the microhole array chips is that all individual wells are fluidically connected together, because they share a single under-pressure chamber in the bottom part of the cartridge. This becomes problematic if even a single membrane gets broken – in such a case the whole liquid including all cells quickly flows through the broken well, and the whole chip becomes unusable. Individual fluidic chambers under each well and a more sophisticated fluid control could resolve this issue.

The square grid arrangement of the cells allows to place up to 192,000 cells on one chip. A hexagonal lattice arrangement, which is the densest packing possible, would result in approximately 15% increase of the number of holes on the same area.

The Si_3N_4 membranes are manufactured using lithography methods, which are slow and expensive. This is worsened by the fact that no cleaning methods are known today, which could reliably remove cells and their debris from the membrane. This makes the expensive chips a single-use material. Moreover, the membranes are very fragile and can be broken by



(a) Raw spectra with signatures of polystyrene. Upper plot is the reference polystyrene spectrum.



(b) EMSC-corrected spectra (upper plot) and their comparison to spectra measured on the conventional Si_3N_4 membrane.

Figure 3.16.: Spectra of `BT-20` cells measured on a polystyrene microhole array chip (see Figure 3.14).

chip bending, strong shaking, or even by a falling water drop.

To overcome this issue, several chips were manufactured using polystyrene as a membrane material. To test the compatibility of these chips with Raman measurements, I sedimented `BT-20` cells onto them and acquired their spectra.

Although the polystyrene signature is present in the Raman spectrum, its intensity is comparable with the one of cell signatures, and can be subtracted using the EMSC[15] technique, as illustrated in the Figure 3.16. EMSC decomposes the observed spectrum into a linear combination of reference spectra that should be known beforehand [2]. Then, the undesirable components can be removed from the signal.

Unfortunately, the presence of the polystyrene signal contributes to the shot noise, so that the measurement requires 5–7 times longer exposure to yield the same SNR[16] as with the Si_3N_4 membrane. Still, plastic membranes have a good potential as a cheaper and easier-to-manufacture material. The undesired polystyrene signal can be further suppressed by a more confocal Raman instrument with a higher Numerical aperture of the objective lens and a smaller pinhole in front of the detector (for more details, see Section 3.1.3 on page 29).

3.4.2. "CanDo" Microfluidic Chip "Cartridge I"

The goal of the CanDo[17] project is to develop an integrated platform for cancer diagnostics that uses multiple methods for CTC[18] capture and enumeration followed by the molecular

[15] *Extended Multiplicative Signal Correction*
[16] *Signal-to-noise ratio*
[17] *Cancer Development Monitor*
[18] *Circulating Tumor Cell*

characterization (nucleic acid analysis). The proposed system takes the whole blood from a cancer patient and isolates CTC either using the *in-vivo* Gilupi *CellCollector* that fetches the cells from the blood stream using a functionalized wire, or, alternatively, the CTCs are isolated *in-vitro* using the inertial separation in a spiral microfluidic chip.

After the enrichment, the cells are injected into the microfluidic chip, shown in Figure 3.17, where they get enumerated using Raman spectroscopy. The Raman spectroscopy allows to identify the type of each measured cells provided that the proper training data for classification models are available, thus the concentration of the tumor cells after the enrichment can be estimated. Next, the cells are lysed and the cell lysate gets analyzed with several biochemical methods for nucleic acid characterization, including qPCR[19] for mutation detection. The role of the Raman spectroscopy is to estimate the number of the CTCs in the sample. For this reason the chip is equipped with a special Raman chamber – an opening covered by a thin glass plate, originally made of quartz. The cells entering the chip are captured with a filter in the Raman chamber. Next, the liquid flow stops, and the cells sediment onto the glass cover slip, where their Raman spectra can be acquired in the inverted microscope configuration.

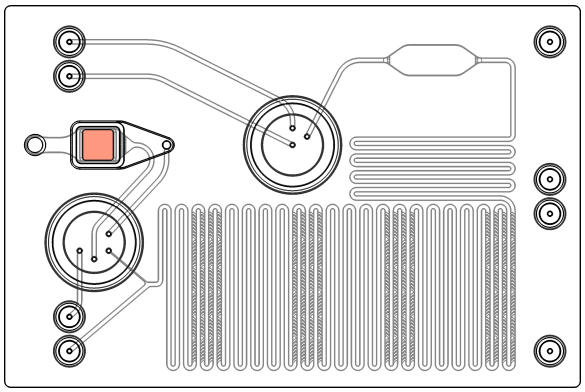


Figure 3.17.: CanDo Cartridge I for CTC detection, enumeration and biochemical analysis. The chip is made of a plastic material, so a special *Raman chamber*, highlighted in red in the image, features an opening that is closed with a quartz or glass cover slip – a material compatible with Raman spectroscopy. The chip is shown in the 1:1 scale.

Two wavelengths of excitation lasers were proposed for the Raman spectroscopy in the CanDo project: near-infrared 785 nm, and 532 nm green. The first one requires use of special cover slips made of quartz glass, because conventional cover slips, which are made of soda-lime glass or borosilicate glass, exhibit strong fluorescence in the NIR region when illuminated by 785 nm light.

Before I joined the CanDo project, Dr. Schie demonstrated that good spectra of cells in the *Cartridge I* covered by a quartz cover slip can be obtained with the 785 nm excitation laser. The question remained open, whether comparably good spectra can be acquired in case of a typical cover slip and a 532 nm excitation laser. The cells could feature some autofluorescence (see Section 3.1.2) that can deteriorate the SNR.

[19] *Quantitative real-time PCR*

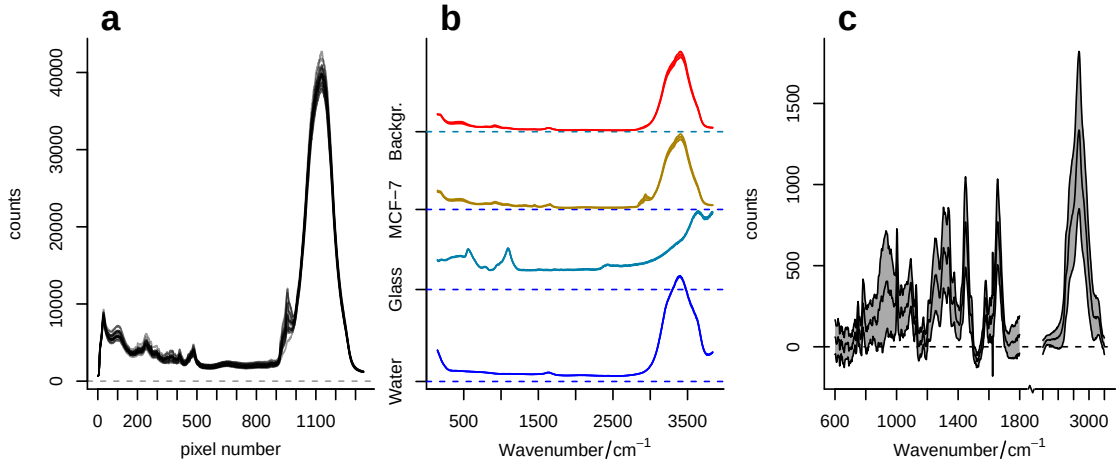


Figure 3.18.: Raman spectra of MCF-7 cells measured in the CanDo *Cartridge I* chip with a 532 nm excitation laser on a glass cover slip. Laser power 120 mW at sample position, exposure time 1 second. **a** – raw spectra as returned by the CCD camera; **b** – spectra of pure water, pure glass, cells and their surrounding medium, calibrated in terms of intensity and wavenumber; **c** – preprocessed (background subtraction, polynomial baseline correction) spectra of cells. The CH-stretching region is downscaled 4 times for the sake of visibility.

To test compatibility of *Cartridge I* with Raman spectroscopy, I used a modified version of the flexible Raman instrument (Section 3.2.1) adopted to 532 nm excitation wavelength. I used Princeton Instruments SpectraPro SP2300 spectrometer with PIXIS100 CCD camera for the registration of the spectra. *Cartridge I* was covered with a conventional soda-lime glass cover slip, MCF-7 cells in PBS were injected into the microfluidic chip and sedimented on the cover slip. 96 cells were located and their spectra were acquired. Additionally, I collected spectra of the adjacent medium, the cover slip itself, and the surrounding PBS (Raman spectrum of water). In the last two cases, the laser was focused either in the bulk glass of the cover slip, or in the water far away from the interface.

Figure 3.18 shows raw and preprocessed spectra of the cells, glass, water, and surrounding medium. The quality of the spectra is very high, and we do not observe any strong fluorescence background, which means that the glass cover slips can be used in place of more expensive quartz cover slips. In contrast to 785 nm, excitation with 532 nm wavelength additionally results in about 450% higher intensity of the Stokes signal, as per Table 3.1.

3.4.3. RoC Quartz Microfluidic Chip

RoC (abbreviated from "Raman on Chip") is a family of microfluidic chips designed and manufactured at IPHT Jena by S. Dochow and his colleagues [44, 68, 91].

The *RoC* was intended to provide a solution for a high-throughput identification of tumor cells, which combines Raman spectroscopy for chemically-specific analysis of cells with optical trapping effect and microfluidics for micro-manipulation of the cells [92].

The chips, shown in the Figure 3.19, are manufactured of two quartz glass wafers. The wafers

are micro-structured using photolithography methods and selective etching, then bound together and cut into individual chips. Each chip has three inputs: in the middle input a PBS with cells is injected at a low flow rate, into the other two a pure PBS is pumped at a much higher flow rate. At the point where all three 70 μm-wide channels meet, the cells get slowly injected between two fast laminar flows of PBS and travel one-by-one along the center of the channel. This is the effect of *hydrodynamic focusing*, which is also visible in the Figure 3.20.

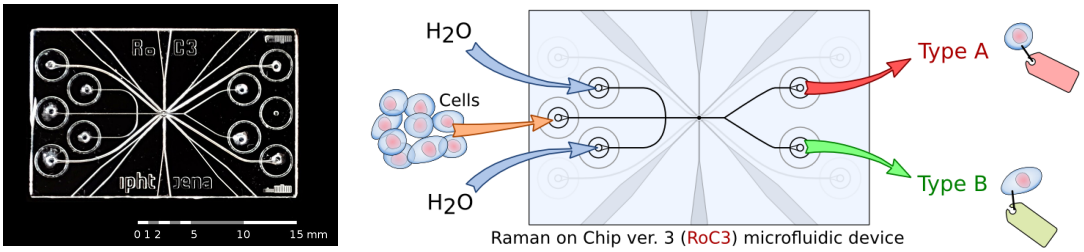
In the center of the chip there is a *Raman chamber*, which is surrounded by six optical fibers (the fibers are inserted into the corresponding meandering channels, filled by a refractive index-matching liquid). Two of them are single-mode fibers used for the optical trapping effect and, optionally, for the excitation of Raman scattering (the vertical channels in Figure 3.19), while the other multi-modal fibers are necessary for the acquisition of the Raman signal. In theory, this configuration could even allow to eliminate the need for a Raman microscope.

A cell passing through the chamber gets trapped by two counter-propagating laser beams [93]. The interaction of the laser light with the molecules contained within the cell results in the Raman scattering that is detected by the (optional) microscope and four multi-modal optical fibers surrounding the Raman chamber.

As soon as a computer, which constantly analyzes the bright-field microscopic image of the Raman chamber, detects a cell trapping event, it triggers the acquisition of a spectrum. Once acquired, the spectrum is immediately preprocessed and the classification model predicts the cell type (e. g. healthy or tumor). Next, the optical trap releases the analyzed cell (the trapping laser is switched off). Finally, the identified cell is sorted to one of the two outputs by varying the flow rates of the liquid going out of the chip. This is achieved by controlling microfluidic syringe pumps attached to the chip outputs.

Previous Results

Dochow *et al.* [44] demonstrated cell trapping in a microfluidic chip using a pair of 1070 nm lasers. The chip was placed under a Raman microscope, and Raman spectra of ~400 trapped cells were collected one by one. Later, Dochow *et al.* [91] reported on-chip excitation and detection of the Raman signal using only optical fibers injected into the RoC2 chip (a prede-



(a) Microfluidic chip RoC3.

(b) Schematic use of the RoC3 chip.

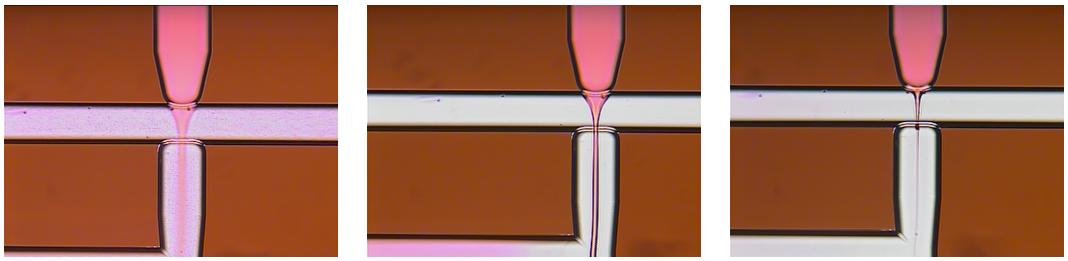
Figure 3.19.: RoC3 microfluidic chip, intended to provide an integrated solution for the Raman-based flow cytometry, i. e. a spectroscopic identification and sorting of cells injected into the chip (one by one). For the details, see paper Dochow *et al.* [44]. Image rights: unpublished own work.

cessor of the RoC3), i. e. without a Raman microscope. This was done, however, not with cells, but with nicotine and urea solutions, as well as with polystyrene beads. These samples feature a way higher Raman cross-section than biological materials, whose measurements are very challenging.

Although the proposed hydrodynamic focusing and microfluidic cell sorting in the RoC3 chip were demonstrated, these methods, especially the sorting, did not work reliably.

Finally, Freitag *et al.* [94] reported fast Raman-based identification of cells in the RoC2 chip at continuous flow. However, the cells were tagged with immuno-multicore SERS labels [95], and the analysis of the spectra was *not* performed in real-time, i. e. the immediate cell sorting could not be possible, even if a reliable cell sorting method would be available.

Experiments with Hydrodynamic Focusing and Cell Trapping



(a) 1:10 ratio, 11 nl/s. **(b)** 1:10 ratio, 1100 nl/s. **(c)** 1:100 ratio, 200 nl/s.

Figure 3.20.: Experiments with hydrodynamic focusing. "'PDG2'" chip, channel width 300 µm. Different ratios of flow between the input channel with the dye and two input channels with the water are shown. The flow value refers to the total flow through all three input channels. Image rights: unpublished own work.

It was necessary to demonstrate the proposed functionality of the RoC3 chip with cells, optimize the bottlenecks, and create an automated Raman-activated cell sorting system based on this chip.

To start with, I experimented with hydrodynamic focusing using two liquids in a simpler microfluidic chip called "PDG2" originally designed for droplet generation (Figure 3.20). Motorized syringes were used to pump the liquid (*neMESYS* from *Cetoni GmbH*) at a precisely controllable flow rate. Under suitable conditions I could observe a laminar flow of both liquids after they were brought into contact, i. e. they did not mix with each for a long time (seconds). For the total flow rate between 20 nl/s and 2000 nl/s the colored liquid formed a filament with a sharp boundary, which was exactly at the center of the channel (Figure 3.20b). The filament was stable at ratio between flow rates ranging from 1:10 to ~1:100, and its width was a function of this ratio. When the total flow rate was below ~20 nl/s, the boundary between the flows was smooth, i. e. they were mixing with each other (Figure 3.20a).

Any shaking of the microfluidic setup seriously affected the flow stability.

During the experiments with the RoC3 chip, a special chip holder [96] was used to connect

tubings and optical fibers to the chip. We used PEEK[20] tubings, as this material has much higher stiffness in comparison to more common PTFE[21]. The tubing stiffness is important, because soft tubings get laterally stretched when the pressure is applied, they store more liquid and act as pressure capacitors. The tubings were kept as short as possible. The experimental setup is shown in Figure 3.21.

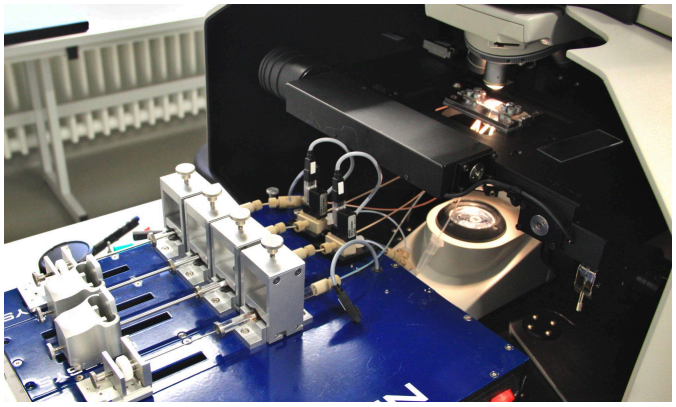


Figure 3.21.: Photo of the microfluidic setup with RoC3 chip under a Raman microscope.

Discussion of RoC3 Quartz Chip

Although the functionality of several individual parts of the RoC3 chip was demonstrated, in particular the hydrodynamic focusing, optical cell trapping, Raman spectroscopy of trapped as well as moving immuno-SERS-labeled cells, the correct interplay of all RoC3 subunits for a truly on-chip identification and sorting of the cells was not achieved.

The meaningful use of the RoC3 chip requires far more than a demonstration of functionality of all its subunits. Each step, be it microfluidics, spectroscopy, image analysis or chemometric cell identification, must work absolutely reliably, and in real time. This requires a complex, fast and precise orchestration of the whole system – image analysis for cell tracking, control of the devices (laser, spectrometer, several microfluidic pumps, bright-field camera with its light source) and a quick processing of Raman spectra for the cell identification.

Inherent Design Problems

First of all, the chip design is still far from optimal. In my opinion, one of the biggest problems is the huge length/width ratio of the microfluidic channels combined with syringe pumps. The channels of the chip are about 20 mm in length and 0.07 mm in diameter. These narrow channels often get blocked by dust particles or cell clusters during the experiment, as illustrated in Figure 3.22. The problem is worsened by the fact, that the microfluidic syringe pumps are big devices (see Figure 3.21), so they are located at least 25 cm away from the chip

[20] *Polyether ether ketone*
[21] *Polytetrafluoroethylene*

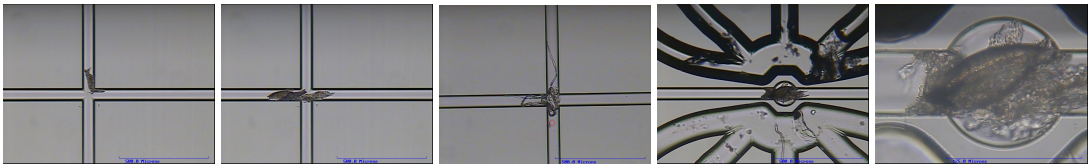


Figure 3.22.: Bright-field images of RoC3 microfluidic device. The narrow (70 μm) channels of the RoC3 chip are easily blocked by dust particles, cell agglomerates, or both. No cleaning procedure is available for such problems.

itself. This adds additional long and narrow (typical diameter between 60 μm and 500 μm) tubes to the microfluidic circuit.

To better understand the implications of this, we have to discuss the underlying physical laws. Microfluidics has a lot of similarities to the electric circuits and can be modeled with them. In fact, the pressure drop ΔP is equivalent to the voltage, and the flow rate Q is equivalent to the current. One can also define hydrodynamic capacitance and hydrodynamic resistance, and apply adapted Ohm's and Kirchhoff's rules to the microfluidic circuits [97, 98].

- The hydrodynamic Ohm's law reads $\Delta P = R_h Q$, where the quantity R_h is called *hydrodynamic resistance*. For a channel of a circular cross-section with a radius r it is given by the Hagen–Poiseuille law [99]:

$$R_h = \frac{8\mu L}{\pi r^4}. \tag{3.5}$$

  Note that R_h depends on the fourth power of the tube radius[22]. Thus, a twice narrower channel has a **16** times higher resistance!

- *Hydrodynamic capacitance* C_h is a quantity that characterizes the change of the volume of the liquid in a microfluidic circuit caused only by the change in pressure. This happens due to both fluid compressibility and the stretching of the elastic fluid channel. The equation $Q = C_h \frac{d\Delta P}{dt}$ is analogous to the one valid for electric circuits, $I = C\frac{dU}{dt}$.

The book of Tabeling [98], page 77, considers the so-called "*microfluidic bottleneck*" problem. It deals with transitional effects and is modeled as an RC-circuit (see Figure 3.23). The problem directly represents the pumping scheme used with the RoC3 chip: it considers a motionless syringe connected to a long narrow channel, both filled with liquid of volume V. The syringe piston begins an abrupt movement creating a pressure ΔP_{in} and a liquid flow with the rate Q_{piston}. Such a situation is encountered during a cell sorting – from two outlets, one outgoing flow must be quickly stopped, and another one suddenly started. The motion of the piston, however, does not *immediately* lead to the flow of the liquid through the channel. Instead, the major part of the displaced liquid remains in the syringe (which is a big elastic tube), presses on it from the inside and elastically stretches it. This hydrodynamic capacitance of the syringe,

[22] This is not the same rule as in the electrical circuits, where the resistance is inversely proportional to the radius squared. The reason is that molecules of the liquid in a pipe behave differently than electrons in a wire.

combined with the hydrodynamic resistance of the channel, affect the rate Q_{chip} of the flow entering the microfluidic chip. In fact, Q_{chip} asymptomatically approaches the steady state with a characteristic time given by

$$\tau = R \cdot C = \frac{8\mu L}{\pi r^4} \cdot KV, \tag{3.6}$$

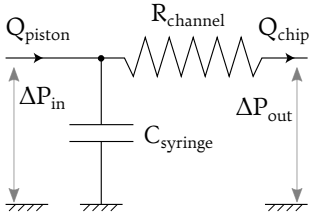


Figure 3.23.: Illustration of the *microfluidic bottleneck* problem. An abrupt start of the pump syringe creates the Q_{piston} flow, but the liquid remains in the syringe and stretches it prior to flowing into the channel. As a result, the flow rate Q_{chip} grows with a characteristic delay time τ until it reaches Q_{piston}. The channel resistance leads to $\Delta P_{\text{out}} \ll \Delta P_{\text{in}}$.

where K is the effective compressibility of both the liquid and the material of the microfluidic circuit. For the geometry used in our experiments (rigid tubes with 64 or 130 μm diameter and about 30 cm in length; 1000 μl volume of the syringe made of borosilicate glass; quartz chip with 70 μm channels of 2 cm length), the characteristic time is about 32 ms for ø130 μm tubes and 323 ms for ø64 μm tubes. Note, that this was calculated for an ideal system without any bubbles, and the elasticity of tubes was neglected. A soft teflon tubing or a tiny air bubble would drastically increase the characteristic time. In the current chip design, an *elastic rubber sealing* is used at the point where tubings and the chip come into contact; this compressible rubber layer additionally worsens the response time of this circuit. For comparison, if a tubing could be completely eliminated, i. e. the pump would inject the liquid directly into the chip, we would get the characteristic time τ = 14 ms, which is about one order of magnitude lower.

Still, the RoC3 channels themselves have a very high hydrodynamic resistance, which in case of ø130 μm tubes contributes about 44% to the total resistance. Therefore, high pressure difference is required between the chip input and output to push the liquid. This high pressure makes water leakages, especially on the interface between the chip and tubings, almost unavoidable. Connectors are still an open recurring problem in the field of microfluidics.

One can argue that a thin ø130 μm tube could be replaced by a bigger one to reduce the hydrodynamic resistance. Unfortunately, the cells are more dense than the surrounding medium and they sediment on the tube walls. This is less likely to happen in narrow tubings, where the liquid moves faster. The strongest sedimentation happens within the syringe containing cells to be analyzed, hence the major part of them remains unavailable for the analysis (Figure 3.24).

In 2008 Lau *et al.* [100] already demonstrated Raman-activated cell detection and sorting on a microfluidic chip, but the cell recovery rate was very low. Their idea was to repeatedly circulate the cells through the interrogation area until they are trapped with a focused laser, identified with Raman spectroscopy, and then pulled by the trapping laser into another "lane" of a parallel laminar flow that leaves the chip. From a high number of cells passing through the interrogation area during the measurement, only a single one gets analyzed and sorted, while the others go to the next loop, where many of them get lost or damaged.

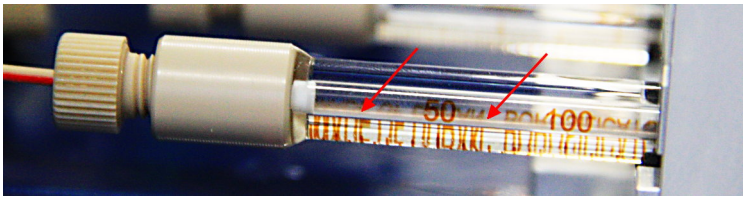


Figure 3.24.: Cells sedimentation in a micro-syringe is observable about 10 minutes after the syringe was filled with the cell suspension.

Outlook — what can be improved in the *RoC* design?

Guidelines recommend to minimize the length of fluidic channels and to refrain from use of syringe pumps. Elastic tubes, glues and sealings should be avoided [98].

Particularly for the RoC3 chip, it would be beneficial to shorten the narrow channels, i. e. the interconnections between the functional parts could be done using much broader fluid channels.

In 2002 Fu *et al.* [101] demonstrated on-chip cell sorting on a T-junction. They used a peristaltic pump located just about 1 mm away from the sorting intersection. Such pump is composed of three monolithic elastomer pneumatic valves [102]. Each valve consists of a small flexible membrane separating the fluidic channel and a pressure control line; the fluidic channel can be blocked by the membrane pushed from the other side by the air. A peristaltic pump has a number of advantages over a piston-based one:

- The small pump is integrated directly into the chip. The fabrication is fairly easy, and the whole laboratory setup is less bulky.
- Since it is integrated, there are no connections between the pump and the chip. As discussed before, the connections contribute to the hydrodynamic capacitance of the system and can leak under high pressures.
- The volume of the displaced liquid is small, and the pump itself is located next to the functional parts of the microfluidic chip. This eliminates the aforementioned "bottleneck" problem and allows for a fast and precise flow control.

Currently, the cells are pumped from the outside into the chip and a lot of them get lost in the syringe or in the tubing. To solve this problem, one could integrate a special chamber for the analyte directly into the chip. It could be located just few millimeters away from the optical interrogation and the cell sorting areas, next to the peristaltic pump. A magnetic steer-bar [103] or magnetic particles [104] placed into this vessel could prevent cells from sedimentation and formation of the clusters. Since the analyte chamber does not stay under pressure, it can have an opening for an easy refilling.

Integration of many components (e. g. peristaltic pump) into a single chip poses some additional challenges, as one has to work with different materials. Unfortunately, Raman spectroscopy imposes strong restrictions on them. Therefore, the whole chip of the discussed here design should be entirely fabricated of a Raman-compatible material (such as quartz), which is challenging, especially for the peristaltic pumps. Alternatively, one can stick to a hybrid

chip – a conventional plastic frame with a small optical interrogation insert made of a Raman-compatible material. In this case the fluidic connection between the main chip and the insert has to be properly designed. The insert layout would be additionally complicated by the need to accommodate optical fibers for the cell trapping. On the other hand, plastic chips can be equipped with standard reliable fluid connectors, such as Luer connector.

To sum up, the RoC3 chip is a blend of innovative ideas, but has a lot of weaknesses that hinder its successful application. There is a number of unresolved issues, and for some of them I have proposed solutions. Their verification would, however, require a careful design and manufacturing of new chips – a costly and lengthy procedure.

3.5. Discussion of Raman Instrumentation

We designed several automated Raman spectroscopic systems for biomedical applications. These prototypes pave a way towards a Raman-based cancer diagnostics and demonstrate what is possible to do. However, several issues have to be considered before implementing such systems into a clinical practice.

Transfer of Cells Between the Instruments

One of the issues that arose in the RamanCTC project is the transfer of the cell sample from one system to another, in particular finding a specific cell under the next instrument.

The microhole array chip (see Section 3.4.1 for details) offers a great way to keep the cells immobilized. The chip features 34 wells, and each of them can accommodate up to ~5600 cells. Although the rectangular grid aids in the search of a particular cell, this search has not been automated so far. The process is done manually, and, given a small field of view in the microscope, it is slow, tedious and very error-prone. The situation is complicated by the fact, that the RS660 and the CellCelector systems observe the membrane from different sides, i. e. they deliver mirrored images of the same sample. Therefore, one of the future goals would be to implement an automated search of specific cells.

There is a demand for a standard representation of the grid coordinates for all instruments in the RamanCTC tool chain, shown in Figure 1.2. We need at least an automated `goto`-routine that would accept the integer grid coordinates and quickly bring a given cell into the center of the field of view. For an efficient processing of the multiple cells we would additionally require some kind of a network communication protocol that can transfer a list of coordinates of the selected cells from one instrument to the next one.

During the sample transport between the instruments the microhole membrane gets shaken, which can result in the displacement of cells. To solve this issue, the distances between the instruments have to be kept short.

An even better solution would be to reduce the number of individual instruments in the tool chain and build multipurpose systems that can sequentially investigate the same sample using several modalities. The fast optical pre-screening of the specimen (first instrument in the RamanCTC workflow) can be combined with the Raman interrogation (RS660). In fact, the RS660 already contains the required hardware for the fast wide-field imaging of the whole

sample, so only the software part has to be adapted to enable a morphological characterization of cells.

Going further, we can consider the integration of all three prototypes, i. e. the wide-field optical image analysis, Raman interrogation, and cell picking, into a single system. Since the cells are sitting atop the microhole membrane, the cell picking can be performed only from above. The wide-field imaging can be done from both sides of the membrane. Finally, Raman measurements have to be preferably done from above, otherwise the plastic cartridge holding the Si_3N_4 membrane would stay in the beam path and could contribute some background signal to the spectra.

The last instrument in the RamanCTC tool chain is the CellCelector, which is based on an inverted fluorescence microscope. It has plenty of space above the motorized slide holder, which is used to position the illumination unit and a robotic micro-pipette. Again, the hardware of the CellCelector allows automated wide-field imaging of the whole sample, and, being controlled by cell recognition and analysis algorithms, can be used for morphological screening. The Raman modality can be either integrated into the inverted fluorescence microscope itself (see, for example Figure B.1), or it can be implemented as a small removable tool coming to the sample from above, interchangeably with the robotic micro-pipette. In either case, **all** three prototypes would be integrated into a single multi-modal instrument, and the need to transfer the specimen and localize the cells with all associated problems would be completely eliminated.

Compatibility Issues

As mentioned in the previous subsection, the RamanCTC tool chain encompasses several instruments that are dependent on each other. Thus, to successfully integrate them together into a single workflow, one has to develop standard interfaces between them, which would allow to exchange data and control commands. Unfortunately, this was not the case here, and each instrument was designed in isolation from the others, which complicated the final integration.

There are some mechanical incompatibilities as well. For example, our available incandescent lamp for the intensity calibration (Kaiser Optical Systems) does not mechanically fit into the RS660 instrument. Another issue is a plastic frame around the microhole array chip that holds the liquid poured onto the chip. It is necessary to fill the chip with ~4–5 mm thick layer of liquid – a prerequisite for a reliable cell immobilization. Unfortunately, such a high frame collides with the objective lens and renders a major part of the chip surface useless. Although the frame is detachable, its removal is associated with an undesirable mechanical stress on the microhole membrane. The problem could be avoided by optimizing the geometrical shape of the frame.

Finally, most of the instrument control is done via proprietary products. The proprietary software, which cannot not be modified by the end user, dramatically hindered the integration of the prototypes into the RamanCTC workflow. This lead to a lot of excess work to find workarounds; an open source approach and carefully designed APIs[23] would make this process

[23] *Application Programming Interfaces*

much more efficient.

Similar compatibility issues between the individual prototypes and tools designed by different collaborators have also arisen in the CanDo project. In general, the problem could be mitigated by a careful planning, direct communication between the people engineering the interfaces, and making the developed specifications, documentation, data and source code available to all involved parties.

4. Integration of Raman Instruments and Data Management

> "Let us change our traditional attitude to the construction of programs: Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to humans what we want the computer to do."
>
> — Donald E. Knuth, Literate Programming [105]

This part describes what methods have been used to keep vast experimental data organized and easily accessible. It describes a database for the storage of the experimental data, as well as the software packages for automated data acquisition, data retrieval from the database and visualization.

4.1. Data Representation in a Spectroscopic Experiment

4.1.1. Multivariate structure of the spectral data

A spectrometer uses a diffractive element to disperse the light, i. e. to spatially separate photons with different wavelength. The spatial intensity variation is a continuous function of the wavelength. However, as soon as the photons get absorbed in the detector, this function gets discretized. For N pixels of the camera, N independent measurements are performed, so each spectrum is a vector in the N-dimensional space.

It is convenient to store spectral data in a tabular form, where columns denote variables, for example individual camera pixels, and rows contain individual observations (what we call spectra). Such representation of the data, when all variables are located in the columns, and all observations in the rows, is called *tidy data*. Tidy data is a data representation form that is very important in data analysis, especially for machine learning and visualization. This is not the only possible data representation; for example, a completely different organization is used in the database tables to store the same data (see Section "Long representation of the spectral data" below).

Since typically the spectrometer alignment is not changed between the measurements, we can associate *wavelength* with the columns, as shown in Table 4.1. Note, that we will refer to this metric as "wavelength" in the future, even though it can be expressed in different units, like pixel numbers, cm^{-1}, nm, eV, etc. In any case the underlying physics is the spatial separation of photons with different "wavelength".

Table 4.1.: Tabular representation of spectral data (*tidy data*). Columns contain different wavelength channels, each row represents one measurement – an individual spectrum.

	Wavelength, nm								
#	602.0	622.0	642.0	662.0	682.0	702.0	722.0	742.0	762.0
1	501.8	524.0	481.3	453.4	482.2	556.3	595.9	607.8	711.3
2	500.5	525.7	480.8	451.3	480.5	560.3	599.9	614.3	716.4
3	466.0	490.1	448.6	419.8	448.9	515.6	557.0	565.2	657.5
4	477.5	501.4	459.7	433.1	456.7	527.4	570.5	582.5	678.1
5	439.4	461.2	421.3	394.7	418.1	485.6	521.8	530.7	623.1
6	435.8	458.2	418.5	392.6	419.8	484.6	526.1	533.7	633.3
7	424.7	445.5	408.0	384.2	408.0	467.8	512.7	518.9	618.9
8	409.2	429.7	392.1	369.0	392.0	453.3	493.0	501.6	603.1

We can associate additional variables with each individual spectrum. We would call them *extra data* or *metadata*. Typically these data describe experimental conditions or labels associated with each spectrum. Examples can include spacial x, y, z coordinates, date/time, sample name, instrument settings, etc. However, these variables are *not* related to the wavelength vector of Table 4.1, and have to be kept separately. Table 4.2 demonstrates this methods of data organization. This is the form, in which data are manipulated by the `R` package `hyperSpec`.

Table 4.2.: Tabular representation of spectral data alongside with the associated extra data.

	Extra data			Wavelength, nm				
	x	y	sample	602.0	622.0	$\cdots$	742.0	762.0
#1	0.1	4.9	A	501.8	524.0	$\cdots$	607.8	711.3
#2	0.2	4.9	A	500.5	525.7	$\cdots$	614.3	716.4
#3	0.3	4.9	A	466.0	490.1	$\cdots$	565.2	657.5
#4	0.4	4.9	A	477.5	501.4	$\cdots$	582.5	678.1
#5	0.1	4.9	A	439.4	461.2	$\cdots$	530.7	623.1
#6	0.2	5.3	A	435.8	458.2	$\cdots$	533.7	633.3
#7	0.3	5.3	A	424.7	445.5	$\cdots$	518.9	618.9
#8	8.7	-3.3	B	409.2	429.7	$\cdots$	501.6	603.1

4.1.2. Long representation of the spectral data

A spectrum can also be saved as a long "tidy" table, where the wavelength values themselves are encoded by a variable. Thus, in the simplest case the spectrum is saved as a list of x, y tuples, where x is the wavelength, and y is the intensity. Additional metadata can be added to this representation, leading to a list of $x, y, v_1, v_2, \ldots, v_n$ tuples, v_i being some additional variable, like time, spatial coordinate, sample name, etc. If the number of wavelength channels or their exact values can vary from spectrum to spectrum, this verbose form becomes beneficial.

A careful reader would note, that this way of data organization is highly redundant, as for each data point in any given spectrum a new row in the table is created. Only x and y variables

have different values from row to row, while the values of other variables (i. e. metadata) get repeated N times without any changes.

4.2. Keeping experimental data organized within a database

A *database* is an organized collection of data stored in a form of tables linked together. A *DBMS*[1] is a software that executes commands and automates data collection, organization and retrieval from the underlying database. A process of designing a database and determining its structure is called *data modeling*.

In this work a large set of experimental parameters related to the measured spectra had to be tracked. It was necessary to keep track of instrument settings, preparation procedure for the sample, instrument calibration, etc.

I decided to use a *relational model* for the database management, where all data records are represented in terms of tables, linked together with *relations*. The collected data get broken apart into pieces that are written into the different tables and then linked together according to the database structure.

I selected a free open-source DBMS called `PostgreSQL` for data management. It meets all our requirements, including high performance and scalability, support of transaction, network access and tight integration with `R` and `Python` programming languages. `PostgreSQL` has been deployed on a server located at Leibniz IPHT in Jena and accessible over an encrypted network connection.

4.2.1. Database Modeling for Storage of Spectral Data

Each spectrum is a set of `(x, y)` pairs, where `y` represents a spectral intensity at wavelength channel `x`. Typically, many spectra share the same wavelength axis. However, after each spectrometer calibration the `x`-axis gets redefined. Therefore, as was mentioned in the Section 4.1.2 before, it makes sense to store the spectral data as (`spectrum_id, x, y`) tuples, where `spectrum_id`, called *foreign key*, is a unique integer number that identifies each spectrum. The table with these data is called `spc_value`. All stored values are arranged by the column `spectrum_id`, then by column `x`. This is by far the largest table in the database, which contains millions of records. For an efficient search, a B-tree index is created on the column `spectrum_id`, which allows to execute searches in logarithmic time.

There is additional information associated with each particular spectrum, such as spatial coordinates of the measurement point, acquisition timestamp, bright field image of the measured area, etc. To avoid redundancy and improve data integrity, these pieces of information have to be saved in a separate table called `spectrum` and containing one row per individual spectrum. The identifier `id` of each spectrum is linked to the column `spectrum_id` of the table `spc_value` (see Figure 4.1). Such a connection allows to uniquely identify a relationship between both tables. The process of data splitting and optimization of the database structure, which helps to avoid redundancy and improve performance, is called `database normalization`. Going further, we create a table `measurement` that contains information as-

[1] *Database Management System*

sociated with a particular experiment, like instrument settings and the measured specimen. Since any specimen can be measured many times, its description has to go into a separate table called `specimen`. Moreover, there is a table `cell_line` that stores general information about cells, from which specimens get prepared. These tables form the core of the database and are extensively used to store the experimental data (see Fig. 4.1).

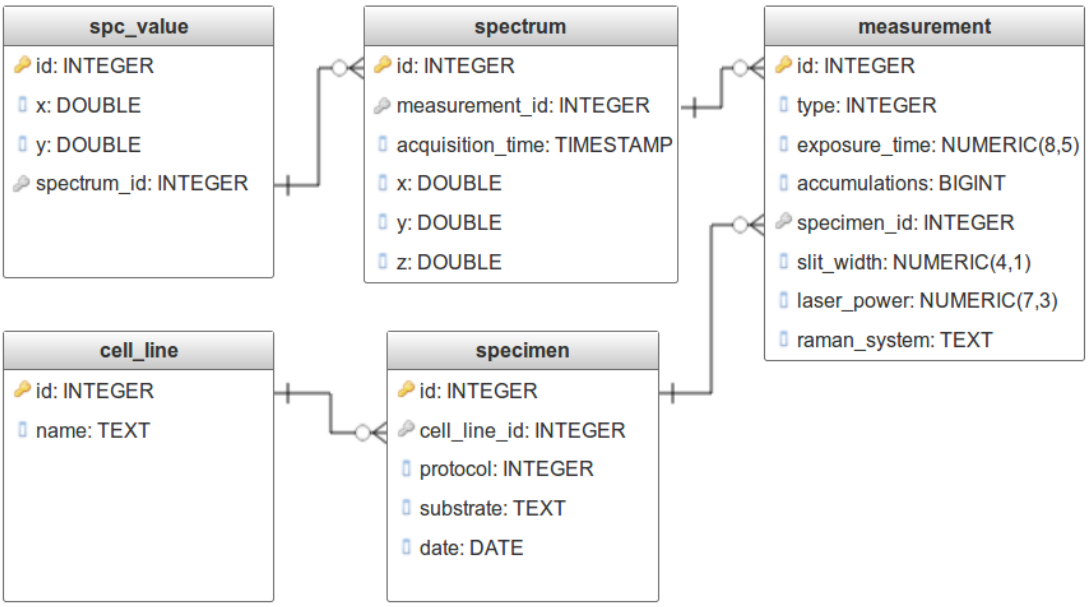


Figure 4.1.: Core tables of the database with several columns shown. For each experiment, a record in the table `measurement` is created. During the experiment many spectra are created. Each row in the table `spectrum` contains data associated with a given spectrum, while the actual data points recorded by the spectrometer go into the table `spc_value`. Tables `specimen` and `cell_line` store the information about specimens and cell lines, correspondingly. The tables are related to each other via foreign keys using field `id`.

There are several additional tables in the database, but they are less important for the design of the experiment, therefore I leave out their description here. A diagram of the whole database structure is provided in the Appendix, Figure C.1 on page 125.

4.2.2. Constraints and mechanism of transactions

Data sanity and consistency are important aspects of a scientific experiment. Missing data, or, even worse, incorrect information can quickly render experimental observations unusable.

The database structure has to reflect the design of the experiment. By applying specific constraints on the table columns, like enforcing a particular data type and/or a range of accepted values, one can ensure that meaningless or incomplete records cannot be inserted into the database in the first place. An example of such constraint is that one cannot create a record about a measurement where information about a specimen to be measured is missing. Furthermore, to create a record about a specimen, one has to indicate a preparation date and

refer to a particular cell line from the table `cell_line`.

An additional layer of cohesion brings a mechanism of *transactions*. A database transaction is an atomic unit of work in a DBMS. A transaction proposes an "all-or-nothing" principle, it groups several individual operations into a single unit of work and ensures that either all of them get executed at once, or none in a case of failure.

We use transactions to ensure, that one cannot create an empty experiment, e. g. a record in the table `measurement` that does not contain any spectra associated with it. Moreover, one cannot create an empty spectrum that contains no data points associated with it in the table `spc_value`. In case of a technical problem the DBMS cancels all pending operations leaving the database intact.

PostgreSQL and SQLite

PostgreSQL and SQLite are two different open-source DBMS, both of which I used in this project. `PostgreSQL` is a feature-rich solution that supports network access, multiuser installations (even with a fine-grained access control), parallel sessions, isolated transactions, etc. For this reason, `PostgreSQL` was used as the single primary DBMS that is accessible from all laboratory or office computers and represents a central data repository which is always in the actualized state.

A shell script taken from the official Wiki of the `PostgreSQL` project is responsible for an automated backup of the database. The script is executed as a `cron` job every day and makes dumps of the database under `/mnt/backup/`.

`SQLite` is a lightweight `DMBS` that stores an entire database in a single file. This allows to conveniently make copies of the main database that can be modified or shared. I developed a `Python` script that loads all spectra from the remote database and creates a local `SQLite` copy of it (the script is a part of the `db2spc` package, see Section 4.5). Note, however, that the script makes copies only of the core tables of the database, which are shown in Figure 4.1.

Although the `db2spc` package, which imports spectra from the database into `R`, can load spectra directly from the remote `PostgreSQL` database, a local database copy dramatically boosts the data retrieval speed as the relatively slow network connection is eliminated.

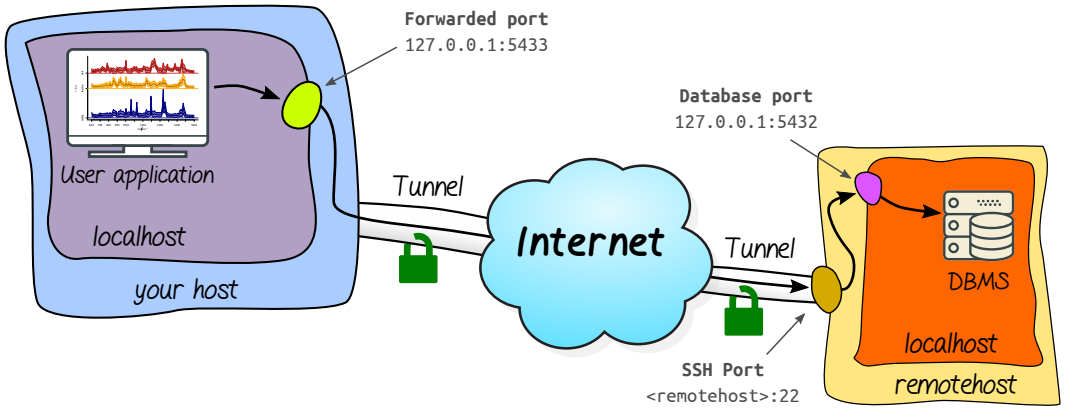
4.2.3. Network security

Since we use Raman instruments to acquire spectra of patient samples, and transmit these spectra over a public network, we have to ensure that the data are being properly encrypted on the way through the network. The computer with the DBMS has been configured in such a way, that it had only single port 22 (*Secure Shell*, SSH) listening to the outside world. Any incoming connection to the SSH port must be properly authenticated by the OpenSSH server; we used 2048 bit asymmetric RSA keys (see Section 4.2.3) for the authentication.

Using `OpenSSH`, one creates a so-called SSH tunnel, which is a secure virtual connection between two communicating computers, which can encapsulate traffic of almost any other network protocol. At the same time, a local port on the client computer is created. This port is forwarded through the secure tunnel to a local port on the server side, on which DBMS is listening. None of these ports are discoverable from any other computer in the network. An

external observer cannot even guess what kind of information is being transmitted through the tunnel across the Internet.

The user application working with the database connects to the local port `5433` and gets access to the `PostgreSQL` database software. Both programs "think" that they are running side by side on the same machine, but in reality all network packages coming to `localhost:5433` get encapsulated into SSH packets and transmitted to the remote server over the Internet in an encrypted form. Then, they get decrypted and delivered to the local port `5432`, which is being listened by the `PostgreSQL` DBMS. Figure 4.2 illustrates this concept. The responses of the DBMS sent back to the user application are handled in exactly the same way.




```
you@your_host$ ssh -L 5433:localhost:5432 remotehost
```

Figure 4.2.: Secure connection between a local application and a remote database over an encrypted virtual tunnel, and a corresponding shell command that establishes this connection. Here, the `-L` option specifies that an encrypted tunnel should be created, mapping an existing port `localhost:5432` at `remotehost` to a newly created port `localhost:5433` at the local machine called `your host`.

The authentication and encryption mechanism, which keeps transmitted data safe, is based on the public-key cryptography, which is addressed in the following section.

Public-key Cryptography

A public-key cryptographic system uses a pair of keys, called *public key* and *private key*. The first one may be distributed widely, while the second one has to be kept secret. Each of these keys allows to encrypt an arbitrary message, and only the corresponding complementary key can be used to decrypt the message back. From the mathematical point of view, each key works as a one way-function – a function that is easy to compute for any input value, but computationally hard to invert and find the input given the output value. This property of keys accomplishes two functions: authentication and protected data transfer.

Two persons, who have exchanged their public keys in advance, can establish a private encrypted communication channel between each other. The message is encrypted using a public key, and only the owner of the matched private key is able to decrypt it. If we assume

that only a single person possesses the matching private key and keeps it secret, then we know that we are communicating with this person. This is the idea of a key-based authentication.

To set up a key-based authentication with OpenSSH, the user generates a key pair and deploys the public key on the remote server. The last action, of course, requires some form of an authorized access to the server, e.g. physical access, password-based or via the system administrator. At the end, a set of public keys from multiple users is registered on the server. On each login attempt the connecting user has to prove the possession of a private key that matches with one of the deployed public keys. Once the authentication is done, a random temporary session key is generated and the further traffic gets encrypted using fast symmetric encryption. OpenSSH supports a number of algorithms for authentication and encryption. In our configuration of the SSH tunnel, I used RSA for authentication and Advanced Encryption Standard (AES) for symmetric encryption, which were considered to be among the most secure algorithms at the moment of writing this text.

4.3. Data acquisition software

I designed and implemented a data acquisition software that orchestrates the spectroscopic experiment and populates the remote database with the acquired data, as illustrated in Figure 4.3. The program is written in Python 2 programming language [106] using the Qt library [107] for the GUI[2]. The software is named spc2db[3].

4.3.1. Graphical user interface

The GUI of the spc2db program was implemented using the "*Qt Designer*" rapid prototyping tool. The application has a number of dialogs that assist the operator during the experiment (see Figure 4.4), and features several important functions:

- The most recent spectrum is displayed in the main window (Figure 4.4a).
- The number of spectra of each type collected today is displayed in the main window.
- A tool-tip with keyboard shortcuts is shown in the main window.
- The acquisition settings are specified, including exposure time, number of accumulations, slit width[4], laser power, as well as information about the sample (see below), comments, and the name of the instrument operator (Figure 4.4b–c).
- By default, the input fields contain values from the latest experiment (except for the date, which is automatically set to today), so usually there is no need to update them, which saves time and efforts.
- The values accepted by the input fields are limited to the reasonable range, thus it is not possible to provide meaningless values.
- There is an editor for the cell samples, which can be used to show and update information about available cell samples or to define a new one (Figure 4.4d).

[2] *Graphical User Interface* of a computer program

[3] The counterpart, which exports data *from* the database, is named "db2spc" and discussed in Section 4.5.

[4] The slit width has to be adjusted manually on the RS785 and then specified in the software for the documentation purposes. The RS660 has a fixed 50 μm slit, so this option is grayed out in the software

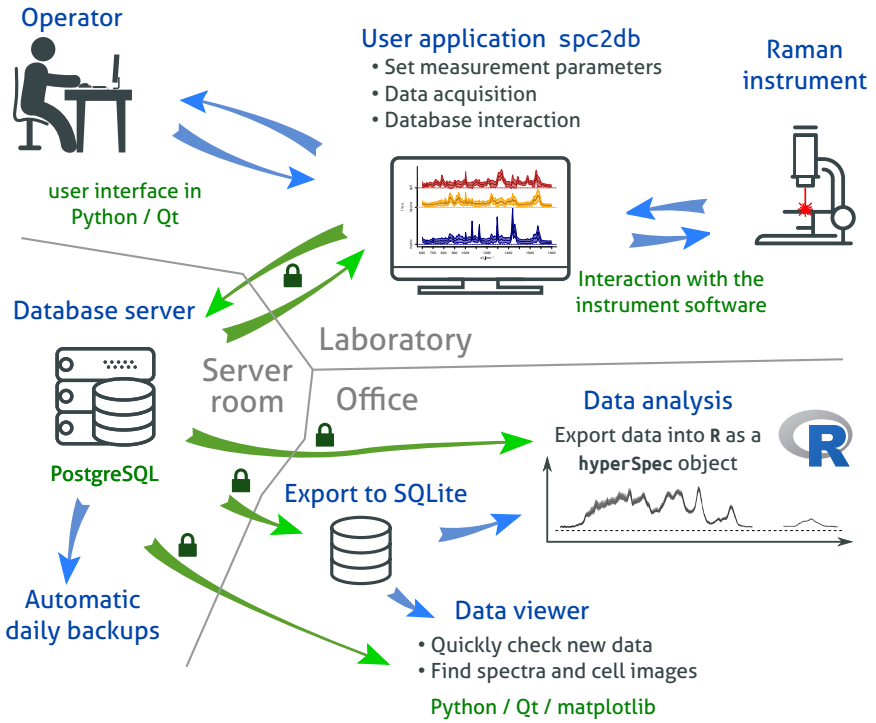


Figure 4.3.: Diagram of the `spc2db` functionality and its interplay with other components of the data management workflow. The application interacts with the user and assists in conducting a spectroscopic experiment. It helps to define a new measurement based on the instrument configuration and the information already available in the database. The software makes a list of cells that the user wants to measure, who selects these cells under the microscope using a joystick to move the motorized stage. Afterwards, the application controls the instrument, and the Raman spectra are collected from the preselected cells in a fully automated manner. The software immediately saves each new spectrum into the remote `PostgreSQL` database.

- There are several dialogs responsible for the correct positioning of the motorized stage, such as calibration of the microhole array membrane position, saving a list of cells to be measured, presenting stage coordinates and microhole indexes, etc. (Figure 4.4e–f).
- Navigation over the microhole array chip using the keyboard is implemented. The numpad keys `1`–`9`, except `5`, are used to jump to adjacent microholes. Vertical, horizontal, as well as diagonal movements are supported. *Shift* and *Ctrl* modifiers change the step size, allowing to jump over five or ten microholes at once, as well as to move to a fraction of the interhole distance.
- In case of a runtime error, the software does not crash, but shows a message box with a detailed information about the problem. The user is asked either to *repeat* the last action, to *abort* it or to *exit* the program (Figure 4.4g–h). The software selects the "*repeat*" option itself after a short timeout, thus allowing an unattended operation.

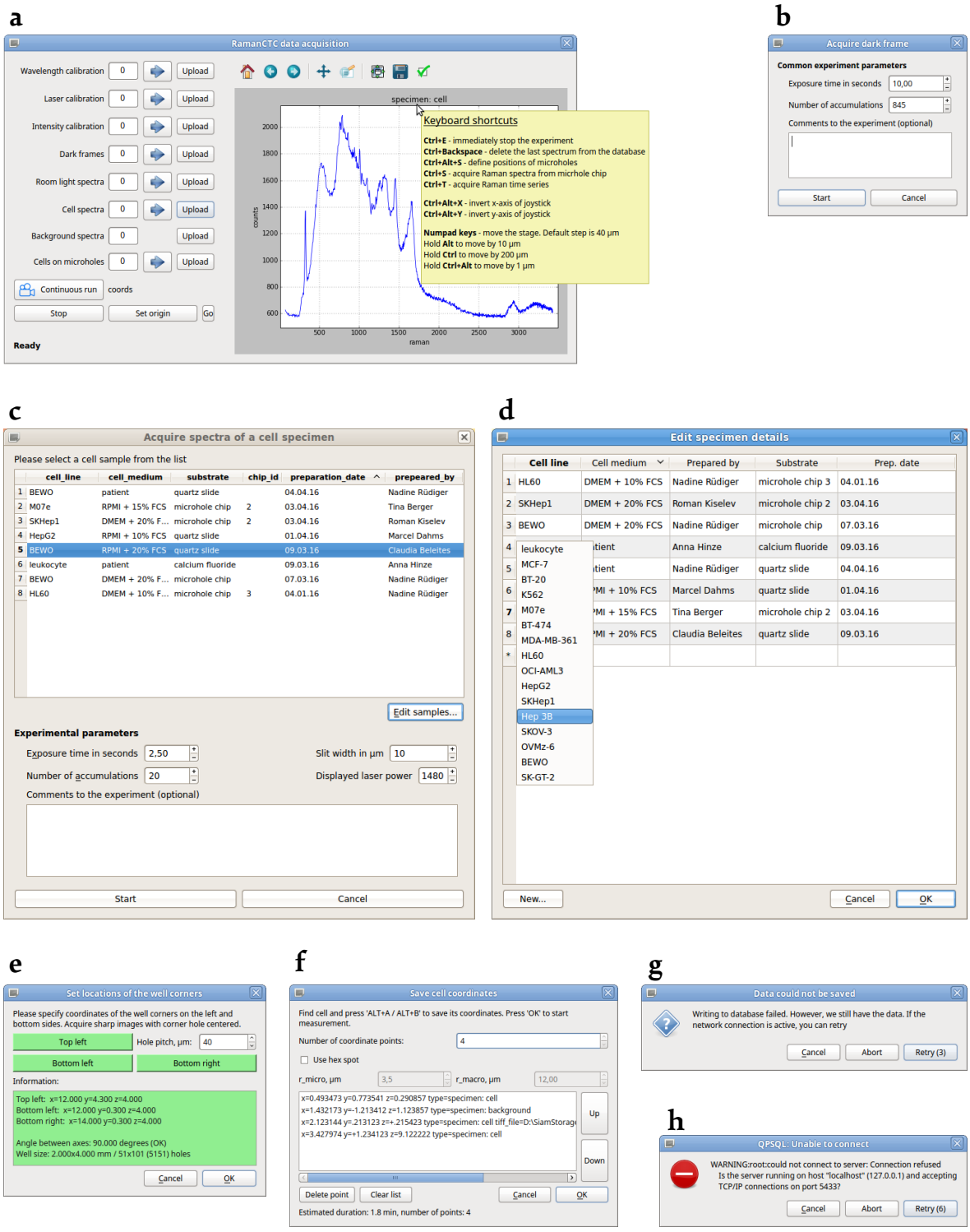


Figure 4.4.: Graphical user interface of the `spc2db` program. **a** – The main window of the program with a tool tip shown. **b** – Experimental parameters for a dark current measurement. **c** – Acquisition of cell spectra: the user can specify instrument settings, select a sample from the database, and provide a comment to the experiment. **d** – Edit available records about cell samples in the database or specify new samples. **e** – Grid specification for the microhole chip. **f** – Dialog with a list of saved coordinates for a later measurement; **g, h** – Dialogs displaying error messages with a timeout. The program automatically repeats the last action, which often helps to resolve the problem (e. g. an interrupted transaction due to a temporary loss of the network connection).

4.3.2. Save/Recall Cell Locations

Typically, 50–70 cell spectra are acquired during an experiment, and the most typical exposure time for an individual cell is 10 seconds with two accumulations, i. e. 20 seconds in total. This pause of 20 seconds, being repeated many times, makes the instrument operator bored and leads to a poor concentration, so that the experimental results could suffer. Therefore it was important to reduce or eliminate this time delay.

For this reason, I implemented the option to preselect the cells before the measurement. After the experimental parameters are specified, the user manually selects several tens of cells that should be measured, and the actual acquisition of Raman spectra follows afterwards. The cells are selected using the joystick to move the motorized stage, and a keystroke "*Alt+A*" to save the current position. Each saved position is listed in the corresponding dialog (see Figure 4.4f). The program estimates the duration of the experiment with the provided settings and the selected cell coordinates. The list elements can be rearranged, or selected with a mouse; in the last case the motorized stage moves to the selected element.

Once the list is complete, the instrument autonomously navigates to each position in the list and acquires a spectrum and a corresponding bright field image of the cell[5]. The acquired data are immediately saved into the database.

4.3.3. Calibration of the Laser Power

The Raman signal is linearly proportional to the laser intensity at the focal spot. Therefore, it is of great interest to track the laser power at the sample during the experiment. The issue is, although both Raman instruments do allow to control the laser power, they do not display it in meaningful units.

The laser controller of the RS785 instrument has a display, where the laser diode current is shown in milliamperes. The power of the laser radiation emitted from a laser diode, however, is a strongly non-linear function of the diode current.

The SIAM software of the RS660 allows to adjust the laser power by setting a value on a 0–100 scale, which essentially controls the rotation angle of the polarizer in the laser beam (Figure 3.7b). Thus, the laser power at the focal spot changes as a $\sin^2(\Theta)$ function, with Θ being the rotation angle of the polarizer.

In order to save the correct values into the database, I had to calibrate the laser power as a function of the displayed value. For each instrument, the laser power at the microscope slide position was measured with an optical power meter (*Thorlabs PM100D* with *S121C* sensor) at several tens of displayed values uniformly distributed across the available range. Then, these values were fitted with a univariate spline. Figure 4.5 shows the results of the fitting. In panel **a**, the diode current is shown on the x-axis, and we clearly see the lasing threshold at approximately 300 mA, followed by a linear range, and a saturation at approximately 950

[5] In case of RS785[6], the bright field image is saved immediately before the acquisition of a spectrum. The RS660[7] instrument has a single sCMOS camera for both modalities, and switching between two modes takes several seconds. Therefore, the image is saved in the moment when the user adds a new element into the list of coordinates, long before the instrument is switched into the spectroscopy mode. This means, that some bright field images could be acquired tens of minutes before the actual Raman measurement.

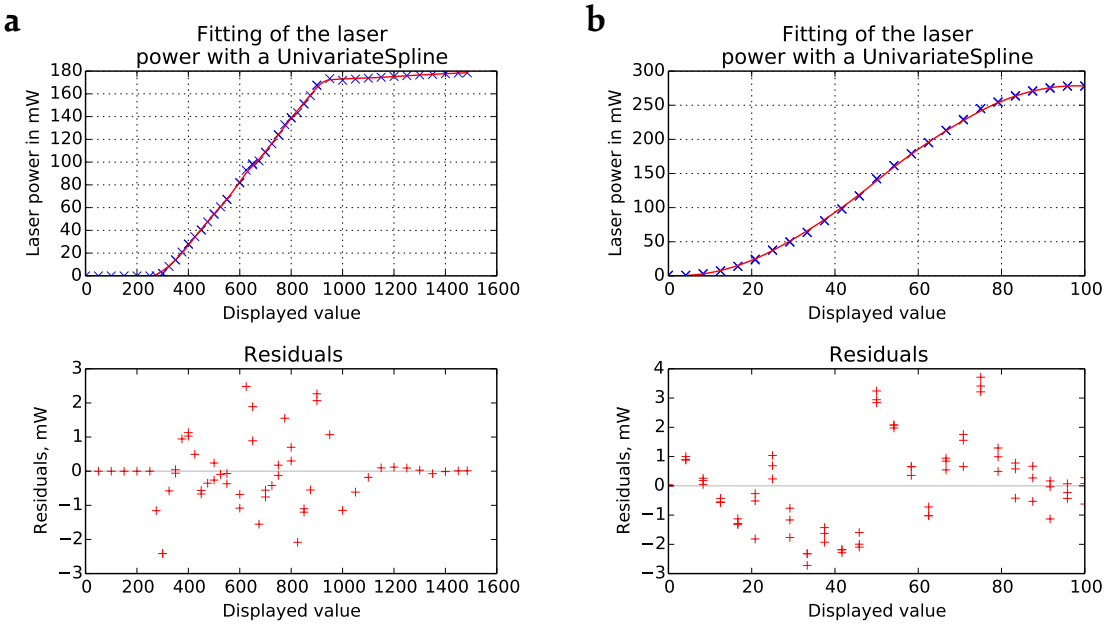


Figure 4.5.: Calibration of the laser power on the microscope slide as a function of the value displayed on the Raman instrument. **a** – RS785 instrument. **b** – RS660 instrument.

mA. In panel **b**, the laser power changes in accordance with the $\sin^2$ law.

The results of the fitting have been integrated into the `spc2db` software. For the RS785 instrument, the user has to manually type in the value of the laser diode current displayed on the laser controller. For the RS660, the user sets the desirable laser power on the 0–100 scale. The `spc2db` software calculates the corresponding real laser power on the sample plane and saves it into the database.

4.3.4. RS785 specific features

The RS785 instrument, named "Roman Raman" in the database, has two components that are managed from the `spc2db` software. These are the motorized stage *Prior OptiScan II* controlled over the `RS232` interface, and the CCD camera *Princeton Instruments PIXIS 256E* that is handled by the `WinSpec` software.

Control of PIXIS Camera over `WinSpec`. I implemented the control of PIXIS Camera using `VBScript` files that are passed to `cscript.exe` (Windows Script Host). These scripts call `WinX/32 Automation` library for `Visual Basic` and instruct `WinSpec` to configure the CCD camera, acquire, display and save the data. The scripts are generated using templates, in which the `spc2db` program substitutes several values, such as exposure time, number of accumulations, shutter operating mode, filename, etc. A care is taken to acquire spectra only when the detector temperature is below $-70°C$.

Since WinSpec saves the spectral data in the SPE format only, it was necessary to implement a function that imports this file format into R. This is described in Appendix on Page 126.

Control of the motorized stage is done using the Python module serial. I implemented a high-level application driver that provides a class named Stage for the hardware control. To connect to the device, the name of the *COM-port* must be provided. The attributes x and y of the class instance correspond to the current stage coordinates; their setting leads to the immediate stage movement:

```
from stage import PriorStage

port = "/dev/ttyS0"  # in Windows use e.g. "COM1"

with PriorStage(port) as stage:
    print stage.x    # get current x-coordinate
    stage.x = 1312   # move to the absolute x-coordinate
    stage.y += 473   # relative move along the y-axis by 0.473 mm
```

The stage driver also provides additional functions that allow to control the movement, such as goto(), rel_move(), is_moving(), stop(), wait_move(). Functions get_position(), set_origin(), invert_x(), invert_y() are used to manipulate the coordinate axes. Finally, it is possible to send/receive custom commands directly to the stage using the send() and recv() functions.

4.3.5. RS660 specific features

The RS660 system is registered in the database under the name "Raman Reader". An embedded computer controls all hardware components of the RS660 instrument, such as the laser power, the operating mode, the sCMOS camera, the motorized stage, the objective lens position, etc. The embedded system communicates over a proprietary protocol with the SIAM software running on the main computer. The SIAM offers a graphical user interface to navigate over the sample, collect mosaic bright-field images or single-point Raman spectra.

Control of the SIAM software is done via an XML protocol – a file that fully defines an experiment, including detector settings, stage position, laser power, acquisition mode and exposure time, etc. The spc2db software generates an XML file from one of the two templates (for Raman experiment and bright-field imaging, correspondingly) and sends this file to a network port on which the SIAM is listening. Then, the SIAM performs the experiment. The SIAM software does not return any results, but only the error code. The measurement result is instead automatically saved into the Siam Storage folder, where it can be fetched from after the experiment is done.

File system monitoring is required to find the results saved by the SIAM software. All acquired experimental data are automatically dumped onto the hard drive in the folder called Siam Storage after **each** action. This also includes intermediate temporary results, which

make the major part of all files. The file path always contains an ordinal number, the date, and a randomly-generated UUID[8].

I implemented a Python module `fs_funcs` that takes care of the file system monitoring. After the XML protocol was sent, the content of the `Siam Storage` folder is checked 10 times per second to detect any new files. When new files appear, their names are parsed using regular expressions to check for the file type. Then, if an appeared file has the correct type (e.g. `spc` file for Raman measurement), the `spc2db` waits until the file size reaches the expected value, meaning that the `SIAM` finished writing to it. Then, the file content can be read and saved into the database.

Additionally, the `fs_funcs` module is used to find the bright field-images and to extract the coordinates from their metadata. This is necessary, because SIAM does not provide any API[9] to get the stage coordinates.

4.4. Algorithm for Automatic Wavelength Calibration

4.4.1. Basics of Wavelength Calibration for Dispersive Spectrometers

A spectrometer disperses the incoming light and typically registers it with an array photodetector, such as CCD or CMOS [76]. A camera pixel records the signal in a small wavelength range. A bit simplified, one can say that each pixel p_i ($p \in \mathbb{N}$) detects the signal at the wavelength λ_i.

A spectroscopist is, however, interested in the wavelength values λ_i, not in the pixel number p_i that are returned by the camera. To do this conversion, one has to calibrate the spectrometer, i. e. to ensure that the $\lambda(p)$ function is known and reasonably correct for each possible value of p.

However, for Raman spectroscopy this is not enough. The units of the x-axis are cm^{-1}. On one hand, they correspond to the energy of the molecular vibration. On the other hand, the detected wavelength λ represents the energy difference between the excitation laser and the molecular vibration, thus the wavenumber is calculated as:

$$\tilde{\nu}\,[\mathrm{cm}^{-1}] = 10^7 \left(\frac{1}{\lambda_0} - \frac{1}{\lambda(p)} \right), \tag{4.1}$$

where λ_0 is the wavelength of the excitation laser in nm. Because the excitation laser wavelength fluctuates, the exact values of λ_0 is not known. Thus, there are two unknown terms that have to be found during the calibration: (a) the $\lambda(p)$ function, and (b) the laser wavelength. Both require two independent calibrations that are based on the measurement of specific chemical substances that are used as a universal calibration standard.

The **wavelength calibration** $\lambda(p)$ requires a measurement of a light source with a precisely known emission spectrum. Gas discharge lamps are ideal sources for such application, because their emission contains sharp spectral lines resulting from electronic transitions. The precise

[8] *Universally unique identifier*, like `42d4d72e-2316-409f-bde1-4744880dbdd6`.
[9] *Application Programming Interface*

values of the emission lines are well known (for example in the NIST[10] AES[11] online database) and can be matched with the observed spectrum to calibrate the corresponding pixels of the array detector. The calibration for the rest of the pixels is interpolated from these values, typically with a polynomial of the 4-th order.

Next, the **laser wavelength** λ_0 has to be calculated. To do this, a Raman spectrum of one of the common standard substances is acquired and matched with the reference from the literature. The most common standard substances are polystyrene, paracetamol (acetaminophen) and cyclohexane [72].

More technical details about calibration of dispersive spectrometers can be found in the article of Tedesco & Davis [108]. There are several issues that make a fully automatic calibration challenging:

1. The function $\lambda(p)$ is not linear, i. e. the measured spectrum is stretched. This is a serious problem for simple signal matching techniques such as cross-correlation.
2. If no calibration is available and the wavelength range of the instrument is not known, it is extremely hard to match the observed spectrum with the reference emission lines.
3. The number of lines and their shape strongly depend on the spectral resolution of the instrument. Many adjacent lines in AES cannot be resolved by common spectroscopic instruments. Thus, it is not correct to claim that an observed peak corresponds to a particular line, if the line is accompanied by other emission lines.
4. The detected spectrum contains a number of weak peaks that can be below the noise level. The peak detection algorithm should nevertheless correctly identify peaks and be immune to the noise.

4.4.2. Peak Detection Using Morphological Operations

The peaks are local maxima in a spectrum. However, simple detection of local maxima would not give correct results, as the experimental data always contain noise.

A very efficient method of noise removal and peak detection offers the so-called *h-dome extraction* described in details in the article of Vincent [109] of morphological grayscale reconstruction. The method, initially proposed for gray-scale images, removes all features from the signal that have a height below the given threshold h, as illustrated in Figure 4.6.

The optimal threshold h can be automatically selected based on the noise level of the spectrum with a safety margin. The noise level is estimated from the spectral regions that do not contain any optical signal, such as silent regions in the Raman spectra. I automatically detect these regions as groups of wavelength channels, whose signal is below the 20-th percentile. The standard deviation of the first derivative of these points is used to estimate the value of h. The derivation removes the constant offset and any slowly varying baseline.

I implemented the h-dome extraction and regional maxima detection for hyperSpec objects [110] based on the source code of the `pymorph` package for morphological image processing in `Python` [111]. The regional maxima are detected in the filtered spectrum after the h-dome extraction, as illustrated in Figure 4.6. An important property of the regional maxima is that

[10] *National Institute for Standartization*
[11] *Atomic Emission Spectra*

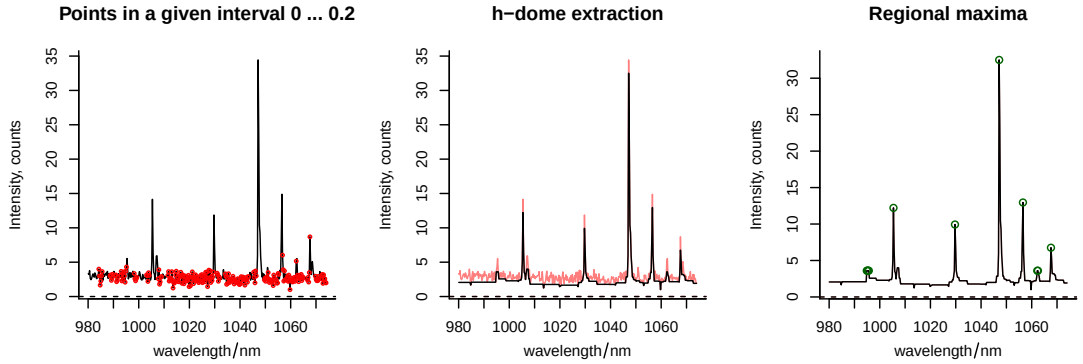


Figure 4.6.: Left – automatic calculation of the noise level in a spectrum. The points whose intensity is below the 20-th percentile and which form reasonably large clusters are used for the calculation of the noise level. In this demonstration a Raman spectrum of a Ne-Ar lamp from the RS660 instrument was used, the estimated noise level is 0.97 counts. **Center** – h-dome extraction applied to the spectrum effectively removes the noise. The red line represents the original spectrum, the black line the filtered one. **Right** – regional maxima detected after the h-dome extraction.

if several local maxima are located next to each other, then only the highest one is selected [109].

This methodology, being applied to AES of a NeAr lamp, measured with the RS785 instrument (1024 data points, spectral range 790–1074 nm), can automatically detect up to ~45 individual peaks. The procedure is relatively immune to the variation in the SNR[12], i. e. it detects lower number of peaks when the spectrum is noisy, but random signal variations are not recognized as peaks. The center of mass is used as the estimated value of the peak position.

The very same feature extraction procedure is applied to the spectra of Raman calibration substance (e. g. paracetamol) to detect peaks for the calculation of laser wavelength λ_0.

4.4.3. Matching of Peaks with Atomic Emission Lines

Once we can match the peaks from an observed spectrum with the specific atomic emission lines, and come up with a regression models, the spectrometer is calibrated. NIST provides an online database of atomic spectra [112]:

> "The Atomic Spectra Database (ASD) contains data for radiative transitions and energy levels in atoms and atomic ions. Data are included for observed transitions and energy levels of most of the known chemical elements."
>
> – NIST Atomic Spectra Database Contents [112].

[12] *Signal-to-noise ratio*

It is possible to programmatically query the ASD to get a list of radiative transitions for a particular substance in a given wavelength range[13]. This means, that **any** gas discharge lamp can be used for the calibration, one only has to know what substance is inside the lamp.

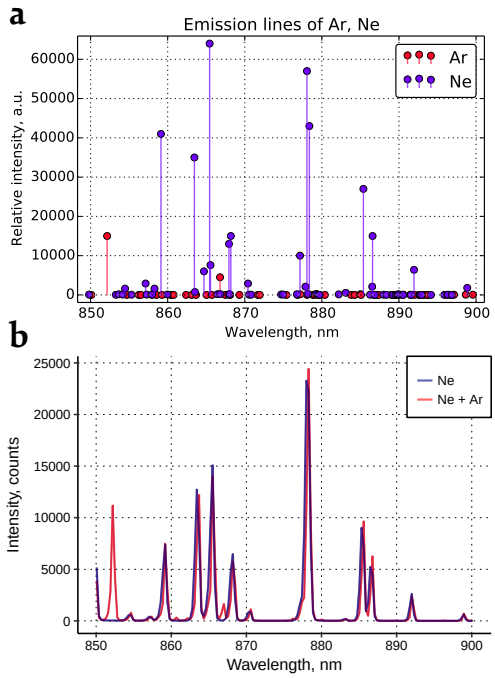


Figure 4.7.: **a** – Spectral lines of Ne and Ar from the reference database. **b** – Measured spectra of Ne and Ne/Ar mixture, acquired with the RS785 instrument.

Often the emission spectrum of a given substance contains a number of closely-located lines, as in case of Ne, shown in Figure 4.7a. It is easy to see, that Ne has a number of lines at ~878 nm, which are so close to each other that they cannot be resolved by our spectrometer. This leads to a *serious problem* – how can we match an observed peak, which is actually a result of convolution of the Ne emission lines with the PSF[14] of the spectrometer, to a particular Ne emission line? Not only the number of observed peaks is lower than the number of actual spectral lines, but also the peak center of mass does not necessarily correspond to any individual spectral line, because multiple lines determine the peak shape together (see Figure 4.7b).

This lead me to the idea, that not the ASD lines themselves should be matched to the measured peaks, but instead they can be used to artificially synthesize the emission spectrum that our instrument is supposed to detect. Then, this artificial "ideal" spectrum can be matched against the observed one to find any deviations in the instrument calibration.

But how can we create this artificial spectrum? To answer this question, we have to understand that the signal on the detector is a superposition of many images of the input slit (or pinhole). The shape of each image is determined by the PSF of the optical system of the spectrograph, and the position is a function of the wavelength[15]. Therefore, the spectrum can be found as the convolution of the ASD emission lines with the PSF. The PSF can be estimated with Airy, Gaussian or Lorentzian function. We used Gaussian function which gave quite descent results. The width of the PSF can be estimated from the geometry of the spectrograph and its slit width. Alternatively, the PSF can be measured by selecting a set of symmetric peaks from the observed AES and calculating an "average" peak, which was done in this work.

Once the PSF is convolved with the emission lines, the observed spectrum is shifted along the wavelength axis until it matches with the simulated one. This is done using the cross-

[13] This is done over the HTTP protocol by creating and GETting a special URL string, which contains the particular arguments to the database. The procedure is documented on the NIST ASD website.

[14] *Point Spread Function*

[15] The shape of the PSF can also slightly vary with the wavelength, but we neglect this effect here

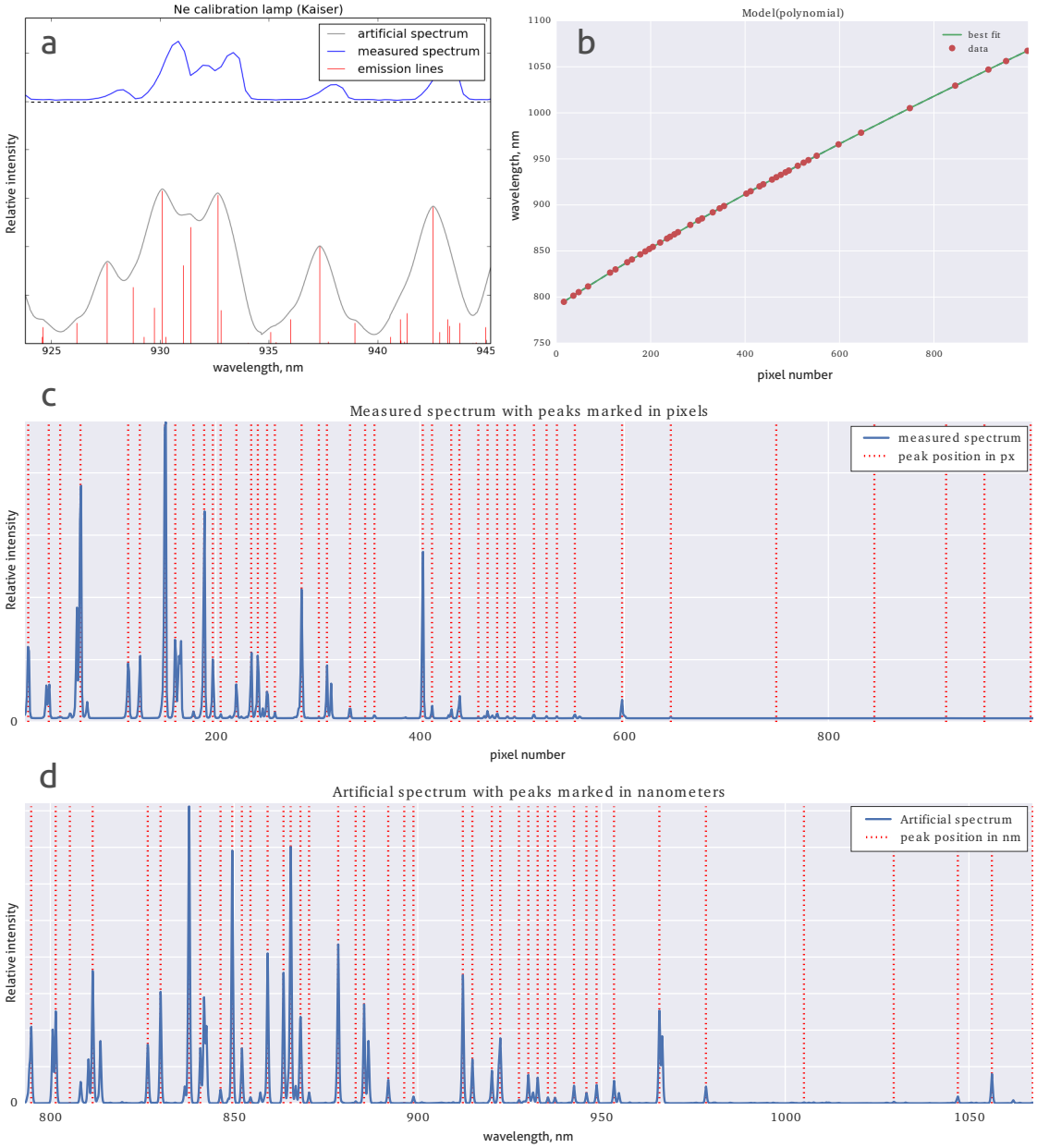


Figure 4.8.: Illustration of the wavelength calibration algorithm. **a** – Convolution of Ne emission lines from the NIST atomic spectra database [112] with a Gaussian that approximates the PSF of the spectrograph results in an artificial spectrum. The measured spectrum of Ne is matched to the expected one. The spectrometer is already coarsely calibrated, so that the measured spectral region more or less corresponds to the spectral range of the emission lines. The relative shift between the spectra is found using the cross-correlation. **b** – Once the peaks from both spectra are detected (see description of the method in the text on page 71 and Figure 4.7) and matched to each other, a polynomial of the 4-th order is used to fit the $\lambda(p)$ function. Most of the fitting residuals are typically below 0.05 nm, some residuals can be as high as 0.2 nm (RS785 instrument). **c, d** – Detected peaks in both spectra.

correlation, which returns the average shift of the signal relative to the reference one. Next, the peaks position in **both** spectra are found using the h-dome extraction, regional maxima detection and calculation of their centers of mass. For each peak in the reference, a corresponding local area in the observed spectrum is analyzed to find the matching peak.

4.4.4. Outlook for Algorithm Improvement

The presented algorithm was developed in `Python` and integrated into the `spc2db` software, but up to date only the RS785 support has been fully implemented and tested. The RS660 instrument has very narrow spectral range, so a lower number of peaks is observed. This requires optimization of some parameters, in particular the PSF model.

The algorithm itself can also be improved. Cross-correlation, which I used to align two spectra, assumes that both signals are shifted with respect of each other, but not stretched or scaled in any way. The developed algorithm works well only if the spectrometer is already coarsely calibrated, i. e. the reference spectrum and the observed one are transformed in a similar manner. The calibration function is, however, non-linear. This means, that the method can adjust the existing calibration, but cannot be used with the raw data (e. g. when abscissa is in pixel numbers).

A more sophisticated approach could be based on the parametric time warping (PTW) [113–115]. This method provides a polynomial transformation of the abscissa axis, i. e. the signal is smoothly stretched until it matches the provided reference, and the limited degree of the polynomial function prevents the overfitting. An additional advantage of PTW is that it returns the $\lambda(p)$ function in an explicit way, so the peak detection procedure becomes redundant.

4.5. Database access with `db2spc`

I developed an `R` package called `db2spc` for the database access. The package offers a convenient way to load specific spectra from the database. There are several important features:

- The package works with both remote PostgreSQL and local SQLite databases.
- A small sample database with 147 spectra is included within the package for the testing and learning purposes.
- The package contains a `Python` script that makes a local SQLite copy of the remote PostgreSQL database (see Section 4.2.2 on page 61 for more details).
- The package is based on `dplyr` and `magrittr` `R` packages, so it offers the full data frame subsetting, grouping and filtering power of `dplyr` combined with the convenient piping syntax "`%>%`" of `magrittr`.
- The loaded spectra are converted into a list of `hyperSpec`[16]. objects, and the whole available experimental metadata gets attached to them.
- This is the second version of the package, which was fully refactored to maximize the performance. The database connections are pooled to keep the number of open sessions

[16] `hyperSpec` is `R` package for handling of hyperspectral data [110].

limited. The spectral information is cached locally to reduce the number of database interactions. The requested spectra are retrieved from the database all together as a single big data frame and further processed locally.
- The package was developed using the "*test-driven development*" paradigm, i. e. the package contains a set of unit tests that check the correct performance of all functions defined in the package. This guarantees the proper functioning.

A short usage example of db2spc is given in the Appendix on page 128.

4.5.1. Automatic Dark Frame Subtraction

The signal from a CCD or CMOS imaging sensor always contains a systematic fixed pattern, consisting of an offset and some fluctuations around it. This pattern is known under the names "black shading", "calibration map", "dark frame", "dark current" etc. The subtraction of the dark frame from each image leads to a considerable image quality improvement, especially under the low-light conditions, as the fixed pattern gets removed. Unfortunately, this pattern depends on the exposure time and the sensor temperature.

To acquire a dark frame, we close the shutter at the spectrometer entrance to keep the sensor in the darkness. At the same time, the image sensor accumulates the charge, which is read out afterwards. We take an average of many measurements to reveal the systematic pattern.

The db2spc package automatically finds the most appropriate dark frame available in the database and subtracts it from the requested spectrum. The dark frame must originate from the same image sensor and (ideally) has the same exposure time as the requested spectrum.

If no dark frames with the requested exposure time e_{spc} are available, then the dark frame is linearly interpolated from the available ones. For this, two dark frames are used:

- one with the highest exposure time below e_{spc}, and
- one with the lowest exposure time above e_{spc}

In case when several dark frames are available, the one with the highest number of accumulations is selected, as it better approximates the fixed pattern. Flowchart 4.9 shows the logic of the db2spc() function, and illustrates how the function selects the most appropriate dark frame spectrum.

The dark frame subtraction improves the SNR of the spectra from the RS660 instrument, especially for the measurements with 10 seconds exposure time.

4.5.2. Use of cell images stored in the database

Furthermore, I implemented an automatic saving of bright field images of cell samples in the database. The idea was that these cell images could later be used to check whether laser was focused on the cell or not, and how the cell looked like during the experiment.

Prior the acquisition of the spectrum, a cell image is captured and saved in the database table spectrum, in the field image. Most of the measurements after August 2016, when the feature was implemented, contain cell images. Each image is saved as a binary block containing a JPEG file. The cell images are currently stored only in the PostgreSQL network database running on

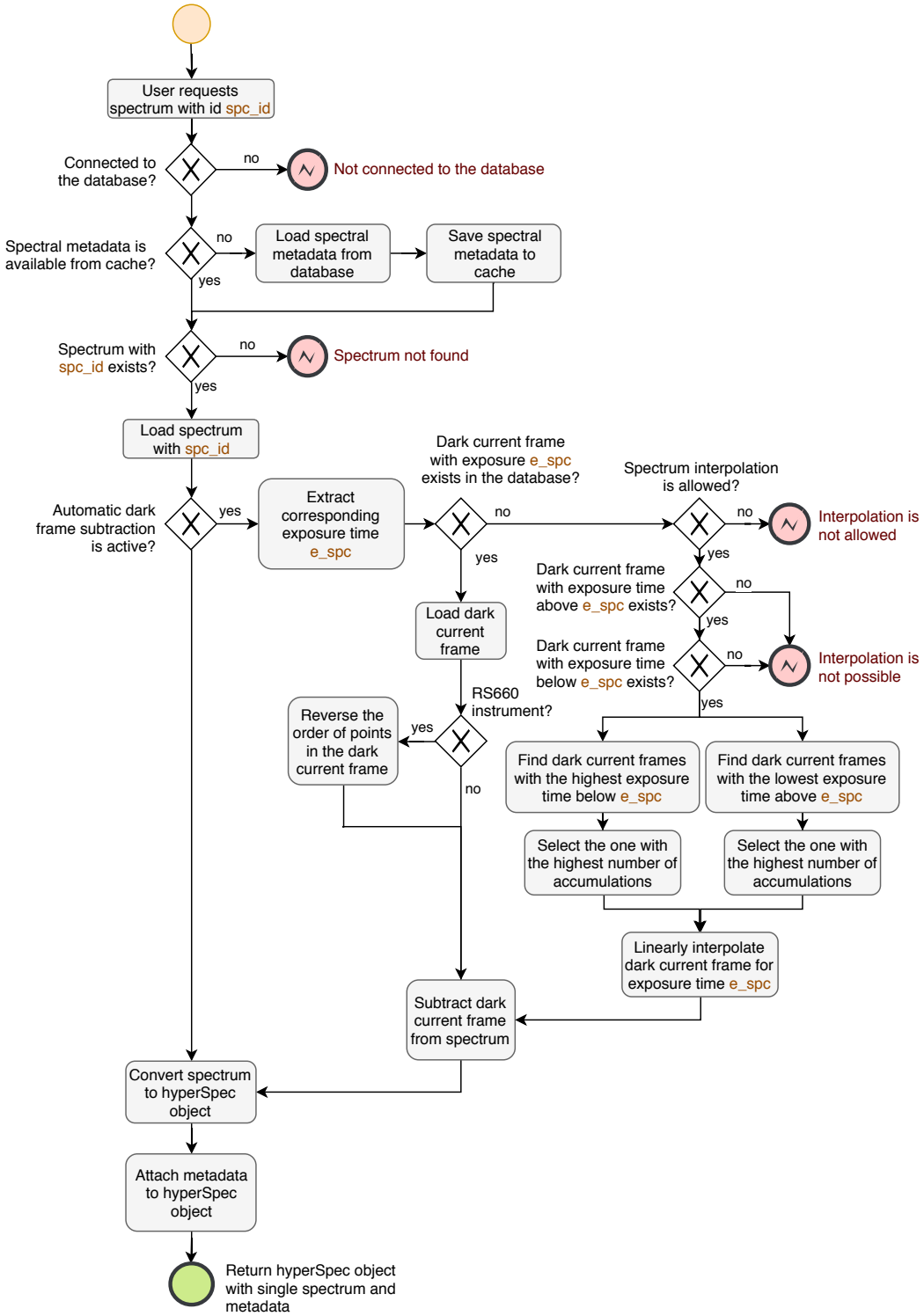


Figure 4.9.: Flow chart that represents how the `db2spc` function loads a single spectrum, including the automatic dark frame subtraction. This is a simplified representation of what happens in reality – `db2spc()` can load many spectra at once, which are processed in parallel to boost the performance.

the server, and are not exported into the SQLite database. This means, one has to connect to the remote database with a call to use_postgre_db(). Appendix F shows 144 images randomly sampled from the database.

The images can be loaded using function db_load_images(), which accepts either a hyperSpec object containing the extra data column id, or directly a vector of id values. The function receives binary blobs with image data, uses R package jpeg[116] to read them, and constructs an image object using R package EBImage [117]. There is also a helper function spc.implot() that plots images of cells corresponding to the provided spectra. The following listing presents an example use of spc.implot(), the result is shown in Figure 4.10.

```
use_postgre_db()
spc.implot(spc[1:8], col=c("red4", "white"))
```

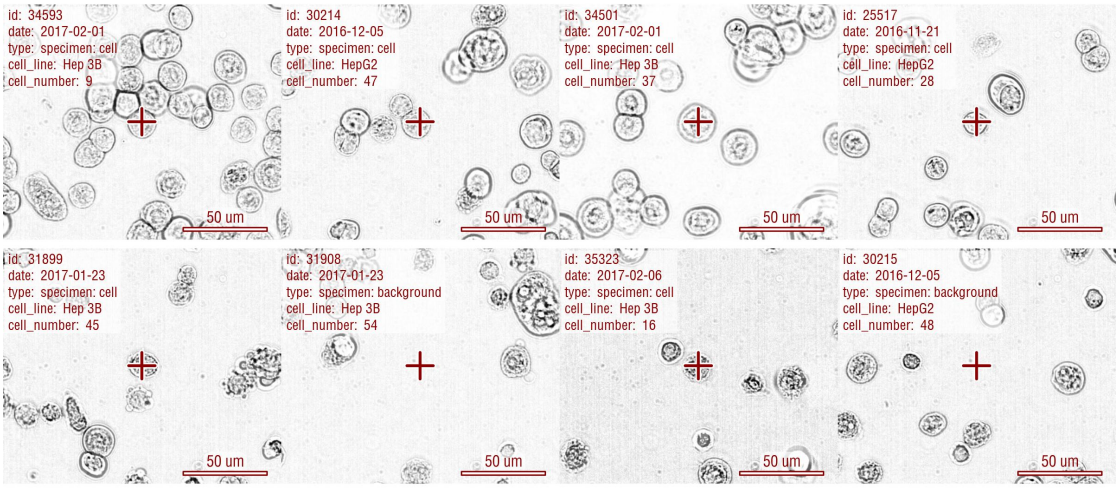


Figure 4.10.: Cell images recorded into the database prior to the acquisition of Raman spectra. These images can be loaded using the function db_load_image() and plotted with the function spc.implot(). Shown here are images corresponding to the first eight spectra from the data set plotted in Figure E.1. A cross in the center of each image marks the position from which the Raman spectrum has been acquired.

The R package EBImage [117] contains a broad number of tools for image processing, such as *otsu* and *adaptive thresholding*, *morphological* operations, *2D convolution* filters, *watershed algorithm*, function for labeling and characterization of segmented images. Using the provided functionality, I developed a simple segmentation algorithm that consists of four important steps (see Figure 4.11):

1. Edge detection,
2. Adaptive thresholding,
3. Restoration of object shapes using morphological operations,
4. Watershed segmentation.

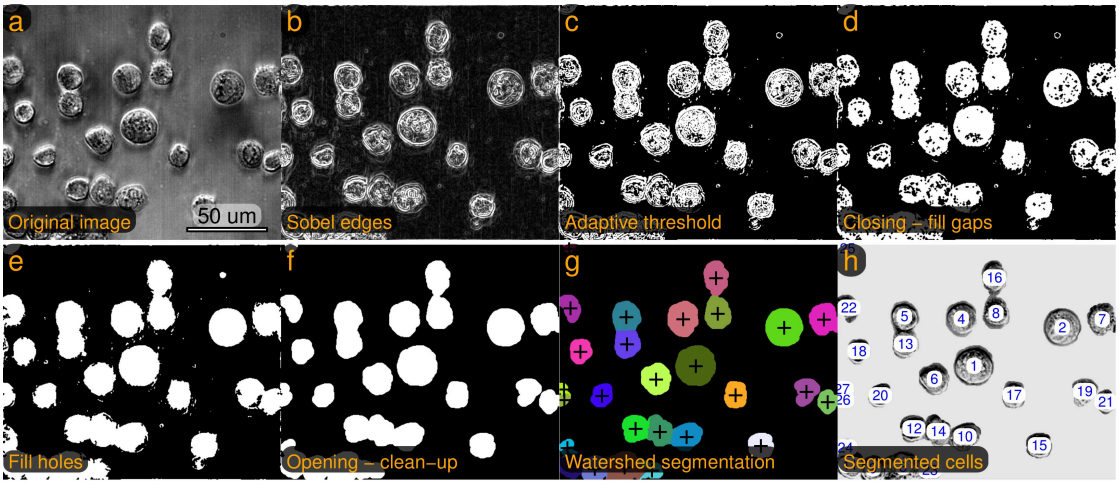


Figure 4.11.: Demonstration of simple cell segmentation algorithm using the functions from R package EBImage [117]. **a** – Original unprocessed image. **b** – Slightly blurred image is convolved with the Sobel operator for the edge detection. **c** – Adaptive threshold returns a binary mask of **b**. **d** – Morphological closing connects adjacent white areas. **e** – Closed regions are filled. **f** – Morphological opening removes small objects and sharp features. **g** – Watershed algorithm is applied to a distance map of the binary mask. It finds ridges that split the mask into individual regions, thus a center of each cell can be calculated. **h** – Segmented and enumerated cells.

Edge detection Bright field microscopy images of unstained cells vary substantially in terms of contrast, brightness, number of objects, their shape and size. However, there is one common feature: the background is more or less uniform, while the cells feature high intensity variations. This allows us to find objects of interest by looking at the intensity gradient.

The intensity gradient is calculated using the Sobel operator, which results in a new image containing emphasized edges. The horizontal and vertical Sobel operators G_x and G_y are defined as

$$G_x = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} +1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix}, \quad \text{and} \quad G_y = G_x^T \tag{4.2}$$

The Sobel operator combines averaging and differentiation operations, which are defined by the first 3×1 and the second 1×3 matrices in 4.2, correspondingly.

Being convolved with an image, each of the G_x and G_y operators produces intensity gradient of the original image along the x or y axis, correspondingly. A rotationally invariant edge detection function $G(\hat{I})$ based on G_x and G_y operators is defined as

$$G(\hat{I}) = \sqrt{(\hat{I} \otimes G_x)^2 + (\hat{I} \otimes G_y)^2}, \tag{4.3}$$

and the corresponding R code is

```
sobel <- function(img, sigma){
  mtx <- matrix(c(1, 2, 1)) %*% matrix(c(1:-1), 1)
```

```
  sqrt( filter2(img, mtx)^2 + filter2(img, t(mtx))^2 )
}
```

Although the Sobel operator intrinsically averages the image, sometimes it can be beneficial to additionally blur the image prior computing the gradient. This helps to ignore too small features that are caused by noise. We do this by convolving the image with a 2D Gaussian, however one could alternatively replace the first matrix in the Sobel operator 4.2 by a bigger Gaussian kernel and do both blurring and edge detection in a single step. The results of the edge detection are presented in Figure 4.11b.

Adaptive thresholding

An adaptive threshold is a powerful method to create a binary mask of an unevenly bright image. It convolves the image with a rectangular window, adds a defined offset, and compares the result with the original image, as shown in Figure 4.11c.

$$\hat{T} = \hat{I} > \hat{I} \otimes \hat{J}_n + y, \tag{4.4}$$

where $\hat{I}$ is the image, y is the constant offset, and $\hat{J}_n$ is a matrix of ones normalized by the sum of its elements, i. e.

$$\hat{J}_n = \frac{1}{n^2} \begin{pmatrix} 1 & 1 & \cdots & 1_{1n} \\ 1 & 1 & \cdots & 1_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 1_{n1} & 1_{n2} & \cdots & 1_{nn} \end{pmatrix} \tag{4.5}$$

The method is implemented in `EBImage` under the name `thresh()`. Applied to the original image, it results in an acceptable thresholding. However, even better result can be achieved by combining it with the edge detection (Figure 4.11c). Here, we use a window $\hat{J}_n$ of size 144×144 (one third of the smallest image dimension), and a small offset of $y = 0.03$. This procedure works well with the majority of the cell images available in the database.

Object shape restoration

The output of the adaptive filter is a binary mask that contains a number of small objects that roughly correspond to the cell area, but contain multiple holes and disconnected regions. In order to find the cells outlines, these regions have to be connected and filled (see Figure 4.11d). This problem is solved using morphological operations, in particular closing. It is a combination of two steps called *dilation* and *erosion*, which grows the segmented regions by m pixels, and then shrink them back by the same number of pixels, as illustrated in Figure 4.12. Morphological operations are based on a convolution with a specific kernel.

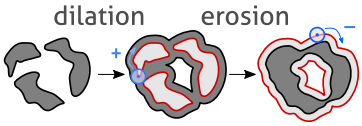


Figure 4.12.: Morphological closing fills the gaps in the mask. The convolution kernel is shown as a blue circle.

When the gaps are closed, the encircled regions can be filled to eliminate the holes in the mask. The resulting image still contains a number of sharp edges and small objects caused by

the noise, which have to be removed (Figure 4.11e). Again, we apply morphological operations, but in the reversed order: first erosion, then dilation. This combination is called *opening*, and it straightens the border shapes, as well as removes small objects that are smaller than the kernel size (see Figure 4.11f).

Watershed segmentation

The result of the previous steps is a binary mask that separates image into the background and the foreground. If the goal is to find and characterize individual objects, then the foreground has to be segmented into individual regions of interest. We calculate a distance map of the binary mask and apply the watershed algorithm, which inverts the distance map and represents the foreground areas as "valleys" that get flooded with "water". The watershed ridge, at which "waters" from two adjacent "basins" meet, is used as a line that separates individual objects. The result of the watershed segmentation is a mask with labeled objects, as shown in Figure 4.11g–h.

The described approach of image segmentation is carried out using the `R` code partially shown below and, being applied on image #35591, results in Figure 4.11:

```
img                                        %>%  plt("a", "Original image", sb=T)

# Sobel edges
edges <- blur(img, 0.9) %>% sobel          %T>% plt("b", "Sobel edges")

# Adaptive thresholding with rectangular window
mask <- thresh(edges, n, n, offset)        %T>% plt("c", "Adaptive threshold")

# Object shape restoration
mask %<>% closing(makeBrush(3, "diamond")) %T>% plt("d", "Closing - fill gaps")
mask %<>% fillHull()                       %T>% plt("e", "Fill holes")
mask %<>% opening(makeBrush(19, 'disc'))   %T>% plt("f", "Opening - clean-up")

# Watershed segmentation
labels <- mask %>% distmap %>% watershed          # "g", "Watershed"
```

There are more than 19,000 cell images in the database, which can be used as an additional information source that complements Raman spectra. For example, it is known that tumor cells are on average bigger in size than leukocytes, so this information can be utilized to support Raman-based cell identification methods. The segmented images can also be used to automatically check whether the laser was focused on the cell during the measurement (assuming that the laser focus was always in the middle of the bright field image).

4.5.3. Interactive Database Viewer

I developed an interactive viewer using `Python` programming language, `Qt` GUI library and a `matplotlib` plotting library. The application is used to quickly find spectra and display spectra from the database. It accepts an `SQL` *`SELECT`* query which is used to filter the data using any of

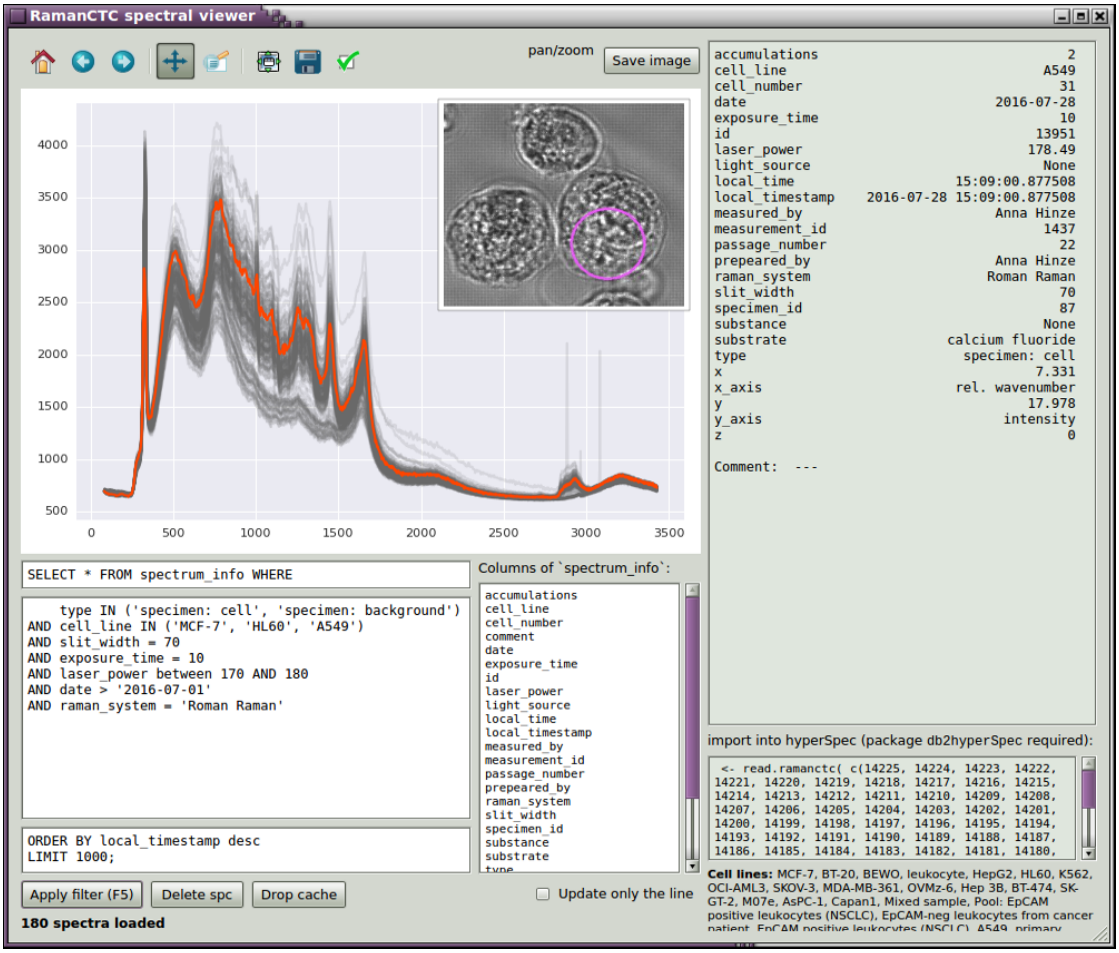


Figure 4.13.: Screenshots of the interactive database viewer, which displays RS785 spectra alongside with the bright field images and experimental metadata.

the numerous data fields, shown in the list called "columns of spectrum_info" (see screenshots in Figure 4.13).

The program loads the corresponding spectra and displays them alongside with the bright field images. The right part of the program window lists the metadata of a particular spectrum. Arrow keys or `F1`/`F2` keys can be used to select the next/previous spectrum in the dataset – this updates the plot, the bright field image and the data window. The program can save the currently plotted spectra or bright field images into a graphic file on the computer.

4.6. Improved Data Management with the Spectra Database

The `PostgreSQL` database that I designed during this work (see Section 4.2 for details) substantially enhanced the organization of the experimental data. Constraints applied onto the fields of the database table ensure that no meaningless records can be saved into the database.

The mechanism of transactions ensures data integrity, as only a whole finished measurement can be saved into the database as single atomic entity.

The database was supplemented by the `spc2db` software running on the lab computer, which assists in the data acquisition. This software automates the experiment by allowing the user to preselect cells for the subsequent automatic measurement. Moreover, it eliminates a lot of issues associated with the data handling, such as the necessity to create meaningful file names, as the experimental results get immediately saved in formally-organized way into the remote database. Finally, the stored data sets can be easily retrieved, visualized, and imported into `hyperSpec` via `db2spc` package for `R`.

Opportunities for Further Enhancements

Despite of the many previously mentioned advantages of the developed data management system, the current database model has some performance bottlenecks. The most prominent of them is the view `spectrum_info` (represented by the tibble[17] returned by the `sm()` function in the `db2spc` package) that is created using a nested `JOIN` statement over a lot of tables. As the number of stored spectra exceeded 10.000, the view generation was taking already several seconds. To correct for this, the view is refreshed once in an hour and stored in a materialized form. The user applications retrieves the data from this materialized view, which, however, does not represent the actual state anymore.

As automated instruments generate more and more data, it becomes challenging to retrieve the them for a subsequent analysis.

Apache Spark is a recently-emerged DBMSs that is probably more suitable for our application – storage of a large number of Raman spectra. `Apache Spark` was designed to handle big data sets and can make efficient parallel calculations directly on server [118]. For Raman spectra, this could be, in the simplest case, several preprocessing steps, such as subtraction of the dark frames, cropping of spectral regions, intensity calibration, calculation of average spectra, etc. In a more sophisticated scenario, one could ask `Apache Spark` not only to select and preprocess a dataset, but also to reduce its dimensionality, extract some features or return only those observations that follow some specific patterns.

`Apache Spark` has bindings to many programming languages, among which are `Python` (module `pyspark`) and `R` (package `sparklyr`). It revolves around the concept of *Resilient Distributed Datasets* [119] and supports distributed machine learning [120]. It can be worth to use `Apache Spark` in place of `PostgreSQL` for applications that work with very large data sets. Raman imaging or high-throughput Raman cell screening are definitely among such applications.

Moreover, the database structure can be optimized in the future. The current database model proved to be redundant, as only several tables are actively used. This happened because initially it was not totally clear what will be recorded during the experiments and what level of granularity is necessary.

[17] *Tibble* is a variant of a data frame in R.

4.7. Overview of the Collected Dataset

The use of the automated acquisition software and data management system allowed us to perform a high number of experiments and collect a big dataset. The total number of spectra stored in the database is 40383. There are measurements of different types, such as Raman spectra of biological specimens, Raman maps, Raman spectra of polystyrene and paracetamol, emission of narrow band gas discharge lamps (pure neon, neon with argon, and mercury with argon), dark frames from the detector, and room light spectra, as shown in Table 4.3.

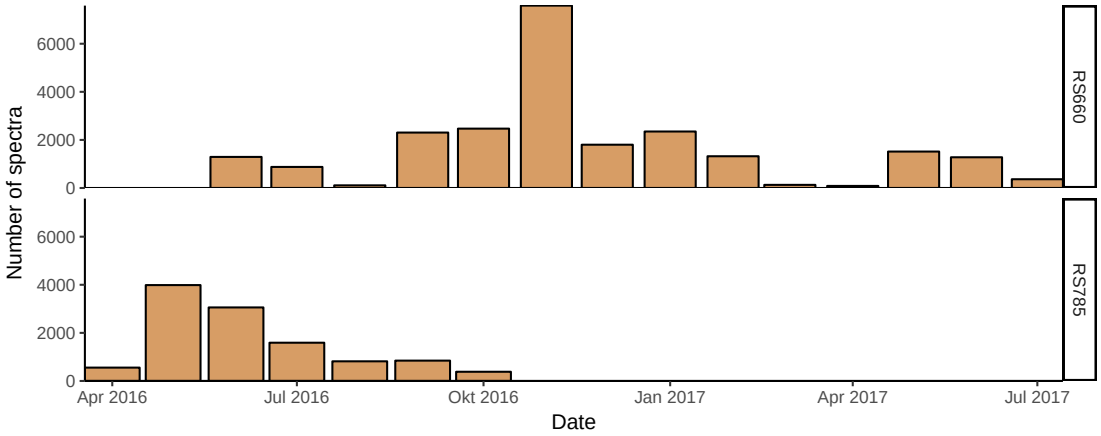


Figure 4.14.: Use of Raman instruments over time for a routine acquisition of cell spectra.

Table 4.3.: Number of spectra in the database, grouped by Raman instrument and measurement type.

Measurement type	RS660	RS785
specimen: cell	19892	9133
raman map	3724	706
specimen: background	3587	2104
dark	19	415
laser calibration	3	676
wavelength calibration	2	111
intensity calibration		5
room light		6

Figure 4.14 shows the distribution of our experimental activity over the time frame of more than one year. The majority of the cell spectra were collected with the exposure time of either 10 seconds or 1.0 second, usually using two accumulations per each spectrum. There are, however, several other exposure times that we used throughout the experiments, listed in Table 4.4.

Table 4.4.: Number of Raman spectra of cells stored in the database, grouped by Raman instrument and exposure time.

Exposure time	RS660	RS785
0.25		66
0.50	196	29
1.00	5756	4099
2.00	1732	7
2.50	2433	
5.00	1596	842
10.00	11711	5947
15.00		174
30.00		73
50.00	55	

Table 4.5.: Number of Raman spectra in the database, grouped by Raman instrument and cell preparation protocol. Effort was taken to develop a single protocol compatible with different cell lines [121]. **Notes:** 2 cell lines from **healthy volunteer** are: "*leukocyte*" and "*Mixed sample*".

Preparation protocol	# Cell lines	RS660	RS785
RPMI + 15% FCS; trypsin; unfixed	16	12826	4880
patient	7	5322	180
RPMI + 15% FCS; untrypsinised; unfixed	7	2683	2944
RPMI + 15% FCS; trypsin; fixed 4% formalin 15 min	3		249
healthy volunteer	2	6119	2539
DMEM + 10% FCS; trypsin; fixed 4% formalin 15 min	2		323
RPMI + 10% FCS; trypsin; fixed 4% formalin 15 min	2		327
not in the list	1	225	
Cell with nanoparticles inside	1	9	
DMEM + 10% FCS; trypsin; unfixed	1		30
RPMI + 10% FCS; trypsin; unfixed	1		471

In total, Raman spectra of 20+ cancer cell lines were acquired, as well as several thousand spectra of healthy leukocytes, fibroblasts and cells from mixed samples (i. e. cell suspensions containing a mixture of healthy and tumor cells without any labeling). The details on the used cell preparation protocols are presented in Table 4.5. The amount of available data per cell line, alongside with cancer type, is summarized in Table 4.6.

Table 4.6.: Number of Raman spectra in the database, grouped by Raman instrument and cell line.

#	Cell line	Cancer type	RS660	RS785
1	BT-20	breast cancer	1762	519
2	BT-474	breast cancer	360	361
3	MCF-7	breast cancer	3300	407
4	MDA-MB-361	breast cancer	480	437
5	BEWO	choriocarcinoma	602	394
6	Hep 3B	hepatocellular carcinoma	746	315
7	HepG2	hepatocellular carcinoma	696	471
8	primary MNCs HCC	hepatocellular carcinoma	1750	
9	103H	large lung carcinoma	1430	119
10	HL60	leukemia	990	684
11	K562	leukemia	360	992
12	M07e	leukemia	362	560
13	OCI-AML3	leukemia	752	614
14	A549	lung cancer	1413	360
15	EpCAM positive leukocytes (NSCLC)	non-small cell lung cancer		4
16	Pool: EpCAM positive leukocytes (NSCLC)	non-small cell lung cancer		74
17	primary MNCs NSCLC	non-small cell lung cancer	3572	
18	primary MNCs NSCLC ex-vivo	non-small cell lung cancer		180
19	SK-GT-2	oesophageal cancer	360	358
20	OVMz-6	ovarian cancer	361	424
21	SKOV-3	ovarian cancer	360	376
22	AsPC-1	pancreatic cancer		699
23	Capan1	pancreatic cancer		134
24	EpCAM-neg leukocytes from cancer patient	–		42
25	HBMEC	–	9	
26	leukocyte	–	3985	2539
27	Mixed sample	–	2557	520
28	primary foreskin fibroblast	–	977	360

Moreover, Raman spectra of about 5.3 thousand cells from 48 individual patients were routinely acquired during the spring and summer of 2017. The majority of these cells were primary MNCs NSCLC (3572 spectra) and primary MNCs HCC (1750 spectra) measured on RS660 instrument, as illustrated by Table 4.7.

The statistics presented here are just a small extraction from a large number of possible data groupings. Since a lot of experimental parameters were tracked, the data can be split into groups using any other criterion that is associated with the spectra.

Table 4.7.: Patient samples. Number of Raman spectra stored in the database, which were acquired from cells obtained from blood of cancer patients, grouped by Raman instrument. In total, blood samples from 48 individual patients were investigated.

Cell line	RS660	RS785
primary MNCs NSCLC	3572	
primary MNCs HCC	1750	
Pool: EpCAM positive leukocytes (NSCLC)		74
Mixed sample		60
EpCAM-neg leukocytes from cancer patient		42
EpCAM positive leukocytes (NSCLC)		4

This enables us to use these data to get answers to many questions that were hard to address before. For example, ANOVA–simultaneous component analysis (ASCA), could shed the light onto the origins of different variances (confounders) that affect classification models [122].

Finally, availability of a high number of statistically independent samples (20+ cell lines) would increase the reliability of two-class classifiers that distinguish between cancer cells and healthy leukocytes in blood samples [123].

5. Data analysis

This chapter describes the analysis of the collected data. I present a new algorithm for detection of outliers in the spectral data, outline several clustering methods and their applications to the spectroscopic data. Furthermore, I present methods of supervised machine learning for automatic cell identification alongside with the validation procedures for classification models. Finally, I demonstrate to what extent classification model can identify tumor cells in the collected data and discuss the results.

5.1. Detection of Outliers in Raman Spectra

An outlier is an observation point that is distant from all other observations. An outlier can occur by chance in any distribution due to variability of the random variable, but more often, especially in a statistically big population, it indicates an experimental error. In case of Raman spectroscopy, the typical reasons for outliers are spikes, sample impurities, or wrong positioning of the sample. The last two reasons corrupt the whole spectrum and often make it unusable. It is important to exclude outlier spectra from the training dataset, unless these outliers can be explained by some inherent variances of the samples, and not by coarse measurement errors.

Spikes

From time to time some pixel of the detector and, to a lesser extent, its direct neighbors, can yield an unreasonably high read out. The reason for this are *cosmic rays*, which are high-energy radiation, mainly originating outside the Solar System. Cosmic rays interact with the Earth's atmosphere and produce a number of secondary particles that sometimes reach the surface and, among other effects, influence functionality of microelectronics.

Upon hitting the detector, a cosmic ray generates a number of electrons in it. These electrons drastically increase the read-out signal of the detector at a given pixel. The spikes appear as narrow intense peaks in the acquired spectra, as shown in Figure 5.1. Their occurrence, intensity and positions are random events.

One of the algorithms for spike detection has been proposed and implemented in `R` by Ryabchykov *et al.* [124], who convolves the spectrum with a Mexican hat wavelet. The width of the wavelet corresponds to the width of spikes, which are much more narrow than the Raman signatures. This operation results in a response that strongly enhances narrow spikes and suppresses all other spectral features, which eventually helps to identify and remove the spikes.

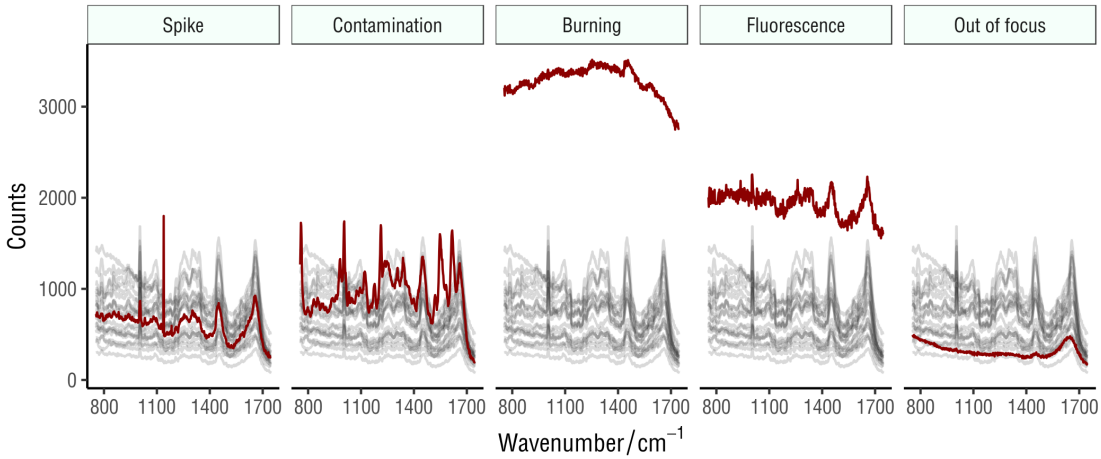


Figure 5.1.: Illustration of different outlier types in Raman spectroscopy. Grey lines represent normal spectra of cancer cells or leukocytes. 114 random spectra were pooled from the database from all registered cell experiments. All spectra were acquired with 10 s exposure time on the RS660 Raman instrument. *Note:* The spectrum of a burning sample is downscaled 4 times for the sake of visibility.

Contamination

Biological samples often contain some small undesired particles, like bacteria, which, due to their different chemical composition, give additional peaks in Raman spectra, as shown in Figure 5.1. These spurious peaks confuse classifiers. The problem is worsened by the fact, that the small objects may get trapped by the excitation laser and move along with it from cell to cell.

Burning

This problems is a direct consequence of the sample contamination. If a contaminating particle has high absorption cross-section, it will rapidly heat up and starts to burn, emitting bright white light with a broad spectral range. As a result, the sample gets damaged, and the spectrum gets a huge vertical offset, often leading to a camera saturation (see Figure 5.1).

Fluorescence

Fluorescence is another source of a broad smooth background, that leads to the overall noise increase (see Figures 5.1 and 3.2). This issue is discussed in more detail in Section 3.1.2.

Out of Focus

Since the excitation laser spot is small and is conjugated with the core of the detection fiber, the Raman instrument is highly confocal. On one hand, this helps to eliminate the undesired background signals, but on the other hand requires a precise focusing on the sample. If the alignment is poor, or if a cell moves away from the laser focus, then no cell signatures are

detected, but only a background originating from the substrate and surrounding media, as shown in Figure 5.1.

Luckily, such spectra are easily detected. One of the methods is to calculate area under the curve (AUC) for a set of peaks characteristic for the cell spectra. If the peaks are weak or absent, then the sample was not in the laser spot.

5.1.1. New Algorithm for Automatic Detection of Outliers in Spectral Data

The most challenging outliers arise due to contamination, fluorescence, and burning. They lead to unpredictable dramatic changes in the shape of the spectrum. Outlier spectra can seriously affect preprocessing steps that are not based on robust statistical measures.

We define an outlier as a spectrum, which has a substantial number of data points far away from a "typical range". I define a typical range individually for each wavelength as a range in which the majority of the data points are located, i. e. between 5-th and 95-th percentiles, if the majority is defined as 90%.

The idea of the proposed algorithm is to penalize data points that go beyond the typical range (plus a small safety margin). The penalties are calculated independently for each wavelength of the spectrum. At the end, some sort of an "average penalty", which characterizes the spectrum as a whole, has to be determined. Based on the distribution of these average penalties, one can find a threshold that separates "good" spectra from the "bad" ones.

Calculation of Penalties

To illustrate the concept, I take a sample dataset, which contains 114 spectra, 5 of which are outliers shown in Figure 5.1. The spectra were normalized by their area under the curve, and the mean value at each wavelength was subtracted to highlight the variance between the spectra. Then, the 5-th and 95-th percentiles were calculated at each wavelength – they mark the "typical" data range. The result is shown in Figure 5.2a.

The inter-spectral variance is different at each data point. Therefore, the penalty should scale with the variance at a given wavelength, i. e. it has to be normalized. Moreover, it should not depend on the average spectral intensity, otherwise it would be hard to select a threshold. We use the metric called *interquartile range*, or IQR for short, which is the difference between the 3-rd and 1-st quartiles, or 75-th and 25-th percentiles. Since quartiles are robust statistic measures of a random distribution, IQR itself is not sensitive to outliers.

Bringing everything together, I come up with a formula that defines a penalty s_i for a data point x_i at wavelength i as:

$$\begin{cases} q_p - x_i & \text{if } x < q_p \\ 0 & \text{if } q_p \leq x_i \leq q_{1-p} \\ x_i - q_{1-p} & \text{if } x > q_{1-p} \end{cases}, \tag{5.1}$$

where q_p and q_{1-p}, typically $q_{0.05}$ and $q_{0.95}$, are the 5-th and the 95-th percentiles. The distance between a given data point x_i and its nearest quantile is normalized by the IQR of

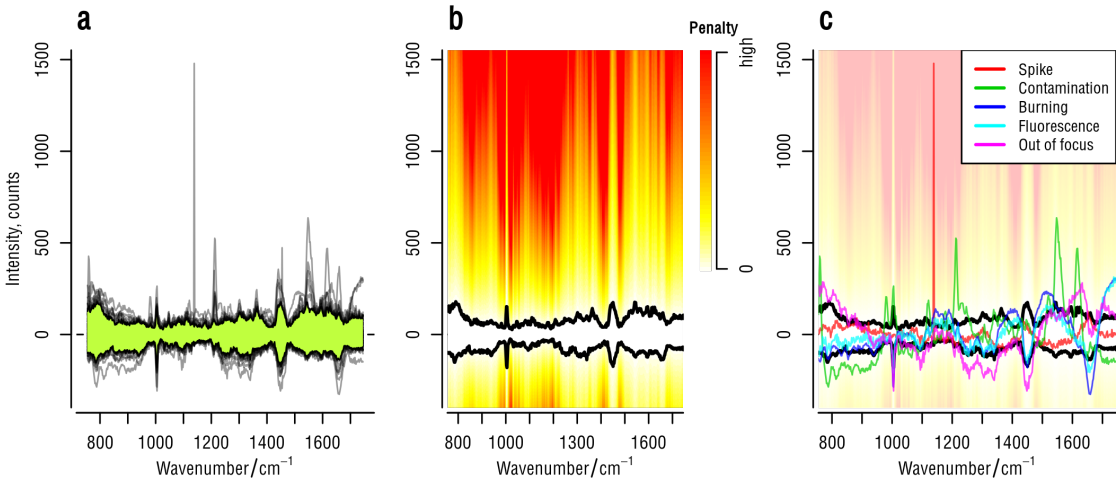


Figure 5.2.: Calculation of penalties for automatic outlier detection. **a** – dataset with 114 spectra after normalization and subtraction of the mean value. Shaded green area shows data points located between percentiles 5 and 95. **b** – points outside the "typical" range get penalized. The penalty increases as the point goes further away, which is shown with the red background. **c** – outlier spectra from Figure 5.1 have high number of penalized data points.

the dataset. This means that the penalty at each wavelength i is normalized by the data spread at the very same wavelength.

Based on s_i, we have to calculate an average penalty $\bar{S}$ for each given spectrum.

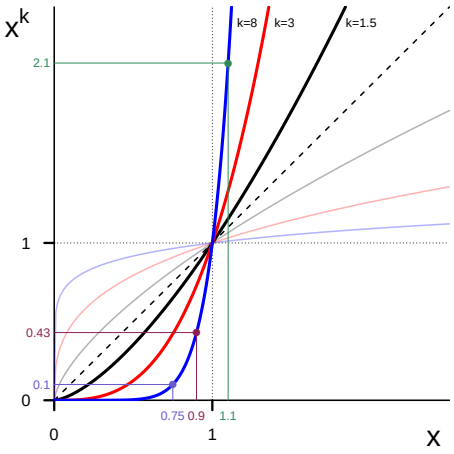


Figure 5.3.: Power function.

5.1.2. Calculation of Average Penalty $\bar{S}$

The simplest, most naive way to calculate an average penalty $\bar{S}$ for a spectrum, is to use a simple *arithmetic mean*:

$$\bar{S} = \frac{1}{N}\sum_{i}^{N} s_i \tag{5.2}$$

This approach has, however, a drawback, which is linearity. Ideally, the penalty should **rapidly** grow when a data point goes beyond the "borderline". That said, a *root mean square* (RMS) could be a better option here:

$$\bar{S} = \sqrt{\frac{1}{N}\sum_{i}^{N} s_i^2} \tag{5.3}$$

With RMS, the penalty rises **quadratically**. We can actually go further and use a *generalized mean*, also known as "power mean", which is defined by the formula

$$\bar{S} = \sqrt[k]{\frac{1}{N}\sum_{i}^{N} s_i{}^{k}}, \tag{5.4}$$

where k can be varied to adjust the behavior of the generalized mean. The arithmetic mean and RMS are just degenerate cases of the generalized mean.

k-value	Meaning
$k = 0$	Geometric mean, i. e. $\sqrt[N]{s_1 s_2 \cdots s_N}$
$k = 1$	Arithmetic mean
$k = 2$	Root mean square
$k = \infty$	Returns the value of the maximum element

For our application, the generalized mean serves as a non-linear moving average which is shifted towards small signal values for small k and emphasizes big signal values for big k, as shown in Figure 5.3.

Meaning of the k Parameter for Outlier Detection

- $0 < k \ll 1$: a spectrum becomes outlier only if it leaves the "typical" range at **many** points. In this case, the shape of the spectrum plays the most important role, while the noise does not contribute much to $\bar{S}$.
- $k \gg 1$: Any single point that has an extremely high or low value would strongly contribute to $\bar{S}$. In this mode, the algorithm is sensitive to spikes.

Implementation of Penalty Calculation

The listing below contains the `R` implementation of the algorithm described above.

```r
library(hyperSpec)

spc_penalties <- function(spc, k = 2, p = 0.05){
  # Calculate quantiles and quartiles
  q <- quantile(spc, c(p, 0.25, 0.75, 1 - p))

  # We are interested in points with too strong signal ...
  penalties <- spc - q[as.character(1 - p)]
  penalties [[penalties < 0]] <- 0
  # ... and too weak signal
  weak <- q[as.character(p)] - spc
  penalties [[weak > 0]] <- weak [[weak > 0]]

  # Normalize by IQR and return generalized mean
  iqr <- q["0.75"] - q["0.25"]
  (rowMeans( (penalties / iqr)^k) )^(1/k)
}
```

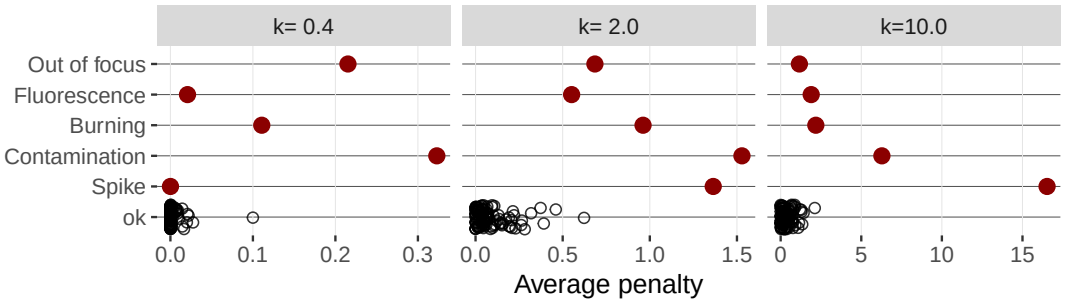


Figure 5.4.: Calculated penalties for different values of the exponent k. 333 spectra in total.

I applied this algorithm to the spectra shown in Figure 5.1. The distribution of the average penalties is provided in Figure 5.4, and we see, that the algorithm works pretty good for the detection of outliers. By adjusting the value of k, one can select, whether spectra with individual "wrong" data points, such as spikes, are detected as outliers (here for $k \geq 2$) or not. Burning and contamination, the most undesired artifacts, are detected really well. Out of focus spectrum is very close to the "typical" range, which is also seen in Figure 5.2. Such issues are better detectable using AUC for specific spectral features. Note, that although we could detect the spectrum with a fluorescent background as outlier, it's penalty is very low. The fluorescence was weak and added some noise to the spectrum, but it did not dramatically change the spectral signatures of cells.

For this particular data set the value $k = 2$ and the threshold value of 0.5 seem to be optimal. However, a different value can be necessary depending on what outliers types and in which proportion are present in the dataset.

5.2. Automatic Cell Identification with Machine Learning

One of the goals of the current thesis was a reliable automatic identification of cancer cells based on their Raman spectra. This is a challenging task, because spectra are complex data sets that contain a number of variations caused not only by the natural intercellular differences, but also by variations in the instrument alignment and sample preparations, as well as noise (see Figure 5.5).

Each spectrum represents a mixture of overlapping signals from thousands of different biomolecules. Even though some of these molecules are different between healthy and cancer cells, the changes are small and distributed in a broad wavenumber range. Thus, cancer cells do not feature any clear spectral markers which would allow to immediately recognize them. This is the main reason why spectroscopic label-free techniques are so challenging in comparison with other methods that rely on antibody tags, such as fluorescent labels.

This complex task requires the use of machine learning algorithms. *Machine learning* is a subfield of computer science, in which a machine learns to recognize specific patterns in data without being explicitly programmed [125, 126]. This means, that the computer derives some statistical measures and trends from a given training dataset and uses this knowledge to make data-driven predictions for new observations it has never seen before.

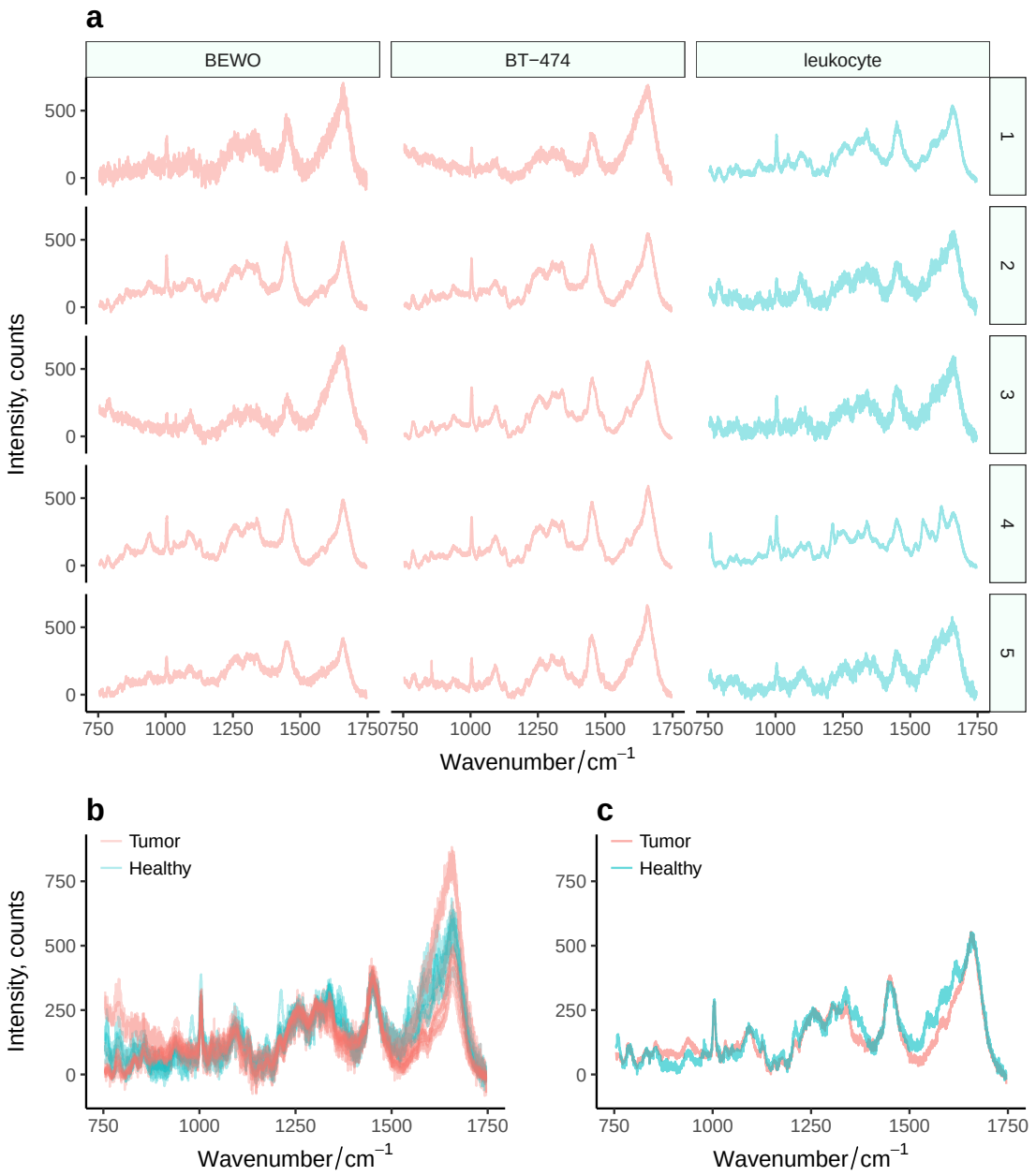


Figure 5.5.: Comparison of Raman spectra from three different cell lines. This picture illustrates the complexity of the data and shows that cell identification is challenging. Apart from different cell lines, all spectra have been acquired under the same experimental conditions (CaF_2 substrate, 10 seconds exposure time, RS660 instrument). For each cell line, 5 spectra have been randomly sampled from the database and normalized to have the same AUC. **a** – Overview of the data set, spectra of each individual cell are visible. The line appears thick due to the high number of data points (2560) and strong pixel-to-pixel variations in the signal. **b** – Overlay of all 15 spectra highlights strong differences between these two cell types. **c** – Comparison of mean spectra of tumor cells and leukocytes reveals consistent variations between the cell lines. This example clearly illustrates, how tremendously big are the variances between individual cells compared to the variances between different cell lines.

Generally all machine learning problems are classified into two categories, called *supervised* and *unsupervised* problems. Below I give a short overview of both categories and discuss their applicability to Raman spectra.

5.3. Unsupervised Methods and their Application to Spectra

This group contains techniques that analyze a given data set in terms of intrinsic variances and try to find hidden relationships between the variables. These methods assume that no ground truth about observed data is available, e. g. we do not know which classes or cases correspond to particular observations.

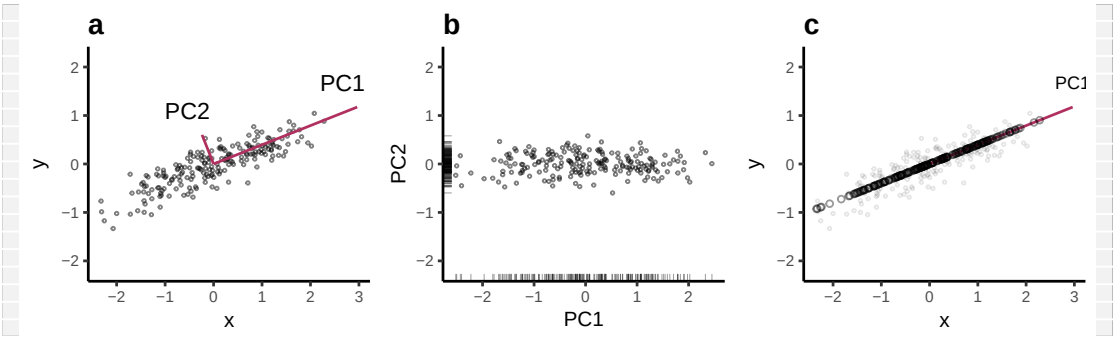


Figure 5.6.: Principal component analysis of a two-dimensional (x, y) data set. **a** – PCA finds a new orthogonal basis formed by vectors PC1 and PC2. New basis vectors are formed as a linear combination of the original x, y variables in such a way, that the variance described by PC1 is greater than those of PC2. **b** – The same data set projected onto a new basis formed by PCs. The point cloud is now parallel to the plot axes. This is just a coordinate transformation, original data can be reconstructed back. **c** – Dimensionality reduction. If we drop the PC2 component, and return to the $(\vec{x}, \vec{y})$ basis, we would get a reasonable approximation of the original data, which, however, contains less variables (here the data set becomes one-dimensional).

Typical examples of methods in this group are *principal component analysis* (PCA), *k-means* clustering, *hierarchical cluster analysis* (HCA), *expectation-maximization* (EM) algorithm, etc. PCA is a statistical technique which transforms a data set into a space of new orthogonal variables called *principal components*. This is done in such a way, that the first PC has the highest possible variance, and each succeeding new variable in turn accounts for the largest possible part of the remaining variance (see Figure 5.6). In simple words, this method finds the most prominent variations in the data set, but it does not care for the cause of these variations. Typically PCA is used to find the most important variables, hidden trends in the data, or to reduce the number of dimensions by dropping some of the least-important PCs. For a given data set, PCA *always* returns the same unique result.

Cluster analysis encompasses formal methods to group objects according to their intrinsic characteristics or similarity. Jain [127] gives a comprehensive review of modern clustering methods and discusses their strengths and weaknesses.

k-means clustering divides a given data set into a defined number of clusters (Figure 5.7a–c). This is an iterative technique, which associates each observation with the nearest cluster

and recomputes cluster average values, or their centers, in each step. The methods starts with randomly chosen cluster centers, so (1) it can end up in a local maximum and (2) it can give different results on two successive runs (Figure 5.7a–b). k-means does not account for the variation within each given cluster, thus the decision border is always the middle line between the centers of adjacent clusters. Often this is not the optimal solution, as shown in Figure 5.7c. Another drawback of k-means is a high computational complexity $\mathcal{O}(n^{dk+1})$, where n is the number of data points, k is the number of classes, and d is the number of dimensions. The reason is that on each iteration k-means computes distances between each data point and each cluster. Therefore, for applications like image analysis derived methods that analyze only local neighborhood, such as superpixels [128], are much more appropriate.

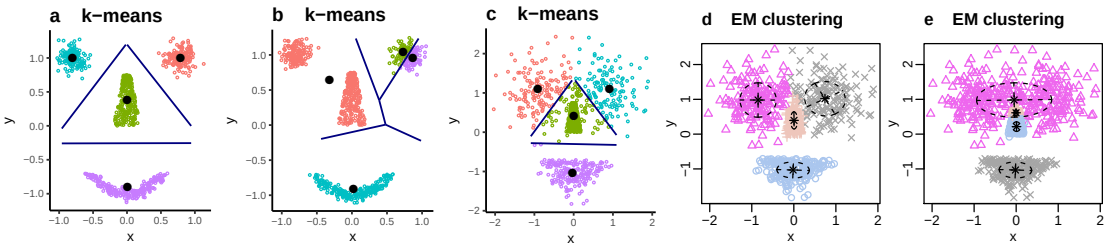


Figure 5.7.: Examples of *k-means* clustering and *Gaussian mixture models* (*EM*-type clustering) applied to artificial data sets. **a** – k-means results in a proper clustering of the dataset with clearly distinct clusters. The result corresponds to an intuitive grouping of the data points. **b** – Due to a random selection of the seed points, during another run the algorithm ends up in a local minimum on the very same dataset. **c** – k-means is not an appropriate method to cluster data with groups that feature very different data spread. In this case, a distribution-based expectation maximization methods (EM), such as Gaussian mixture models, shown in **d**, are more appropriate. **e** – EM models, if used in the unsupervised mode, can also by chance return an incorrect clustering of the data set.

EM algorithm is a distribution-based iterative method, which tries to fit a given number of multidimensional distribution functions to the available data points. At each step, the fitting parameters are varied and the likelihood of the estimation is calculated. The algorithm iteratively looks for those distribution parameters that result in the maximum likelihood of the distribution. A popular subtype of the EM algorithm is called "Gaussian mixture models". The EM algorithm method accounts for the internal variance in the dataset, so that the cluster size corresponds to the data spread. This behavior is advantageous over the simple k-means clustering, and works particularly well for low number of variables and high number of observations, as shown in Figure 5.7d. However, EM algorithm suffers from several drawbacks. First of all, it is computationally expensive, with complexity quickly rising with more variables. If the number of variables is close to or even lower than the number of observations, which is often the case in spectroscopy, the algorithm does not converge, because it does not have enough data points to fit multivariate distribution functions. Second, the method is sensitive to the initial placement of the seed points, and with a high number of dimensions it is likely to end up in a local minimum (see Figure 5.7e). However, with a manual placement of the seed points, this drawback can be eliminated. Note, that in this case a prior knowledge about the data is used, moving this algorithm to the category of (semi)-supervised methods.

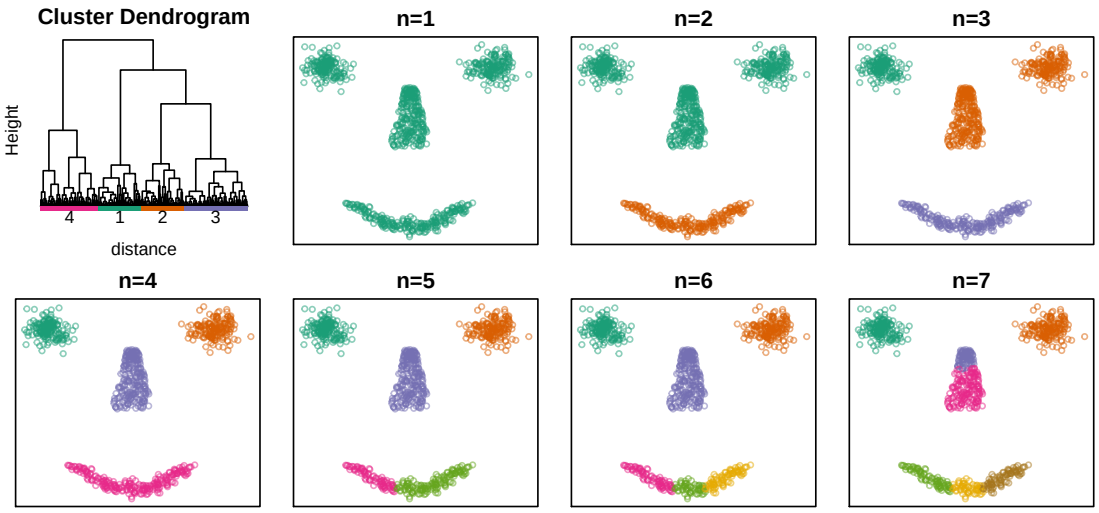


Figure 5.8.: Hierarchical clustering applied onto an artificial smiley dataset. The upper left panel shows a cluster dendrogram, from which we see that four big clusters feature large distances from their parents. The remaining panels show clustering result with the number of clusters varying from 1 to 7.

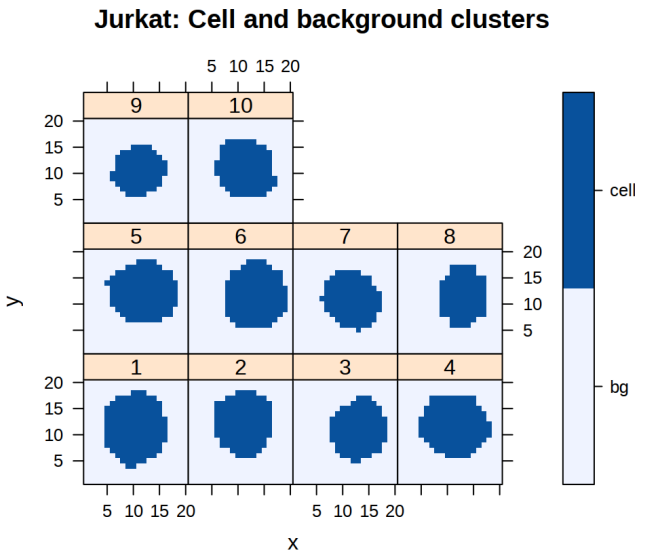
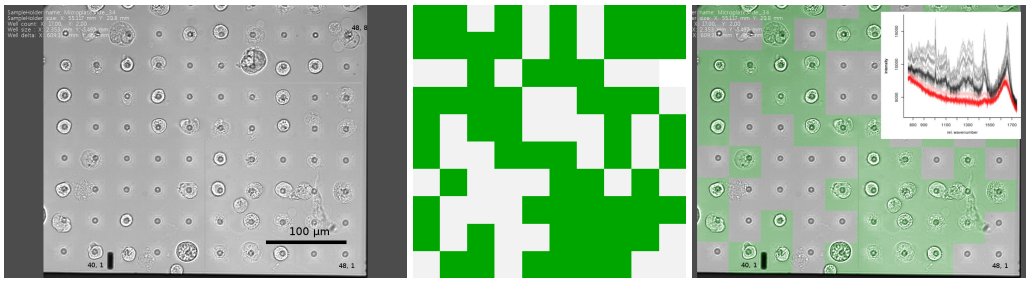


Figure 5.9.: k-means clustering separates cell spectra from background ones in Raman maps of ten cells [69].

Hierarchical cluster analysis is a technique that tries to split a data set into a hierarchically organized tree of clusters based on their similarity. The similarity is derived from a distance matrix containing the differences between each pair of the values in the data set. Each tree node has two children, and the number of nodes grows as one travels from the root node down the tree. Once the hierarchy is built, one can cut the tree at an arbitrary level into a desired number of clusters. The benefit of HCA is that the distance matrix is calculated once for a data set, and the number of clusters can be varied later, as illustrated in Figure 5.8. The (dis)similarity of the clusters is calculated using one of many mathematical metrics, such as euclidean distance, maximum distance, Manhattan distance, etc. Unfortunately, distance calculation for each pair of data points is computationally expensive.

Clustering techniques are broadly used in many areas of science to split the available data into distinct groups. In biomedical spectroscopy there are several applications as well. For example, in "RamanCTC" project a microhole array has been used to capture cells and immo-



(a) Overview of the measured area **(b)** HCA clustering **(c)** Overlay with the image and inset with the clustered spectra

Figure 5.10.: Application of hierarchical clustering to check occupation of microholes by cells. The cells have been immobilized on a microhole array chip [90], and a Raman spectrum has been acquired from each hole. The immobilization of the cells is a stochastic process, so we do not know in advance which holes would be occupied and which not.

bilize them on predefined locations [90]. I used cluster analysis to identify microholes that are occupied by cells[1] (Figure 5.10). A similar problem is shown in Figure 5.9: here, I identified what individual spectra in Raman maps contain cell signatures, and what do not.

Often clustering techniques are used to process spectroscopic imaging data. Suppose it is known that n different substances give the major contribution to the sample composition, but their spatial distribution in the measured region of interest is not known. Since these substances yield different spectra, we can separate the data set into n clusters and assign each cluster to a given substance.

Particularly useful here are the *N-FINDR* [87, 88, 129] and *Vertex Component Analysis* [130] (VCA) algorithms, which search for the most distinct spectra in a given data set, a process called "endmember extraction". This technique is analogous to a selection of the most extreme points of an n-dimensional scatter plot. The "purest" components would be the spectra of (almost) pure substances that compose the sample, and it is possible to calculate their relative concentration at each point of the Raman map, as illustrated in Figure 5.11. This process is also known under the name "spectral unmixing".

5.4. Supervised Machine Learning for Cell Identification

Supervised machine learning is a class of techniques that establish mathematical models describing labeled training data, and use these models later to make data-driven predictions for new observations that are not part of the training data set. There are two major classes of super-

[1] It is an inefficient method from the experimental point of view, because time-consuming acquisition of Raman spectrum is done at *each* microhole regardless of whether it is occupied or not. A much more reasonable workflow would involve taking a picture of the microhole array, using image processing to identify occupied holes, and measuring only those holes that are occupied. This procedure can also be used for a fast screening: out of about 200,000 cells, a much smaller subset of large suspicious cells could be selected. Unfortunately, this was not implemented during our experiments.

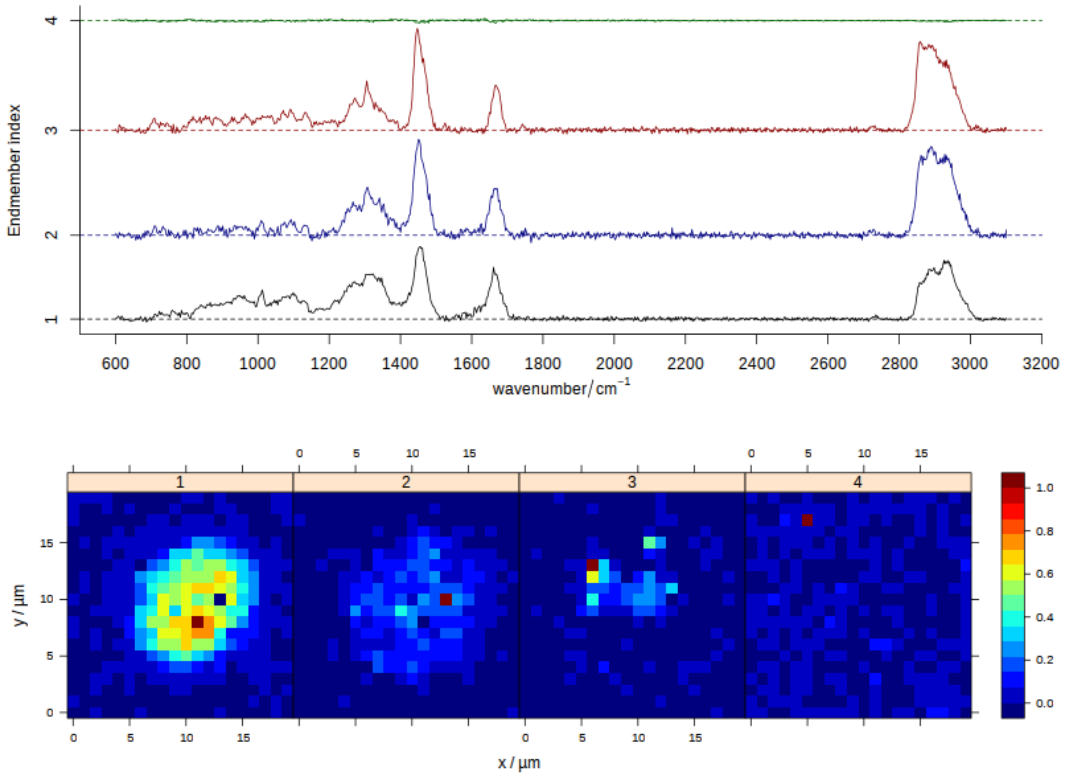


Figure 5.11.: Four endmembers extracted with N-FINDR algorithm from a Raman image of a tumor cell, and their corresponding concentration maps. Data from [69].

vised problems: *regression* and *classification*. In the former case, the model describes a relation between two continuous variables in the form $y = f(x)$, and tries to estimate the function f. Then the estimation f_e of the function f is used to predict values of the variable y for new observations x', i. e. it returns $y'_e = f_e(x')$ which is an approximation of $y' = f(x')$. An example of regression model is a Gaussian fitting, which establishes a model giving a numeric estimate for any value of the independent variable. Another class of problems is *classification*, which encompasses models called classifiers that work with categorical labels, as opposed to numeric ones. A relevant example would be a classifier that predicts a cell type based on its Raman spectrum. Such classifier analyses numeric data, but gives a categorical value as the output.

Further, there is a class of techniques called *reinforcement learning*. Here, a constant feedback in a form of reward or punishment is provided to the model on each prediction that it makes. The correct input/output pairs are not present, instead, the model balances between the exploration of new cases and the exploitation of the obtained knowledge. This leads to a constant optimization of the model parameters [131]. I did not use such models in this work, although they might be useful for our application as well.

5.4.1. Validation

"All models are wrong, but some are useful" – George Box [132]

It is not enough to just train a classifier that can make predictions on new data – the classifier performance must also to be proven, e. g. objectively measured and characterized. The process of the formal classifier testing is called *validation*. It is important to know the *accuracy* of the model, e. g. how probable it is, that the classifier would correctly assign a label to a new observation that it has never seen before. The corresponding property of regression models is the *root mean square error of prediction* (RMSE), which is the standard deviation of the differences between the observed values and the predicted ones.

For a binary classifier[2] the *sensitivity* and *specificity* values are often of interest. Sensitivity shows, how probable it is, that the model can correctly recognize positive cases as such. The specificity, on contrary, is the proportion of the negatives that were correctly identified as negatives.

To perform a classifier validation, one could acquire new data, i. e. perform the experiment one more time, and ask the model to predict the type of each new observation, given the fact that the type is known. The percentage of correct predictions can be used to estimate the accuracy. Unfortunately, due to a limited number of test cases, this estimate could be biased. In fact, the test cases could be just by chance different from the training ones or from those that would later occur during the classifier use. Moreover, it is not always practical or even possible to perform new experiments and collect more data just to validate the model.

Therefore, usually a model is validated with already available data that get split into stratified training and testing sets. A new so-called *surrogate* classification model is trained with one data set and is validated with the other one in the assumption, that the final classifier would be better, but not worse, than the surrogate model[3]. This process has to be repeated many times, as the calculated accuracy value will always fluctuate around the "real one" due to the limited number of observations and their random sampling. It is important to note, that all data processing steps that *depend on the values themselves have to be done inside of* ***each*** *training loop*. This guarantees that the testing data set remains independent from the training set and does not bias the surrogate model.

The model performance is usually characterized with *cross-validation* or *bootstrapping* procedures, as well as their variations, which are described elsewhere [133, 134].

Sample Size Planning

It is generally accepted that a high number of observations is mandatory to build a good classifier and to properly validate it. However, it is often forgotten that not the quantity of observations *per se* is important, but the number of possibly different cases that they cover. In bio-spectroscopic studies this is the number of **statistically independent** samples (in our case these are individual patients with CTCs[4] and healthy volunteers), which ultimately affects the

[2] *Binary classifier* assigns each observation to one of two possible classes, i. e. A/B or *Positive*/*Negative*.

[3] This assumption comes from the fact, that the final model uses the whole data set, e. g. more training data, than each surrogate model.

[4] *Circulating Tumor Cells*

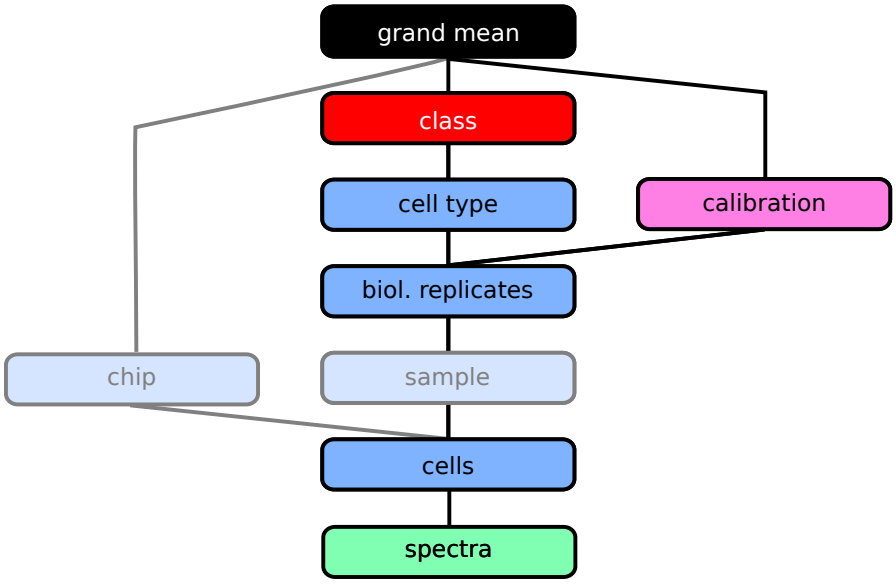


Figure 5.12.: A *Hasse diagram* representing the hierarchical structure of our spectroscopic data. Image credit: Dr. Claudia Beleites [122].

ability of the classifier to correctly identify new cases [134]. The availability of such samples is, unfortunately, always limited, as properly annotated and statistically independent specimens are very costly, with the patient biopsies being the extreme case.

There are many confounding factors in our spectroscopic experiments that influence the performance of the classifier. In addition to the variations between the actual classes that we want the classifier to predict, there is a measurement noise, the influence of the sub-cellular location, the variations between individual cells (biological replicates), as well as the fluctuations between the instrument calibrations and the used substrates, as shown in Figure 5.12.

Thus, a proper number of observations is required to cover the variances on each level of the data hierarchy. Only in this case the classifier would "see" enough data to decide which patterns correspond to the actual characteristic features between the classes, and which are caused by confounding factors and thus have to be ignored.

The problem of the sample size planning occurs over and over again in bio-spectroscopic studies. The paper of Beleites *et al.* [123] goes into the details on this topic. Briefly, even if many individual spectra were acquired during the experiment, the low number of statistically independent samples can just by chance result in an overoptimistic accuracy value during the model validation. It is possible, however, to estimate a CI[5] for the learning curve based on the experimental design (and not on the actual data). Because the CI is typically very broad due to the sample scarcity, the *superiority of one classifier over another one **often** cannot be demonstrated* – this task would require a practically unreachable volume of statistically independent samples.

In this work, we acquired spectra of 20 individual cell lines tracing back to 6 different cancer

[5] *Confidence interval*

types[6]. Each cell line originates from an individual patient, thus it allows us to have twenty statistically independent "patients" in our study, which is a comparably high number. Note, however, that the cell lines are *just a model* of real CTCs, and large clinical trials would be necessary to demonstrate the performance of the workflow in the future.

For each cell line, several batches were prepared, and at least 50 individual cells (biological replicates) were analyzed with Raman spectroscopy from each batch. Some of the cell lines were measured with both RS785[7] and RS660[8] instruments and on two different substrates (calcium fluoride or Si_3N_4 microhole array chip). Thus, a more or less reasonable quantity of samples is available on each level of the data hierarchy.

5.5. Classification Models for Cell Identification

Images 5.13 and 5.14 show preprocessed spectra of leukocytes and different incubated cell lines from RS660 and RS785 instruments, correspondingly. The cell lines were grouped according to the corresponding cancer type. The preprocessing included subtraction of the dark frame and the spectrum of the surrounding medium, piece-wise polynomial baseline correction and area normalization. Visual inspection does not reveal any significant differences, and machine learning methods are required for the cell identification.

Biospectroscopic data typically contain high number of variables and a scarce amount of statistically independent samples. This circumstance renders many of the popular machine learning algorithms useless, as they expect quite the opposite: the number of independent observations greatly exceeding the number of features[9].

Simple linear models, such as *linear discriminant analysis* (LDA), are superior for such kind of problems, as they make their predictions based on strictly defined data transformations, i. e. they have a very limited number of degrees of freedom. These methods can discover only general trends in the data and often do not get influenced by occasional anomalies in the data set. In addition to LDA, it is worth to mention *lasso and elastic-net regularized generalized linear models* (glmnet) technique, as well as *random forest* (RF) and *k-nearest neighbors* (kNN).

I tested all these methods for the cell identification (about 1200 individual spectra, two-class problem, i. e. tumor vs healthy), and all of them provided reasonably good classification accuracy above 98% in most of the cases. Among them LDA is by far the fastest one, as it benefits from the hardware-accelerated matrix multiplication. Moreover, for our ill-posed problem of low number of statistically-independent observations it is actually impossible to prove that any of these techniques is superior to another one [123, 134]. Therefore, LDA was used further in this work as a fast and reasonably good machine learning technique.

[6] Breast cancer, leukemia, non-small cell lung cancer, hepatocellular carcinoma, pancreatic cancer and ovarian cancer.

[7] *Raman System with 785 nm excitation laser and a custom microscope, also known as "Roman Raman" (see Section 3.2.3)*

[8] *Raman System from company* Till ID *with 660 nm excitation laser, also known as "Raman Reader" (see Section 3.2.2)*

[9] Typical rule of thumb in machine learning is to use at least five independent samples for each observed feature. For the RS660 instrument, which returns 2560 data points per each measurement, in ideal case a training of a reasonably good classifier would require at least 12800 individual patients that donated blood for the Raman-spectroscopic tumor cells analysis.

Figure 5.13.: Preprocessed spectra of cancer cell lines present in the database, measured with the **RS660** instrument. Mean and standard deviation of all spectra within each group are shown. The cell lines are grouped according to the tumor type they originate from. The dataset contains 10632 individual spectra.

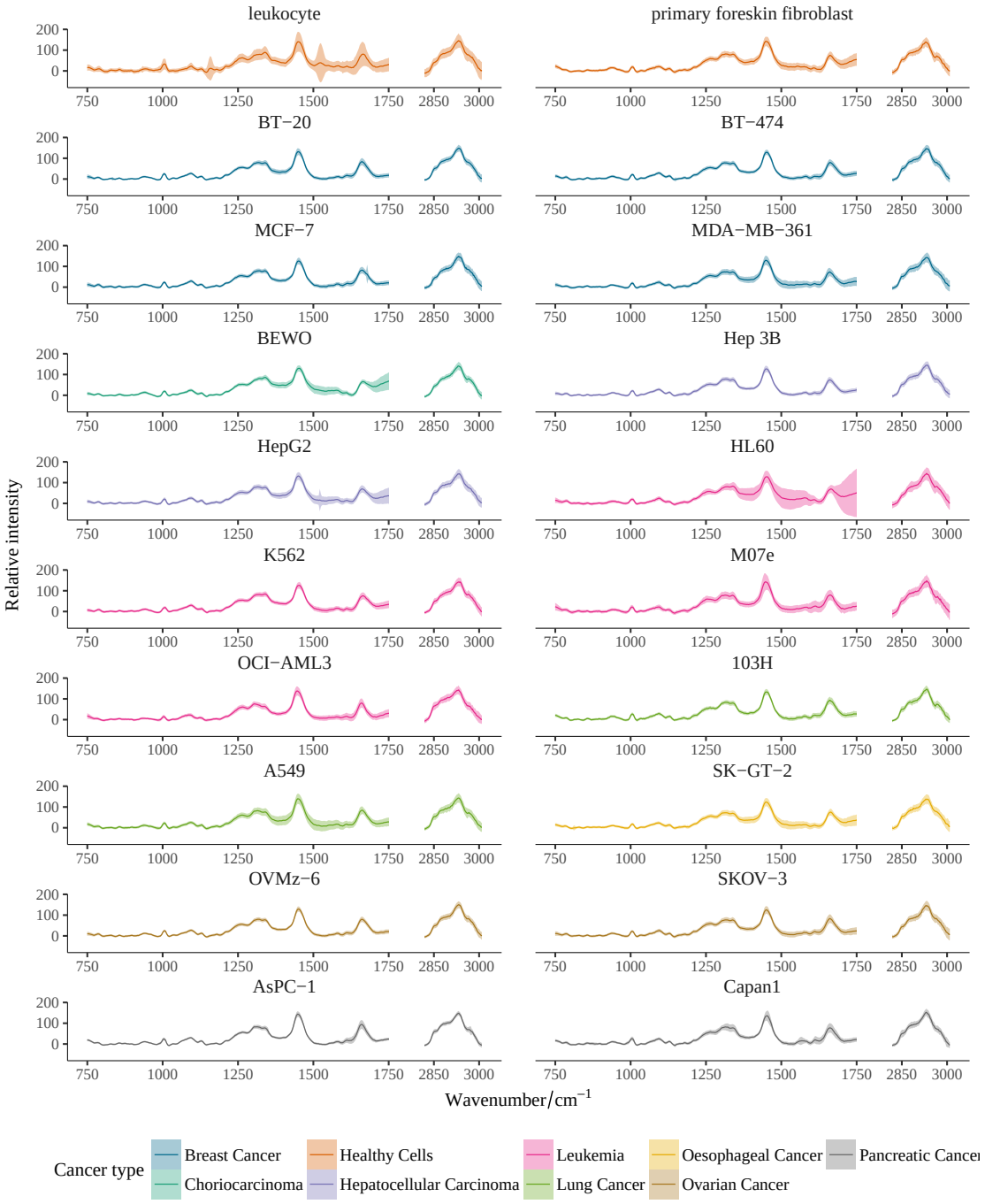


Figure 5.14.: Preprocessed spectra of cancer cell lines present in the database, measured with the **RS785** instrument. Mean and standard deviation of all spectra within each group are shown. The cell lines are grouped according to the tumor type they originate from. The CH-stretching region is downscaled 4 times for the sake of visibility. The dataset contains 7631 individual spectra.

LDA, as many other techniques, has to be preceded by a data dimensionality reduction step. It is typically carried out using *principal component analysis* (PCA) or *partial least squares regression* (PLSR), which is in turn internally based on PCA. It is important to select the proper number of new latent variables at this stage. Reduced dimensionality serves two purposes. First, it results in a faster model training. Second, it provides a more generalized projection of the initial dataset, which helps to prevent overfitting.

5.5.1. Selection of the Optimal Number of Principal Components

PCA/LDA classifier is a combined model that uses the unsupervised method of principal component analysis to find a set of new variables that describe the most prominent variances in the data set. The purpose of this method is to reduce the data dimensionality, as the number of new variables, called *principal components*, *latent variables* or *loadings*, which are essentially linear combinations of the original predictors, can be drastically lowered. For more information see Section 5.3.

It is quite challenging to estimate the number of principal components that should be retained without compromising the trends hidden behind the data. There are, however, several empirical approaches to this problem. First of all, one can check the proportion of the explained variance, as shown in Figure 5.15. With the decision threshold of 2% of the explained variance the first 6 PCs in the RS785 data and the first 6 PCs in the RS660 spectra appear to be significant. However, it is hard to decide what threshold value has to be used.

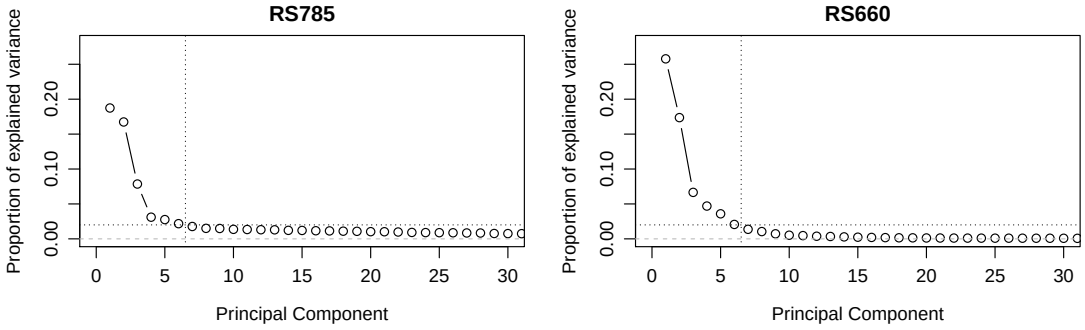


Figure 5.15.: Proportion of explained variance by principal componentsin Raman spectra from RS785 (7631 spectra) and RS660 (10632 spectra) instruments.

However, we have to keep in mind that PCA is an unsupervised method that does not necessarily find what one is searching for. It can happen, that the most prominent variances are **not** caused by the different cell types, but instead by some other confounding factors. Therefore, to be on the safe side, I additionally checked the visual appearance of the loadings to see whether the variances are caused by Raman peaks or by some systematic observation errors. Finally, I tested how the prediction accuracy, or the performance of the combined PCA/LDA model actually depends on the number of retained principal components.

Figure 5.16 shows, that PCA loadings from RS660 do not reveal any significant Raman signatures after PC_{16}, so we would not expect that the use of more than 16 principal components

would lead to any improvement of the classification accuracy. On the other hand, PCA loadings of the RS785 data set contain Raman bands at least up to PC_{38}. It is questionable though, whether that many PCs are required to train a good classifier, as the high number of variables can readily lead to an overfitted model.

It is also interesting to note, that the PC_4 of the RS660 PCA displays some kind of a "noisy" pattern, which follows by visually less "noisy" lines in subsequent PC_5–PC_9. This is most probably caused by the inherent pixel-to-pixel differences of the sCMOS sensor, as in contrast to a CCD each pixel of the sCMOS sensor contains an individual amplifier. Although much of this variation was compensated by subtraction of the dark frame, we obviously evidence some residual pattern.

We see, that the results from both empiric methods for the selection of the optimal number of principal components contradict with each other. By looking at the variances, we conclude that 6 PCs are enough, whereas looking at the loadings suggests to use about 40 and 16 PCs, correspondingly.

A more reliable approach is to actually test the influence of the PCA step on the prediction accuracy of a two-class LDA classifier. To do this, I did the following steps:

1. I pulled a fraction of data (about 15%)[10] with replacement from the whole dataset;
2. I applied the principal component analysis onto the data, and kept only the first N components, where number N was varied;
3. I used the new dataset with N variables to train the LDA classifier and evaluate its prediction accuracy. The accuracy of the model for each number N is estimated using a 3-times repeated 10-fold cross-validation.

The whole calculations, i. e. the steps 1–3, were repeated 5 times to account for the sampling variances. The results are shown in Figure 5.17. From this I conclude, that the optimal number of PCs would be about 30 for the RS660 data, and about 150 for the RS785 data[11]. These values were used in the subsequent PCA/LDA models.

5.5.2. Accuracy of Classifiers Depending on the Exposure Time

The database contains spectra that were acquired using different exposure times (see Table 4.4), the most often used values were 1.0 second and 10.0 seconds.

This allows to investigate two important practical questions:

1. Does a longer exposure time (e. g. 10 seconds) results in a major increase of classification accuracy as opposed to experiments with just 1 second exposure time?

[10] I did not use the whole dataset to reduce the computational costs. This means that this approach is actually a combination of bootstrapping with cross-validation.

[11] This high number of PCs could be caused by the automatic wavelength calibration routine (Section 4.4). Although it helped to keep the calibration correct, minor shifts of the wavelength axis from experiment to experiment could have been detected as important variables by PCA.

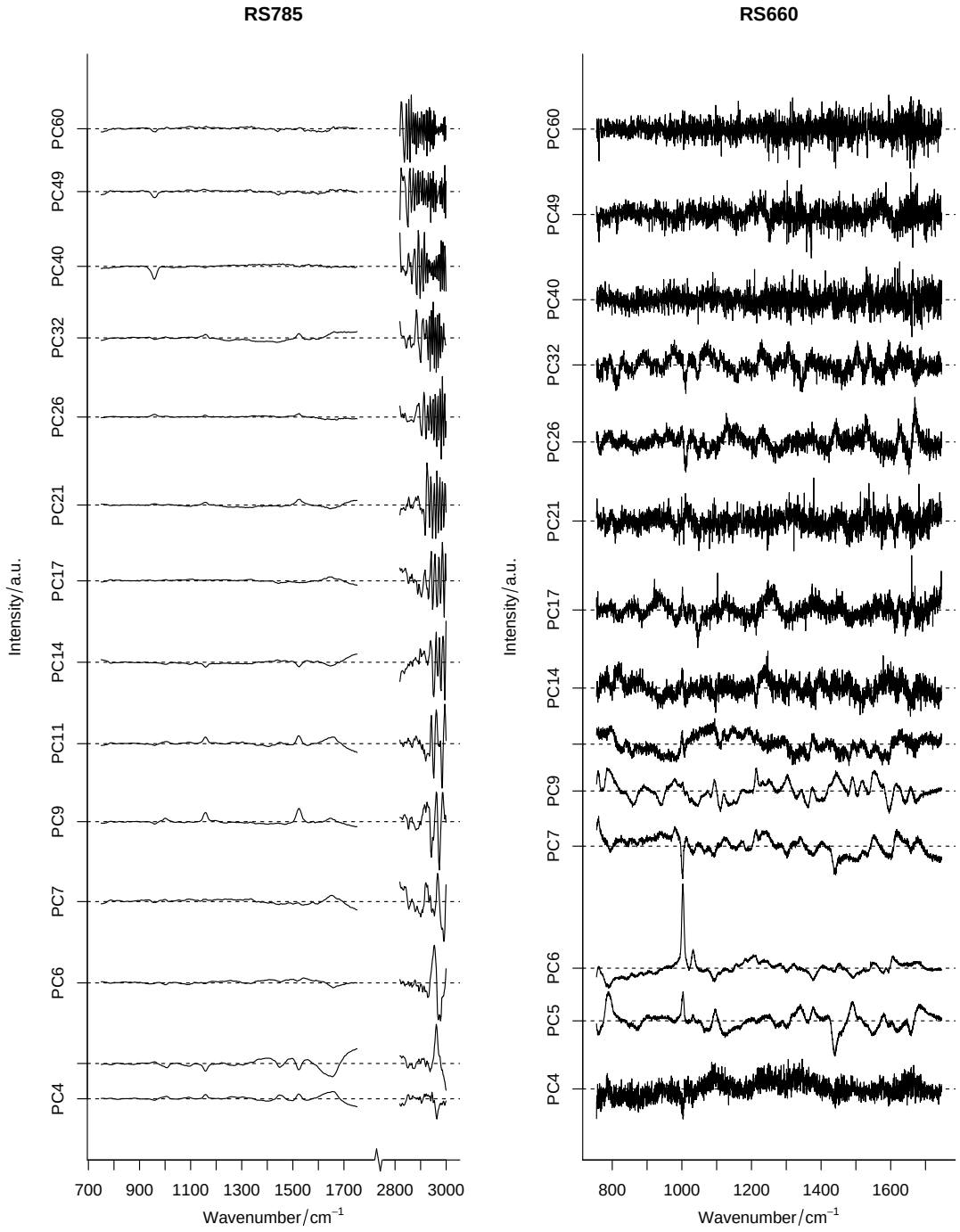


Figure 5.16.: Visual representation of PCA loadings calculated from the RS785 and RS660 datasets. In the dataset from the RS785 instrument the Raman signatures are present in many loadings, at least up to PC_{38}. On the other hand, in PCA loadings of the RS660 dataset one can observe spectroscopic features only up to the loading PC_{16}; loadings beyond PC_{16} contain only a systematic noise pattern on the sCMOS detector.

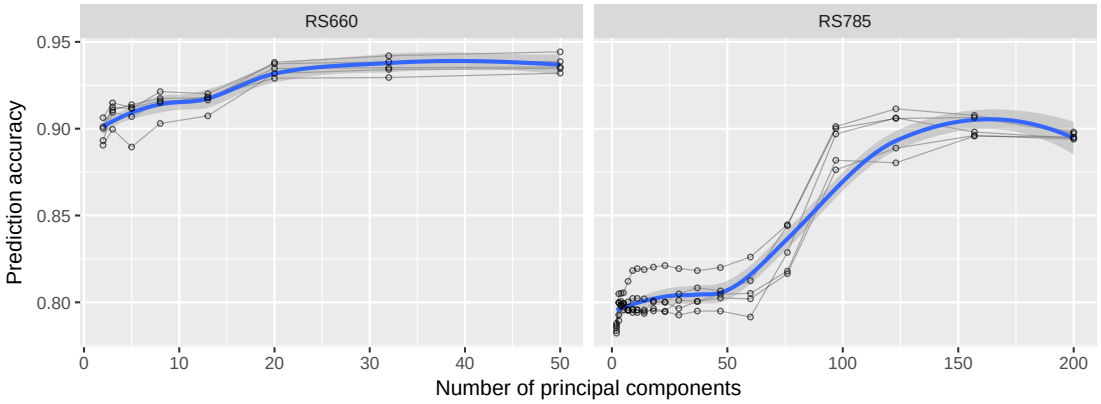


Figure 5.17.: Accuracy of the PCA/LDA classifier depending on the number of principal components. The shaded area shows the 95% confidence interval of the blue spline approximation line.

2. Can a classifier trained on the data collected with one exposure time be reliably used to make predictions on the data collected with another exposure time?

To answer this question, I investigated two datasets shown in Figures 5.13 and 5.14. The distribution of the spectra over exposure times is provided in the Table 5.2. Two-class (i. e. tumor/healthy) PCA/LDA classifier was trained and cross-validated to investigate how the model accuracy depends on the acquisition time during the Raman experiments.

Table 5.2.: Distribution of spectra over exposure time. Same dataset as in Figures 5.13–5.14.

Exposure time, s	RS660	RS785
1.0	3547	3002
2.0	197	3
5.0	787	470
10.0	6101	3955
15.0	–	150
30.0	–	51

First, I built a model using the whole dataset and estimated the classifier performance using 10× repeated 10-fold cross-validation. The classification model demonstrated prediction accuracy of about 91.3% (RS785) / 94.0% (RS660), as shown in Table 5.3.

In the next step, I used only the 10-second data to train the classifier, and used the model to predict the cell type using spectra acquired with 1 second exposure time. The classifier accuracy for the 10-second data turned out to be slightly better than in the previous case, in particular 94.2% (RS785) / 94.6% (RS660). This shows, that the more pronounced noise due to a worse signal-to-noise ratio in the 1-second data confuses the classifier. However, this effect is very weak. I used the same classification model to predict the cell type based on spectra acquired with 1 second exposure time (the so-called "testing dataset"). This resulted in prediction accuracy values of 81.7% and 93.0% for RS785 and RS660, correspondingly.

Additionally, I did a complementary test: the classifier was trained on the 1-second spectra and tested on the 10-second data. The prediction accuracy of 88.8% (RS785) turned out to be lower than in the previous experiment, but I did not observe any dramatic degradation of the prediction accuracy that one could expect. Surprisingly, the cross-validated accuracy for the RS660 data (95.8%) was even higher, than in the other two cases. The models were able

to predict the cell type on the 10-second data with reasonably good accuracies of 79.7% and 92.3% for RS785 and RS660, correspondingly.

To sum up, I did not observe a dramatic change in the prediction accuracy when the exposure time of the spectrum varied between ten seconds and one second. In fact, with the quite broad IQR of accuracy values, and low number of biological replicates, it becomes questionable, whether an exposure time above one second provides any substantial advantage to the Raman-based cell identification with the instrumentation used here.

Table 5.3.: Prediction accuracy (median and interquartile range, IQR) of a two-class PCA/LDA classifier depending on the exposure time. The accuracy was determined using 10× repeated 10-fold cross-validation. See description in the text for more details.

	Training dataset			Testing dataset	
		CV Accuracy			
Instrument	Exposure, s	median	IQR	Exposure, s	Accuracy
RS785	all	91.3	90.8 – 92.0	–	–
RS785	10	94.2	93.6 – 94.9	1	81.7
RS785	1	88.8	87.6 – 89.7	10	79.7
RS660	all	94.0	93.6 – 94.4	–	–
RS660	10	94.6	94.1 – 95.1	1	93.0
RS660	1	95.8	94.9 – 96.1	10	92.3

Moreover, as I could not prove that 10-second exposure time is reasonably better than the 1-second exposure time with our hardware, the use of 1-second exposure time would be more efficient, because about ten times more spectra can be collected within the same time frame. This would allow to perform further experiments more efficiently, as more cells could be analyzed with the Raman spectroscopy.

Finally, I demonstrated that data from different exposure times can be intermixed together without any substantial issues. This is a promising result on the way towards spectroscopic model transfer. It means, that during the data collection the exposure time can be selected as a trade-off between the number of cells to be investigated and the time frame available for the experiment. Note, however, that the exposure time should not be very low or unreasonably high. In the first case, the SNR[12] would deteriorate as the camera read-out noise becomes the major noise source, followed by the shot noise. In the second scenario, the dark current of the sensor starts to show a non-linear behavior. There is also a risk of overexposure, i. e. saturated detector, which leads to the trimmed peaks. This case is particularly bad, as the information at specific wavelengths gets lost. Simple linear classification models, such as LDA, cannot correctly handle such cases.

[12] *Signal-to-noise ratio*

5.6. Use of Classification Models to Identify Specific Tumor Type

As it was shown above, classification models (in this case combined PCA/LDA model) can reliably distinguish between healthy and cancer cells. We can, however, try to additionally identify the particular type of cancer, as this question is also very relevant for the clinical diagnostics. This capability could have an important practical application, as the identification of a particular cancer type based on CTC detection and analysis would result in a substantial improvement in the cancer monitoring and early diagnostics.

To investigate this issue, I used several data analysis methods. I started with an exploratory data analysis and applied principal component analysis onto the preprocessed dataset. This simple unsupervised technique was already sufficient to reveal some trends and groupings in the available experimental data, as illustrated in Figure 5.18. This gives a feeling, that particular individual spectra tend to form clusters when being projected into the space formed by the first two principal components PC_1 and PC_2. The clusters start in the common origin (which is a "noisy" spectrum that does not contain any cell signal) and point into slightly different directions. Probably, these clusters can be associated with different cancer types.

Further, I evaluated the performance of the PCA/LDA and PCA/RF[13] classification models, as they work quite well even under such unfavorable data quantities[14]. From the two tested machine learning methods, one cannot reliably say which one is superior, as their accuracies are approximately equal in all tests that are discussed in this section.

The classifier was trained on spectra of tumor cells only. Spectra of cultivated cancer cells, measured with the RS660 instrument, were grouped by cancer type in accordance with the Table 4.6. Five times repeated five-fold cross-validation was used to estimate the model accuracy. The classification model was trained to distinguish between seven different tumor types, in particular *breast cancer*, *choriocarcinoma*, *hepatocellular carcinoma*, *leukemia*, *lung cancer*, *oesophageal cancer*, and *ovarian cancer*.

Unfortunately, the accuracy of the classifier was very low, about 64.2% (IQR 63.2–65.3%), which means that for the most of the time it was just guessing. Then, I completely removed one of the classes *(Lung Cancer)*, and repeated the model training and validation steps. This resulted in a small increase of prediction accuracy of about 5%.

I kept removing classes further one by one, and observed a constant grow of the prediction accuracy (Table 5.4). This implies, that tumor cells from the spectroscopic point of view are similar to each other, even though they belong to different tumor types.

[13] Random Forest

[14] Most of the machine learning methods require the number of independent variables to be much lower than the number of statistically independent observations, which is not the case in our study and the majority of other similar biospectroscopic studies.

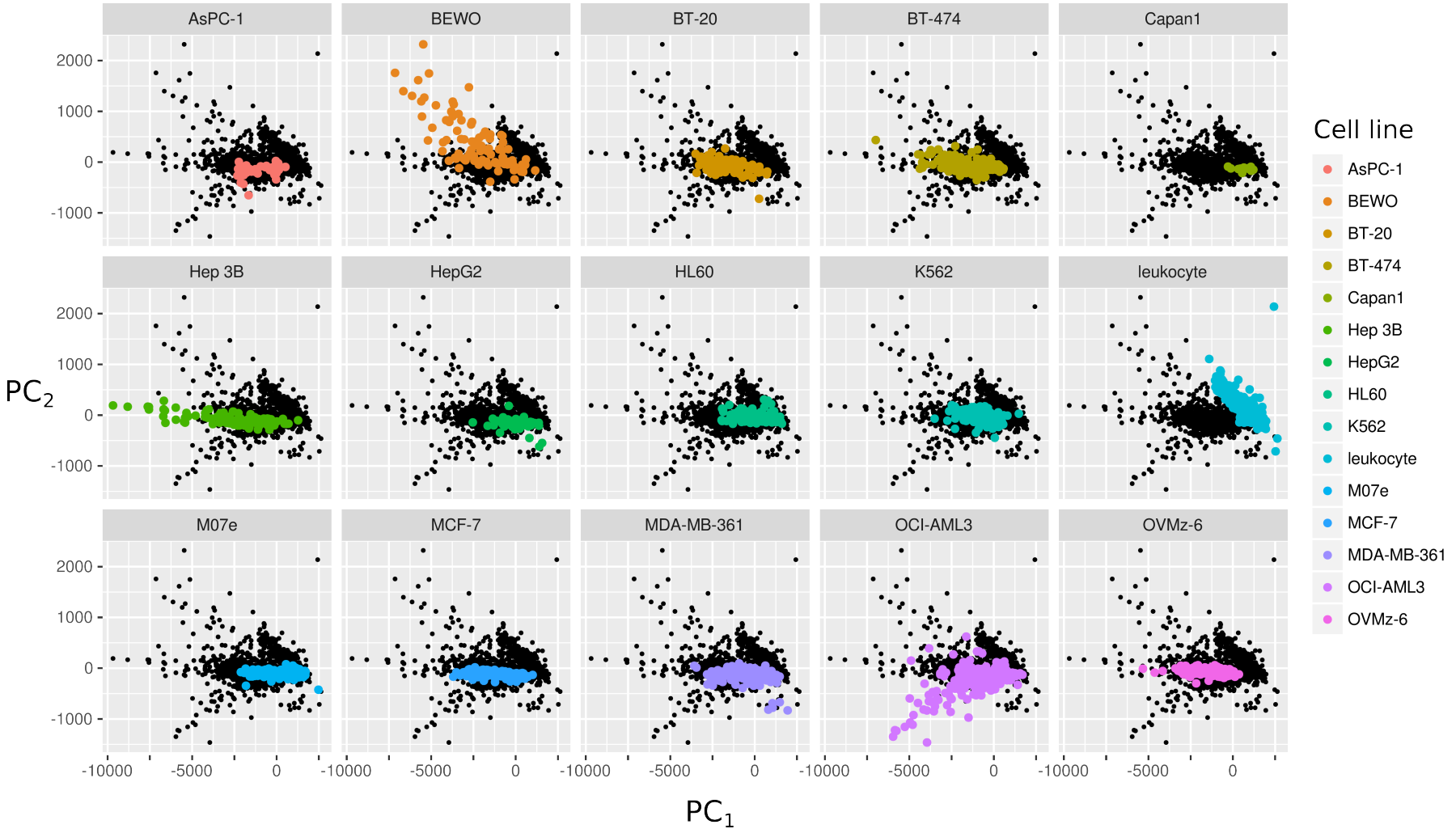


Figure 5.18.: Principal component analysis of the preprocessed cell spectra from the database. The figure shows the distribution of spectra in the coordinate space formed by the PC_1/PC_2 latent variables. The spectra are faceted and colored according to the cell line. It is easy to notice that individual spectra from one group tend to form point clouds of specific shapes, with leukocytes being the most distinct from all other groups. *Image credit: Dr. Claudia Beleites.*

Table 5.4.: Prediction accuracy (median and interquartile range, IQR) of PCA/LDA classification model trained to distinguish between different number of specific tumor types. The model was validated using 10× repeated 10-fold cross-validation. Data from RS660 instrument. See description in the text for more details.

Breast Cancer	Choriocarcinoma	Hepatocellular Carcinoma	Leukemia	Lung Cancer	Oesophageal Cancer	Ovarian Cancer	Prediction Accuracy	
							Median, %	IQR, %
2231	448	1042	1678	1840	285	552	64.2	63.2 – 65.3
2231	448	1042	1678	–	285	552	69.3	68.4 – 70.2
2231	448	1042	1678	–	–	552	72.0	70.9 – 72.9
2231	–	1042	1678	–	–	552	74.9	73.7 – 76.0
2231	–	1042	1678	–	–	–	81.4	79.8 – 82.2
2231	–	–	1678	–	–	–	91.4	90.3 – 92.1
Number of individual spectra								

The observed situation is also illustrated in Figure 5.19, which shows data distribution in the coordinate space formed by linear discriminants LD_1 and LD_2 of a three-class PCA/LDA model. During the cross-validation the data set was repeatedly split into training and testing folds; for each fold, a new surrogate PCA/LDA model with it's own latent variables was trained and validated. For each surrogate model I calculated the projection of all data points from the training folds (shaded background) and the testing fold (crosses) into the new coordinate space formed by the linear discriminants (LD-space); the plot shows the averaged distribution of these data points over all surrogate models. This illustration indicates, that in the LD-space the first latent variable LD_1 discriminates between leukocytes and of tumor cells, and the separation is very good, especially for the data from the RS660 instrument. The differences between the two cancer cell lines are encoded in the LD_2 variable, and there is no clear separation between them, although they belong to different cancer types (*A549* is lung cancer, *Hep 3B* is hepatocellular carcinoma).

The statement, that spectroscopic differences between different tumor cell lines are much smaller than between tumor cells and leukocytes, is additionally supported by Figure 5.18, where leukocytes form a clearly distinct cluster in the PC_1/PC_2 coordinate space.

This behavior, on one hand, complicates the diagnostic of a particular tumor type, but, on the other hand, it suggests the possibility of using cultivated tumor cell lines as a convenient substitute of real patient CTCs when training classification models. Cell lines are readily available and, if they can be used alongside with the patient data, they can partially solve the problem of scarce samples available for the model training. This is especially reasonable with two-class models that try to distinguish between healthy cells and CTC. The following section introduces such an experiment.

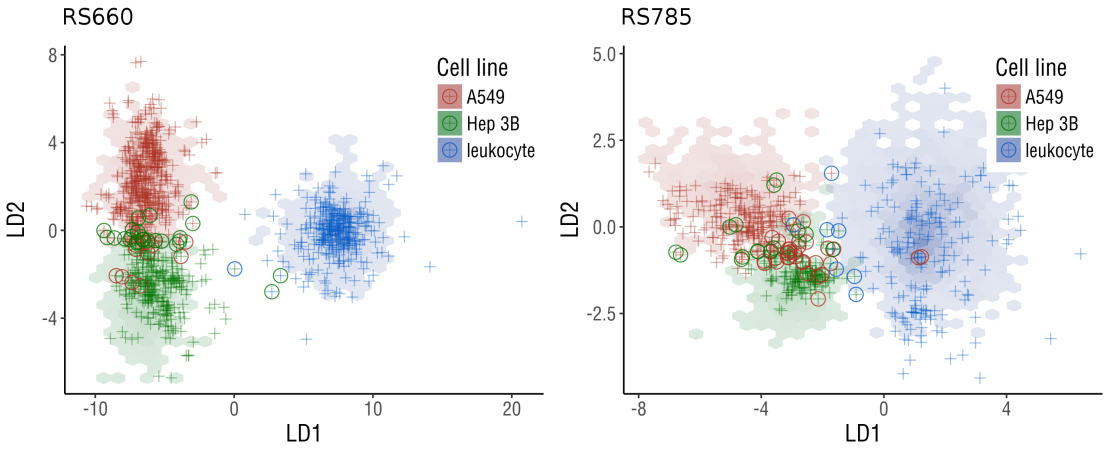


Figure 5.19.: Visualization of performance of the PCA/LDA classifier for a three-class problem. The image is an overlay of five folds from the cross-validation loop. The shaded background represents a distribution of the training spectral data being projected into the coordinate space formed by the linear discriminants. Correctly classified individual spectra are represented by crosses; misclassified spectra are marked with circles (the color of the circle shows the true class of the predicted cell). Note a very good separation of leukocytes (blue) from two clusters of tumor cells, that are relatively hard to distinguish from each other.

5.6.1. Prediction of Patient Samples

As mentioned in Section 4.7 and Table 4.7, Raman spectra were collected not only from the cultivated tumor cell lines, but also from primary human cells. This includes leukocytes donated by healthy donors as well as blood samples from oncological patients. As we use incubated cells as a model of the real CTCs, it was of great interest to estimate whether a classification model trained on incubated cells and leukocytes could also perform reasonably well on the data obtained from the real patients, probably involving some CTCs.

I trained a classification model using spectra of incubated cancer cell and healthy leukocytes. The dataset in the database contains about nine thousand cells labeled as "tumor" and about two thousand "leukocytes". This ratio does not reflect the occurrence of the real CTCs and it can bias the classifier, as the classification model expects a similar 9:2 proportion of CTCs in the testing data. To reduce this effect, I artificially changed the aforementioned 9:2 proportion in the training dataset towards 1:5. For the model training and validation I used all available spectra of leukocytes, but took only 5% of tumor cell line spectra. These 5% were randomly sampled from the database. The 1:5 proportion is still not realistic, because we are looking for "one cell in a million", as it is being often colloquially said. Unfortunately, the use of realistic CTC concentrations would leave the classifier basically with no training data. Any further data balancing is also not feasible, as low number of spectra in the "CTC" class would not sufficiently represent the variance between tumor cells.

The total number of spectra used for the classifier training was 3424. The classifier was validated with a 10-times repeated 10-fold cross-validation and demonstrated a very high prediction accuracy, in particular the median sensitivity value of 93.71% (IQR 92.81–94.7%), and

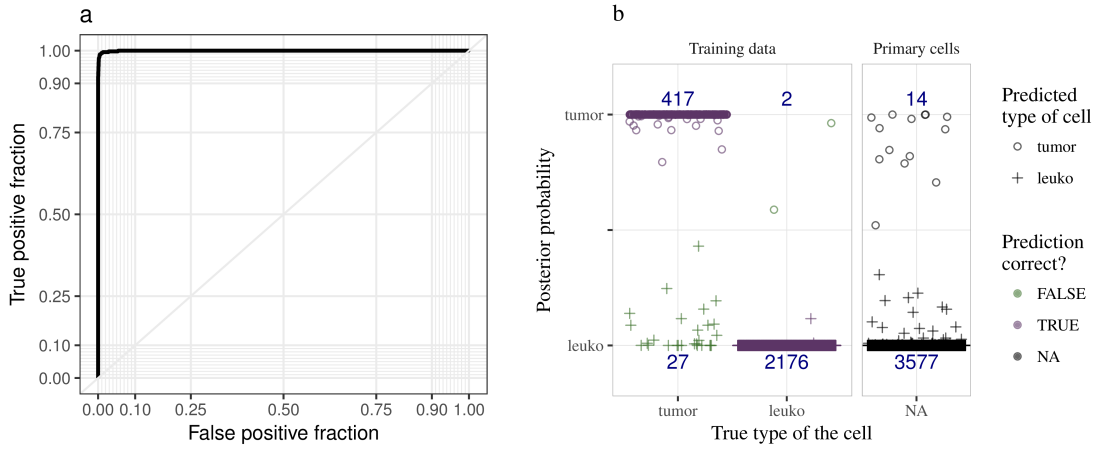


Figure 5.20.: Use of PCA/LDA classification model trained on cultivated cancer cell lines and leukocytes from healthy donors to identify CTCs in patient samples.
The classifier was trained on cell spectra obtained with the RS660 instrument from cell lines A549, 103H, MCF-7, Hep 3B, BT-20, BT-474, HepG2, HL60, M07e, MDA-MB-361, BEWO, SK-GT-2, OCI-AML3, SKOV-3, K562, OVMz-6, leukocyte. 30 principal components were used to train the classifier that was validated with 25-times repeated bootstrapping. Panel a shows the ROC curve of the model, which demonstrates median sensitivity value of 93.71median specificity value of 100The left part of the panel **b** shows the model predictions for the same data that were used for the training, where we see low number of false positives/negatives. The right part contains the prediction results for the spectra of individual blood cells from patients. Unfortunately, the ground truth in this experiment is not available, as the measured cells were neither enriched nor labeled or recognized by any other technique.

median specificity value of 100% (IQR 99.88–100%), as illustrated by the *receiver operating characteristic curve (ROC-curve)* in Figure 5.20.

Then, this classifier was used to detect CTCs from the spectra of individual cells isolated from the blood of oncological patients. In total, 3591 cells were investigated, which originated from 48 individual patients. The cell samples were not processed using any CTC-enrichment technique nor they were labeled by any means, so, unfortunately, no ground truth is available for this experiment.

Our goal was to observe, in which proportions the predicted cell type would be distributed, and whether these proportions make any sense. Our expectation for a reasonable number of detected CTCs was from zero and up to several tens among 3.6 thousand measured cells, as the concentration of CTCs in blood is very low. It turned out, that the classification model recognized only 14 cells to be CTCs, or about 0.53% of the total number of cells, which corresponds very well with the expected concentration of the CTCs.

5.7. Experimental Demonstration of Cell Identification Workflow

On 22-nd February 2017 me and my colleagues successfully demonstrated the most challenging technological steps proposed in the RamanCTC project.

First of all, we demonstrated cell immobilization on a microhole chip, as shown in Figure 5.22. A mixed sample has been prepared, which consisted of *BT-20* breast tumor cells labeled with fluorescent anti-EpCAM markers, and unlabeled leukocytes. This mixture mimicked a patient blood sample coming from the enrichment step.

Using the RS660 instrument, we created a big bright-field mosaic image of the microhole array with cells immobilized on the holes. Then, we calibrated the position of the the microhole array in the data acquisition software and selected the region of interest (ROI). The instrument collected spectra from each microhole in the ROI (16×21 microholes, 336 spectra) in a fully automated fashion and transmitted them to the remote database.

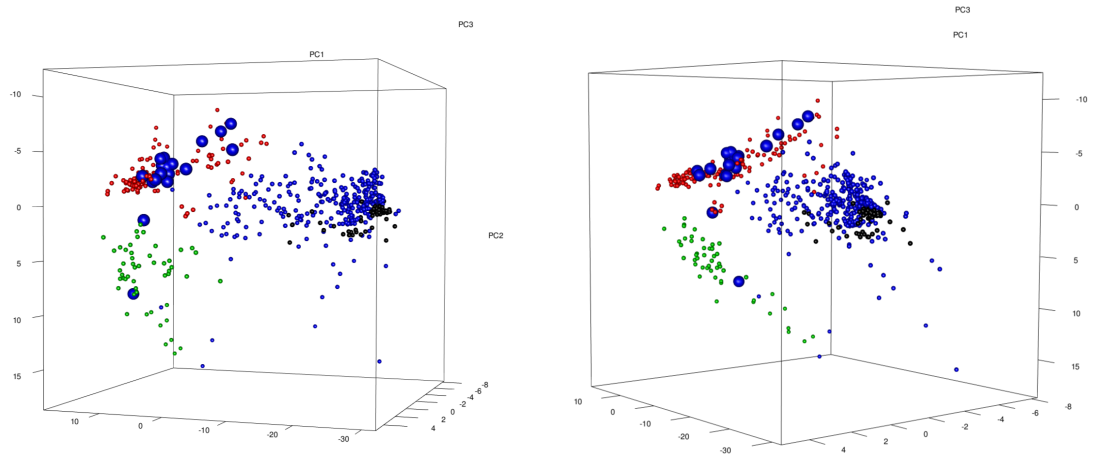


Figure 5.21.: Visualization of the spectral data acquired during the demonstration of the Raman-based CTC identification workflow (see Figure 5.22 for the experiment description). The distribution of points is shown from two different angles. The image shows a projection of the acquired spectra onto the coordinate space formed by PCA latent variables. PCA was calculated using a training dataset with 230 spectra of incubated "BT-20" cells and leukocytes from healthy donors, which were acquired beforehand. **Color coding:** training spectra: *red* – tumor cells, *green* – leukocytes, *black* – empty holes; predicted spectra shown in blue: *points* – leukocytes or empty holes, *spheres* – true tumor cells.

The spectra were imported from the database into the R environment. There, I calculated a projection of the collected spectra into a new 3D space formed by the first three principal component analysis loadings of a bigger training dataset collected and analyzed beforehand. The projected cells from the mixed sample formed three distinct clusters, corresponding to the empty holes, tumor cells and leukocytes (see Figure 5.21). Most of the tumor cells have been correctly projected into the point cloud formed by training spectra of tumor cells. This shows that even with such simple unsupervised technique as PCA one can separate tumor and healthy cells apart based on their spectral differences.

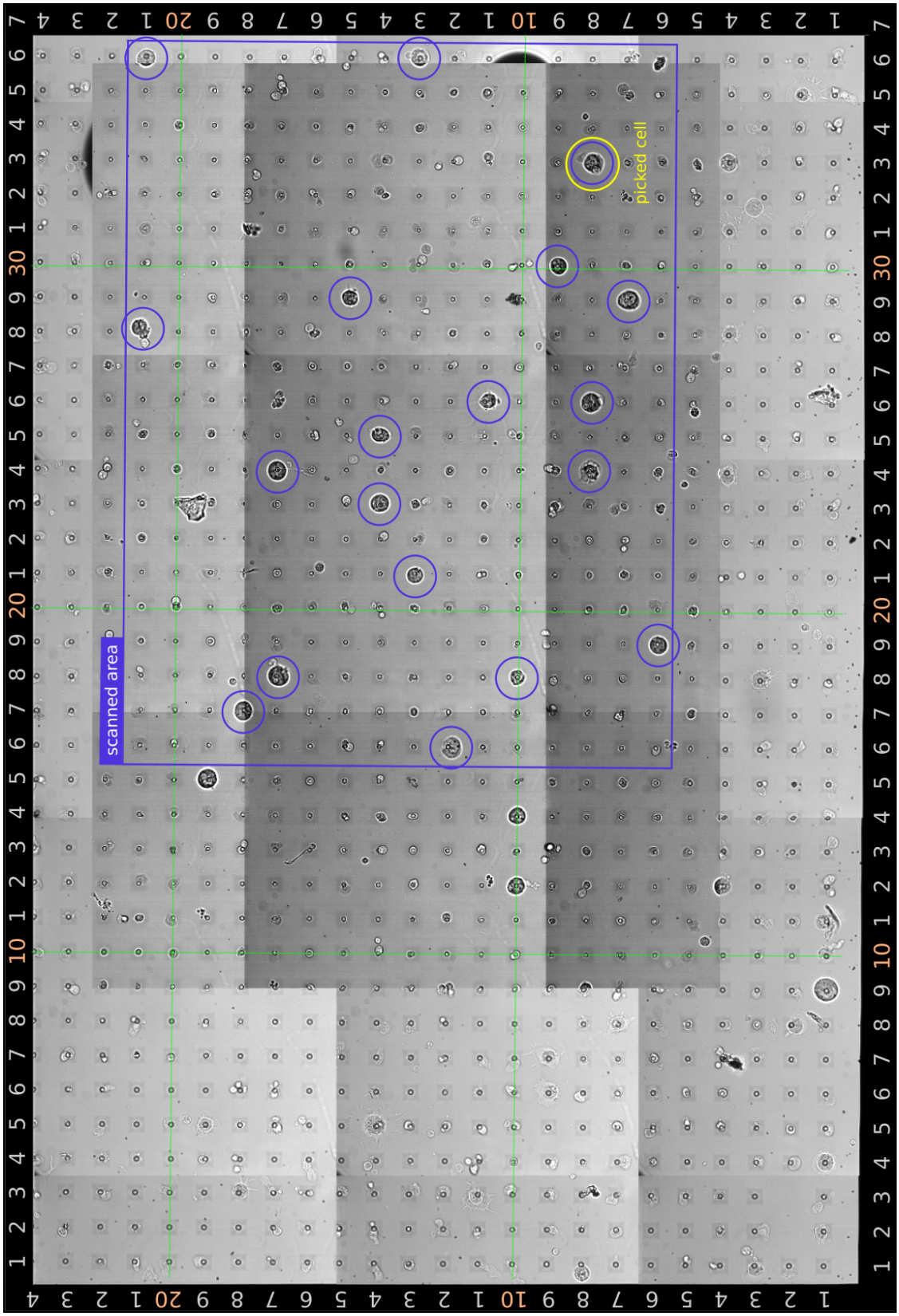


Figure 5.22.: Demonstration of the Raman-based CTC identification workflow, developed during the RamanCTC project. A cell mixture was prepared, consisting of cultivated 'BT-20' cells and leukocytes from healthy donors. The cells were sedimented onto a microhole array chip. The cultivated tumor cells (highlighted with blue circles) have much bigger size than the leukocytes that are barely visible on the membrane. The RS660 instrument automatically acquired spectra from each of the 336 holes in the region of interest (blue rectangle). The yellow circle labels the cell that was successfully picked with a robotic micropipette and transferred into another vessel for a subsequent biochemical analysis (next generation single-cell sequencing).

Finally, the microhole chip was transported to the CellCollector instrument. There, exactly the same ROI was manually found, and one of the cells identified as cancer positive was picked using a motorized micropipette. It was transferred into another test tube and it's genome was analyzed using single-cell PCR.

We used fluorescent labels to have a ground truth for the validation of the Raman-based cell recognition (Note that RS660 instrument does not have a fluorescence modality, but CellCelector does). To the best of our knowledge, used fluorescent labels do not influence Raman spectra and thus should not be able to bias the PCA projection.

This successful experiment showed the feasibility of Raman spectroscopy for cancer diagnostics. The mixed sample served as a surrogate of an enriched blood sample from a patient. It contained a high number of tumor cells alongside with leukocytes, as one would expect from a sensitive, but unspecific enrichment procedure.

5.8. Automation and Data Management Pave the Way Towards Biospectroscopic Clinical Cancer Diagnostics

Complex bio-spectroscopic studies produce a lot of multivariate hierarchical data. A high number of complementary information (e. g. patient data, preparation protocols, sample handling, used equipment and its settings, etc.) is associated with each measurement. A clinical translation makes data management even more challenging, as the number of auxiliary metadata fields further increases, and the mistakes in the data handling are not tolerated.

Therefore, it is crucial to develop, optimize and deploy efficient methods for spectroscopic data organization that allow to properly conduct complex bio-spectroscopic studies. For future clinical applications, the whole workflow has to be made automatic, reliable and fast. Moreover, the collected data have to be analyzed and interpreted in an automated manner, as physicians need answers, not complex entangled data sets.

Still, more instrumental development is necessary to increase the technology readiness level of the clinical Raman spectroscopy. As patient samples are costly and rare, we used artificially-incubated cell lines as surrogate of real CTCs. Unfortunately, this is not exactly the same as real tumor cells. For the validation of this technique and demonstration of its maturity, bigger clinical studies would be required. Only a reliably working and clinically-proven diagnostic instrument is allowed to be integrated into the daily routine of oncological clinics.

Bibliography

1. Cantarero, A. *FP7 CanDo - A step forward in monitoring pancreatic cancer. http://www.fp7cando.eu/* Molecular Science Institut (ICMOL). (2014).
2. Martens, H., Nielsen, J. P. & Engelsen, S. B. Light scattering and light absorbance separated by extended multiplicative signal correction. Application to near-infrared transmission analysis of powder mixtures. *Analytical Chemistry* **75,** 394–404 (2003).
3. Hsieh, H. B. *et al.* High speed detection of circulating tumor cells. *Biosensors and Bioelectronics* **21,** 1893–1899 (2006).
4. Ferlay, J. *et al. GLOBOCAN 2012: Cancer incidence and mortality worldwide: IARC CancerBase no. 11. Lyon, France: International Agency for Research on Cancer* 2013.
5. Bernard, W. & Christopher, P. World cancer report 2014. international agency for research on cancer. *World Health Organization, Lyon, France* (2014).
6. Wu, A., Paganini-Hill, A., Ross, R. & Henderson, B. Alcohol, physical activity and other risk factors for colorectal cancer: a prospective study. *British journal of cancer* **55,** 687–694 (1987).
7. Howe, G. R. *et al.* Dietary factors and risk of breast cancer: combined analysis of 12 case-control studies. *J. Natl. Cancer Inst.* **82,** 561–569 (1990).
8. Miller, A., Howe, G., Jain, M., Craib, K. & Harrison, L. Food items and food groups as risk factors in a case-control study of diet and colo-rectal cancer. *Int. J. Cancer* **32,** 155–161 (1983).
9. Giovannucci, E. *et al.* Physical activity, obesity, and risk for colon cancer and adenoma in men. *Annals of internal medicine* **122.** 01074, 327–334 (1995).
10. Holmes, M. D., Chen, W. Y., Feskanich, D., Kroenke, C. H. & Colditz, G. A. Physical activity and survival after breast cancer diagnosis. *Jama* **293.** 01505, 2479–2486 (2005).
11. Lee, I.-M. Physical activity and cancer prevention – data from epidemiologic studies. *Medicine and science in sports and exercise* **35.** 00505, 1823–1827 (2003).
12. Warburton, D. E., Nicol, C. W. & Bredin, S. S. Health benefits of physical activity: the evidence. *Can. Med. Assoc. J.* **174.** 04899, 801–809 (2006).
13. Begley, C. G. & Ellis, L. M. Drug development: Raise standards for preclinical cancer research. *Nature* **483,** 531–533 (2012).
14. Buchen, L. Cancer: Missing the mark. *Nature* **471,** 428–432 (2011).
15. Gorges, T. M. & Pantel, K. Circulating tumor cells as therapy-related biomarkers in cancer patients. *Cancer Immunol Immunother* **62,** 931–939 (May 2013).
16. Bacac, M. & Stamenkovic, I. Metastatic Cancer Cell. *Annual Review of Pathology: Mechanisms of Disease* **3,** 221–247. ISSN: 1553-4006, 1553-4014 (Feb. 2008).
17. Galler, K., Bräutigam, K., Große, C., Popp, J. & Neugebauer, U. Making a big thing of a small cell – recent advances in single cell analysis. *The Analyst* **139,** 1237. ISSN: 0003-2654, 1364-5528 (2014).
18. Ashworth, T. A case of cancer in which cells similar to those in the tumours were seen in the blood after death. *Aust Med J.* **14,** 146 (1869).
19. Fehm, T. *et al.* Cytogenetic evidence that circulating epithelial cells in patients with carcinoma are malignant. *Clinical Cancer Research* **8,** 2073–2084 (2002).
20. Marrinucci, D. *et al.* Circulating tumor cells from well-differentiated lung adenocarcinoma retain cytomorphologic features of primary tumor type. *Archives of pathology & laboratory medicine* **133,** 1468–1471 (2009).
21. Fidler, I. J. Metastasis: Quantitative Analysis of Distribution and Fate of Tumor Emboli Labeled With 125I-5-Iodo-2□-deoxyuridine 2 3. *Journal of the National Cancer Institute* **45,** 773–782 (1970).

22. Langley, R. R. & Fidler, I. J. The seed and soil hypothesis revisited—The role of tumor-stroma interactions in metastasis to different organs. *Int. J. Cancer* **128,** 2527–2535. ISSN: 1097-0215 (2011).

23. Zhe, X., Cher, M. L. & Bonfil, R. D. Circulating tumor cells: finding the needle in the haystack. *American journal of cancer research* **1,** 740 (2011).

24. Miller, M. C., Doyle, G. V. & Terstappen, L. W. Significance of circulating tumor cells detected by the CellSearch system in patients with metastatic breast colorectal and prostate cancer. *Journal of oncology* **2010** (2009).

25. Ghossein, R. A., Bhattacharya, S. & Rosai, J. Molecular detection of micrometastases and circulating tumor cells in solid tumors. *Clinical Cancer Research* **5,** 1950–1960 (1999).

26. Rosenberg, R. *et al.* Comparison of two density gradient centrifugation systems for the enrichment of disseminated tumor cells in blood. *Cytometry Part A* **49,** 150–158 (2002).

27. Vona, G. *et al.* Isolation by size of epithelial tumor cells: a new method for the immunomorphological and molecular characterization of circulating tumor cells. *The American journal of pathology* **156,** 57–63 (2000).

28. Stathopoulou, A. *et al.* Real-Time Quantification of CK-19 mRNA-Positive Cells in Peripheral Blood of Breast Cancer Patients Using the Lightcycler System. *Clinical Cancer Research* **9,** 5145–5151. ISSN: 1078-0432 (2003).

29. Millner, L. M., Linder, M. W. & Valdes, R. Circulating tumor cells: a review of present methods and the need to identify heterogeneous phenotypes. *Annals of Clinical & Laboratory Science* **43,** 295–304 (2013).

30. Paterlini-Brechot, P. & Benali, N. L. Circulating tumor cells (CTC) detection: clinical impact and future directions. *Cancer letters* **253,** 180–204 (2007).

31. Zwirglmaier, K. Fluorescence in situ hybridisation (FISH)–the next generation. *FEMS microbiology letters* **246,** 151–158. ISSN: 0378-1097 (2 2005).

32. Miltenyi, S., Müller, W., Weichel, W. & Radbruch, A. High gradient magnetic cell separation with MACS. *Cytometry* **11,** 231–238 (1990).

33. Harb, W. *et al.* Mutational analysis of circulating tumor cells using a novel microfluidic collection device and qPCR assay. *Translational oncology* **6,** 528IN1–538 (2013).

34. Kraeft, S.-K. *et al.* Reliable and sensitive identification of occult tumor cells using the improved rare event imaging system. *Clinical Cancer Research* **10,** 3020–3028 (2004).

35. Chen, S. *et al.* Catch and Release: rare cell analysis from a functionalised medical wire. *Scientific Reports* **7** (2017).

36. Scherag, F. D. *et al.* Highly selective capture surfaces on medical wires for fishing tumor cells in whole blood. *Analytical Chemistry* **89,** 1846–1854 (2017).

37. Gorges, T. M. *et al.* Enumeration and molecular characterization of tumor cells in lung cancer patients using a novel in vivo device for capturing circulating tumor cells. *Clinical Cancer Research,* clincanres–1416 (2015).

38. Gossett, D. R. *et al.* Label-free cell separation and sorting in microfluidic systems. *Analytical and bioanalytical chemistry* **397,** 3249–3267 (2010).

39. Fabbri, F. *et al.* Detection and recovery of circulating colon cancer cells using a dielectrophoresis-based device: KRAS mutation status in pure CTCs. *Cancer letters* **335,** 225–231 (2013).

40. Huh, D. *et al.* Gravity-driven microfluidic particle sorting device with hydrodynamic separation amplification. *Analytical chemistry* **79,** 1369–1376 (2007).

41. Hyun, J.-c., Choi, J., Jung, Y.-g. & Yang, S. Microfluidic cell concentrator with a reduced-deviation-flow herringbone structure. *Biomicrofluidics* **11,** 054108 (2017).

42. Seo, J., Lean, M. H. & Kole, A. Membrane-free microfiltration by asymmetric inertial migration. *Applied Physics Letters* **91,** 033901 (2007).

43. Hou, H. W. *et al.* Isolation and retrieval of circulating tumor cells using centrifugal forces. *Scientific reports* **3** (2013).

44. Dochow, S. *et al.* Quartz microfluidic chip for tumour cell identification by Raman spectroscopy in combination with optical traps. *Analytical and Bioanalytical Chemistry* **405,** 2743–2746. ISSN: 1618-2642, 1618-2650 (Jan. 2013).

45. Raman, C. V. A new radiation (1928).

46. Long, D. A. *The Raman effect: a unified treatment of the theory of Raman scattering by molecules* ISBN: 0-471-49028-8 978-0-471-49028-9 (Wiley, Chichester; New York, 2002).

47. Harris, D. C. & Bertolucci, M. D. *Symmetry and spectroscopy: an introduction to vibrational and electronic spectroscopy* (Courier Corporation, 1989).

48. Diem, M. *et al.* Molecular pathology via IR and Raman spectral imaging. *Journal of Biophotonics.* ISSN: 1864-0648 (2013).

49. Eberhardt, K., Stiebing, C., Matthäus, C., Schmitt, M. & Popp, J. Advantages and limitations of Raman spectroscopy for molecular diagnostics: an update. *Expert Review of Molecular Diagnostics,* 1–15. ISSN: 1473-7159, 1744-8352 (Apr. 2015).

50. Kong, K., Kendall, C., Stone, N. & Notingher, I. Raman spectroscopy for medical diagnostics—From in-vitro biofluid assays to in-vivo cancer detection. *Advanced drug delivery reviews* **89,** 121–134 (2015).

51. Assaf, A., Cordella, C. B. Y. & Thouand, G. Raman spectroscopy applied to the horizontal methods ISO 6579:2002 to identify Salmonella spp. in the food industry. *Anal Bioanal Chem* **406,** 4899–4910 (Aug. 2014).

52. Bielecki, C. *et al.* Classification of inflammatory bowel diseases by means of Raman spectroscopic imaging of epithelium cells. *J Biomed Opt* **17,** 076030 (July 2012).

53. Harz, M. *et al.* Direct analysis of clinical relevant single bacterial cells from cerebrospinal fluid during bacterial meningitis by means of micro-Raman spectroscopy. *Journal of biophotonics* **2,** 70–80. ISSN: 1864-0648 (1-2 2009).

54. Harz, M. *et al.* Micro-Raman spectroscopic identification of bacterial cells of the genus Staphylococcus and dependence on their cultivation conditions. *Analyst* **130,** 1543–1550 (Nov. 2005).

55. Harz, M., Rösch, P. & Popp, J. Vibrational spectroscopy-A powerful tool for the rapid identification of microbial cells at the single-cell level. *Cytometry Part A* **75A,** 104–113. ISSN: 15524922, 15524930 (Feb. 2009).

56. Saar, B. G., Contreras-Rojas, L. R., Xie, X. S. & Guy, R. H. Imaging drug delivery to skin with stimulated Raman scattering microscopy. *Molecular pharmaceutics* **8,** 969–975 (2011).

57. Assmann, C. *et al.* Identification of vancomycin interaction with Enterococcus faecalis within 30 min of interaction time using Raman spectroscopy. *Anal Bioanal Chem* **407,** 8343–8352 (Nov. 2015).

58. Hung, K.-K., Stege, U. & Hore, D. K. Ir absorption, raman scattering, and ir-vis sum-frequency generation spectroscopy as quantitative probes of surface structure. *Applied Spectroscopy Reviews* **50,** 351–376 (2015).

59. Mandair, G. S. & Morris, M. D. Contributions of Raman spectroscopy to the understanding of bone strength. *BoneKEy reports* **4** (2015).

60. De Souza, R. A. *et al.* Raman spectroscopy detection of molecular changes associated with two experimental models of osteoarthritis in rats. *Lasers Med Sci* (Aug. 2013).

61. Kallaway, C. *et al.* Advances in the clinical application of Raman spectroscopy for cancer diagnostics. *Photodiagnosis and photodynamic therapy* **10,** 207–219 (2013).

62. Jermyn, M. *et al.* A review of Raman spectroscopy advances with an emphasis on clinical translation challenges in oncology. *Physics in medicine and biology* **61,** R370 (2016).

63. Beleites, C. *et al.* Raman spectroscopic grading of astrocytoma tissues: using soft reference information. *Anal Bioanal Chem* **400,** 2801–2816 (2011).

64. Beljebbar, A., Dukic, S., Amharref, N. & Manfait, M. Ex vivo and in vivo diagnosis of C6 glioblastoma development by Raman spectroscopy coupled to a microprobe. *Anal Bioanal Chem* **398,** 477–487. ISSN: 1618-2650 (1 Sept. 2010).

65. Mavarani, L. *et al.* Spectral histopathology of colon cancer tissue sections by Raman imaging with 532 nm excitation provides label free annotation of lymphocytes, erythrocytes and proliferating nuclei of cancer cells. *Analyst* **138,** 4035–4039 (July 2013).

66. Neugebauer, U., Bocklitz, T., Clement, J. H., Krafft, C. & Popp, J. Towards detection and identification of circulating tumour cells using Raman spectroscopy. *Analyst* **135,** 3178–3182 (Dec. 2010).

67. Latka, I., Dochow, S., Krafft, C., Dietzek, B. & Popp, J. Fiber optic probes for linear and nonlinear Raman applications – Current trends and future development. *Laser & Photonics Reviews,* n/a–n/a. ISSN: 1863-8899 (2013).

68. Dochow, S. *Faser- und chipbasierte Raman-Detektionssysteme für biomedizinische Anwendungen* PhD thesis (Friedrich-Schiller Universität Jena, 2013).

69. Kiselev, R., Schie, I. W., Aškrabić, S., Krafft, C. & Popp, J. Design and first applications of a flexible Raman micro-spectroscopic system for biological imaging. *Biomedical Spectroscopy and Imaging* **5,** 115–127. ISSN: 22128808, 22128794 (Mar. 2016).

70. Kubitscheck, U. *Fluorescence microscopy: from principles to biological applications* (John Wiley & Sons, 2017).

71. Claxton, N. S., Fellers, T. J. & Davidson, M. W. Laser scanning confocal microscopy. *Department of Optical Microscopy and Digital Imaging, Florida State University, Tallahassee, http://www. olympusconfocal. com/theory/LSCMIntro. pdf.* (Visited on 10/07/2015) (2006).

72. McCreery, R. L. *Raman Spectroscopy for Chemical Analysis* ISBN: 978-0-471-23187-5 (John Wiley & Sons, Mar. 2005).

73. De Veld, D. C. *et al.* Clinical study for classification of benign, dysplastic, and malignant oral lesions using autofluorescence spectroscopy. *Journal of biomedical optics* **9,** 940–950 (2004).

74. Aubin, J. E. Autofluorescence of viable cultured mammalian cells. *Journal of Histochemistry & Cytochemistry* **27,** 36–43 (1979).

75. Bigas, M., Cabruja, E., Forest, J. & Salvi, J. Review of CMOS image sensors. *Microelectronics journal* **37,** 433–451 (2006).

76. Litwiller, D. Ccd vs. cmos. *Photonics Spectra* **35,** 154–158 (2001).

77. *EMVA Standard 1288, Standard for Characterization of Image Sensors and Cameras.* European Machine Vision Association, Dec. 2016.

78. Zimmermann, H. K. *Integrated Silicon Optoelectronics* ISBN: 978-3-642-01520-5 3-642-01520-4 978-3-642-01521-2 3-642-01521-2. (Visited on 01/30/2014) (Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 2010).

79. Saleh, B. E. A., Teich, M. C., Saleh, B. E. A. & Teich, M. C. *Fundamentals of Photonics* ISBN: 0-471-83965-5 0-471-21374-8. (Visited on 02/04/2014) (John Wiley & Sons, Inc., New York, USA, Aug. 1991).

80. Gåsvik, K. J. *Optical metrology* ISBN: 0-470-84300-4. (Visited on 02/05/2014) (Wiley. com, 2003).

81. Sibarita, J.-B. Deconvolution microscopy. *Microscopy Techniques,* 1288–1291 (2005).

82. Krauss, S. D. *et al.* Colocalization of fluorescence and Raman microscopic images for the identification of subcellular compartments: a validation study. *Analyst* **140,** 2360–2368 (7 2015).

83. *Shutter Operations for CCD and CMOS Image Sensors* AND9195/D. ON Semiconductor (May 2016).

84. *Rolling Shutter vs. Global Shutter - Technical note* Q Imaging (2014).

85. Schie, I. W., Kiselev, R., Krafft, C. & Popp, J. Rapid acquisition of mean Raman spectra of eukaryotic cells for a robust single cell classification. *The Analyst* **141,** 6387–6395. ISSN: 0003-2654, 1364-5528 (2016).

86. Schie, I. W., Krafft, C. & Popp, J. Cell classification with low-resolution Raman spectroscopy (LRRS). *Journal of Biophotonics.* ISSN: 1864063X. (Visited on 08/18/2016) (Aug. 2016).

87. Dowler, S. W., Takashima, R. & Andrews, M. Reducing the complexity of the N-FINDR algorithm for hyperspectral image analysis. *IEEE Trans Image Process* **22,** 2835–2848 (July 2013).

88. Du, Q., Raksuntorn, N., Younan, N. H. & King, R. L. End-member extraction for hyperspectral image analysis. *Appl. Opt.* **47,** F77–F84 (Oct. 2008).

89. Heintzmann, R. & Sheppard, C. J. The sampling limit in fluorescence microscopy. *Micron* **38,** 145–149 (2007).

90. Neugebauer, U. *et al.* Raman-Spectroscopy Based Cell Identification on a Microhole Array Chip. *Micromachines* **5,** 204–215. ISSN: 2072-666X (Apr. 2014).

91. Dochow, S. *et al.* Raman-on-chip device and detection fibres with fibre Bragg grating for analysis of solutions and particles. *Lab on a Chip* **13,** 1109. ISSN: 1473-0197, 1473-0189 (2013).

92. Dochow, S. *et al.* Tumour cell identification by means of Raman spectroscopy in combination with optical traps and microfluidic environments. *Lab on a Chip* **11,** 1484. ISSN: 1473-0197, 1473-0189 (2011).

93. Guck, J. *et al.* The optical stretcher: a novel laser tool to micromanipulate cells. *Biophysical journal* **81,** 767–784 (2001).

94. Freitag, I. *et al.* Recognition of tumor cells by immuno-SERS-markers in a microfluidic chip at continuous flow. *The Analyst* **141,** 5986–5989. ISSN: 1364-5528 (2016).

95. Freitag, I. *et al.* Preparation and characterization of multicore {SERS} labels by controlled aggregation of gold nanoparticles. *Vibrational Spectroscopy* **60,** 79–84. ISSN: 0924-2031 (2012).

96. Dochow, S., Krafft, C. & Uhlemann, W. *Adjustable receiving device for micro-fluidic chip with optical fiber for optical microscope, has chip shuttle which is arranged between side pieces and spacers by adjustable screw along long sides of side piece on base carrier* DE Patent 102,010,050,679. Mar. 2012.

97. Bruus, H. *Theoretical microfluidics* (Oxford university press Oxford, 2007).

98. Tabeling, P. *Introduction to microfluidics* (Oxford University Press on Demand, 2005).

99. Sutera, S. P. & Skalak, R. The history of Poiseuille's law. *Annual Review of Fluid Mechanics* **25,** 1–20 (1993).

100. Lau, A. Y., Lee, L. P. & Chan, J. W. An integrated optofluidic platform for Raman-activated cell sorting. *Lab Chip* **8,** 1116–1120. ISSN: 1473-0189 (June 2008).

101. Fu, A. Y., Chou, H.-P., Spence, C., Arnold, F. H. & Quake, S. R. An integrated microfabricated cell sorter. *Analytical chemistry* **74,** 2451–2457 (2002).

102. Unger, M. A., Chou, H.-P., Thorsen, T., Scherer, A. & Quake, S. R. Monolithic microfabricated valves and pumps by multilayer soft lithography. *Science* **288,** 113–116 (2000).

103. Ryu, K. S., Shaikh, K., Goluch, E., Fan, Z. & Liu, C. Micro magnetic stir-bar mixer integrated with parylene microfluidic channels. *Lab on a Chip* **4,** 608–613 (2004).

104. Lee, S. H., van Noort, D., Lee, J. Y., Zhang, B.-T. & Park, T. H. Effective mixing in a microfluidic chip using magnetic particles. *Lab on a Chip* **9,** 479–482 (2009).

105. Knuth, D. *Literate Programming* ISBN: 9780937073803 (Cambridge University Press, 1992).

106. Van Rossum, G. & Drake, F. L. *Python language reference manual, release 2.5* ISBN: 978-0-9541617-8-1 (Network Theory Limited, Bristol, United Kingdom, 2006).

107. Summerfield, M. *Rapid GUI programming with python and Qt: the definitive guide to PyQt programming* ISBN: 978-0-13-439333-9 (Prentice Hall, Upper Saddle River, NJ Boston Indianapolis San Francisco New York Toronto Montreal London, 2015).

108. Tedesco, J. M. & Davis, K. L. *Calibration of dispersive Raman process analyzers* in (Feb. 1999), 200–212. (Visited on 09/23/2015).

109. Vincent, L. Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms. *IEEE transactions on image processing* **2,** 176–201 (1993).

110. Beleites, C. & Sergo, V. *hyperSpec: a package to handle hyperspectral data sets in R, [Online] at https://github.com/cbeleites/hyperSpec* R package version 0.99-20180528 (2018).

111. Dougherty, E. R. & Lotufo, R. A. *Hands-on Morphological Image Processing* 00587. ISBN: 978-0-8194-4720-3 (SPIE Press, 2003).

112. Kramida, A., Ralchenko, Y., Reader, J. & NIST ASD Team. *NIST Atomic Spectra Database (version 5.5.1), [Online] at https://physics.nist.gov/asd* (2017).

113. Wehrens, R., Bloemberg, T. & Eilers, P. Fast parametric time warping of peak lists. *Bioinformatics,* btv299. ISSN: 1367-4803, 1460-2059 (May 2015).

114. Bloemberg, T. G. *et al.* Improved parametric time warping for proteomics. *Chemometrics and Intelligent Laboratory Systems* **104,** 65–74 (2010).

115. Eilers, P. H. Parametric time warping. *Analytical chemistry* **76,** 404–411 (2004).

116. Urbanek, S. *jpeg: Read and write JPEG images* R package version 0.1-8 (2014).

117. Pau, G., Fuchs, F., Sklyar, O., Boutros, M. & Huber, W. EBImage—an R package for image processing with applications to cellular phenotypes. *Bioinformatics* **26,** 979–981 (2010).

118. Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S. & Stoica, I. Spark: Cluster computing with working sets. *HotCloud* **10,** 95 (2010).

119. Zaharia, M. *et al. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing* in *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation* (2012), 2–2.

120. Sparks, E. R. *et al. MLI: An API for distributed machine learning* in *Data Mining (ICDM), 2013 IEEE 13th International Conference on* (2013), 1187–1192.

121. Dahms, M. *Untersuchung der Kompatibilität von Fluoreszenzmarkern zum Einzelzellnachweis mittels Raman-Spektroskopie* PhD thesis (FSU Jena, Mar. 2017).

122. Beleites, C. *et al. ASCA for Design of Classifier Training Experiments* in *Analytica Conference* (May 2016).

123. Beleites, C., Neugebauer, U., Bocklitz, T., Krafft, C. & Popp, J. Sample size planning for classification models. *Anal Chim Acta* **760,** 25–33 (2013).

124. Ryabchykov, O. *et al.* Automatization of spike correction in Raman spectra of biological samples. *Chemometrics and Intelligent Laboratory Systems* **155,** 1–6 (2016).

125. Michie, D., Spiegelhalter, D. J. & Taylor, C. C. Machine learning, neural and statistical classification (1994).

126. Kotsiantis, S. B., Zaharakis, I. & Pintelas, P. *Supervised machine learning: A review of classification techniques* 2007.

127. Jain, A. K. Data clustering: 50 years beyond K-means. *Pattern recognition letters* **31,** 651–666 (2010).

128. Achanta, R. *et al.* SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence* **34,** 2274–2282 (2012).

129. Winter, M. E. *N-FINDR: an algorithm for fast autonomous spectral end-member determination in hyperspectral data* in *SPIE's International Symposium on Optical Science, Engineering, and Instrumentation* (International Society for Optics and Photonics, 1999), 266–275. (Visited on 01/05/2016).

130. Nascimento, J. & Dias, J. Vertex component analysis: a fast algorithm to unmix hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing* **43,** 898–910. ISSN: 0196-2892 (Apr. 2005).

131. Busoniu, L., Babuska, R. & De Schutter, B. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, And Cybernetics-Part C: Applications and Reviews, 38 (2), 2008* (2008).

132. Box, G. E. Robustness in the strategy of scientific model building. *Robustness in statistics* **1,** 201–236 (1979).

133. Venables, W. N. & Ripley, B. D. *Modern Applied Statistics with S* Fourth. ISBN 0-387-95457-0 (Springer, New York, 2002).

134. Beleites, C. *Raman-spektroskopische Diagnostik von primären Hirntumoren mit Hilfe weicher chemometrischer Klassifikationsmethoden* PhD thesis (Friedrich-Schiller-Universität Jena, Sept. 2014).

135. *PIXIS 400: 1340 pixel CCD array* Princeton Instruments spectroscopy group (Oct. 2014).

136. *PIXIS 256: 1024 pixel CCD array* Princeton Instruments spectroscopy group (Dec. 2015).

137. *Zyla sCMOS - Speed and Sensitivity for Physical Science Imaging and Spectroscopy* Andor (Sept. 2017).

138. *WinSpec Princeton Instruments Spectroscopic Software, User Manual, Version 2.6 A* Roper Scientific (Roper Scientific. January, Jan. 2012).

139. Wickham, H. *ggplot2: Elegant Graphics for Data Analysis* ISBN: 978-0-387-98140-6 (Springer-Verlag New York, 2009).

140. Xie, Y. in *Implementing Reproducible Computational Research* ISBN 978-1466561595 (Chapman and Hall/CRC, 2014).

A. Quantum efficiency of the used cameras

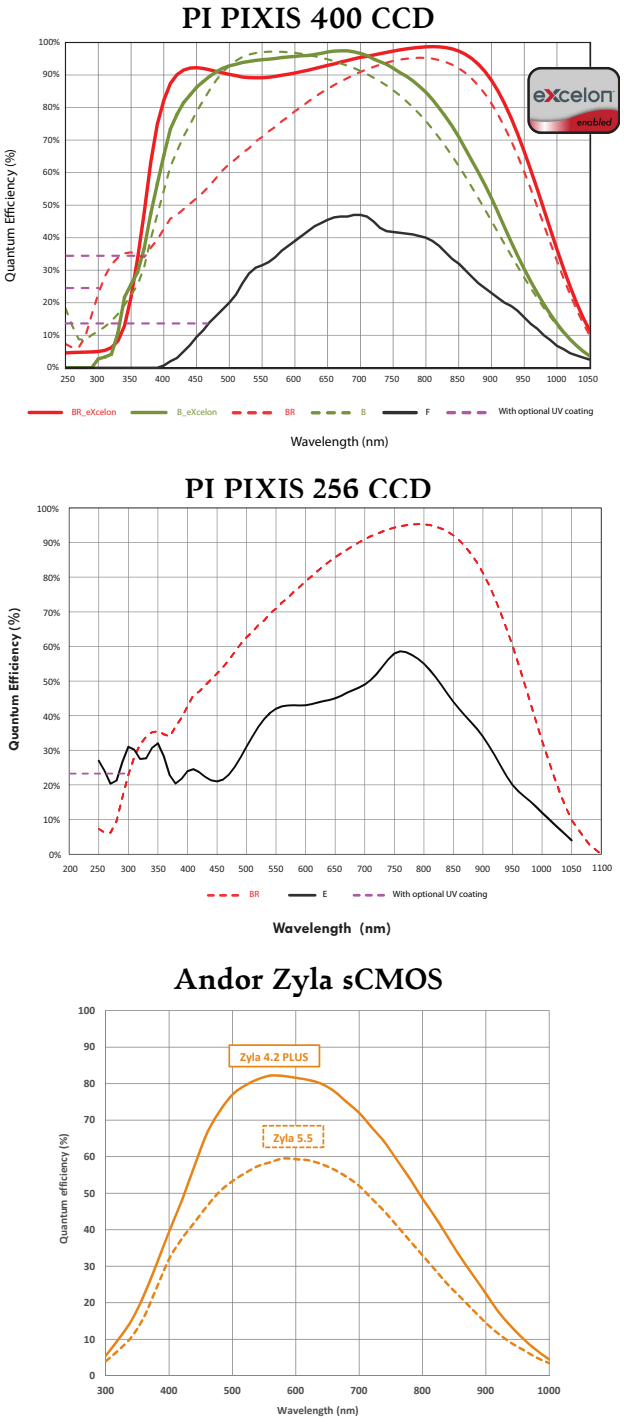


Figure A.1.: QE of detectors used in Raman instruments. Reprinted from specifications [135–137].

B. Fluorescence Attachment for Raman System

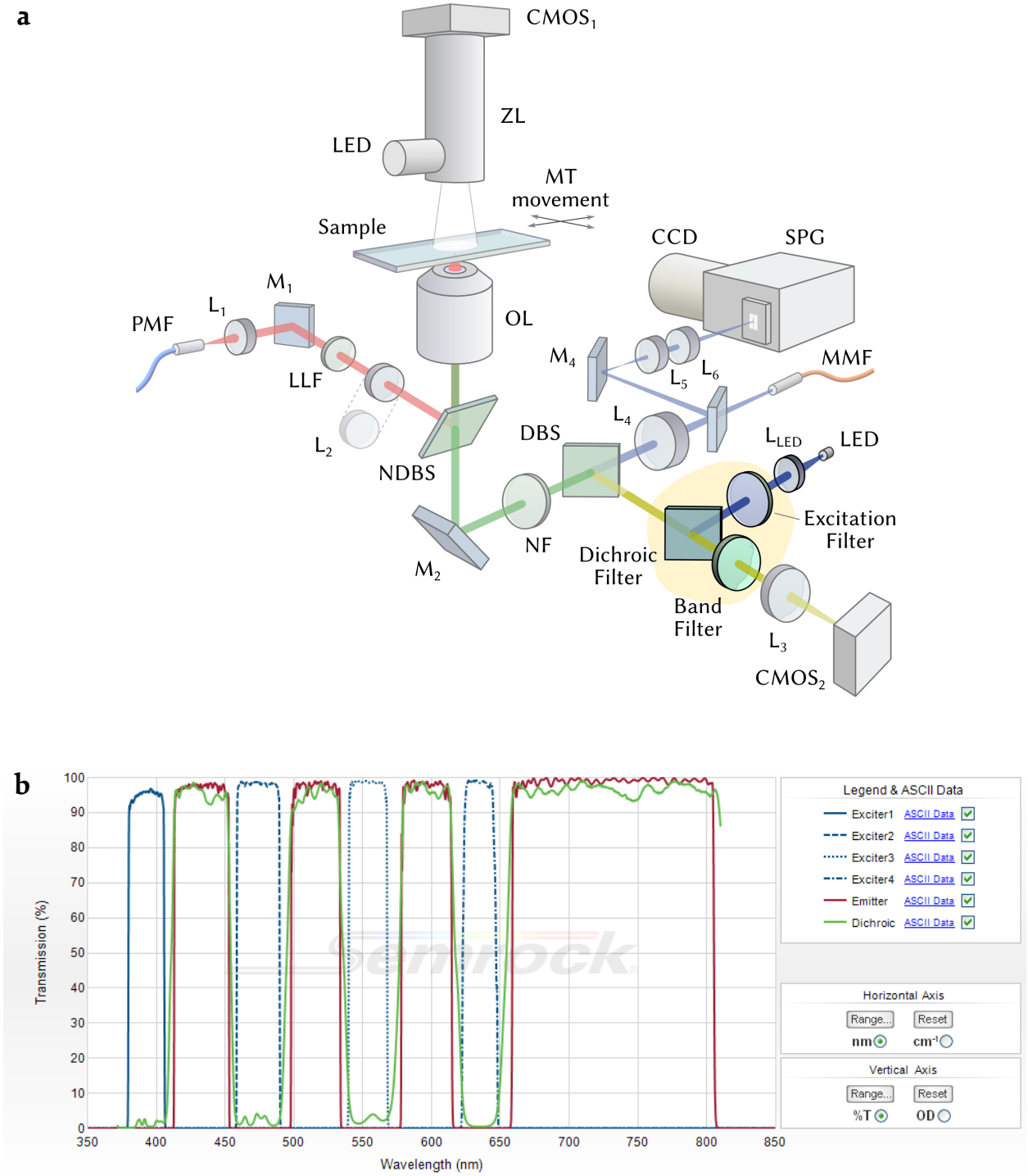


Figure B.1.: Fluorescence wide-field image module for the flexible Raman instrument. **a** – A filter cube, containing a dichroic filter and a band emitter filter, is inserted in the parallel beam path in front of the $CMOS_2$ camera. The dichroic beam splitter is necessary to couple the fluorescence excitation light source into the microscope (LED, L_{LED} and the excitation filter). **b** – Transmission profiles of multi-band emission and dichroic filters, as well as four exciter filters for *"Alexa 488"/"DAPI"*, *"FITC"*, *"Hoecst 33342"/"TRITC"* and *"Alexa 674"/"Cy5"* dyes, correspondingly. Shown here is the filter set Semrock "`LED-DA/FI/TR/Cy5-4X-A-000`", image courtesy Semrock). The Raman excitation laser wavelength is 785 nm, the Raman detection window is 790–1070 nm. For more details see Page 33.

C. Full Database Structure

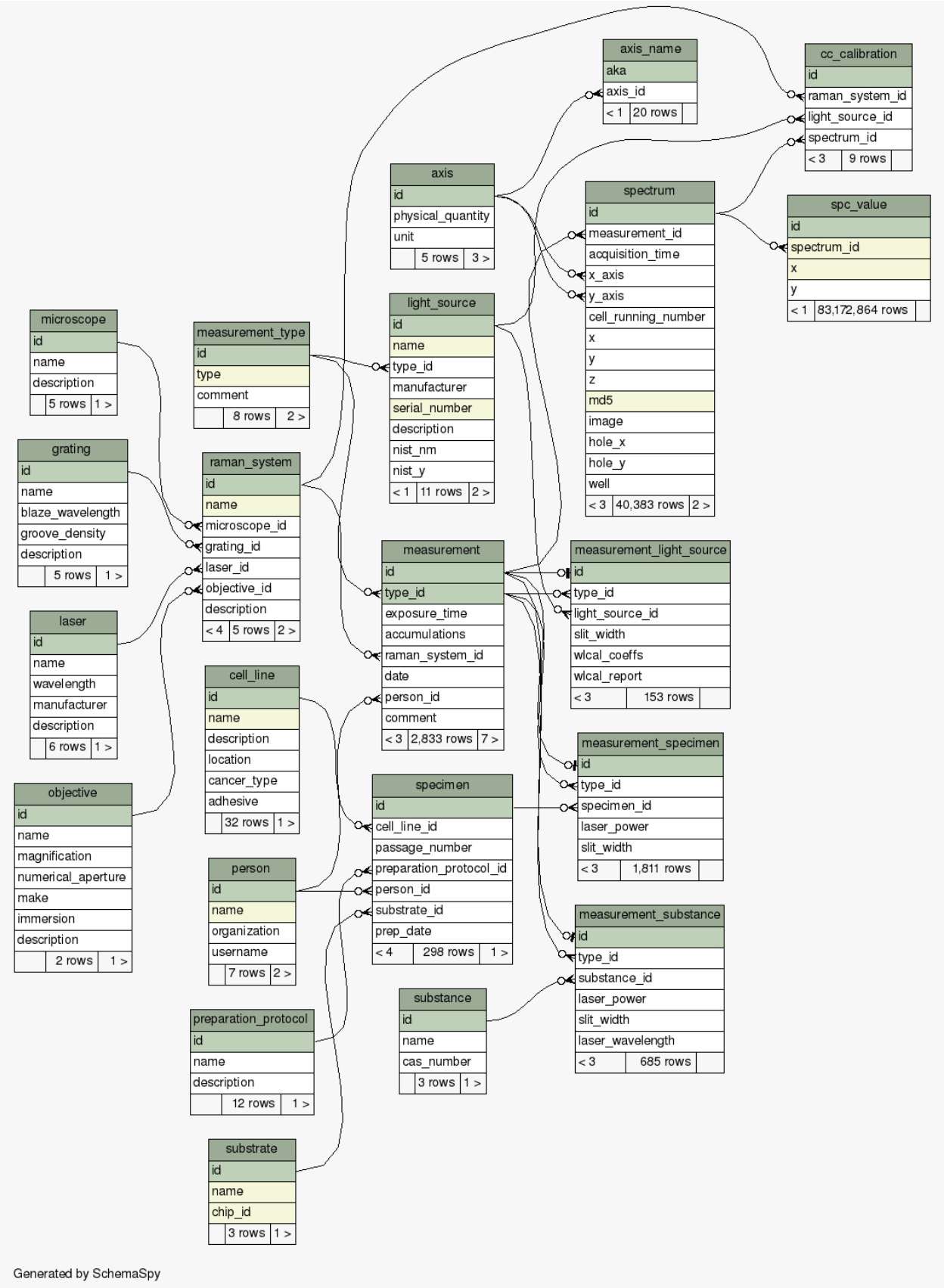


Figure C.1.: Full schema of the database for Raman spectra.

D. Development of import filters for SPE file format

Two Raman instruments that we used in the lab are based on cameras from Princeton Instruments that are controlled by the WinSpec software. Only one file format, called SPE, is supported by the software. Thus, we had to find methods how to read the spectral data contained in these files. SPE is a binary file format that is documented in the *LightField* user manual, available online [138]. The file consists of three parts. The first one is the 4100 bytes long binary header with fields of fixed width that describes the data organization and some experiment conditions. It is followed by a binary block representing the readout (1D for single spectra and 2D for multiple ROI or imaging) at each CCD pixel. The third part is the XML footer containing extended file metadata. The XML footer was added in version 3.0, previous versions of SPE file contain only the header and the data block.

Import into Python.

The instrument supplier did not provide tools for reading of SPE neither for R nor for Python. I could find several different free Python scripts that read SPE files, and used the one called pyWinSpec developed by Anton Loukianov at University of Michigan, available at GitHub. The script reads a given file, extracts all header fields and properly converts them from the binary form to named attributes of a class SpeFile. The spectral data points are returned in a form of numpy array. I embedded the script into our spc2db program (see Section 4.3).

Although this was enough to read the intensity values, the values of the wavelength axis were not available, because they are not stored in the SPE file. Instead, the file header contains the so-called "X Calibration Structure", which states whether the instrument is calibrated and how. I had to develop functions that reconstruct the wavelength axis based on the polynomial coefficients, laser wavelength and x-axis units stored in the aforementioned structure.

Import into hyperSpec.

I could not find any working script for importing of SPE files into R, so I had to develop it ourselves. The script parses the file header according to file specification version 3.0, reads the spectral values and reconstructs the x-axis based on the polynomial coefficients saved in the file. The defined function has been documented and integrated into the source code of hyperSpec[15] under the name read.spe().

The function constructs a single hyperSpec object of all spectra saved in the file, which additionally contains extra data from the SPE file, in particular exposure_sec, LaserWavelen, accumulCount, numFrames, darkSubtracted. The second argument to function allows to specify whether the calibration data saved in the file should be used (xaxis="file") or not (xaxis="px"). This results in a change of the wavelength axis and the x-label of the resulting hyperSpec object, as shown in Figure D.1a–b.

A helper function spe.showcalpoints() reads SPE file, makes a plot of the spectrum and displays which peaks have been used in WinSpec for calibration, as illustrated in Figure D.1c. Ad-

[15] hyperSpec is R package for handling of hyperspectral data [110].

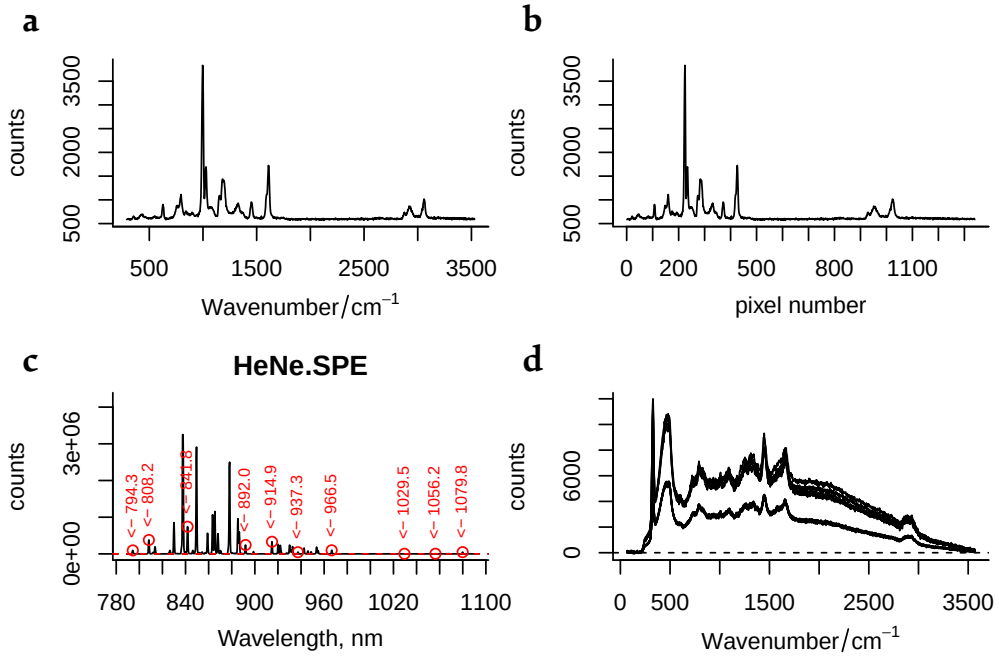


Figure D.1.: Demonstration of `read.spe()` function to load SPE file(s). The code is shown in the corresponding listing. **a** – The function is called with default parameters: the axis labels and the calibration data are loaded from the file itself. **b** – Second argument `"px"` enforces units of the x-axis to be in pixel numbers. **c** – Helper function `spe.showcalpoints()`. **d** – Vectorized version of `read.spe()` used to load many files at once.

ditionally, there are functions `read.spe.header()` and `read.spe.xml()` that can retrieve metadata from the file header and footer, correspondingly.

`read.spe()` can read only a single file. To load multiple files from a folder, it is convenient to get their names using a wildcard, apply `read.spe()` on each of them, then collapse the result into a single `hyperSpec` object. With the package `magrittr` this can be conveniently wrapped in a custom function (see Figure D.1d and the corresponding code listing for an example).

```
read.spe("../data/spe/polystyrene.SPE") %>% plot          # Subfigure (a)
read.spe("../data/spe/polystyrene.SPE", "px") %>% plot    # Subfigure (b)
spe.showcalpoints("../data/spe/HeNe.SPE")                 # Subfigure (c)

# Define a vectorized function
read.spe.files <- . %>% Sys.glob() %>%
  lapply(. %>% read.spe("raman")) %>% collapse

# Use vectorized function to load all leukocyte spectra
read.spe.files("../data/spe/leuko*.SPE") %>% plot()       # Subfigure (d)
```

E. Usage examples of db2spc package

The R package which is responsible for the access to the database with Raman spectra is called db2spc. First of all, it has to be loaded. The db2spc automatically loads hyperSpec, dplyr and magrittr packages.

```
library(db2spc)
```

Before we can access the database, we have to connect to it. This is done using a family of functions use_..._db(), i. e. use_sample_db() to access a sample database shipped with the package (contains 147 spectra), use_sqlite_db() to work with a locally-saved database or use_postgre_db() to work with PostgreSQL. Internally these functions call use_custom_db(), which can also be accessed directly for non-standard cases.

```
use_sqlite_db("../data/raman_ctc_spectra.sqlite")
```

```
Observations: 40,383
Variables: 29
$ id              <int> 35798, 35797, 35796, 35795, 35794, 35793, 35792, 35791, 3579...
$ measurement_id  <int> 3365, 3365, 3365, 3365, 3365, 3365, 3365, 3365, 3365, 3365, ...
$ type            <chr> "raman map", "raman map", "raman map", "raman map", "raman m...
$ local_timestamp <chr> "2017-02-09 15:24:57.093569", "2017-02-09 15:24:34.632457", ...
$ raman_system    <chr> "Raman Reader", "Raman Reader", "Raman Reader", "Raman Reade...
$ cell_line       <chr> "BT-20", "BT-20", "BT-20", "BT-20", "BT-20", "BT-20", "BT-20...
$ substrate       <chr> "microhole chip", "microhole chip", "microhole chip", "micro...
$ passage_number  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
$ cell_number     <dbl> 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0, 91, 75, 64...
$ x               <dbl> 3.050683, 3.010749, 2.970814, 3.050206, 3.010272, 2.970337, ...
$ y               <dbl> 1.655577, 1.656055, 1.656533, 1.615714, 1.616192, 1.616670, ...
$ z               <dbl> 4.398000, 4.398000, 4.398000, 4.398034, 4.398034, 4.398034, ...
$ hole_x          <dbl> 6, 5, 4, 6, 5, 4, 6, 5, 4, 6, 5, 4, 6, 5, 4, 9, 10, 9, 10, 9...
$ hole_y          <dbl> 5, 5, 5, 4, 4, 4, 3, 3, 3, 2, 2, 2, 1, 1, 1, 8, 9, 1, 8, 2, ...
$ well            <chr> "B9", "B9", "B9", "B9", "B9", "B9", "B9", "B9", "B9", "B9", ...
$ prepeared_by    <chr> "Anna Hinze", "Anna Hinze", "Anna Hinze", "Anna Hinze", "Ann...
$ specimen_id     <dbl> 245, 245, 245, 245, 245, 245, 245, 245, 245, 245, 245, 245, ...
$ substance       <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
$ light_source    <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
$ exposure_time   <dbl> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, ...
$ accumulations   <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, ...
$ laser_power     <dbl> 278.22, 278.22, 278.22, 278.22, 278.22, 278.22, 278.22, 278....
$ slit_width      <dbl> 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, ...
$ measured_by     <chr> "Roman Kiselev", "Roman Kiselev", "Roman Kiselev", "Roman Ki...
$ comment         <chr> "Mostly holes with cells. See image 2017.02.09_15.20.53__233...
$ date            <chr> "2017-02-09 00:00:00.000000", "2017-02-09 00:00:00.000000", ...
$ local_time      <chr> "15:24:57.093569", "15:24:34.632457", "15:24:12.233688", "15...
$ x_axis          <chr> "rel. wavenumber", "rel. wavenumber", "rel. wavenumber", "re...
$ y_axis          <chr> "intensity", "intensity", "intensity", "intensity", "intensi...
```

Listing 1: Experimental metadata tracked during Raman experiments, which can be used for filtering and subsetting. Shown here is the output of the command sm() %>% glimpse().

From this moment, we can access metadata of all spectra as `dplyr` tibble using the `sm()` function, which caches the results for a faster retrieval of the metadata.[16] An overview of data returned by `sm()` gives Listing 1.

Search for spectra

It is very convenient to use `dplyr filter()` function to find a spectrum of interest. Let's say, we want to select 100 random spectra of cells whose name starts with `"Hep"`, acquired with 10 seconds exposure time, at least two accumulations, with the `"Raman Reader"` instrument. This search can be done as follows:

```
subset <- sm() %>%
  filter(cell_line %like% "Hep*" &
         exposure_time == 10 &
         accumulations >= 2 &
         date > '2016-09-01' &
         raman_system == "Raman Reader") %>%
  collect() %>%
  sample_n(100)
```

Using the command shown below, it is easy to check how many spectra for each cell line are there. The command result is the Table E.1):

```
subset %>% count(cell_line, type)
```

Table E.1.: Count of cell spectra

cell_line	type	n
Hep 3B	specimen: background	10
Hep 3B	specimen: cell	46
HepG2	specimen: background	4
HepG2	specimen: cell	40

Load spectra from the database into R

Using function `db2spc()`, we get the spectra from the database and convert them into a list of `hyperSpec` objects. The dark frame is already subtracted from these spectra (see Section 4.5 for more details). We collapse the spectra together into a single `hyperSpec` object, so they share the same wavelength axis, and plot them (see Figure E.1).

```
spc@label$.wavelength <- expression(Wavenumber/cm^-1)
spc %>% qplotspc(spc.nmax=50, alpha=0.5) +
  aes(col=type) +
  facet_grid(cell_line ~ .) +
  plot_theme
```

[16] If you don't want to use the cache, because you need a direct access to the database that reflects its state in real-time, use function `spectral_metadata()`.

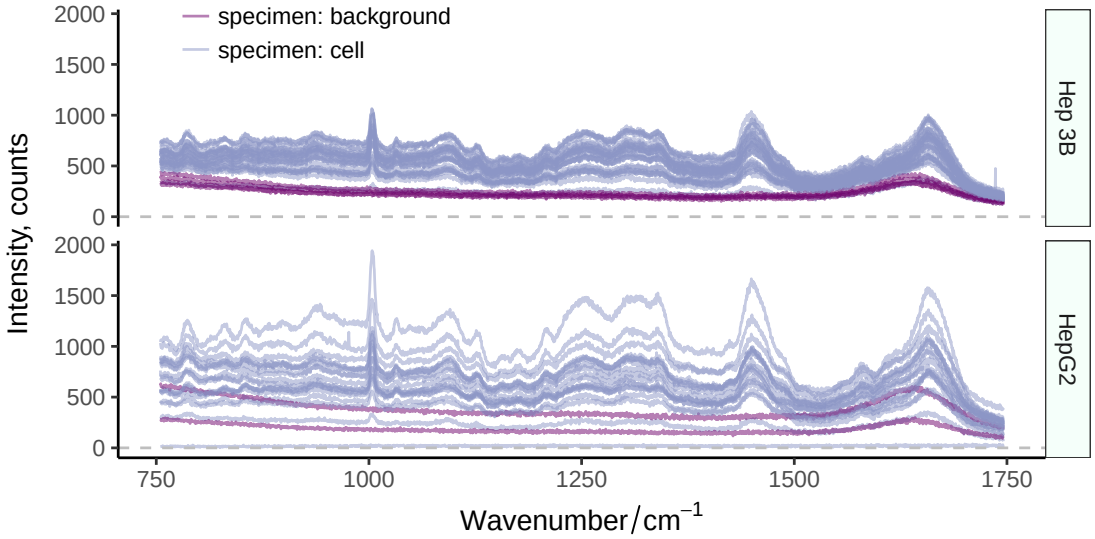


Figure E.1.: Plotting of spectra loaded from a database.

The `db2spc` package modifies the string representation of the `hyperSpec` objects to fit more information onto the screen. Internally this is done using the `dplyr::glimpse()` function. This results in a representation similar to one show in Listing 1.

Convenient filtering functions

There is a set of functions whose name starts with "`f.`" – they are used together with `sm()` to select the spectra of interest and can save a lot of typing. Table Table E.2 lists all of them. See documentation of each specific function for more details and use examples.

```
sm() %>% f.rs(660) %>% f.exp(10) %>% f.type("cell") %>% nrow
```

```
## [1] 9827
```

Table E.2.: Convenience functions for filtering and selection of data from the database of Raman spectra.

Name	Argument type	Description
`f.comment`	String	Filter spectra by a specific comment
`f.date`	Date	Filter by date
`f.exp`	Numeric	Filter by exposure time
`f.hexspot`	Boolean	Filter by hexspot or point acquisition
`f.id` / `f.mid`	Integer	Filter by spectrum id / measurement id
`f.rs`	String or numeric	Filter by Raman system name
`f.type`	String	Filter by measurement type

F. Examples of images stored in the database

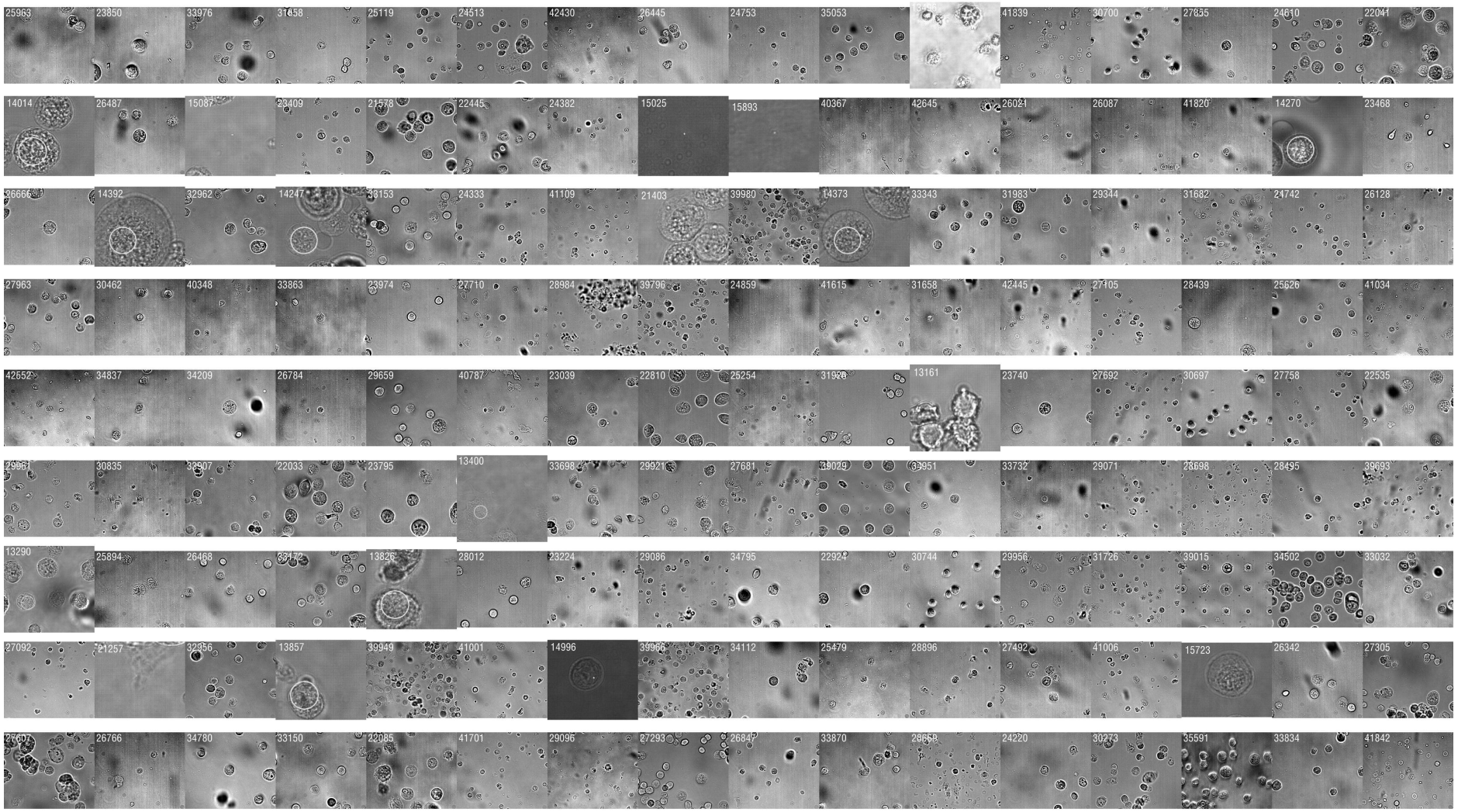


Figure F.1.: 144 randomly selected bright field images from the RamanCTC database. Number in the corner is the `id` of the image and spectrum.

Acknowledgments

This work would not be possible without the support of many people. First of all, I want to express my gratitude to Dr. Claudia Beleites, who not only helped me to enter into the field of spectroscopy and chemometrics, but also was always inspiring me to use new tools that increase my productivity. I owe here the knowledge of the `R` programming language, `Git` version control, and `SQL` databases. She was always ready to give me hints when something gets broken, and to answer even the most weird "why?!" questions.

I am very thankful to Anna Hinze, Jelena Mihailovic, Dr. Merlin-Verena Luetke-Eversloh and Dr. Nadine Rüdiger, who cultivated a tremendous amount of cells for the spectroscopic experiments and acquired thousands of cell spectra. Without their efforts it would not be possible to fill the spectral database with such a diverse dataset.

I acknowledge the invaluable support from Dr. Sebastian Dochow, Dr. Iwan Schie and Dr. Claudia Beleites, who spent a lot of time with me in the lab helping to optimize my experiments. For further assistance I acknowledge Mohamed Hassoun, Dr. Clara Stiebing, Jan Rüger, Dr. Christoph Krafft, Anka Schwuchow, Darya Davydova, and Dr. Katharina Eberhardt. Special thanks to Dr. Stefan Laimgruber who provided an efficient technical support with for the RS660 Raman instrument.

For help with experiment planning, generation of creative of ideas, and lots of fruitful discussions, I would like to thank Dr. Claudia Beleites, Dr. Christoph Krafft, Prof. Dr. Joachim Clement, Dr. Iwan Schie, Dr. Anna Medyukhina, Mohamed Hassoun, and Jan Rüger.

I thank Dr. Christoph Krafft, Prof. Dr. Jürgen Popp, Dr. Anna Medyukhina, Pavel Kliuiev, Dr. Jan Rüger, Dr. Clara Stiebing, and Dr. Tatiana Kirchberger-Tolstik for proof-reading of this thesis and acknowledge their constructive critics.

I am very thankful to my friends and family for their unconditional support, optimism and trust. My colleagues created a very nice and constructive working environment, they helped me to stay motivated and keep on going, and it was a lot of of fun to work together.

The major part of this work was done at Leibniz Institute of Photonic Technologies. I acknowledge support of numerous colleagues with whom I had honor to work together.

This work received funding from *Bundesministerium für Bildung und Forschung* and *European Union's Seventh Framework Programme for research, technological development and demonstration.* I also acknowledge the open-source community that provided beautiful toos for the data analysis. Most of the graphics were created using the `ggplot2` package [139]. The thesis was written using `rmarkdown` and `knitr` – a dynamic report generation tool [140] that keeps the data analysis reproducible.

Selbstständigkeitserklärung

Ich erkläre, dass ich die vorliegende Arbeit selbstständig und unter Verwendung der angegebenen Hilfsmittel, persönlichen Mitteilungen und Quellen angefertigt habe.

Jena, 9. April 2019 _____

Roman Kiselev

Erklärung

Ich erkläre,

- dass mir die geltende Promotionsordnung der Fakultät bekannt ist;
- dass ich die Dissertation selbst angefertigt, keine Textabschnitte eines Dritten oder eigener Prüfungsarbeiten ohne Kennzeichnung übernommen und alle von mir benutzten Hilfsmittel, persönlichen Mitteilungen und Quellen in meiner Arbeit angegeben habe;
- dass mich folgende Personen bei der Auswahl und Auswertung des Materials sowie bei der Herstellung des Manuskripts unterstützt haben: siehe Danksagung in der Dissertation;
- dass die Hilfe eines Promotionsberaters nicht in Anspruch genommen wurde und dass Dritte weder unmittelbar noch mittelbar geldwerte Leistungen von mir für Arbeiten erhalten haben, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen;
- dass ich die Dissertation noch nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht habe;
- dass ich nicht die gleiche, eine in wesentlichen Teilen ähnliche oder eine andere Abhandlung bei einer anderen Hochschule als Dissertation eingereicht habe.

Jena, 9. April 2019 _____

Roman Kiselev