

DOCTORAL THESIS

User-friendly Analysis of Droplet Experiments

Immanuel Sanka

TALLINN UNIVERSITY OF TECHNOLOGY
DOCTORAL THESIS
47/2023

User-friendly Analysis of Droplet Experiments

IMMANUEL SANKA



TALLINN UNIVERSITY OF TECHNOLOGY

School of Science

Department of Chemistry and Biotechnology

This dissertation was accepted for the defence of the degree of Doctor of Philosophy in Gene Technology on 15/09/2023

Supervisor:

Associate Professor Ott Scheler, Ph.D.
Department of Chemistry and Biotechnology
Tallinn University of Technology
Tallinn, Estonia

Co-supervisor:

Associate Professor Olli-Pekka Smolander, Ph.D.
Department of Chemistry and Biotechnology
Tallinn University of Technology
Tallinn, Estonia

Opponents:

Professor (Assistant) Tomasz Kamiński
Institute of Biochemistry
Faculty of Biology
University of Warsaw
Warsaw, Poland

Professor (Associate) Jose Fonseca
Department of Electrical and Computer Engineering
School of Science and Technology
NOVA University Lisbon
Lisbon, Portugal

Defence of the thesis: 27/10/2023, Tallinn

Declaration:

Hereby I declare that this doctoral thesis, my original investigation and achievement, submitted for the doctoral degree at Tallinn University of Technology has not been submitted for doctoral or equivalent academic degree.

Immanuel Sanka

signature



European Union
European Regional
Development Fund



Investing
in your future

Copyright: Immanuel Sanka, 2023

ISSN 2585-6898 (publication)

ISBN 978-9916-80-049-2 (publication)

ISSN 2585-6901 (PDF)

ISBN 978-9916-80-050-8 (PDF)

Printed by Koopia Niini & Rauam

TALLINNA TEHNIKAÜLIKOO
DOKTORITÖÖ
47/2023

Kasutajasõbralik tilga eksperimentide analüüs

IMMANUEL SANKA



Contents

List of publications	6
Author's contributions to the publications.....	7
Introduction	8
Abbreviations	9
1 LITERATURE OVERVIEW	10
1.1 Droplet emulsion	10
1.1.1 Polydisperse droplets by vortexing	10
1.1.2 Droplet generation using a microfluidics setup	11
1.1.3 Applications of droplet emulsions in different types of experiments.....	13
1.2 Droplet detection and content(s) analysis.....	14
1.2.1 Imaging in droplet experiments	15
1.2.2 Image analysis pipeline and workflow	16
1.2.3 Elements of image analysis	17
1.2.4 User-friendly image analysis software	20
1.2.5 Important image analysis steps for droplets and content detection	21
1.2.6 Analytical platforms for droplet data analysis	22
2 AIMS OF THE STUDY	24
3 MATERIALS AND METHODS	25
4 RESULTS AND DISCUSSIONS	26
4.1 Droplet detection is an important step in droplet experiments (Publication II) ...	26
4.1.1 There are many types of software suitable for droplet detection	26
4.1.2 Each type of software performs differently in droplet detection	27
4.2 Droplet classification can be done manually or automatically (Publication I).....	28
4.3 A developed web application (EasyFlow) quickly provides analysis and visualization of droplet experiment data (Publication III)	30
4.4 Analytical pipelines for common droplet experimental scenarios and their applications (Publication III)	31
4.4.1 Different droplet and object detection using user-friendly software	31
4.4.2 Applications of droplet emulsions with user-friendly pipelines.....	34
5 CONCLUSIONS	37
References	38
Acknowledgments.....	50
Abstract.....	51
Lühikokkuvõte	52
Appendix 1	53
Appendix 2	63
Appendix 3	75
Curriculum vitae.....	87
Elulookirjeldus.....	89

List of publications

The list of author's publications, on the basis of which the thesis has been prepared:

- I S. Bartkova, M. Vendelin, **I. Sanka**, P. Pata, O. Scheler. (2020). Droplet image analysis with user-friendly freeware CellProfiler. *Analytical Methods*, 12, 2287–2294
- II **I. Sanka**, S. Bartkova, P. Pata, O.P. Smolander and O. Scheler. (2021). Investigation of Different Free Image Analysis Software for High-Throughput Droplet Detection. *ACS Omega*, 6 (35), 22625–22634
- III I. **Sanka**, S. Bartkova, P. Pata, M. Ernits, M. Meinberg, N. Agu, V. Aruoja, O.-P. Smolander, & O. Scheler. (2023). *User-friendly analysis of droplet array images*. *Analytica Chimica Acta*, 1272: 341397

Author's contributions to the publications

Contributions to the papers in this thesis:

- I The author contributed to pipeline testing, data analysis and article writing.
- II The author initiated the project, designed the experiments, constructed different pipelines, analyzed the data, and wrote the article.
- III The author initiated the project, developed the software, constructed different pipelines, tested the pipelines, analyzed the data, and wrote the article.

Introduction

Droplet emulsions have led to various high-throughput biological and chemical experiments. The droplet emulsion technique divides bulk reactions into droplets with micro- to picolitre liquids in volume. The technique allows researcher to perform massive parallelization without performing tedious work. It requires at least two immiscible liquids, such as water and oil. By adding a surfactant to the liquids, droplets will stay in their proper form without coalescing to each other. Droplets can be generated by using a microfluidic chip and a vortexing method. These methods generate droplets in different sizes, resulting in homogeneous and heterogeneous sizes, respectively. Furthermore, droplet emulsions have improved existing methods: the digital droplet polymerase chain reaction (ddPCR), high-throughput drug screening, molecular detection, etc. Even though the method is robust, it demands sophisticated setups and equipment, and a custom analytical pipeline.

Imaging is one of the most used techniques in scientific laboratories, including for analyzing droplets. Image acquisition is usually performed using brightfield or fluorescence microscopy, where droplets are observed and captured in the form of image data. This data then needs to be processed using image analysis software by calculating the pixels that are captured in the image acquisition process. A pixel is the smallest unit of image data which represents the bit-depth (color information) of the captured object. Pixel manipulation can be performed in image analysis software to find the expected object(s); in this case, the objects are droplets in two-dimensional pixel arrays. Using image analysis software, objects can be determined and measured, including to determine droplets and objects of interest which are encapsulated during droplet generation. The objects which have been studied ranging from genetic materials (e.g., DNA and RNA) to microorganisms (bacteria, microalgae, yeasts, etc.). Though imaging is a common method to retrieve data from experimental results using droplet emulsions, detection requires sophisticated pipelines and often demands programming skills. Yet, this part is not described clearly in published articles. Therefore, this analytical part also creates more obstacles for new researchers or other users who are interested in performing high-throughput experiments using droplet emulsions, especially for users with no image analysis background.

There are different types of user-friendly software available online. Unfortunately, each type of software requires pipeline construction with module exploration, in which the user needs to adjust the modules and make them suitable for their experiments. Based on these problems, analytical tools for droplet-based experiments become important, especially when a new researcher or user does not have the time or resources for a steep learning curve of pipeline development for detecting droplets.

In this thesis, there are four important aims covering 1) the exploration of a droplet detection platform for different droplet experiments, 2) the search for a suitable platform for droplet classification, 3) the simplification of data analysis from droplet array images, and 4) the development of user-friendly analytical pipelines for droplet experiments in different scenarios. Each of the goals involves important steps in which the user does not need to engage in complex exploration to detect droplets or objects of interest. Moreover, full workflows were added and could be used to analyze droplets after acquiring image data.

Abbreviations

AST	Antimicrobial susceptibility test
aTc	Anhydrotetracycline
CP	CellProfiler
CPA	CellProfiler Analyst
CRISPR	Clustered Regularly Interspaced Short Palindromic Repeats
csv	Comma separated value
ddELISA	Digital droplet enzyme-linked immunosorbent assay
ddIA	Digital droplet immunoassay
ddPCR	Digital droplet polymerase chain reaction
DNA	Deoxyribonucleic acid
ESI-MS	Electrospray ionization – mass spectrometry
GABA	Gamma-aminobutyric acid
IJ	ImageJ
Ila	Ilastik
LAMP	Loop mediated isothermal amplification
MALDI-MS	Matrix-assisted laser desorption/ionization – mass spectrometry
MK	Megakaryocytes
NAD	Nicotinamide adenine dinucleotide
QP	QuPath
RNA	Ribonucleic acid
XLSX	Microsoft Excel Open XML Spreadsheet

1 LITERATURE OVERVIEW

Droplet emulsion has become a powerful tool for performing high-throughput experiments and to make it possible to reduce the amount of experimental reaction. By using imaging, a droplet-based experiment can also be performed in scientific laboratories which have microscopes. However, analyzing droplets in image data often requires bioimage analysts or individuals with programming backgrounds. Even though image analysis was developed decades ago, there have been a limited number of studies that involve providing pipelines for analyzing droplet image data.

1.1 Droplet emulsion

Droplet emulsion can be used to divide bulk experiments into micro to nano liter droplets. Droplet emulsion can be generated only by using two or more immiscible liquids which cannot be mixed or dissolve into each other, e.g., water-in-oil or oil-in-water (Binks & Lumsdon, 2000). A surfactant is usually added to help the formation of emulsion. It contains both hydrophobic (water-repelling) and hydrophilic (water-attracting) surfaces (Baret, 2012). The surfactant plays important roles in droplet emulsion generation: stabilizing the interface of the droplet (K. Xu et al., 2017), controlling the droplet size (Fernandez et al., 2004), reducing shear force between droplets (Assadi et al., 2012), etc. Moreover, droplet emulsion can be generated using a simple method, such as vortexing or manual shake, and by using a microfluidics setup (Byrnes, Chang, et al., 2018).

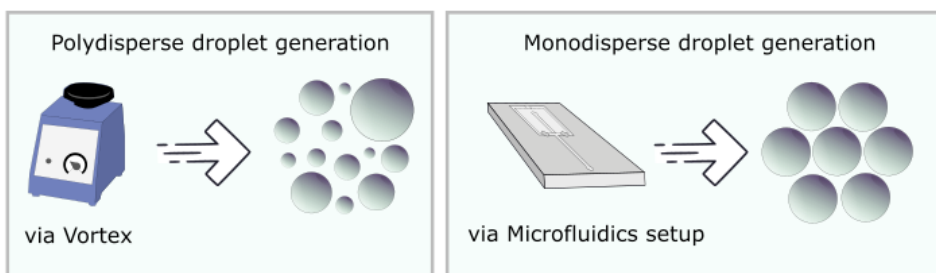


Figure 1. Droplet generation can be performed via vortexing and a microfluidics chip to produce polydisperse and monodisperse droplets, respectively.

1.1.1 Polydisperse droplets by vortexing

The generation of droplet emulsions can be performed by using the vortexing method or manual shake (Devenish et al., 2013). The vigorous mixing of two or more immiscible liquids leads to droplet formation without the need for sophisticated setups. This method has been used to accommodate high-throughput experiments, e.g., a cell-free biological reaction and ribozyme selection (Agresti et al., 2005; Tawfik & Griffiths, 1998). However, this method generates a large variety of droplet sizes. Even though the droplets may vary in size, polydisperse droplets do not require sophisticated setups for their generation.

Polydisperse droplets are still rarely used due to their irregularity in size and randomness. However, some researchers have used polydisperse droplets to perform high-throughput experiments. For instance, Byrnes et al. (Byrnes, Chang, et al., 2018) have used the method to perform a digital droplet polymerase chain reaction (ddPCR) without using sophisticated tools. They also validated their methods by providing a statistical framework which can be used to assess randomness and irregular size

distribution (Byrnes, Phillips, et al., 2018). In a recent study, polydisperse droplets were used to develop the amplification-free detection of viral DNA and RNA (Xue et al., 2023). The method also was utilized to perform an immunoassay (a digital droplet immunoassay – ddIA) experiment for disease identification and monitoring (Byrnes et al., 2020). Although this method does not require sophisticated setups, the droplet analysis requires more attention. For example, an experiment by Chen et al. (Chen et al., 2022) used polydisperse droplets to perform loop-mediated isothermal amplification (LAMP), which required an image analysis with a scripted program to detect and analyze the droplets (Figure 2).

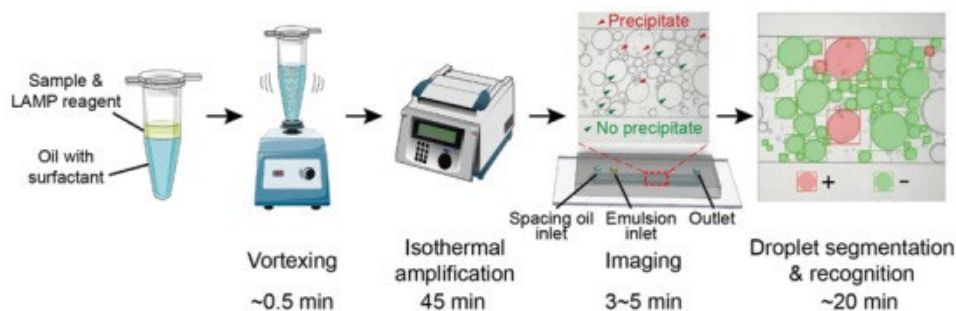


Figure 2. Polydisperse droplets can be generated by vortexing immiscible liquids in a microtube. The example above shows the utilization of polydisperse droplets for isothermal amplification and requires a less sophisticated setup. Reproduce from Chen et al., (2022) under the Creative Commons CC-BY.

1.1.2 Droplet generation using a microfluidics setup

The use of microfluidics for droplet generation has become popular since the method not only generates droplets of homogeneous size but also allows for the integration of different systems, e.g., analytical detection (Y. Zhu & Fang, 2013). The principle of droplet microfluidics comes from employing physical phenomena to control and adjust flow rates and properties of immiscible liquids (Sangam et al., 2020; Srisa-Art et al., 2007). As is shown in Figure 3, droplets can be generated using a microfluidic chip which is adapted e.g., a T-junction, flow focusing, or co-flow feature (Matuła et al., 2020). Each droplet generation using a microfluidics approach relies on fundamental physics and chemistry: fluid dynamics, viscous forces, interfacial tension, continuous phases, channel dimension, etc. (Pang et al., 2020). These variables contribute directly to the droplet's profile e.g., the droplet's size and stability. However, the droplet generation mechanism can be explained as a passive or active method (P. Zhu & Wang, 2016). The passive method requires no external stimuli (e.g., depending on capillary, viscous or inertial forces), while the active method requires additional forces (electrical, magnetic etc.). Therefore, microfluidics can be used to generate a homogeneous size of droplets, usually called monodisperse droplets (P. Zhu & Wang, 2016). In addition to droplet generation, microfluidics has been used for various applications, including for droplet manipulation (merging, splitting, re-loading, incubation, detection, and sorting) (Matuła et al., 2020).

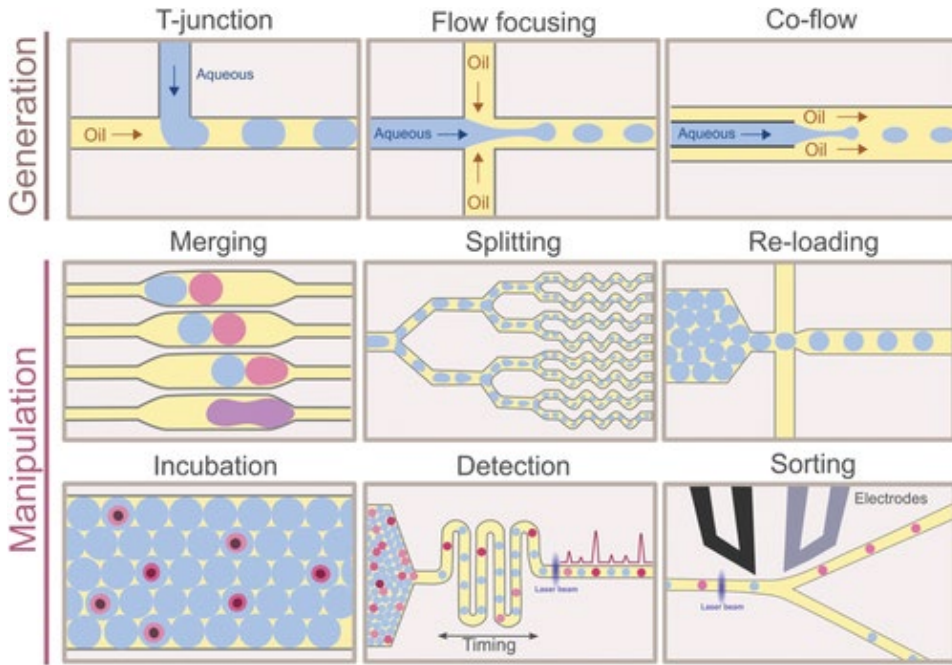


Figure 3. A microfluidic chip can be used to perform droplet generation and manipulation. Each function above requires a specific microfluidic chip and can be integrated into one system. In droplet generation, there are T-junction, flow focusing and co-flow principle which are usually implemented in the microfluidics chip. Reproduced from Matuła et al., (2020) under the Creative Commons CC-BY-NC-ND license.

The microfluidics method provides an important advantage to calculate the probability of content encapsulation. The technique adopts a Poisson distribution, which represents a discrete distribution and quantifies the likelihood of a certain number of occurrences taking place within a defined time frame (Kissell & Poserina, 2017). This distribution can be explained by this formula:

$$p(k, \lambda) = \frac{(\lambda^k e^{-\lambda})}{k!}$$

where k is the number of objects in a droplet and λ (lambda) is the mean or average number of contents per droplet (Collins et al., 2015). For example, Najah et al. (Najah et al., 2012) applied a value of $\lambda = 1$ in the Poisson distribution. Using this, they were able to determine probabilities of 37% for the compartment with zero objects, 37% for the compartment with one object, and 26% for the compartment with more than one object. This provides a control to manipulate the encapsulation which increases the probability of getting only one object per droplet, e.g., by reducing the λ value.

Even though droplet microfluidics provides precise droplet generation, there are some limitations which require more attention. For instance, this technique demands a complex device setup that assures the performance of a droplet microfluidics platform (W. Zheng et al., 2022). Droplet microfluidics also require experts to fabricate a microfluidic chip, set up the equipment, which consists of a syringe, pumps, and a microfluidic chip, and it demands analytical setups to produce results from the experiments (Shang et al., 2017).

1.1.3 Applications of droplet emulsions in different types of experiments

Droplet emulsions have been extensively utilized to perform high-throughput experiments to detect and measure various objects of study, from small molecules to cells (Figure 4). The applications can also be found in studies in different fields: microbiology, biotechnology, chemistry etc. To provide a better understanding of each application, the following paragraphs present detailed examples.

In microbiology, droplet emulsions are used to perform various microbial studies, for instance:

- Antimicrobial activities (Kaminski et al., 2016; Postek & Garstecki, 2022; Ruszczak et al., 2023; Watterson et al., 2020),
- Hetero-resistance in microbes (Scheler et al., 2020),
- Bacterial isolation and screening (F. Xu et al., 2023; Yin et al., 2022),
- Microbiota interactions (Tauzin et al., 2020; Yu et al., 2022), etc.

Applications of droplet-based microfluidics are also widespread in chemistry experiments and some researchers have used them to detect different compounds and molecules, including:

- Dopamine (Alizadeh & Salimi, 2019),
- Ionic analytes (R. Wang et al., 2021),
- Small molecules with biochemical activities (Ha et al., 2021), etc.

Moreover, droplet technologies have been utilized to ease massive parallelization in biotechnology methods, including:

- Different protein detections (Giuffrida et al., 2018; Kebriaei & Basu, 2021),
- Digital droplet enzyme-linked immunosorbent assays (ddELISA) (Cohen et al., 2020; Yi et al., 2022),
- Digital droplet polymerase chain reactions (ddPCR) (B. J. Hindson et al., 2011; C. M. Hindson et al., 2013),
- Single-cell DNA and RNA sequencing methods (Bageritz & Raddi, 2019; De Rop et al., 2022; Lan et al., 2017; Nesterenko et al., 2021; Salomon et al., 2019; Zilionis et al., 2017).

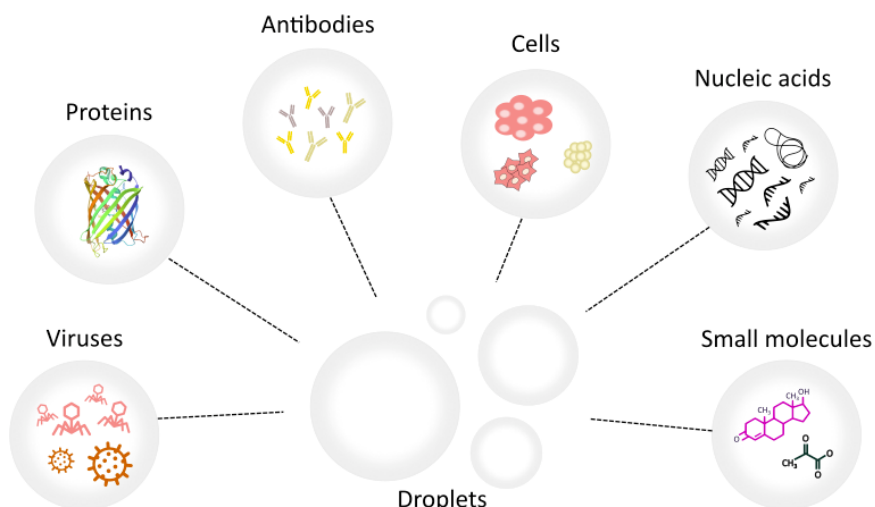


Figure 4. Droplet emulsions are used in various applications, from detecting small molecules to monitoring cell-cell interactions.

In addition to the examples above, droplet-based experiments have also facilitated different studies:

- Mammalian cell observation (De Cesare et al., 2021),
- Lipid production in microalgae (G. Zheng et al., 2022),
- Single-cell metabolomics (Feng et al., 2022),
- Yeast screening methods with incubation and detection technology (Ahmadi et al., 2019),
- Detection of γ -aminobutyric acid (GABA) by integrating droplet microfluidics with matrix-assisted laser desorption/ionization-mass spectrometry (MALDI)-MS and Raman spectroscopy (Bell et al., 2021),
- Biocatalyst analysis using electrospray ionization mass spectrometry (ESI-MS) (Diefenbach et al., 2018).

All of the examples above show the flexibility of the methods, which can be adapted for numerous experimental scenarios.

1.2 Droplet detection and content(s) analysis

There are various detection methods that have been used for different droplet experiments. In general, these detection methods can be classified into two groups: 1) optical detection and 2) non-optical detection. In the optical detection method, the droplet characterization is performed by detecting objects based on their optical properties, including light absorption, scattering, or emission (Zhou et al., 2022). This includes imaging-based detection (Rutkowski et al., 2022), absorbance detection (Medcalf et al., 2023), fluorescence tracing (Vallejo et al., 2019), light scattering (Pacocho et al., 2022), Raman spectroscopy (Yue et al., 2022) and interferometric detection (Zamboni et al., 2021). In non-optical detection methods, the signals emanating from droplets are detected by electrical sensors (Fu et al., 2017), mass property sensors (Kempa et al., 2020), capillary electrophoresis (Liénard-Mayor et al., 2021), or electrochemical reaction sensors (Delahaye et al., 2021). Both methods have advantages and disadvantages (Figure 5). In brief, optical detection requires less sophisticated equipment, versatile, non-destructive, and contact free (Kalantarifard et al., 2018). The use of common equipment, such as microscopy, provides more accessibility than with non-optical methods (Y. Zhu & Fang, 2013). However, the versatility cannot compete with the high specificity which can be achieved by performing non-optical detection, e.g., the use of matrix-assisted laser desorption/ionization-mass spectrometry (MALDI)-MS to detect droplets containing γ -aminobutyric acid (GABA) (Bell et al., 2021). Yet, optical detection is often preferred since the equipment is usually available in biology or chemistry laboratories. Even with the accessibility of detection equipment, analyzing optical detection results requires expert knowledge, e.g., transforming image data into tabular data (shape and size profiles of droplets and their contents) using image analysis.

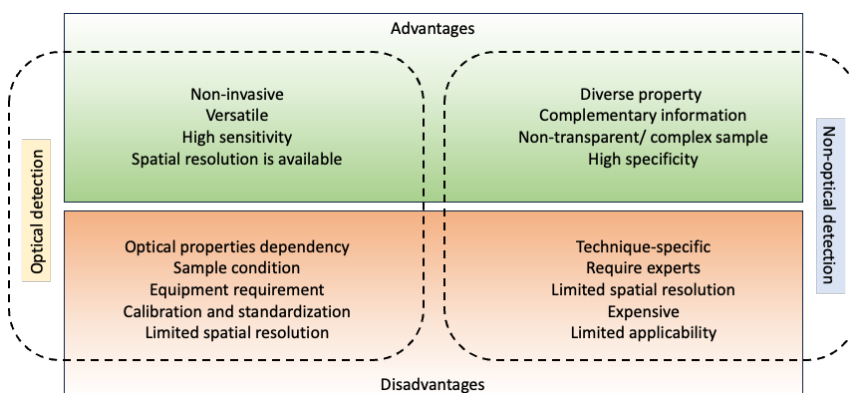


Figure 5. Pros and cons of optical and non-optical detection methods for droplet detection and analysis. In brief, both optical detection and non-optical detection methods provide different purpose and capacity.

1.2.1 Imaging in droplet experiments

One of the optical detection methods, imaging, is known to be the most accessible approach for acquiring experimental data (on droplets and their contents) (Liu & Zhu, 2020). The acquired data is in the form of image data which can be processed later using image analysis software. Therefore, many researchers have used imaging to perform different types of droplet-based experiments. There are various microscopy techniques which are usually used in such experiments, including light and fluorescence microscopies (Saateh et al., 2019) and high-resolution microscopy (e.g., electron microscopy) (Y. Zhu & Fang, 2013). Recently, Szydlowski et al. (Szydlowski et al., 2020) were able to utilize a smartphone device to capture images in a droplet-based experimental study and performed detections. Although data acquisition from the imaging method is accessible, an advanced analytical method is essential and often it demands some programming skills and prior image analysis knowledge.

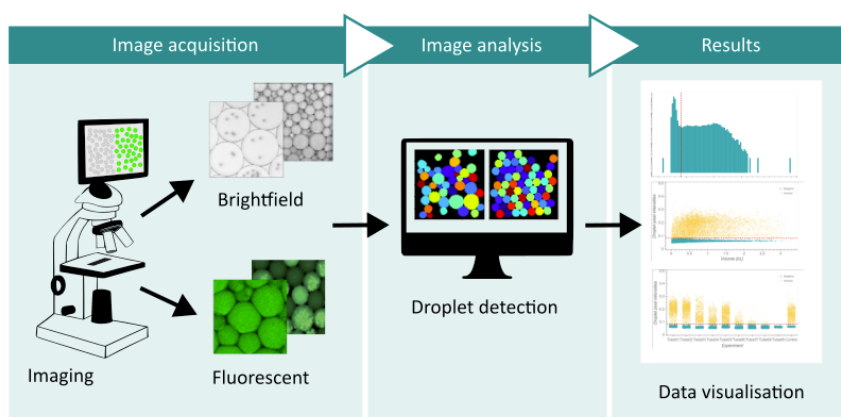


Figure 6. Illustration of droplet imaging and the image analysis workflow in acquiring droplet data and results for data visualization. It starts from image data from microscopy which can result brightfield or fluorescent image data. The data then can be analyzed to find droplets and visualizations can be performed afterward.

1.2.2 Image analysis pipeline and workflow

Image analysis is usually referred to as a method to discover, identify, and understand the patterns in image data (Gonzalez & Woods, 2008). In Miura (Miura, 2020), the image analysis of a biological object can be seen as bioimage analysis, where the purpose of identification was to find biological components and perform measurements in an unbiased way. In a droplet-based experiment, image analysis focuses more on the identification of droplets and objects, which can be biological or other types of samples, e.g., microplastic beads. Image analysis is an important step in acquiring data from images. It typically involves several steps, including acquisition, pre-processing, segmentation, feature extraction, classification, and data interpretation (Hartig, 2013). In Miura and Sladoje (Klemm & Miura, 2022; Miura et al., 2020), the workflow started with biological image data and resulted in numbers, plots, statistics, and visualizations. However, the important part (a pipeline) came from the components in the middle of the workflow (Figure 7). Each component had specific features, command, or algorithm, which could be used to process images or groups of pixels. The pipeline consisted of different components (usually referred as a collection), which are usually readily available in image analysis software.

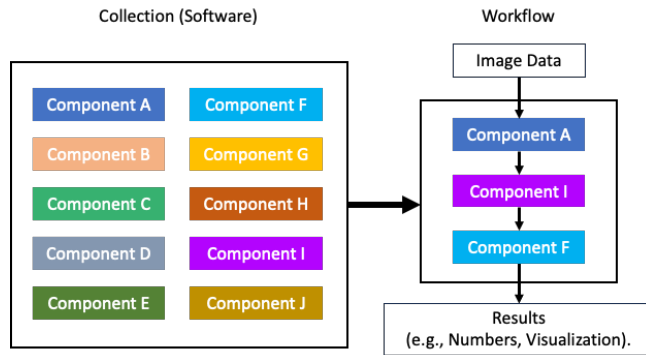


Figure 7. Image analysis workflow comprising component assembly from a software library collection. The workflow encompasses image data retrieval, a pipeline built from diverse components, and anticipated results from the actual object. Adopted with modification from Miura et al. (2020) under the Creative Commons CC-BY.

In a droplet-based experiment, the workflow usually starts with the experiment and type of droplets, which are the objects of interest and provide scientific questions to answer. As mentioned in the previous section, for example, the use of the LAMP method for nucleic acid quantification includes imaging as a data acquisition method (Figure 2). This method then uses a specific image analysis pipeline with a convolutional neural network (CNN) model that detects each droplet and evaluates the detection (Chen et al., 2022). There are some other pipelines that are used to find microbial community interactions across different environments (Hsu et al., 2019) and that assess multi-gene expression in *E. coli* (Sierra et al., 2022). Unfortunately, these pipelines require programming skills in either Python (Python.org, 2023) or MATLAB (MATLAB, 2023).

Table 1 shows various research projects which have utilized different software to detect droplets and objects. Both monodisperse and polydisperse droplets have been used to encapsulate varied objects, e.g., nucleic acids, beads, bacteria, antibodies, and diverse cells. There are various types of software which have been utilized to detect both droplets and objects.

Table 1. Previous research projects that describe the software used for droplet and object detection.

Type of droplets	Objects	Software	References
Monodisperse	DNA/ nucleic acid	Mathematica	(Baccouche et al., 2017; Genot et al., 2016)
	Nicotinamide adenine dinucleotide (NAD), Bacteria, Chinese hamster ovary, K562 cells	LabView and/or MATLAB	(Beneyton et al., 2018; Lyu et al., 2015; Sesen & Whyte, 2020; Vitor et al., 2018; Vo et al., 2017)
	Particle-templated emulsification object, bacteria, beads with fluorescence	ImageJ/ Fiji	(Demaree et al., 2018; Kao et al., 2019; Pan et al., 2015; Rakszewska et al., 2016)
	Oil droplets	CellProfiler	(Scheuble et al., 2017)
	Beads	Python (OpenCV)	(B. Li et al., 2020; Svensson et al., 2018)
	Not available (NA)	C++ (OpenCV)	(Zang et al., 2013)
Polydisperse	Sarcoma condensates, antibodies	ImageJ	(Avni et al., 2022; Byrnes et al., 2020)
	Bacteria, antibodies	MATLAB	(Byrnes et al., 2020; Byrnes, Phillips, et al., 2018)
	DNA, double emulsion droplets	Python	(Chen et al., 2022; Durve et al., 2022)

The table above shows that there are different workflows and methods to detect droplets and their contents (Table 1). Most of the previous studies used script-based software, e.g., C++ (OpenCV- C++, 2023), MATLAB (MATLAB, 2023), Python (Python.org, 2023), and Mathematica (Wolfram Documentation, 2023). There are some user-friendly types of software, such as ImageJ (Schindelin et al., 2015; Schneider et al., 2012), CellProfiler (Lamprecht et al., 2007; McQuin et al., 2018; Stirling, Swain-Bowden, et al., 2021), and LabView (LabVIEW, 2023). However, the documentations of these usages in the research articles are unclear. Moreover, LabView is only available commercially.

1.2.3 Elements of image analysis

There are various image analysis elements which are important in analyzing images. In this section, there are four important elements which are required to bring the foundation to this thesis, including a) pixels and image data, b) filters and segmentation, c) batch processing, and d) feature extraction.

a) Pixels and image data

Imaging method generates image data containing information which usually is detected by a detector in different optical device or microscope (Kuswandi et al., 2007). There are various sensors which can be used to image droplets, including Charge-Coupled Device (CCD) or Complementary Metal Oxide Semiconductor (CMOS) sensors (Pärnamets et al., 2021). From this sensors, droplet images will be translated as pixels, acronym of picture elements, which can be defined as the smallest units in image data (Zhang & Gourley, 2009). Each pixel represents color or bit-depth value (or color information) with coordinates where usually is described as a grid-square with specific color (Bull, 2014; Floyd et al., 1986). It contains bit-depth which varies in each pixel, for example, an 8-bit image has 256 numerical values, ranging from 0 (black) to 255 (white) (Bankhead, 2023). The bit-depth also varies depending on the image acquisition method or sensor that capture the image, e.g., 16-bit with 65536 color variations or 24-bit with 16 millions color variations. In high bit-depth image, some colors are usually used to show the images in a more detailed manner, e.g., 16-bit with 0-255 range values but using three different colors such as red, green, and blue (Tan & Jiang, 2018). This different colors are called as channels where each pixel values represent the intensity of each color channel. To show the number of pixels and its intensity, lookup table (LUT) is usually used (Figure 8). This LUT can also be used to convert an image in different color without changing the pixel value (Bankhead, 2023).

Image data also hosts metadata such as channel, time, space, microscopy details, etc. One example, a format Open Microscopy Environment (OME) Remote Object (OMERO) and Bio-Formats project (OME, 2023). These formats contain multidimensional data which suitable to image analysis software. In addition to these formats, there are various filetypes which are .lif (Leica), .oif (Olympus), .nd and .nd2 (Nikon), .dv (DeltaVision), and .czi or .lsm (Zeiss). There are also generic lossless or lossy image data such as TIFF or JPEG, respectively (Tan & Jiang, 2018).

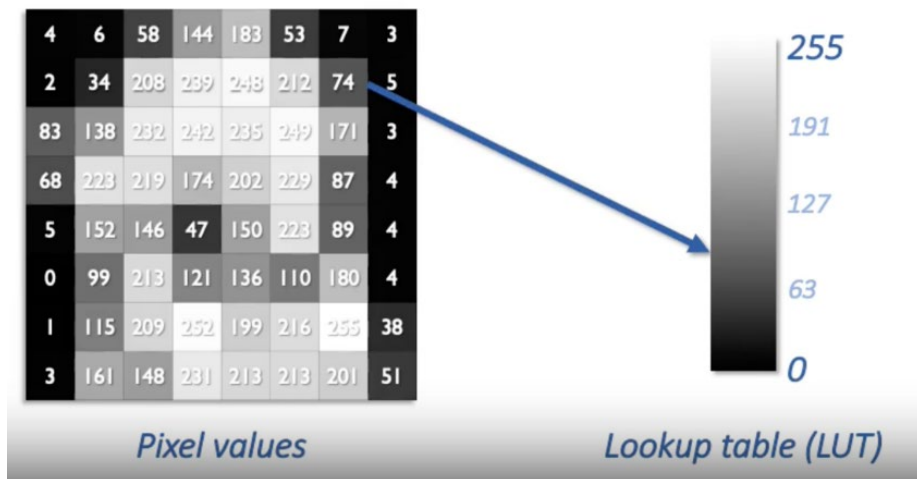


Figure 8. Look up table for greyscale image which has a pixel's value from 0-255. This value may vary depending on the image data. This range is used to describe the 8-bit greyscale image data. In 16-bit greyscale image, the value is ranging from 0-65,535. Reproduced from Bioimagebook by Bankhead (2023) under the Creative Commons CC-BY.

b) Filters and segmentation

As previously explained, components hold an important part in processing image data. Filters and segmentation become essential components in image analysis (Oliveira et al., 2016). These components can help detecting regions or objects with specific criteria, including droplets and objects in it. Filters are commonly used to remove noises (Tan & Jiang, 2018). The filtering components are usually used to enhance or modify images, e.g., noise removal, enhance edges, suppress specific pixels, etc. (Flamary et al., 2012). Some filters are developed by transforming the pixels with specific distribution, for instances, Gaussian filter, median filter, Gabor filter, and mean/average filter (Georgeson & Meese, 1999; Tan & Jiang, 2018). Each of the filter is effective for specific case, for examples, Gabor filters are commonly used for texture analysis and median filters are effective in reducing noise (Britos & Ojeda, 2019; Reiser et al., 2007).

On the other hand, segmentation is more toward a process to partition image into regions or objects (Oliveira et al., 2016; Szeliski, 2011). In general, segmentation method can be classified into two groups, semantic and direct segmentation. In semantic segmentation, identified pixels are assigned into the same generic class while in direct segmentation, each assigned pixels group is defined as specific group (Bazin et al., 2011; Liu et al., 2021). For example, in semantic segmentation, detection of droplets in an image data can be identified of one same circles without considering the different features in each droplet. In direct segmentation, droplets can be grouped into different labels depending on the annotations/ given labels, e.g., droplets with object(s) in it and empty droplets. The segmentation methods can be implemented in various components, e.g., thresholding, watershed, edge detection, etc. As the simplest segmentation method, thresholding is used to classify pixels based on the specific value as the limit between each group (Rogowska, 2000). This thresholding method can be performed to group image as a whole image data (global) or as a local/determined windows (adaptive). Usually, it will be followed by watershed algorithm where the thresholded image will be modeled as a topographic landscape, and the intensity values represent elevations. Watershed is applied to separate the regions and identify the boundaries between objects (Tlig et al., 2020). There is also edge detection which is a technique in image processing that identifies points in a digital image where there are sharp changes in brightness, known as edges or boundaries (Tan & Jiang, 2018).

In thresholding, there are different algorithms which can be implemented to group the pixels based on different calculations, e.g., Otsu, Entropy, Robust Background, Savuola, etc (Lamprecht et al., 2007; Rogowska, 2000). For instance, in Otsu's algorithm, pixels classification where two classes, foreground, and background are separated based on the grayscale intensity values of its pixels (Rogowska, 2000; Xia et al., 2019). Implementation for Otsu's algorithm can be defined the variance between the two classes/ regions, formulated as:

$$\sigma_B^2(k) = \frac{[w_1(k) \times \mu_1(k) - w_2(k) \times \mu_2(k)]^2}{w_1(k) \times w_2(k)}$$

where:

- $\sigma_B^2(k)$ represents the between-class variance at threshold k ,
- $w_1(k) \times w_2(k)$ are the weights of class 1 (foreground) and class 2 (background) at threshold,
- $\mu_1(k)$ and $\mu_2(k)$ are the means of class 1 and class 2 at threshold k respectively, calculated as the cumulative means up to threshold k .

c) Batch processing

Image analysis can accommodate more than single image data. In droplet-based experiment, thousands of droplets need to be captured in different images (Herbert, 2023). Typically, videos are used, and image analysis is performed on each frame to count and obtain information about the droplets (Bull, 2014). To process these images, batch processing is usually required. Batch processing involves handling multiple images simultaneously, streamlining the analysis workflow (Nanes, 2015). Each of image analysis software is equipped to handle a set of images. However, performing this additional process may require additional steps or programming skills. For instance, in ImageJ, batch processing requires a set of macro script which executes the working components and loop the process for multiple images (Klemm & Miura, 2022). Nevertheless, the components can be recorded during the analysis of a single image (Herbert, 2023). Despite the recorded macro, some adjustments are usually necessary before performing the batch processing and this requires steep learning curve.

d) Feature extraction

Feature extraction plays a crucial role in obtaining results from segmentation and object detection. It involves extracting various features commonly utilized in image analysis, such as pixel intensity, size, and shape (Plá et al., 1993). These features serve a significant purpose in determining the distribution of detected objects. For instance, in droplet detection, features like size distribution plays a vital role in assessing the monodispersity in droplet generation (Gawryszewski et al., 2018). These features are derived from the detected pixels that represent the identified object, enabling the determination of average pixel intensity, maximum radius, and area of detection (Z. Li & Liao, 2022). In the context of droplet detection, feature extraction is commonly employed to obtain crucial insights and measurements, facilitating various tasks and analyses (Gawryszewski et al., 2018).

1.2.4 User-friendly image analysis software

In recent developments, different types of image analysis software have been used, including those that are user-friendly and widely accessible. User-friendly software can be defined as a software which emphasize simplicity other than complex system in providing its feature to the user (Omotayo, 1984). This kind of software aims to ease user to access and use the software without a steep learning curve. These types of software are available online and are ready to use, such as ImageJ (Schindelin et al., 2015), CellProfiler (Lamprecht et al., 2007; Stirling, Swain-Bowden, et al., 2021), Ilastik (Berg et al., 2019), QuPath (Bankhead et al., 2017) and Icy (De Chaumont et al., 2012). The types of software are meant to accommodate different kinds of bioimage experiments. For example, CellProfiler can be used to detect and quantify types of cells, including megakaryocytes (MK), during thrompoiesis (Figure 9) (Salzmann et al., 2018), human tumor cells (Elkabets et al., 2011), small-molecule inhibitors of leukemia stem cells (Hartwell et al., 2013), etc. There are some examples of the use of ImageJ for detecting nuclei and prostate cancer cells, which have been extensively described (Hartig, 2013). To accommodate complex bioimage cases, there are some examples where two types of software are needed. Ellen et al. (Dobson et al., 2021) described two different cases of studying cell morphology and migration in time-lapse datasets and whole plate analysis to detect nucleus and whole-cell images.

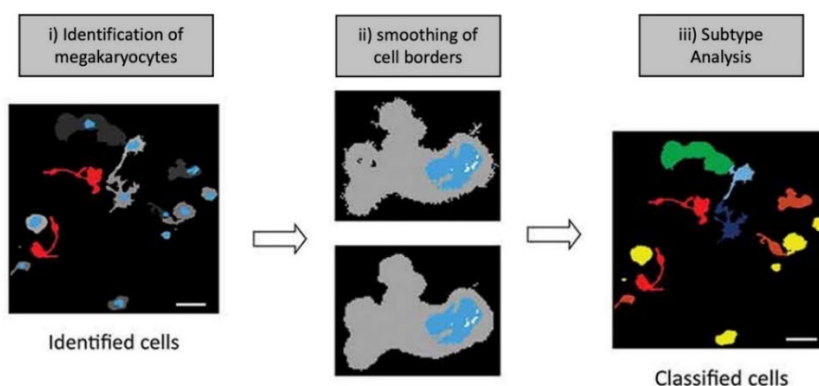


Figure 9. Image analysis pipeline for megakaryocyte (MK) detection using CellProfiler™. The pipeline starts by identifying both nucleated and unnucleated cells. After the identification, borders in the selected object are smoothed. At the end, the objects are classified into different subtypes based on their morphology. Adoption with modification from Salzmann et al. (2018) under the Creative Commons CC-BY.

However, the above-mentioned pipelines and examples showed that there are types of software which are already capable of accommodating different types of analysis, including droplet-based experiments. Recently, some articles have appeared on the use of ImageJ for droplet analysis. As previously mentioned, ImageJ has been used to analyze monodisperse and polydisperse droplets that contain cells (Zielke et al., 2022) and biomolecule reactions (Avni et al., 2022), respectively. However, a detailed description is often not available in the publications, which may lead to confusion among new researchers who are interested in reproducing the workflow. Moreover, droplet detection is usually followed by classification and analysis, and these are rarely described in any workflow. Therefore, there is a need to provide such a workflow and details.

1.2.5 Important image analysis steps for droplets and content detection

In principle, droplet and content detections, also called object detections, in image data require a selection of specific pixels that differ from the background pixels. To retrieve the correct detections, image data are usually reviewed through different steps, including pre-processing, processing, and post-processing.

Pre-processing is the initial step in object detection and involves enhancing the quality of the image to improve the accuracy of detection. This may include such operations as noise reduction and geometric transformation (Fan et al., 2019; Sonka et al., 1993). Pre-processing aims to reduce problems with image data, including noise reduction, optimized contrast, and brightness, which may hinder detection (Park et al., 2021).

In the processing step, images are analyzed to detect and classify objects of interest. This includes various techniques, including thresholding, implementation of watershed and segmentation algorithms, where pixels are partitioned and grouped based on color, intensity, or texture (Liang et al., 2014). The aim of the processing step is to detect objects of interest and retrieve information on the objects (Uchida, 2013). The extracted information includes shape, size, and the object's behavior. For instance, images can be divided into two or more groups, e.g., objects as "foreground" and a microscope's slide as "background" (Hartig, 2013). In this thesis, the objects of interests are droplets and encapsulated objects.

Post-processing is the final step in object detection and involves refining the results obtained from the detection algorithm. This may include filtering out false positives, grouping or merging detections, and tracking objects across frames in a video sequence (Sabater et al., 2020). Post-processing techniques aim to improve the accuracy and reliability of object detection results (Savargaonkar et al., 2021). Moreover, this step is usually added to standardize outputs, including as image data and interpreting results (Schulz-Menger et al., 2020).

Overall, object detection in image analysis involves a combination of processing, pre-processing, and post-processing steps to accurately detect and classify objects within an image or video sequence. These steps are essential for achieving robust and efficient object detection results.

1.2.6 Analytical platforms for droplet data analysis

A droplet-based experiment generates large amounts of data that require data processing and visualization. Using an imaging approach, droplet experimental results are presented as pixel profiles from image analysis software. In an example from a CellProfiler Analyst (CPA) (Dao et al., 2016; Jones et al., 2008), it can provide data exploration and visualization to gain insights regarding the processed data from CellProfiler (McQuin et al., 2018). This software is also equipped with a classifier to train a machine-learning model from a detected object (Jones et al., 2009). In the newest version of CPA, the software is also equipped with a dimensional reduction tool to minimize non-significant dimensions or features in the data set (Stirling, Carpenter, et al., 2021). Since a study of droplet-based experiments using a CellProfiler does not exist, the use of this software remains untested. Furthermore, the software is only able to find the input of property files available from a CellProfiler.

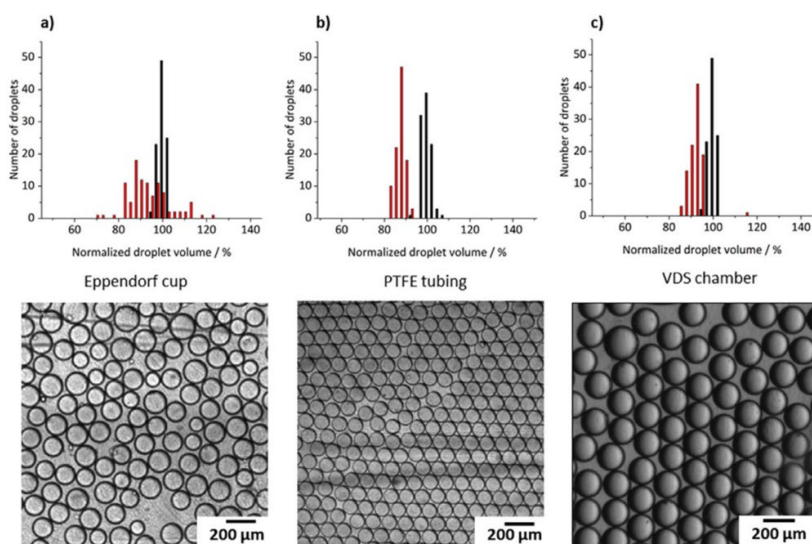


Figure 10. Examples of droplet analysis from image data which was performed to show the effect of droplet storage in droplet's size variation. Each of the droplet image is visualized as a normalized droplet volume in percentage (%) histogram that indicates droplets after generation (black bars) in comparison to after a transfer process (red bars). This figure is reproduced from Grösche et al. (2019) under copyright 2023 by the publisher. Used with permission.

Image analysis software generates pixel data which can be translated into different variables, including shapes and sizes. In droplet analysis, there are different important variables which are usually shown in a publication, e.g., volume distribution, coefficient variations and relative signal distribution. As shown in Figure 10, Grösche et al. (2019) used size distribution to assess the effects of different storage methods. In Scheler et al. (2020), droplet data are shown to find the signal distribution among droplets, including in different effects of serial antibiotic concentration.

As mentioned previously, this data processing is mostly performed individually and there is no standard in assessing droplet distribution with a user-friendly tool from image analysis results, especially without the need for programming.

2 AIMS OF THE STUDY

The general goal of this study was to develop user-friendly and low-learning-curve tools for analyzing droplet arrays that do not require programming skills of the end user. To reach the goal, there were four specific aims:

- Aim 1 Provide a droplet detection platform for different droplet experiments.
- Aim 2 Find a suitable platform for droplet classification which is needed to group droplets in the experiment.
- Aim 3 Simplify the data analysis of droplet array images, which usually requires an image analyst and programming skills.
- Aim 4 Develop user-friendly analytical pipelines for different droplet experiment scenarios.

3 MATERIALS AND METHODS

The materials and methods which are used in this thesis are explained in detail in the published articles. This section will mention only a key message for each step necessary to follow the thesis.

Publication I

- A CellProfiler™ pipeline was developed and applied to detect droplets in 64 images, which were acquired using a fluorescence wide-field microscope.
- The distinction between empty droplets and droplets containing bacteria was determined using a CellProfiler Analyst (CPA) and a database generated by the CellProfiler™ pipeline.

Publication II

- User-friendly types of software were explored through online search and manual tests to perform droplet detection.
- Pipelines were generated using the four most popular types of software determined by a Scopus search and Twitter assessment (ImageJ, CellProfiler, Ilastik, and QuPath).
- Pipeline comparisons were carried out to compare the different logics of each software.
- Accuracy and precision were performed to assess the capability of the four compared types of software in detecting droplets.
- In addition, software processing time was examined to determine detection efficiency.

Publication III

- Detection of polydisperse droplets and fluorescent objects inside droplets were accomplished by using CellProfiler™. This included the measurement of each object.
- Detection of monodisperse droplets and fluorescence objects in were carried out with CellProfiler™. This pipeline has additional modules to enhance detection and to build a relationship between droplets and objects.
- In brightfield images, we used a combination of Ilastik and CellProfiler™ to detect object(s) in droplets which do not have fluorescence capability (e.g., microplastic beads). CellProfiler™ was used to analyze and obtain the relationship data and their pixel profiles.
- EasyFlow is able to visualize droplet signals and size distributions, the relationship between droplet signals and sizes, and to describe the experimental conditions based on signal distribution. EasyFlow was invented to simplify the data processing and visualization of droplet image analysis, which was generated by image analysis software as comma separated value (.csv).
- EasyFlow was developed in Python using a Streamlit framework and important libraries, such as Pandas, NumPy, Matplotlib and Bokeh.
- We have provided examples of full detection and analysis pipelines for different scenarios by providing images as input data and visualizations with necessary tables and statistics as output.

4 RESULTS AND DISCUSSION

4.1 Droplet detection is an important step in droplet experiments (Publication II)

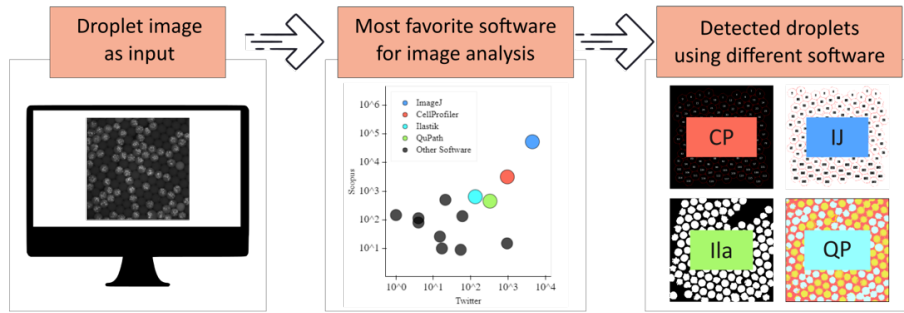
This section addresses the need to provide a droplet detection platform for different droplet experiments.

4.1.1 There are many types of software suitable for droplet detection

As mentioned in the literature section of the image analysis pipeline and workflow, image analysis software can be defined as a collection (Miura, 2020). The collection or types of software have been developed for different purposes, and searching for the right software for detecting objects, including droplets, requires a steep learning curve and some exploration, especially with no prior image analysis knowledge or programming skills. There are numerous types of image analysis software available online. To evaluate the most discussed software in available publications, Twitter and Scopus keyword search was performed (on February 11, 2021). Based on the results, the most discussed types of software used for image analysis are ImageJ, CellProfiler, Ilastik, and QuPath. These four types of software are able to detect droplets (Figure 11A). The pipelines from each type of software were also identified and showed the key variables that distinguish each step of droplet detection. The variables are grouped into three levels of abstraction that cover the fundamental principle of each software workflow (Figure 11B). The detailed abstractions for each type of software are explained in the figure below and in Publication II (Sanka et al., 2021).

The use of three-level abstractions provides a clearer understanding of each component in the software which needs to be compiled as a pipeline. For instance, in CellProfiler (CP with red box – Figure 11B), to detect droplets in fluorescence images, the pre-processing steps require image data preparation that includes image upload, metadata detection, name types which will be used in the pipeline, and groups of images. The processing step usually involves pixel intensities, texture, edges, etc. (Liang et al., 2014; Uchida, 2013). In CP, the module only requires an IdentifyPrimaryObject module, which contains different components, including thresholding, smoothing, segmentation, and automatic selection. For post-processing, there are four steps used to display data as images and documents (spreadsheet in .csv format). These include “OverlayOutlines”, “OverlayObject”, “DisplayDataOnImage”, and “ExportToSpreadsheet” modules. Further explanation regarding other types of software is provided both in the publication and in the Supporting Information (Sanka et al., 2021).

A) Droplet detection comparison using the most favorite software for image analysis



B) Image-based droplet detection pipeline has three degrees of abstraction

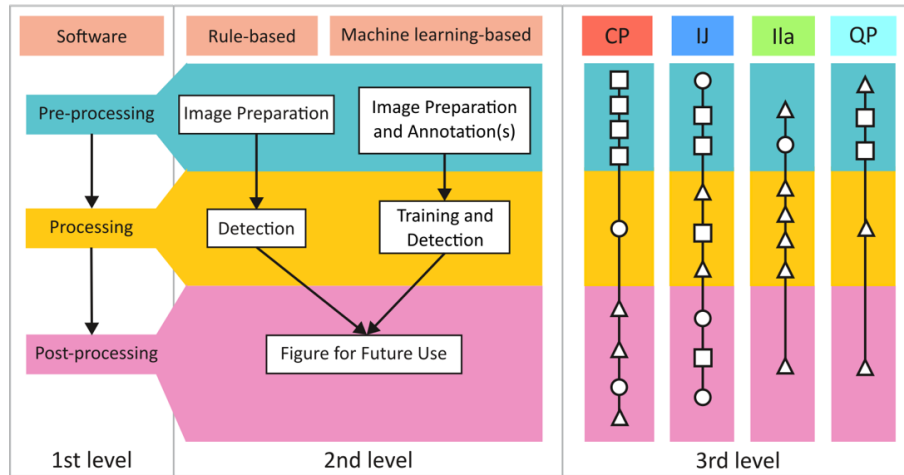


Figure 11. Diagram illustrating droplet detection in single image data. A) Using the most popular software based on Scopus and Twitter, we were able to identify droplets in a single fluorescence image (CellProfiler-CP, ImageJ-IJ, Ilastik-Ila, and QuPath-QP). B) In principle, each type of software shares an identical logic, including pre-processing, processing, and post-processing. The second level contains two categories: rule-based, where users specify some parameters (e.g., size and threshold value), and machine learning-based, where users classify/annotate groups of pixels in an image. The third level describes the steps that must be taken in each type of software. Rectangles, circles, and triangles are used to identify modules with one, one to eight, and more than eight alternatives, respectively (Publication II - Sanka et al., 2021).

4.1.2 Each type of software performs differently in droplet detection

Each type of software performs differently in terms of accuracy, precision, and processing time. To show the capabilities in detecting droplets, accuracy and precision tests were performed by comparing manual counts as a “ground truth” with each detection from each type of software. In the experiment, sensitivity and specificity tests using True Positive – positive droplets (TP), False Positive – false droplet detection/underestimation (FP), and False Negative – software could not detect droplets/overestimation (FN) were used (Gaddis & Gaddis, 1990). Using these calculations, precision and accuracy were determined, explaining the ratio of correct detection among the total and the probability of recreating correct detection, respectively (Bland & Altman, 1995, 1999). Figure 12A

shows that CellProfiler™ had the highest accuracy and precision in detecting droplets. Moreover, counting the errors in each type of software showed that both rule-based types of software (CP and ImageJ) produced fewer errors than the machine-learning based types of software (Ilastik and QuPath) (Figure 12B). The high number of errors mostly involved filtering the droplets that touched borders and wrong segmentation that resulted in joint droplets (Sanka et al., 2021). The performance of each type of software was also assessed in terms of processing time (Figure 12C). Based on the processing time measurement, in general Java-based software (CellProfiler and Ilastik) performed almost 10-20 times faster than Python-based software (QuPath and ImageJ). This processing time experiment gave unexpected results since we expected that machine-learning based software would provide longer processing times due to training and feature implementation (Frank et al., 2020). However, we concluded that the processing time was longer because of the programming language used to make the software. This was supported by a previous study which compared Java-based with other types of software, including Python-based software (Fourment & Gillings, 2008).

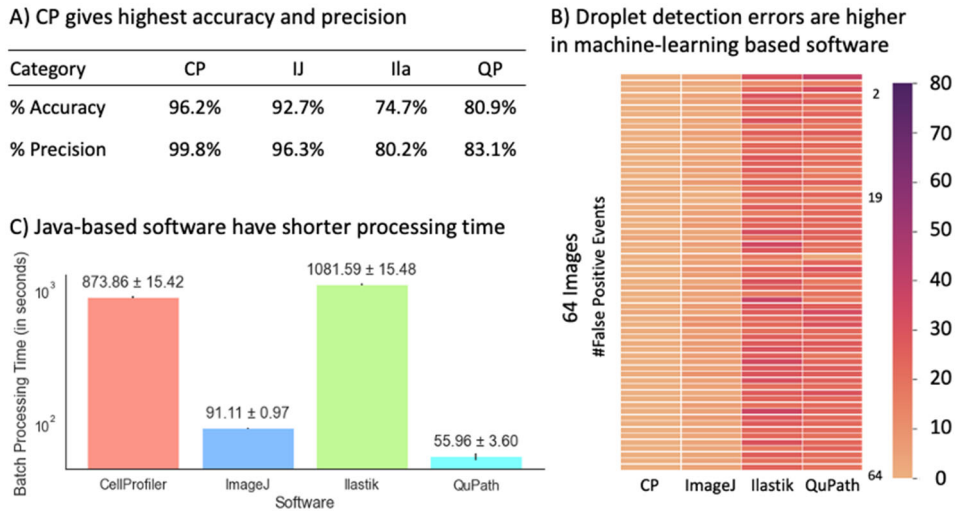


Figure 12. User-friendly software shows different accuracy, precision, and processing times for detecting droplets. A) CellProfiler™ (CP) demonstrated high accuracy and precision compared to the other software (ImageJ, Ilastik, and QuPath). B) In a diagnostic test, the false positive events were higher in the machine-learning-based software compared to the rule-based, where each block represents an event or image error detection (high numbers in the scale indicate more errors). C) The Java-based software (ImageJ and QuPath) displayed faster processing times in detecting droplets. The same 64 images with 10 replicates were used for this evaluation (Publication II - Sanka et al., 2021).

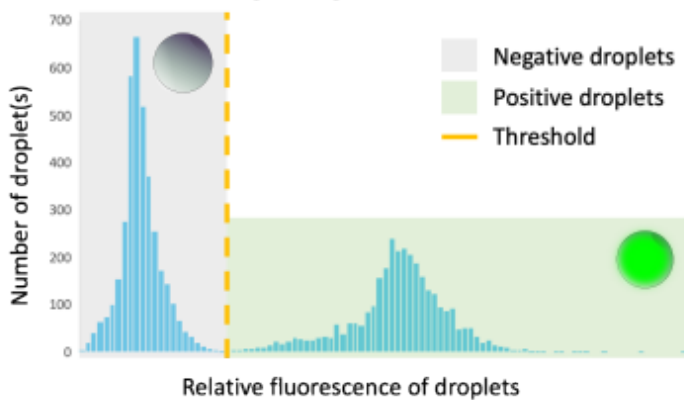
4.2 Droplet classification can be done manually or automatically (Publication I)

This section addresses the need to find a suitable platform for the classification of group droplets in experiments.

Droplet detection is usually followed by intensity measurement, where gray signals that represent each droplet are measured. This measurements can then be transformed into sizes or other measurement variables, e.g., maximum/minimum ferret diameter, mean diameter, radius, mean pixel intensity or maximum pixel intensity (Liang et al.,

2014). Using relative fluorescence or pixel intensity, droplets can be classified into two or more groups. In Figure 13A, two types of droplets are classified based on the detected mean intensity in the first local minima after the first peak. In this case, two groups of droplets that represent empty droplets and droplets with objects which have fluorescence were classified. There is also a method which can be used to distinguish droplets by implementing a supervised machine learning algorithm, such as CellProfiler Analyst (CPA) (Figure 13B). The software can host a database file from CellProfiler that contains detected objects. In CPA, the user only needs to train on a model based on the selection of the image, e.g., positive droplets with bright green fluorescence and negative droplets with dark pixels. In CPA, the result can also be assessed using a confusion matrix, which cross validates the classification results with predefined classes (droplets with and without bacteria/fluorescence) (Dao et al., 2016).

A) Manual thresholding using relative fluorescence



B) Automatic thresholding can be performed using CellProfiler Analyst with different supervised machine learning algorithm

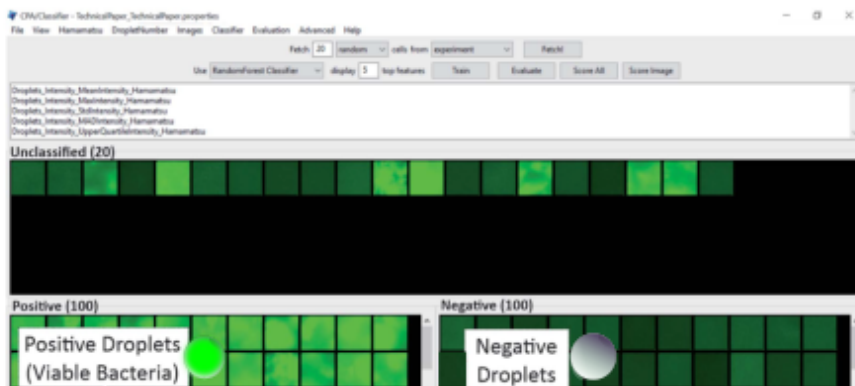


Figure 13. Droplet classification techniques (manual and automatic) use pixel profiles as the parameters to group droplets into two or more different types. A) Manual thresholding provides flexibility to define the intensity value characteristic of the desired group(s). B) Automatic thresholding categorizes based on visually inspected examples which are used to build a classification model. Manual thresholding is done by establishing a threshold in the relative fluorescence droplet distribution and the automatic classification using CellProfiler Analyst (CPA) with an available classification algorithm, e.g., Random Forest (Publication I - Bartkova et al., 2020).

4.3 A developed web application (EasyFlow) quickly provides analysis and visualization of droplet experiment data (Publication III)

This section deals with simple data analysis of droplet array images without the need for programming skills.

Every type of image analysis software is able to produce pixel profiles from image data. Unfortunately, not every image is designed for droplet analysis. EasyFlow is built to simplify the workflow of image-based droplet detection analysis and can host comma separated value (.csv) or excel file format (.xlsx or .xls) from image analysis software (Figure 14A). Using the mentioned file, the user can directly upload it onto EasyFlow and it will generate four important visualizations for droplet experiments, including droplet size and signal distributions, the relationship between size and signal pattern, and a comparison of experimental conditions (Figure 14B). EasyFlow also provides tuneable thresholding, where the user can define the manual threshold to classify two types of droplets.

Easyflow generates the necessary graphs usually used to assess droplet-based experiments. These include:

- Droplet size distribution (Sun et al., 2019)
To determine size homogeneity, especially in assessing mono-dispersity in droplet generation using a microfluidics setup.
- Droplet signal distribution (Rutkowski et al., 2022)
To distinguish two or more types of objects, including droplets with and without encapsulated objects.
- Relationship between droplet size and signal
To assess whether encapsulation is clumped in specific volumes or shows some distributions.
- Comparison of experimental condition (Sanka et al., 2021; Scheler et al., 2020)
To show the distribution of signals in different labels which are usually used for different droplet experiments, e.g., an antimicrobial susceptibility test (AST).

A) EasyFlow is a web application which can process output data from image analysis software



B) EasyFlow provides quick data processing and is equipped with four important visualizations

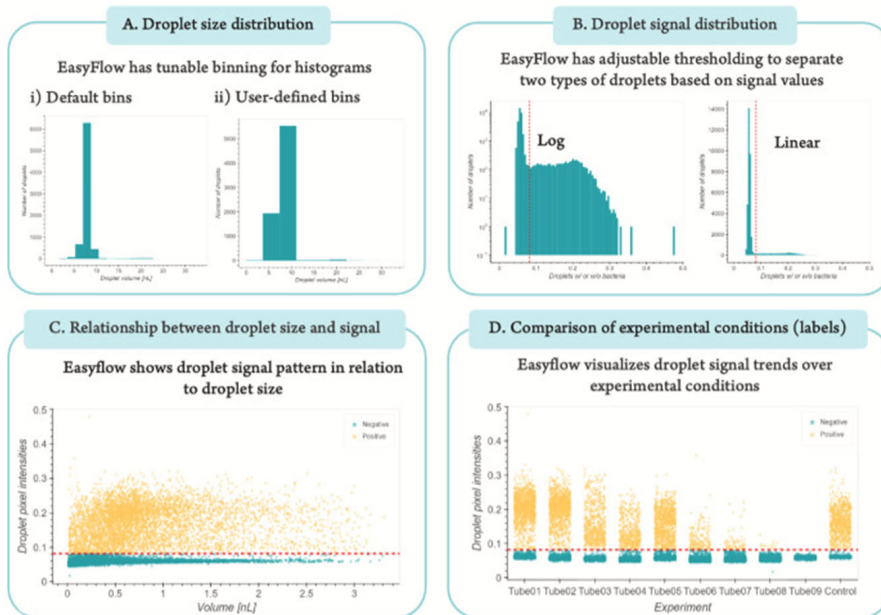


Figure 14. EasyFlow simplifies data processing and the visualization of image analysis software output. A) Image analysis software generates Comma Separated Values (.csv) as the result of droplet detection. Label, signal, and size data can be used to produce results in EasyFlow. B) EasyFlow produces essential graphs that represent droplet profiles and provide necessary data processing. This includes 1) droplet size distribution histograms, 2) the distribution of droplet signals, 3) the relationship between droplet size and signal data, and 4) a comparison of experimental condition (label) graphs. EasyFlow is equipped with adjustable binning and tuneable thresholding. Thresholding is required to group two types of droplets and generate color-coded plots (especially for relationship and comparison graphs) (Publication III - Sanka et al., 2023).

4.4 Analytical pipelines for common droplet experimental scenarios and their applications (Publication III)

This section provides the results for showcasing user-friendly analytical pipelines for different droplet experiment scenarios.

4.4.1 Different droplet and object detection using user-friendly software

In droplet-based experiments, there are various types of image-based analyses which are usually performed in different settings. For instance, it may depend on the types of droplets, microscopy techniques, etc. (Figure 15). Here, we have provided different pipelines for analyzing droplet image data that could be used as examples. Each of the pipelines adopts the principle of pre-processing, processing, and post-processing, as described in (Sanka et al., 2021).

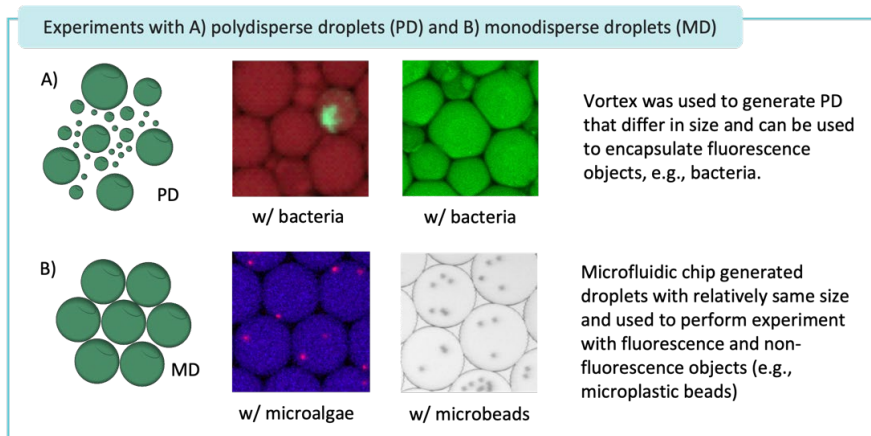


Figure 15. Typical droplets and objects of study in image data. These image data capture signals from droplets and our objects of interest. A) show polydisperse droplets that encapsulate bacteria which give different fluorescence signals and B) show monodisperse droplets that encapsulate fluorescence and non-fluorescence objects (Publication III - Sanka et al., 2023).

The first pipeline works in polydisperse droplet detection, which accommodates fluorescence object measurement (Figure 16A). In this case, the fluorescence represents whether the droplet has objects or not. In brief, this pipeline starts with channel splitting and splits the input image, which has false coloring – red, green, and blue – for visualization. Using only an IdentifyPrimaryObject module, droplets can be detected with ease. However, an additional filtering module is required to eliminate faulty segmentation. The filtering module is added by considering different variables which are important for eliminating non-cyclic objects, including eccentricity, solidity, and form factors (Andersson et al., 2019; Tiemeijer et al., 2021).

For the second pipeline, the detection is performed to find fluorescence objects in monodisperse droplets in fluorescent image data (Figure 16B). This pipeline is also able to distinguish how many objects are encapsulated in each droplet. The pipeline starts with color splitting using a ColorToGray module, similar to Pipeline 1. However, in this pipeline, two modules of IdentifyPrimaryObject were used to detect droplets and fluorescence objects separately. This includes a filtering module to eliminate imperfect segmentation. Once both droplets and objects were detected, the relationship between both detections was determined under a RelateObject module.

For the third pipeline, the detection can be implemented in brightfield image data and is meant to detect monodisperse droplets which have objects in them (Figure 16C). This pipeline uses a combination of Ilastik and CellProfiler™ (CP) to detect both droplets and the objects of interest. Briefly, Ilastik was needed to detect objects in the droplets using Pixel Classification and the Object Identification module. This detection resulted in a probability map which could be used to perform object re-detection in CP. On the other hand, droplets were detected using CP with some additional modules, e.g., EnhanceOrSuppress, IdentifyPrimaryObject, and RelateObject modules.

From each of the pipelines, both droplets and objects were measured using MeasureObjectIntensity and MeasureSizeShape to retrieve the pixel data. All of these results were exported as .csv files to suit EasyFlow. Complete pipelines are explained more thoroughly in (Sanka et al., 2023).

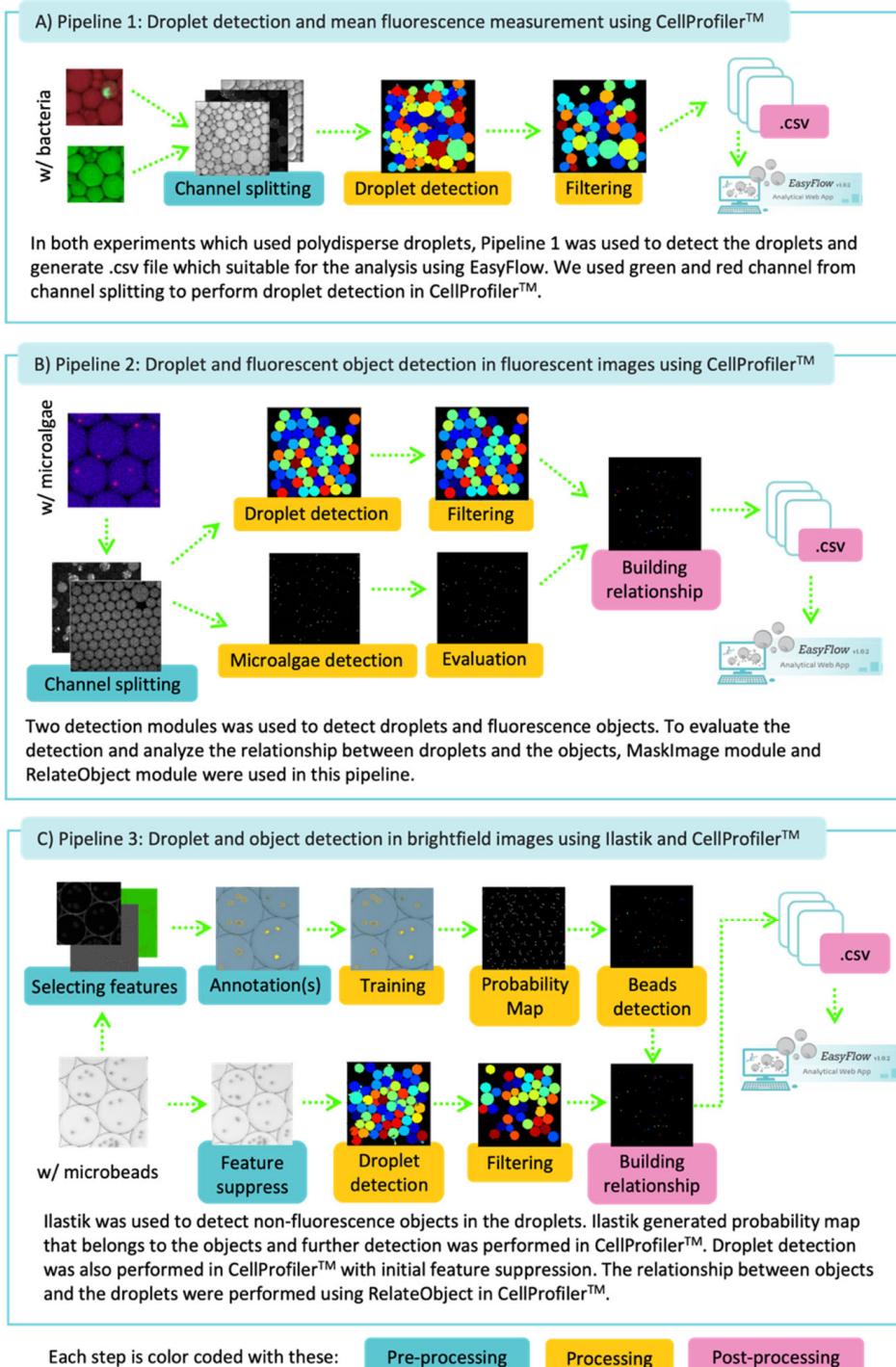


Figure 16. Droplet image analysis pipelines for typical droplet experiment scenarios. These pipelines cover polydisperse and monodisperse droplet detection in fluorescent image data, polydisperse droplets in brightfield image data, and different objects in droplets (Publication III - Sanka et al., 2023).

4.4.2 Applications of droplet emulsions with user-friendly pipelines

We tested the pipelines to perform different microbiology research, e.g., controlled growth of *E. coli* using a CRISPR system, an antimicrobial heteroresistance test in *E. coli*, microalgae growth observation, and a microplastic bead (microbeads) droplet stability test.

a) Bacteria with an aTc CRISPR system can grow in high temperatures

We tested Pipeline 1 by performing an analysis on CRISPR-based system bacteria, where the growth could be adjusted by anhydrotetracycline (aTc). Gardner et al. (Gardner et al., 2000) introduced a "toggle switch" that can be activated in the presence of aTc, inhibiting bacterial growth, while its deactivation in the absence of aTc changes the toggle switch to the OFF state, allowing bacterial growth to resume. Moreover, the CRISPR system is temperature sensitive so that the bacterial growth will stop if the temperature is high. As we can see in Figure 17, the droplet pixel intensities at 42°C with an aTc or anhydrotetracycline "toggle switch" are higher than at 37°C. This shows the "toggle switch" works by restarting the replication and proliferation after switching to 42°C. We can also see that both bacterial growths remained stable without additional aTc. This corresponds to the result from Wiktor et al. (Wiktor et al., 2016), where replication and proliferation could be triggered by switching the temperature to 42°C.

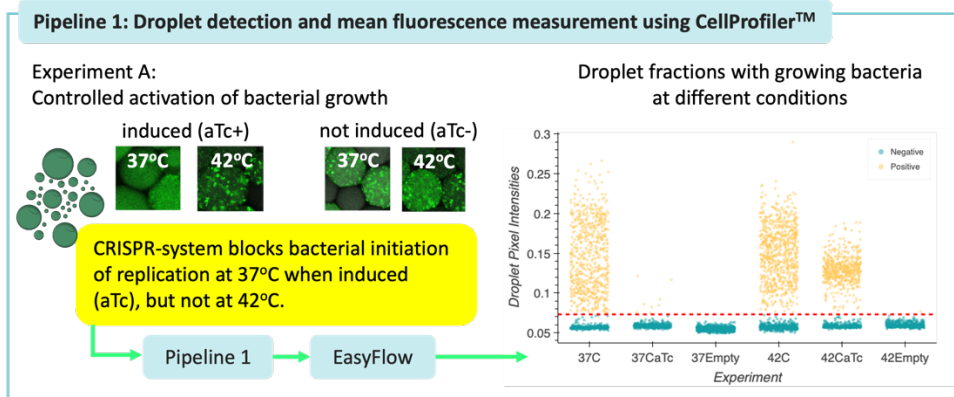


Figure 17. Controlled activation of bacterial growth using an aTc knock-out in polydisperse droplets shows different growth rates at 37°C and 42°C. (Publication III - Sanka et al., 2023)

b) Antibiotic activity in the B-period of the cell cycle

Pipeline 1 was also used to monitor the antibiotic activity in growth-synchronized bacteria. Bacterial growth consists of three periods, including the preparation of DNA replication (B-period), replication to termination (C-period), and the end of the termination and division of bacteria cells (D-period) (J. D. Wang & Levin, 2009). In this experiment, bacterial replication was "arrested" using serine hydroxamate (SHX) during the B-period before adding antibiotics. This means the bacteria could have synchronized growth and we wanted to see whether different stages of the bacterial growth period affect the overall response of antibiotic treatment (Figure 18). However, based on the results, we could not see any difference between non-arrested and arrested bacteria. Both groups have the same viability, and this indicates that the B-period of the cell cycle does not affect the antibiotic response to the bacteria.

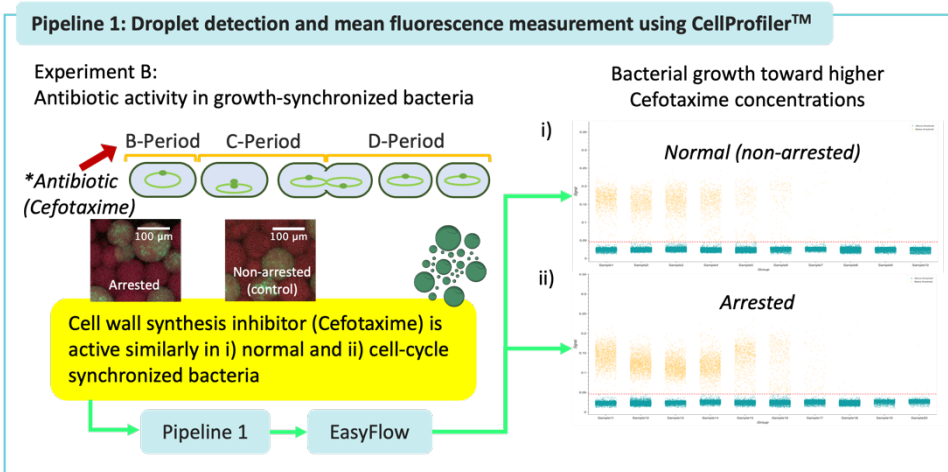


Figure 18. Both normal and cell-cycle synchronized growth bacteria show similar resistance responses (Publication III - Sanka et al., 2023).

c) Microalgae growth remains stable in droplets

In this experiment, we used *Raphidocelis subcapitata* microalgae cells for these droplet-based experiments. The microalgae are important in performing toxicology experiments (Suzuki et al., 2018). Therefore, it is important to see whether the microalgae can grow in droplets. Using Pipeline 2, we observed that the microalgae's growth remained stable for over 72 hours (Figure 19). Multiplication occurred after 24 hours and we also observed that the growth became slower after reaching a specific density, for example between 48 hours and 72 hours. It is known that microalgae growth slows down after a certain density is reached (Lananan et al. 2013). Therefore, the analysis corresponds to the mentioned previous research. In this case, growth is slower after 48 hours.

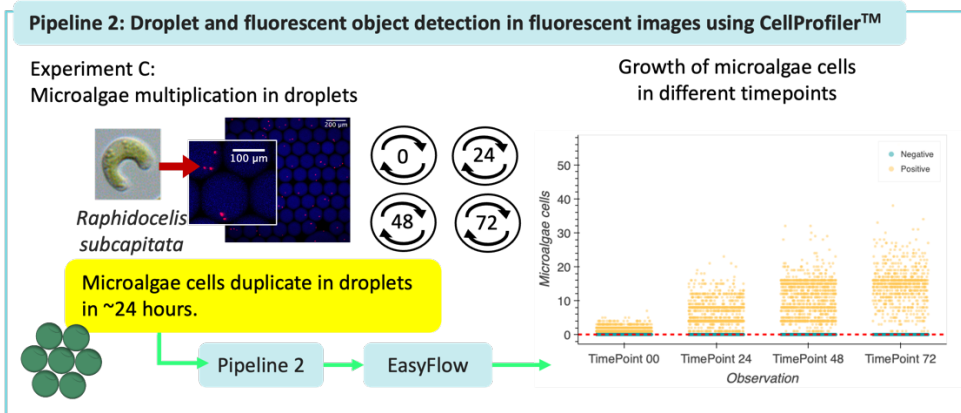


Figure 19. Microalgae multiply in droplets over a 72-hour incubation period (Publication III - Sanka et al., 2023).

d) Microplastic bead encapsulation in droplets

Microplastic beads are suitable for simulating environmental pollution, which has become a serious problem worldwide. Microplastics have some implications in remote and marine ecosystems (Horton & Barnes, 2020). In this experiment, we tried to encapsulate the beads in droplets to see whether the droplets would remain stable over time. As we can see in Figure 20, observation after 24 hours showed different numbers of microplastic beads were encapsulated in the droplets. Most of the droplets had three to five beads in each droplet but the encapsulation rate showed that the droplet could encapsulate from one to 14 bead(s). Moreover, this encapsulation follows the theoretical Poisson distribution, which is explained in detail in the Supplementary Information of Sanka et al. (Sanka et al., 2023).

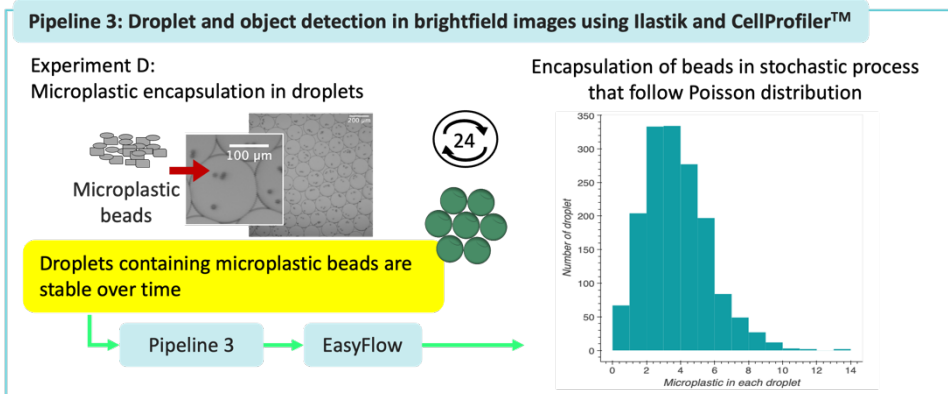


Figure 20. Droplets with different numbers of microplastic beads are stable overnight (Publication III - Sanka et al., 2023).

5 CONCLUSIONS

In this thesis, four important goals were reached. Briefly, each of the important conclusions that reflect the aims are described below:

- Aim 1 Provide a droplet detection platform for different droplet experiments.
- Platform exploration resulted in the discovery of the four most used types of user-friendly software (ImageJ, CellProfiler, Ilastik and QuPath). The types of software are capable of detecting droplets. The software adopt rule-based and machine learning-based principles which help the user to perform the pre-processing, processing, and post-processing steps. All these types of software are user-friendly and provide simple pipelines for non-experienced users.
 - Despite the user-friendliness, each type of software performed differently in detecting droplets. Using 64 images, CellProfiler gives the highest accuracy and precision. However, ImageJ and QuPath have faster processing times.
- Aim 2 Find a suitable platform for droplet classification, which is needed to group droplets in experiments.
- Droplet classification can be performed by grouping average pixel intensity through manual thresholding and machine-learning-based classification software.
 - In manual thresholding classification, the grouping of two types of droplets (e.g., empty and object-encapsulated droplets) can be performed by putting a threshold value on the first local minima after the first local maxima. Values below threshold indicate empty droplets and values above threshold indicate object-encapsulated droplets.
 - Machine-learning-based classification software, such as CellProfiler Analyst, is able to classify two types of droplets, e.g., droplets with and without bacteria (with fluorescence labels).
- Aim 3 Simplify data analysis of droplet array images, which usually requires image analyst and programming skills.
- Data analysis can be simplified by developing the user-friendly data processing and visualization analytical web application EasyFlow. This web application eases data processing and the visualization of droplet data which are produced by image analysis software. It is open-source and users do not need specific hardware. Furthermore, EasyFlow can be used by many researchers with no prior experience, and it has a gradual learning curve. Additionally, EasyFlow also can host any data that is stored as .csv or .xlsx files containing labels, sizes, and signals.
- Aim 4 Develop user-friendly analytical pipelines for different droplet experiment scenarios.
- Different pipelines were developed and tested to perform bacteria-based experiments, microalgae growth experiments, and microplastic bead experiments. These pipelines work with droplets of the same size (monodisperse) and of different sizes (polydisperse). All of the pipelines are user-friendly and require no prior image analysis experience or programming skills.

References

- Agresti, J. J., Kelly, B. T., Jäschke, A., & Griffiths, A. D. (2005). Selection of ribozymes that catalyse multiple-turnover Diels–Alder cycloadditions by using in vitro compartmentalization. *Proceedings of the National Academy of Sciences*, 102(45), 16170–16175. <https://doi.org/10.1073/pnas.0503733102>
- Ahmadi, F., Samlali, K., Vo, P. Q. N., & Shih, S. C. C. (2019). An integrated droplet-digital microfluidic system for on-demand droplet creation, mixing, incubation, and sorting. *Lab on a Chip*, 19(3), 524–535. <https://doi.org/10.1039/C8LC01170B>
- Alizadeh, N., & Salimi, A. (2019). Polymer dots as a novel probe for fluorescence sensing of dopamine and imaging in single living cell using droplet microfluidic platform. *Analytica Chimica Acta*, 1091, 40–49. <https://doi.org/10.1016/J.ACA.2019.08.036>
- Andersson, D. I., Nicoloff, H., & Hjort, K. (2019). Mechanisms and clinical relevance of bacterial heteroresistance. *Nature Reviews Microbiology* 2019 17:8, 17(8), 479–496. <https://doi.org/10.1038/s41579-019-0218-1>
- Assadi, Y., Farazadeh, M. A., & Bidari, A. (2012). Dispersive Liquid–Liquid Microextraction. *Comprehensive Sampling and Sample Preparation: Analytical Techniques for Scientists*, 181–212. <https://doi.org/10.1016/B978-0-12-381373-2.00051-X>
- Avni, A., Joshi, A., Walimbe, A., Pattanashetty, S. G., & Mukhopadhyay, S. (2022). Single-droplet surface-enhanced Raman scattering decodes the molecular determinants of liquid-liquid phase separation. *Nature Communications* 2022 13:1, 13(1), 1–13. <https://doi.org/10.1038/s41467-022-32143-0>
- Baccouche, A., Okumura, S., Sieskind, R., Henry, E., Aubert-Kato, N., Bredeche, N., Bartolo, J.-F., Taly, V., Rondelez, Y., Fujii, T., & Genot, A. J. (2017). Massively parallel and multiparameter titration of biochemical assays with droplet microfluidics. *Nature Protocols*, 12(9), 1912–1932. <https://doi.org/10.1038/nprot.2017.092>
- Bageritz, J., & Raddi, G. (2019). Single-Cell RNA Sequencing with Drop-Seq. *Methods in Molecular Biology (Clifton, N.J.)*, 1979, 73–85. https://doi.org/10.1007/978-1-4939-9240-9_6
- Bankhead, P. (2023). *Introduction to Bioimage Analysis — Introduction to Bioimage Analysis*. <https://bioimagebook.github.io/>
- Bankhead, P., Loughrey, M. B., Fernández, J. A., Dombrowski, Y., McArt, D. G., Dunne, P. D., McQuaid, S., Gray, R. T., Murray, L. J., Coleman, H. G., James, J. A., Salto-Tellez, M., & Hamilton, P. W. (2017). QuPath: Open source software for digital pathology image analysis. *Scientific Reports*, 7(1), 1–7. <https://doi.org/10.1038/s41598-017-17204-5>
- Baret, J. C. (2012). Surfactants in droplet-based microfluidics. *Lab on a Chip*, 12(3), 422–433. <https://doi.org/10.1039/C1LC20582J>
- Bartkova, S., Vendelin, M., Sanka, I., Pata, P., & Scheler, O. (2020). Droplet image analysis with user-friendly freeware CellProfiler. *Analytical Methods*, 12(17), 2287–2294. <https://doi.org/10.1039/d0ay00031k>
- Bazin, P. L., Ye, C., Bogovic, J. A., Shiee, N., Reich, D. S., Prince, J. L., & Pham, D. L. (2011). Direct Segmentation of the Major White Matter Tracts in Diffusion Tensor Images. *NeuroImage*, 58(2), 458. <https://doi.org/10.1016/J.NEUROIMAGE.2011.06.020>
- Bell, S. E., Park, I., Rubakhin, S. S., Bashir, R., Vlasov, Y., & Sweedler, J. V. (2021). Droplet Microfluidics with MALDI-MS Detection: The Effects of Oil Phases in GABA Analysis. *ACS Measurement Science Au*, 1(3), 147–156. <https://doi.org/10.1021/ACSMEASURESCIAU.1C00017>

- Beneyton, T., Krafft, D., Bednarz, C., Kleineberg, C., Woelfer, C., Ivanov, I., Vidaković-Koch, T., Sundmacher, K., & Baret, J. C. (2018). Out-of-equilibrium microcompartments for the bottom-up integration of metabolic functions. *Nature Communications*, 9(1), 1–10. <https://doi.org/10.1038/s41467-018-04825-1>
- Berg, S., Kutra, D., Kroeger, T., Straehle, C. N., Kausler, B. X., Haubold, C., Schiegg, M., Ales, J., Beier, T., Rudy, M., Eren, K., Cervantes, J. I., Xu, B., Beuttenmueller, F., Wolny, A., Zhang, C., Koethe, U., Hamprecht, F. A., & Kreshuk, A. (2019). ilastik: interactive machine learning for (bio)image analysis. *Nature Methods*, 16(12), 1226–1232. <https://doi.org/10.1038/s41592-019-0582-9>
- Binks, B. P., & Lumsdon, S. O. (2000). Influence of Particle Wettability on the Type and Stability of Surfactant-Free Emulsions. *Langmuir*, 16(23), 8622–8631. <https://doi.org/10.1021/la000189s>
- Bland, J. M., & Altman, D. G. (1995). Comparing methods of measurement: why plotting difference against standard method is misleading. *The Lancet*, 346(8982), 1085–1087. [https://doi.org/10.1016/S0140-6736\(95\)91748-9](https://doi.org/10.1016/S0140-6736(95)91748-9)
- Bland, J. M., & Altman, D. G. (1999). Measuring agreement in method comparison studies. *Statistical Methods in Medical Research*, 8, 135–160. <https://doi.org/10.1191/096228099673819272>
- Britos, G. M., & Ojeda, S. M. (2019). Robust estimation for spatial autoregressive processes based on bounded innovation propagation representations. *Computational Statistics*, 34(3), 1315–1335. <https://doi.org/10.1007/S00180-018-0845-4/TABLES/1>
- Bull, D. R. (2014). Digital Picture Formats and Representations. *Communicating Pictures*, 99–132. <https://doi.org/10.1016/B978-0-12-405906-1.00004-0>
- Byrnes, S. A., Chang, T. C., Huynh, T., Astashkina, A., Weigl, B. H., & Nichols, K. P. (2018). Simple Polydisperse Droplet Emulsion Polymerase Chain Reaction with Statistical Volumetric Correction Compared with Microfluidic Droplet Digital Polymerase Chain Reaction. *Analytical Chemistry*, 90(15), 9374–9380. <https://doi.org/10.1021/acs.analchem.8b01988>
- Byrnes, S. A., Huynh, T., Chang, T. C., Anderson, C. E., McDermott, J. J., Oncina, C. I., Weigl, B. H., & Nichols, K. P. (2020). Wash-Free, Digital Immunoassay in Polydisperse Droplets. *Analytical Chemistry*, 92(5), 3535–3543. <https://doi.org/10.1021/ACS.ANALCHEM.9B02526>
- Byrnes, S. A., Phillips, E. A., Huynh, T., Weigl, B. H., & Nichols, K. P. (2018). Polydisperse emulsion digital assay to enhance time to detection and extend dynamic range in bacterial cultures enabled by a statistical framework. *Analyst*, 143(12), 2828–2836. <https://doi.org/10.1039/c8an00029h>
- Chen, L., Ding, J., Yuan, H., Chen, C., & Li, Z. (2022). Deep-dLAMP: Deep Learning-Enabled Polydisperse Emulsion-Based Digital Loop-Mediated Isothermal Amplification. *Advanced Science*, 9(9). <https://doi.org/10.1002/ADVS.202105450>
- Cohen, L., Cui, N., Cai, Y., Garden, P. M., Li, X., Weitz, D. A., & Walt, D. R. (2020). Single Molecule Protein Detection with Attomolar Sensitivity Using Droplet Digital Enzyme-Linked Immunosorbent Assay. *ACS Nano*, 14(8), 9491–9501. <https://doi.org/10.1021/ACS.NANO.0C02378>
- Collins, D. J., Neild, A., deMello, A., Liu, A. Q., & Ai, Y. (2015). The Poisson distribution and beyond: Methods for microfluidic droplet production and single cell encapsulation. In *Lab on a Chip* (Vol. 15, Issue 17, pp. 3439–3459). Royal Society of Chemistry. <https://doi.org/10.1039/c5lc00614g>

- Dao, D., Fraser, A. N., Hung, J., Ljosa, V., Singh, S., & Carpenter, A. E. (2016). CellProfiler Analyst: interactive data exploration, analysis and classification of large biological image sets. *Bioinformatics (Oxford, England)*, 32(20), 3210–3212. <https://doi.org/10.1093/bioinformatics/btw390>
- De Cesare, I., Zamora-Chimal, C. G., Postiglione, L., Khazim, M., Pedone, E., Shannon, B., Fiore, G., Perrino, G., Napolitano, S., Di Bernardo, D., Savery, N. J., Grierson, C., Di Bernardo, M., & Marucci, L. (2021). ChipSeg: An Automatic Tool to Segment Bacterial and Mammalian Cells Cultured in Microfluidic Devices. *ACS Omega*, 6(4), 2473–2476. <https://doi.org/10.1021/acsomega.0c03906>
- De Chaumont, F., Dallongeville, S., Chenouard, N., Hervé, N., Pop, S., Provoost, T., Meas-Yedid, V., Pankajakshan, P., Lecomte, T., Le Montagner, Y., Lagache, T., Dufour, A., & Olivo-Marin, J. C. (2012). Icy: an open bioimage informatics platform for extended reproducible research. *Nature Methods* 2012 9:7, 9(7), 690–696. <https://doi.org/10.1038/nmeth.2075>
- De Rop, F. V., Ismail, J. N., Bravo González-Blas, C., Hulselmans, G. J., Flerin, C. C., Janssens, J., Theunis, K., Christiaens, V. M., Wouters, J., Marcassa, G., de Wit, J., Poovathingal, S., & Aerts, S. (2022). Hydrop enables droplet-based single-cell ATAC-seq and single-cell RNA-seq using dissolvable hydrogel beads. *ELife*, 11, e73971. <https://doi.org/10.7554/eLife.73971>
- Delahaye, T., Lombardo, T., Sella, C., & Thouin, L. (2021). Electrochemical assessments of droplet contents in microfluidic channels. Application to the titration of heterogeneous droplets. *Analytica Chimica Acta*, 1155, 338344. <https://doi.org/https://doi.org/10.1016/j.aca.2021.338344>
- Demaree, B., Weisgerber, D., Dolatmoradi, A., Hatori, M., & Abate, A. R. (2018). Direct quantification of EGFR variant allele frequency in cell-free DNA using a microfluidic-free digital droplet PCR assay. In *Methods in Cell Biology* (1st ed.). Elsevier Inc. <https://doi.org/10.1016/bs.mcb.2018.10.002>
- Devenish, S. R. A., Kaltenbach, M., Fischlechner, M., & Hollfelder, F. (2013). Droplets as reaction compartments for protein nanotechnology. *Methods in Molecular Biology*, 996, 269–286. https://doi.org/10.1007/978-1-62703-354-1_16/COVER
- Diefenbach, X. W., Farasat, I., Guetschow, E. D., Welch, C. J., Kennedy, R. T., Sun, S., & Moore, J. C. (2018). Enabling Biocatalysis by High-Throughput Protein Engineering Using Droplet Microfluidics Coupled to Mass Spectrometry. *ACS Omega*, 3(2), 1498–1508. <https://doi.org/10.1021/acsomega.7b01973>
- Dobson, E. T. A., Cimini, B. A., Klemm, A. H., Wählby, C., Carpenter, A. E., & Eliceiri, K. W. (2021). ImageJ and CellProfiler: Complements in Open-Source Bioimage Analysis. *Null*. <https://doi.org/10.1002/cpz1.89>
- Durve, M., Tiribocchi, A., Bonaccorso, F., Montessori, A., Lauricella, M., Bogdan, M., Guzowski, J., & Succi, S. (2022). DropTrack - Automatic droplet tracking with YOLOv5 and DeepSORT for microfluidic applications. *Physics of Fluids*, 34(8), 82003. <https://doi.org/10.1063/5.0097597/2846690>
- Elkabets, M., Gifford, A. M., Scheel, C., Scheel, C., Nilsson, B., Reinhardt, F., Bray, M.-A., Carpenter, A. E., Jirström, K., Magnusson, K., Ebert, B. L., Pontén, F., Weinberg, R. A., & McAllister, S. S. (2011). Human tumors instigate granulysin-expressing hematopoietic cells that promote malignancy by activating stromal fibroblasts in mice. *Journal of Clinical Investigation*. <https://doi.org/10.1172/jci43757>

- Fan, L., Zhang, F., Fan, H., & Zhang, C. (2019). Brief review of image denoising techniques. *Visual Computing for Industry, Biomedicine, and Art*, 2(1), 1–12. <https://doi.org/10.1186/S42492-019-0016-7>
- Feng, D., Li, H., Xu, T., Zheng, F., Hu, C., Shi, X., & Xu, G. (2022). High-throughput single cell metabolomics and cellular heterogeneity exploration by inertial microfluidics coupled with pulsed electric field-induced electrospray ionization-high resolution mass spectrometry. *Analytica Chimica Acta*, 1221, 340116. <https://doi.org/10.1016/J.ACA.2022.340116>
- Fernandez, P., André, V., Rieger, J., & Kühnle, A. (2004). Nano-emulsion formation by emulsion phase inversion. *Colloids and Surfaces A: Physicochemical and Engineering Aspects*, 251(1–3), 53–58. <https://doi.org/10.1016/J.COLSURFA.2004.09.029>
- Flamary, R., Tuia, D., Labbé, B., Camps-Valls, G., & Rakotomamonjy, A. (2012). Large margin filtering. *IEEE Transactions on Signal Processing*, 60(2), 648–659. <https://doi.org/10.1109/TSP.2011.2173685>
- Floyd, C. E., Jaszczak, R. J., & Coleman, R. E. (1986). Image Resampling on a Cylindrical Sector Grid. *IEEE Transactions on Medical Imaging*, 5(3), 128–131. <https://doi.org/10.1109/TMI.1986.4307761>
- Fourment, M., & Gillings, M. R. (2008). A comparison of common programming languages used in bioinformatics. *BMC Bioinformatics*, 9(1), 1–9. <https://doi.org/10.1186/1471-2105-9-82>
- Frank, M., Drikakis, D., & Charissis, V. (2020). Machine-learning methods for computational science and engineering. In *Computation* (Vol. 8, Issue 1, p. 15). MDPI Multidisciplinary Digital Publishing Institute. <https://doi.org/10.3390/computation8010015>
- Fu, H., Zeng, W., Li, S., & Yuan, S. (2017). Electrical-detection droplet microfluidic closed-loop control system for precise droplet production. *Sensors and Actuators A: Physical*, 267, 142–149. <https://doi.org/https://doi.org/10.1016/j.sna.2017.09.043>
- Gaddis, G. M., & Gaddis, M. L. (1990). Introduction to biostatistics: Part 3, sensitivity, specificity, predictive value, and hypothesis testing. *Annals of Emergency Medicine*, 19(5), 591–597. [https://doi.org/10.1016/S0196-0644\(05\)82198-5](https://doi.org/10.1016/S0196-0644(05)82198-5)
- Gardner, T. S., Cantor, C. R., & Collins, J. J. (2000). Construction of a genetic toggle switch in *Escherichia coli*. *Nature* 2000 403:6767, 403(6767), 339–342. <https://doi.org/10.1038/35002131>
- Gawryszewski, K., Rana, Z. A., Jenkins, K. W., Ioannou, P., & Okonkwo, D. (2018). An automatic image analysis methodology for the measurement of droplet size distributions in liquid–liquid dispersion: round object detection. *International Journal of Computers and Applications*, 41(5), 329–342. <https://doi.org/10.1080/1206212X.2018.1542555>
- Genot, A. J., Baccouche, A., Sieskind, R., Aubert-Kato, N., Bredeche, N., Bartolo, J. F., Taly, V., Fujii, T., & Rondelez, Y. (2016). High-resolution mapping of bifurcations in nonlinear biochemical circuits. *Nature Chemistry*, 8(8), 760–767. <https://doi.org/10.1038/nchem.2544>
- Georgeson, M. A., & Meese, T. S. (1999). Adaptive Filtering in Spatial Vision: Evidence from Feature Marking in Plaids. *Perception*, 28(6), 687–702. <https://doi.org/10.1068/P2836>
- Giuffrida, M. C., Cigliana, G., & Spoto, G. (2018). Ultrasensitive detection of lysozyme in droplet-based microfluidic devices. *Biosensors and Bioelectronics*, 104, 8–14. <https://doi.org/10.1016/J.BIOS.2017.12.042>

- Gonzalez, R. C., & Woods, R. E. (2008). *Digital Image Processing* (3rd ed.). Pearson.
- Grösche, M., Korvink, J. G., Rabe, K. S., & Niemeyer, C. M. (2019). Comparison of Storage Methods for Microfluidically Produced Water-in-Oil Droplets. *Chemical Engineering & Technology*, 42(10), 2028–2034. <https://doi.org/10.1002/CEAT.201900075>
- Ha, B., Kim, T. J., Moon, E., Giaccia, A. J., & Pratz, G. (2021). Flow radiocytometry using droplet optofluidics. *Biosensors and Bioelectronics*, 194, 113565. <https://doi.org/10.1016/J.BIOS.2021.113565>
- Hartig, S. M. (2013). Basic Image Analysis and Manipulation in ImageJ. *Current Protocols in Molecular Biology*, 102(1), 14.15.1-14.15.12. <https://doi.org/10.1002/0471142727.MB1415S102>
- Hartwell, K. A., Miller, P., Mukherjee, S., Mukherjee, S., Mukherjee, S., Kahn, A. R., Stewart, A. L., Stewart, A. L., Stewart, A. L., Logan, D. J., Negri, J., Negri, J., Negri, J., Duvet, M., Järås, M., Puram, R. V., Dančík, V., Al-Shahrour, F., Kindler, T., ... Golub, T. R. (2013). Niche-based screening identifies small-molecule inhibitors of leukemia stem cells. *Nature Chemical Biology*. <https://doi.org/10.1038/nchembio.1367>
- Herbert, A. (2023). *ImageJ Batch Processing*. <http://rsbweb.nih.gov/ij/plugins/>
- Hindson, B. J., Ness, K. D., Masquelier, D. A., Belgrader, P., Heredia, N. J., Makarewicz, A. J., Bright, I. J., Lucero, M. Y., Hiddessen, A. L., Legler, T. C., Kitano, T. K., Hodel, M. R., Petersen, J. F., Wyatt, P. W., Steenblock, E. R., Shah, P. H., Bousse, L. J., Troup, C. B., Mellen, J. C., ... Colston, B. W. (2011). High-throughput droplet digital PCR system for absolute quantitation of DNA copy number. *Analytical Chemistry*, 83(22), 8604–8610. <https://doi.org/10.1021/ac202028g>
- Hindson, C. M., Chevillet, J. R., Briggs, H. A., Gallichotte, E. N., Ruf, I. K., Hindson, B. J., Vessella, R. L., & Tewari, M. (2013). Absolute quantification by droplet digital PCR versus analog real-time PCR. *Nature Methods*, 10(10), 1003–1005. <https://doi.org/10.1038/nmeth.2633>
- Horton, A. A., & Barnes, D. K. A. (2020). Microplastic pollution in a rapidly changing world: Implications for remote and vulnerable marine ecosystems. *Science of The Total Environment*, 738, 140349. <https://doi.org/10.1016/J.SCITOTENV.2020.140349>
- Hsu, R. H., Clark, R. L., Tan, J. W., Ahn, J. C., Gupta, S., Romero, P. A., & Venturelli, O. S. (2019). Microbial Interaction Network Inference in Microfluidic Droplets. *Cell Systems*, 9(3), 229-242.e4. <https://doi.org/10.1016/J.CELS.2019.06.008>
- Jones, T. R., Carpenter, A. E., Lamprecht, M. R., Moffat, J., Silver, S. J., Silver, S. J., Silver, S. J., Grenier, J. K., Castoreno, A. B., Eggert, U. S., Root, D. E., Golland, P., & Sabatini, D. M. (2009). Scoring diverse cellular morphologies in image-based screens with iterative feedback and machine learning. *Proceedings of the National Academy of Sciences of the United States of America*. <https://doi.org/10.1073/pnas.0808843106>
- Jones, T. R., Kang, I. H., Wheeler, D. B., Lindquist, R. A., Papallo, A., Sabatini, D. M., Golland, P., & Carpenter, A. E. (2008). CellProfiler Analyst: data exploration and analysis software for complex image-based screens. *BMC Bioinformatics*. <https://doi.org/10.1186/1471-2105-9-482>
- Kalantarifard, A., Saateh, A., & Elbuken, C. (2018). Label-Free Sensing in Microdroplet-Based Microfluidic Systems. *Chemosensors*, 6(2). <https://doi.org/10.3390/chemosensors6020023>

- Kaminski, T. S., Scheler, O., & Garstecki, P. (2016). Droplet microfluidics for microbiology: techniques, applications and challenges. *Lab on a Chip*, 16(12), 2168–2187. <https://doi.org/10.1039/C6LC00367B>
- Kao, Y., Kaminski, T. S., Postek, W., Makuch, K., Ruszczak, A., Stetten, F. Von, Zengerle, R., & Garstecki, P. (2019). Gravity-driven microfluidic assay for digital enumeration of bacteria and for antibiotic susceptibility testing. *Lab on a Chip*. <https://doi.org/10.1039/C9LC00684B>
- Kebriaei, R., & Basu, A. S. (2021). Microfluidic protein detection and quantification using droplet morphology. *Microfluidics and Nanofluidics*, 25(5), 1–11. <https://doi.org/10.1007/S10404-021-02443-W>
- Kempa, E. E., Smith, C. A., Li, X., Bellina, B., Richardson, K., Pringle, S., Galman, J. L., Turner, N. J., & Barran, P. E. (2020). Coupling Droplet Microfluidics with Mass Spectrometry for Ultrahigh-Throughput Analysis of Complex Mixtures up to and above 30 Hz. *Analytical Chemistry*, 92(18), 12605–12612. <https://doi.org/10.1021/ACS.ANALCHEM.0C02632>
- Kissell, R., & Poserina, J. (2017). Chapter 4 - Advanced Math and Statistics. In R. Kissell & J. Poserina (Eds.), *Optimal Sports Math, Statistics, and Fantasy* (pp. 103–135). Academic Press. <https://doi.org/10.1016/B978-0-12-805163-4.00004-9>
- Klemm, A., & Miura, K. (2022). *Batch Processing Methods in ImageJ*. 7–27. https://doi.org/10.1007/978-3-030-76394-7_2
- Kuswandi, B., Nuriman, Huskens, J., & Verboom, W. (2007). Optical sensing systems for microfluidic devices: A review. *Analytica Chimica Acta*, 601(2), 141–155. <https://doi.org/10.1016/J.ACA.2007.08.046>
- LabVIEW. (2023). *What is LabVIEW? Graphical Programming for Test & Measurement - NI*. <https://www.ni.com/en-us/shop/labview.html>
- Lamprecht, M. R., Sabatini, D. M., & Carpenter, A. E. (2007). CellProfiler™: Free, versatile software for automated biological image analysis. *BioTechniques*, 42(1), 71–75. <https://doi.org/10.2144/000112257>
- Lan, F., Demaree, B., Ahmed, N., & Abate, A. R. (2017). Single-cell genome sequencing at ultra-high-throughput with microfluidic droplet barcoding. *Nature Biotechnology*, 35(7), 640–646. <https://doi.org/10.1038/NBT.3880>
- Li, B., Li, Y., Manz, A., & Wu, W. (2020). Miniaturized Continuous-Flow Digital PCR for Clinical-Level Serum Sample Based on the 3D Microfluidics and CMOS Imaging Device. *Sensors*, 20(9), 2492. <https://doi.org/10.3390/s20092492>
- Li, Z., & Liao, L. (2022). Bright Field Droplet Image Recognition Based on Fast Hough Circle Detection Algorithm. *2022 IEEE 14th International Conference on Computer Research and Development, ICCRD 2022*, 234–238. <https://doi.org/10.1109/ICCRD54409.2022.9730227>
- Liang, Y., Zhang, M., & Browne, W. N. (2014). Image segmentation: A survey of methods based on evolutionary computation. In *Simulated Evolution and Learning. SEAL 2014. Lecture Notes in Computer Science* (Dick, G., Vol. 8886, pp. 847–859). Springer Verlag. https://doi.org/10.1007/978-3-319-13563-2_71
- Liénard-Mayor, T., Taverna, M., Descroix, S., & Mai, T. D. (2021). Droplet-interfacing strategies in microscale electrophoresis for sample treatment, separation and quantification: A review. *Analytica Chimica Acta*, 1143, 281–297. <https://doi.org/10.1016/J.ACA.2020.09.008>

- Liu, W. wen, & Zhu, Y. (2020). "Development and application of analytical detection techniques for droplet-based microfluidics"-A review. *Analytica Chimica Acta*, 1113, 66–84. <https://doi.org/10.1016/J.ACA.2020.03.011>
- Liu, W., Zhang, Y., Yan, J., Zou, Y., & Cui, Z. (2021). Semantic Segmentation Network of Remote Sensing Images with Dynamic Loss Fusion Strategy. *IEEE Access*, 9, 70406–70418. <https://doi.org/10.1109/ACCESS.2021.3078742>
- Lyu, F., Xu, M., Cheng, Y., Xie, J., Rao, J., & Tang, S. K. Y. (2015). Quantitative detection of cells expressing BlaC using droplet-based microfluidics for use in the diagnosis of tuberculosis. *Biomicrofluidics*, 9(4), 44120. <https://doi.org/10.1063/1.4928879>
- MATLAB. (2023). MATLAB. <https://www.mathworks.com/products/matlab.html>
- Matuła, K., Rivello, F., & Huck, W. T. S. (2020). Single-Cell Analysis Using Droplet Microfluidics. *Advanced Biosystems*, 4(1), 1900188. <https://doi.org/10.1002/adbi.201900188>
- McQuin, C., Goodman, A., Chernyshev, V., Kamentsky, L., Cimini, B. A., Karhohs, K. W., Doan, M., Ding, L., Rafelski, S. M., Thirstrup, D., Wiegraabe, W., Singh, S., Becker, T., Caicedo, J. C., & Carpenter, A. E. (2018). CellProfiler 3.0: Next-generation image processing for biology. *PLOS Biology*, 16(7), e2005970. <https://doi.org/10.1371/journal.pbio.2005970>
- Medcalf, E. J., Gantz, M., Kaminski, T. S., & Hollfelder, F. (2023). Ultra-High-Throughput Absorbance-Activated Droplet Sorting for Enzyme Screening at Kilohertz Frequencies. *Analytical Chemistry*, 95(10), 4597–4604. <https://doi.org/10.1021/acs.analchem.2c04144>
- Miura, K. (2020). Bioimage Data Analysis Workflows. In *Learning Materials in Biosciences*.
- Miura, K., Paul-Gilloteaux, P., Tosi, S., & Colombelli, J. (2020). Workflows and Components of Bioimage Analysis. In *Bioimage Data Analysis Workflows* (pp. 1–7). Springer, Cham. https://doi.org/10.1007/978-3-030-22386-1_1
- Najah, M., Griffiths, A. D., & Ryckelynck, M. (2012). Teaching single-cell digital analysis using droplet-based microfluidics. *Analytical Chemistry*, 84(3), 1202–1209. <https://doi.org/10.1021/ac202645m>
- Nanes, B. A. (2015). Slide Set: Reproducible image analysis and batch processing with ImageJ. *BioTechniques*, 59(5), 269–278. <https://doi.org/10.2144/000114351>
- Nesterenko, P. A., McLaughlin, J., Cheng, D., Bangayan, N. J., Sojo, G. B., Seet, C. S., Qin, Y., Mao, Z., Obusan, M. B., Phillips, J. W., & Witte, O. N. (2021). Droplet-based mRNA sequencing of fixed and permeabilized cells by CLInt-seq allows for antigen-specific TCR cloning. *Proceedings of the National Academy of Sciences of the United States of America*, 118(3), e2021190118. <https://doi.org/10.1073/PNAS.2021190118>
- Oliveira, G. L., Valada, A., Bollen, C., Burgard, W., & Brox, T. (2016). Deep learning for human part discovery in images. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 1634–1641. <https://doi.org/10.1109/ICRA.2016.7487304>
- OME. (2023). Home | Open Microscopy Environment (OME). <https://www.openmicroscopy.org/>
- Omotayo, O. (1984). Designing user-friendly software systems. *Data Processing*, 26(5), 16–18. [https://doi.org/10.1016/0011-684X\(84\)90303-4](https://doi.org/10.1016/0011-684X(84)90303-4)
- OpenCV C++. (2023). Introduction - OpenCV Tutorial C++. <https://www.opencv-srf.com/p/introduction.html>
- Pacocha, N., Zapotoczna, M., Makuch, K., Bogusławski, J., & Garstecki, P. (2022). You will know by its tail: a method for quantification of heterogeneity of bacterial populations using single-cell MIC profiling. *Lab on a Chip*, 22(22), 4317–4326. <https://doi.org/10.1039/D2LC00234E>

- Pan, M., Lyu, F., & Tang, S. K. Y. (2015). Fluorinated Pickering Emulsions with Non-adsorbing Interfaces for Droplet-based Enzymatic Assays. *Analytical Chemistry*, 87(15), 7938–7943. <https://doi.org/10.1021/acs.analchem.5b01753>
- Pang, Y., Zhou, Q., Wang, X., Lei, Y., Ren, Y., Li, M., Wang, J., & Liu, Z. (2020). Droplets generation under different flow rates in T-junction microchannel with a neck. *AIChE Journal*, 66(10), e16290. <https://doi.org/10.1002/AIC.16290>
- Park, S.-H., Lim, H., Bae, B. K., Hahm, M. H., Chong, G. O., Jeong, S. Y., & Kim, J.-C. (2021). Robustness of magnetic resonance radiomic features to pixel size resampling and interpolation in patients with cervical cancer. *Cancer Imaging*, 21(1), 19. <https://doi.org/10.1186/s40644-021-00388-5>
- Pärnamets, K., Pardy, T., Koel, A., Rang, T., Scheler, O., Le Moullec, Y., & Afrin, F. (2021). Optical Detection Methods for High-Throughput Fluorescent Droplet Microflow Cytometry. *Micromachines* 2021, Vol. 12, Page 345, 12(3), 345. <https://doi.org/10.3390/M12030345>
- Plá, F., Juste, F., & Ferri, F. (1993). Feature extraction of spherical objects in image analysis: an application to robotic citrus harvesting. *Computers and Electronics in Agriculture*, 8(1), 57–72. [https://doi.org/10.1016/0168-1699\(93\)90058-9](https://doi.org/10.1016/0168-1699(93)90058-9)
- Postek, W., & Garstecki, P. (2022). Droplet Microfluidics for High-Throughput Analysis of Antibiotic Susceptibility in Bacterial Cells and Populations. *Accounts of Chemical Research*, 55(5), 605–615. <https://doi.org/10.1021/acs.accounts.1c00729>
- Python.org. (2023). *Welcome to Python.org*. <https://www.python.org/>
- Rakszewska, A., Stolper, R. J., Kolasa, A. B., Piruska, A., & Huck, W. T. S. (2016). Quantitative Single-Cell mRNA Analysis in Hydrogel Beads. *Angewandte Chemie International Edition*, 1–5. <https://doi.org/10.1002/anie.201601969>
- Reiser, K. M., Bratton, C. G., Yankelevich, D. R., Knoesen, A., Rocha-Mendoza, I., & Lotz, J. C. (2007). Quantitative analysis of structural disorder in intervertebral disks using second harmonic generation imaging: comparison with morphometric analysis. *Journal of Biomedical Optics*, 12(6), 064019. <https://doi.org/10.1117/1.2812631>
- Rogowska, J. (2000). Overview and Fundamentals of Medical Image Segmentation. *Handbook of Medical Imaging*, 69–85. <https://doi.org/10.1016/B978-012077790-7/50009-6>
- Ruszczak, A., Jankowski, P., Vasantham, S. K., Scheler, O., & Garstecki, P. (2023). Physicochemical Properties Predict Retention of Antibiotics in Water-in-Oil Droplets. *Analytical Chemistry*, 95(2), 1574–1581. <https://doi.org/10.1021/ACS.ANALCHEM.2C04644>
- Rutkowski, G. P., Azizov, I., Unmann, E., Dudek, M., & Grimes, B. A. (2022). Microfluidic droplet detection via region-based and single-pass convolutional neural networks with comparison to conventional image analysis methodologies. *Machine Learning with Applications*, 7, 100222. <https://doi.org/10.1016/J.MLWA.2021.100222>
- Saateh, A., Kalantarifard, A., Celik, O. T., Asghari, M., Serhatlioglu, M., & Elbuken, C. (2019). Real-time impedimetric droplet measurement (iDM). *Lab on a Chip*, 19(22), 3815–3824. <https://doi.org/10.1039/C9LC00641A>
- Sabater, A., Montesano, L., & Murillo, A. C. (2020). Robust and efficient post-processing for video object detection. *IEEE International Conference on Intelligent Robots and Systems*, 10536–10542. <https://doi.org/10.1109/IROS45743.2020.9341600>
- Salomon, R., Kaczorowski, D., Valdes-Mora, F., Nordon, R. E., Neild, A., Farbehi, N., Bartonicek, N., & Gallego-Ortega, D. (2019). Droplet-based single cell RNAseq tools: a practical guide. *Lab on a Chip*, 19(10), 1706–1727. <https://doi.org/10.1039/C8LC01239C>

- Salzmann, M., Hoesel, B., Haase, M., Mussbacher, M., Schrottmaier, W. C., Kral-Pointner, J. B., Finsterbusch, M., Mazharian, A., Assinger, A., & Schmid, J. A. (2018). A novel method for automated assessment of megakaryocyte differentiation and proplatelet formation. *Platelets*, 29(4), 357–364. <https://doi.org/10.1080/09537104.2018.1430359>
- Sangam, S., Jaligam, M. M., Javed, A., Dubey, S. K., & Goel, S. (2020). Droplet Based Microfluidic Electrochemical Detection of Uric Acid, Ascorbic Acid and Dopamine. *ECS Meeting Abstracts*, MA2020-02(57), 3888. <https://doi.org/10.1149/MA2020-02573888MTGABS>
- Sanka, I., Bartkova, S., Pata, P., Ernits, M., Meinberg, M. M., Agu, N., Aruoja, V., Smolander, O.-P., & Scheler, O. (2023). User-friendly analysis of droplet array images. *Analytica Chimica Acta*, 1272, 341397. <https://doi.org/https://doi.org/10.1016/j.aca.2023.341397>
- Sanka, I., Bartkova, S., Pata, P., Smolander, O. P., & Scheler, O. (2021). Investigation of Different Free Image Analysis Software for High-Throughput Droplet Detection. *ACS Omega*, 6(35), 22625–22634. <https://doi.org/10.1021/ACSOMEGA.1C02664>
- Savargaonkar, M., Chehade, A., & Rawashdeh, S. (2021). *RMOPP: Robust Multi-Objective Post-Processing for Effective Object Detection*. <https://arxiv.org/abs/2102.04582v1>
- Scheler, O., Makuch, K., Debski, P. R., Horka, M., Ruszczak, A., Pacocha, N., Sozański, K., Smolander, O., Postek, W., & Garstecki, P. (2020). Droplet digital antibiotic susceptibility screen reveals single-cell clonal heteroresistance pattern in an isogenic bacteria population. *Scientific Reports*, 10, 3282. <https://doi.org/10.1038/s41598-020-60381-z>
- Scheuble, N., Iles, A., Wootton, R. C. R., Windhab, E. J., Fischer, P., & Elvira, K. S. (2017). Microfluidic Technique for the Simultaneous Quantification of Emulsion Instabilities and Lipid Digestion Kinetics. *Analytical Chemistry*, 89(17), 9116–9123. <https://doi.org/10.1021/acs.analchem.7b01853>
- Schindelin, J., Rueden, C. T., Hiner, M. C., & Eliceiri, K. W. (2015). The ImageJ ecosystem: An open platform for biomedical image analysis. In *Molecular Reproduction and Development* (Vol. 82, Issues 7–8, pp. 518–529). John Wiley and Sons Inc. <https://doi.org/10.1002/mrd.22489>
- Schneider, C. A., Rasband, W. S., & Eliceiri, K. W. (2012). NIH Image to ImageJ: 25 years of image analysis. *Nature Methods*, 9(7), 671–675. <https://doi.org/10.1038/nmeth.2089>
- Schulz-Menger, J., Bluemke, D. A., Bremerich, J., Flamm, S. D., Fogel, M. A., Friedrich, M. G., Kim, R. J., Von Knobelsdorff-Brenkenhoff, F., Kramer, C. M., Pennell, D. J., Plein, S., & Nagel, E. (2020). Standardized image interpretation and post-processing in cardiovascular magnetic resonance - 2020 update: Society for Cardiovascular Magnetic Resonance (SCMR): Board of Trustees Task Force on Standardized Post-Processing. *Journal of Cardiovascular Magnetic Resonance*, 22(1), 1–22. <https://doi.org/10.1186/S12968-020-00610-6>
- Sesen, M., & Whyte, G. (2020). Image-Based Single Cell Sorting Automation in Droplet Microfluidics. *Scientific Reports*, 10(1). <https://doi.org/10.1038/S41598-020-65483-2>
- Shang, L., Cheng, Y., & Zhao, Y. (2017). Emerging Droplet Microfluidics. *Chemical Reviews*, 117(12), 7964–8040. <https://doi.org/10.1021/acs.chemrev.6b00848>
- Sierra, A. M. R., Arold, S. T., & Grünberg, R. (2022). Efficient multi-gene expression in cell-free droplet microreactors. *PLoS ONE*, 17(3). <https://doi.org/10.1371/JOURNAL.PONE.0260420>

- Sonka, M., Hlavac, V., & Boyle, R. (1993). Image pre-processing. *Image Processing, Analysis and Machine Vision*, 56–111. https://doi.org/10.1007/978-1-4899-3216-7_4
- Srisa-Art, M., DeMello, A. J., & Edel, J. B. (2007). High-throughput DNA droplet assays using picoliter reactor volumes. *Analytical Chemistry*, 79(17), 6682–6689. <https://doi.org/10.1021/AC0709870>
- Stirling, D. R., Carpenter, A. E., & Cimini, B. A. (2021). CellProfiler Analyst 3.0: Accessible data exploration and machine learning for image analysis. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btab634>
- Stirling, D. R., Swain-Bowden, M. J., Lucas, A., Lucas, A. M., Carpenter, A. E., Cimini, B. A., & Goodman, A. (2021). CellProfiler 4: improvements in speed, utility and usability. *BMC Bioinformatics*. <https://doi.org/10.1186/s12859-021-04344-9>
- Sun, M., Li, Z., & Yang, Q. (2019). μ droPi: A Hand-Held Microfluidic Droplet Imager and Analyzer Built on Raspberry Pi. *Journal of Chemical Education*, 96(6), 1152. <https://doi.org/10.1021/ACS.JCHEMED.8B00975>
- Suzuki, S., Yamaguchi, H., Nakajima, N., & Kawachi, M. (2018). *Raphidocelis subcapitata* (=Pseudokirchneriella subcapitata) provides an insight into genome evolution and environmental adaptations in the Sphaeropleales. *Scientific Reports 2018* 8:1, 8(1), 1–13. <https://doi.org/10.1038/s41598-018-26331-6>
- Svensson, C., Shvydkiv, O., Dietrich, S., Mahler, L., Weber, T., Choudhary, M., Tovar, M., Figge, M. T., & Roth, M. (2018). Coding of Experimental Conditions in Microfluidic Droplet Assays Using Colored Beads and Machine Learning Supported Image Analysis. *Small*, 1802384, 1802384. <https://doi.org/10.1002/sml.201802384>
- Szeliski, R. (2011). *Computer Vision*. <https://doi.org/10.1007/978-1-84882-935-0>
- Szydlowski, N. A., Jing, H., Alqashmi, M., & Hu, Y. S. (2020). Cell phone digital microscopy using an oil droplet. *Biomedical Optics Express*, 11(5), 2328. <https://doi.org/10.1364/BOE.389345>
- Tan, L., & Jiang, J. (2018). Digital signal processing: Fundamentals and applications. *Digital Signal Processing: Fundamentals and Applications*, 1–903. <https://doi.org/10.1016/C2017-0-02319-4>
- Tauzin, A. S., Pereira, M. R., Van Vliet, L. D., Colin, P.-Y., Laville, E., Esque, J., Laguerre, S., Henrissat, B., Terrapon, N., Lombard, V., Leclerc, M., Doré, J., Hollfelder, F., & Potocki-Veronese, G. (2020). Investigating host-microbiome interactions by droplet based microfluidics. *Microbiome*, 8(1), 141. <https://doi.org/10.1186/s40168-020-00911-z>
- Tawfik, D. S., & Griffiths, A. D. (1998). Man-made cell-like compartments for molecular evolution. *Nature Biotechnology*, 16(7), 652–656. <https://doi.org/10.1038/NBT0798-652>
- Tiemeijer, B. M., Sweep, M. W. D., Sleeboom, J. J. F., Steps, K. J., van Sprang, J. F., De Almeida, P., Hammink, R., Kouwer, P. H. J., Smits, A. I. P. M., & Tel, J. (2021). Probing Single-Cell Macrophage Polarization and Heterogeneity Using Thermo-Reversible Hydrogels in Droplet-Based Microfluidics. *Frontiers in Bioengineering and Biotechnology*, 9, 953. <https://doi.org/10.3389/FBIOE.2021.715408/BIBTEX>
- Tlig, L., Bouchouicha, M., Tlig, M., Sayadi, M., & Moreau, E. (2020). A Fast Segmentation Method for Fire Forest Images Based on Multiscale Transform and PCA. *Sensors 2020, Vol. 20, Page 6429, 20(22)*, 6429. <https://doi.org/10.3390/S20226429>
- Uchida, S. (2013). Image processing and recognition for biological images. In *Development Growth and Differentiation* (Vol. 55, Issue 4, pp. 523–549). Dev Growth Differ. <https://doi.org/10.1111/dgd.12054>

- Vallejo, D., Nikoomanzar, A., Paegel, B. M., & Chaput, J. C. (2019). Fluorescence-Activated Droplet Sorting for Single-Cell Directed Evolution. *ACS Synthetic Biology*, 8(6), 1430–1440. <https://doi.org/10.1021/acssynbio.9b00103>
- Vitor, M. T., Sart, S., Barizien, A., Torre, L. G. D. La, & Baroud, C. N. (2018). Tracking the Evolution of Transiently Transfected Individual Cells in a Microfluidic Platform. *Scientific Reports*, 8(1), 1–9. <https://doi.org/10.1038/s41598-018-19483-y>
- Vo, P. Q. N., Husser, M. C., Ahmadi, F., Sinha, H., & Shih, S. C. C. (2017). Image-based feedback and analysis system for digital microfluidics. *Lab on a Chip*, 17(20), 3437–3446. <https://doi.org/10.1039/c7lc00826k>
- Wang, J. D., & Levin, P. A. (2009). Metabolism, cell growth and the bacterial cell cycle. *Nature Reviews. Microbiology*, 7(11), 822. <https://doi.org/10.1038/NRMICRO2202>
- Wang, R., Zhou, Y., Ghanbari Ghalehjoughi, N., Mawaldi, Y., & Wang, X. (2021). Ion-Induced Phase Transfer of Cationic Dyes for Fluorescence-Based Electrolyte Sensing in Droplet Microfluidics. *Analytical Chemistry*, 93(40), 13694–13702. <https://doi.org/10.1021/ACS.ANALCHEM.1C03394>
- Watterson, W. J., Tanyeri, M., Watson, A. R., Cham, C. M., Shan, Y., Chang, E. B., Eren, A. M., & Tay, S. (2020). Droplet-based high-throughput cultivation for accurate screening of antibiotic resistant gut microbes. *ELife*, 9, 1–22. <https://doi.org/10.7554/ELIFE.56998>
- Wiktor, J., Lesterlin, C., Sherratt, D. J., & Dekker, C. (2016). CRISPR-mediated control of the bacterial initiation of replication. *Nucleic Acids Research*, 44(8), 3801. <https://doi.org/10.1093/NAR/GKW214>
- Wolfram Language. (2023). *Image Processing & Analysis—Wolfram Language Documentation*. <https://reference.wolfram.com/language/guide/ImageProcessing.html>
- Xia, D., Xu, X., Zhang, Y., Yao, X., Li, Z., Liu, L., Niu, Z., & Li, H. (2019). Research and analysis of threshold segmentation algorithms in image processing. *Journal of Physics: Conference Series*, 1237(2), 022122. <https://doi.org/10.1088/1742-6596/1237/2/022122>
- Xu, F., Liu, C., Xia, M., Li, S., Tu, R., Wang, S., Jin, H., & Zhang, D. (2023). Characterization of a Riboflavin-Producing Mutant of *Bacillus subtilis* Isolated by Droplet-Based Microfluidics Screening. *Microorganisms*, 11(4), 1070. <https://doi.org/10.3390/MICROORGANISMS11041070/S1>
- Xu, K., Zhu, P., Colon, T., Huh, C., & Balhoff, M. (2017). A Microfluidic Investigation of the Synergistic Effect of Nanoparticles and Surfactants in Macro-Emulsion-Based Enhanced Oil Recovery. *SPE Journal*, 22(02), 459–469. <https://doi.org/10.2118/179691-PA>
- Xue, Y., Luo, X., Xu, W., Wang, K., Wu, M., Chen, L., Yang, G., Ma, K., Yao, M., Zhou, Q., Lv, Q., Li, X., Zhou, J., & Wang, J. (2023). PddCas: A Polydisperse Droplet Digital CRISPR/Cas-Based Assay for the Rapid and Ultrasensitive Amplification-Free Detection of Viral DNA/RNA. *Analytical Chemistry*, 95(2), 966–975. <https://doi.org/10.1021/ACS.ANALCHEM.2C03590>
- Yi, J., Gao, Z., Guo, Q., Wu, Y., Sun, T., Wang, Y., Zhou, H., Gu, H., Zhao, J., & Xu, H. (2022). Multiplexed digital ELISA in picoliter droplets based on enzyme signal amplification block and precisely decoding strategy: A universal and practical biodetection platform. *Sensors and Actuators B: Chemical*, 369, 132214. <https://doi.org/https://doi.org/10.1016/j.snb.2022.132214>

- Yin, J., Chen, X., Li, X., Kang, G., Wang, P., Song, Y., Ijaz, U. Z., Yin, H., & Huang, H. (2022). A droplet-based microfluidic approach to isolating functional bacteria from gut microbiota. *Frontiers in Cellular and Infection Microbiology*, 12. <https://doi.org/10.3389/FCIMB.2022.920986>
- Yu, Y., Wen, H., Li, S., Cao, H., Li, X., Ma, Z., She, X., Zhou, L., & Huang, S. (2022). Emerging microfluidic technologies for microbiome research. *Frontiers in Microbiology*, 13. <https://doi.org/10.3389/fmicb.2022.906979>
- Yue, S., Fang, J., & Xu, Z. (2022). Advances in droplet microfluidics for SERS and Raman analysis. *Biosensors and Bioelectronics*, 198, 113822. <https://doi.org/10.1016/J.BIOS.2021.113822>
- Zamboni, R., Zaltron, A., Chauvet, M., & Sada, C. (2021). Real-time precise microfluidic droplets label-sequencing combined in a velocity detection sensor. *Scientific Reports* 2021 11:1, 11(1), 1–12. <https://doi.org/10.1038/s41598-021-97392-3>
- Zang, E., Brandes, S., Tovar, M., Martin, K., Mech, F., Horbert, P., Henkel, T., Figge, M. T., & Roth, M. (2013). Real-time image processing for label-free enrichment of Actinobacteria cultivated in picolitre droplets. *Lab on a Chip*, 13(18), 3707–3713. <https://doi.org/10.1039/c3lc50572c>
- Zhang, A. B., & Gourley, D. (2009). Digitising material. *Creating Digital Collections*, 55–72. <https://doi.org/10.1016/B978-1-84334-396-7.50005-5>
- Zheng, G., Gu, F., Cui, Y., Lu, L., Hu, X., Wang, L., & Wang, Y. (2022). A microfluidic droplet array demonstrating high-throughput screening in individual lipid-producing microalgae. *Analytica Chimica Acta*, 1227, 340322. <https://doi.org/10.1016/J.ACA.2022.340322>
- Zheng, W., Zhao, S., Yin, Y., Zhang, H., Needham, D. M., Evans, E. D., Dai, C. L., Lu, P. J., Alm, E. J., & Weitz, D. A. (2022). High-throughput, single-microbe genomics with strain resolution, applied to a human gut microbiome. *Science*, 376(6597). <https://doi.org/10.1126/SCIENCE.ABM1483>
- Zhou, P., He, H., Ma, H., Wang, S., & Hu, S. (2022). A Review of Optical Imaging Technologies for Microfluidics. *Micromachines*, 13(2). <https://doi.org/10.3390/MI13020274>
- Zhu, P., & Wang, L. (2016). Passive and active droplet generation with microfluidics: a review. *Lab on a Chip*, 17(1), 34–75. <https://doi.org/10.1039/C6LC01018K>
- Zhu, Y., & Fang, Q. (2013). Analytical detection techniques for droplet microfluidics-A review. In *Analytica Chimica Acta* (Vol. 787, pp. 24–35). Elsevier. <https://doi.org/10.1016/j.aca.2013.04.064>
- Zielke, C., Gutierrez Ramirez, A. J., Voss, K., Ryan, M. S., Gholizadeh, A., Rathmell, J. C., & Abbyad, P. (2022). Droplet Microfluidic Technology for the Early and Label-Free Isolation of Highly-Glycolytic, Activated T-Cells. *Micromachines*, 13(9). <https://doi.org/10.3390/MI13091442/S1>
- Zilionis, R., Nainys, J., Veres, A., Savova, V., Zemmour, D., Klein, A. M., & Mazutis, L. (2017). Single-cell barcoding and sequencing using droplet microfluidics. *Nature Protocols*, 12(1), 44–73. <https://doi.org/10.1038/nprot.2016.154>

Acknowledgments

Above all, I would like to express my deepest gratitude to God for His abundant blessings and guidance throughout my PhD program. His presence has been evident in every aspect of my journey, providing me with motivation, strength, and miraculous interventions during the writing period. I am truly grateful for His support.

I am immensely thankful to my beautiful wife, who entered my life during my PhD study. Her presence has been a source of inspiration and a pillar of strength. I am amazed by the way our marriage has strengthened my spirit, allowing me to maintain focus and stay on track with my studies. Her support and presence through all the ups and downs of this journey have been invaluable, and I cannot fully express my gratitude for her kindness and love.

I extend my heartfelt appreciation to my parents and parents-in-law for their support and encouragement. Their constant reassurance and prayers have boosted my motivation and reminded me that everything will eventually be fine. I am truly blessed to have such loving and supportive family members in my life.

Special gratitude goes to my supervisors, Prof. Ott Scheler and Prof. Olli-Pekka Smolander, whose unwavering support, invaluable guidance, and expertise have been instrumental in shaping the direction and quality of this research journey. I consider myself fortunate to have had the opportunity to work under their mentorship. Additionally, I would like to acknowledge the funding sources that have supported my research, including TTU Development Program 2016–2022 (project no. 2014–2020.4.01.16.0032), and Estonian Research Council grant PRG620.

I would like to express my sincere appreciation to my colleagues, collaborators, and fellow researchers who have been constant sources of motivation, inspiration, and intellectual exchange, including the Lab on A Chip group and the Smart Analytics group. I am especially grateful to Simona Bartkova, Ph.D., whose collaboration, and contributions have played a crucial role in achieving the results presented in my articles and thesis. Without her support, I believe my progress would not have been as great.

To all my friends in Tallinn and my Synthetic Biology community (Synbio.id) back in my home country, I am deeply grateful for their friendship and support. Their presence and the enjoyable activities we shared have made my PhD study journey more enjoyable and meaningful. I would like to extend my gratitude to Aulia, Nurul, Alfi, Farhan, and all the advisors and executive committee members of Synbio.id for their valuable contributions.

I would also like to acknowledge the insightful and inspiring individuals from different regions and time zones who have provided support. Their positive influence and constructive discussions have motivated me to continuously strive for excellence in my project. I am thankful for their presence in my life.

Lastly, I want to dedicate this thesis to myself, acknowledging the hard work and dedication I have invested throughout this journey. The countless long nights and sacrificed weekends were a testament to my determination and commitment. This achievement is a reflection of my perseverance and growth.

The completion of this doctoral thesis has been made possible through the collective support, guidance, and contributions of the individuals mentioned above. Their belief in me and their invaluable presence have shaped my academic and personal development. I am forever grateful for their contributions to my success.

Abstract

User-friendly analysis of droplet experiments

Droplet emulsion techniques have revolutionized high-throughput experiments in biological and chemical research. By encapsulating reactions within droplets, researchers can achieve parallelization on a massive scale, saving time and effort. This method involves the use of immiscible liquids, typically water and oil, with the addition of a surfactant to stabilize the droplets. This technique has greatly improved existing methods, such as ddPCR and high-throughput drug screening, and has led to advancements in molecular detection. However, despite its robustness, the droplet emulsion method often requires sophisticated setups, equipment, and custom analytical pipelines to understand experimental results.

Imaging is one of the most accessible methods for data acquisition, especially in droplet-based experiments, particularly using brightfield or fluorescence microscopy. Although imaging is a widely used method, the analysis of droplet images poses challenges due to the need for complex pipelines, which are not always clearly described in published articles, as well as programming skills. These obstacles present difficulties for new researchers and users without a background in image analysis.

This doctoral thesis focuses on providing a comprehensive platform for the detection, classification, and analysis of droplets in various experiments. The aim is to simplify and streamline the process for researchers with different levels of expertise and to develop user-friendly analytical pipelines for different droplet experiment scenarios.

Recognizing the importance of analytical tools for droplet-based experiments, especially for users with limited time and resources, this thesis accomplished four goals: 1) identified the most user-friendly types of software which can be used to detect droplets and assessed their performance based on accuracy, precision, and detection time, 2) determined two approaches to classifying droplets by manual thresholding and the user-friendly droplet classification CellProfilerAnalyst, 3) developed the user-friendly analytical web application EasyFlow to process droplet array images, and 4) tested analytical pipelines which were developed to analyze bacteria-based experiments, microalgae growth experiments, and microplastic bead experiments. These goals eliminate the need for complex exploration and enable users to detect droplets or objects of interest more easily.

Lühikokkuvõte

Kasutajasõbralik analüütiline lähenemine tilgapõhiste katsetele

Droplet-emulsiooni tehnikad on muutnud suure läbilaskevõimega katsed bioloogilistes ja keemilistes uuringutes palju kättesaadavamaks. Reaktsioonide kapseldamine tilkadesse võimaldab teadlastel saavutada kõrget katsete paralleelsust, mis omakorda säästab aega ja vaeva. Selle meetodi kasutamiseks on vaja segunematuid vedelikke, tavaliselt vett ja õli, ning pindaktiivset ainet (surfaktanti), mis stabiliseerib tilgad. Tilkade emulsioonitehnika on märkimisväärselt täiustanud juba olemasolevaid meetodeid, nagu ddPCR (digitaalne tilkpõhine polümeraasi ahelreaktsioon) ja suure läbilaskevõimega ravimite sõelumine, ning võimaldanud olulisi edusamme molekulaardiagnostikas. Siiski nõuab tilkade emulsioonimeetod tihti keerukaid lahendusi, spetsiifilist varustust ja analüütilisi meetodeid eksperimentide tulemuste analüüsiks.

Tilgapõhiste katsete tulemuste analüüsimisel kasutatakse tihti pildistamist, näiteks helevälja või fluorestsentsmikroskoopiat. Kui pildistamine on laialdaselt kasutatav meetod, siis saadud piltide analüüs on keerukam kuna vajab tihti keerukaid meetodeid, mida avaldatud artiklites alati selgelt ei kirjeldata ning kohati ka programmeerimis oskust. See tekitab raskusi uutele teadlastele ja kasutajatele, kellel puudub vastav pildianalüüsi taust.

Antud doktoritöö keskendub tervikliku platvormi väljatöötamisele tilkade tuvastamise, klassifitseerimise ja analüüsimise jaoks erinevates katsetes. Eesmärgiks on lihtsustada ja tõhustada protsessi erineva taustaga teadlastele ning luua kasutajasõbralikud analüütilised töövood erinevate katsestsenaariumide jaoks, mis kasutavad eksperimentides tilkasid.

Mõistes analüütiliste tööriistade tähtsust tilgapõhistes katsetes, eriti arvestades kasutajatel tihti olevat piiratud aega ja muid resursse, tõi see doktoritöö neli tulemust: i) tilkade tuvastamise platvormi erinevate tilkade katsete jaoks, ii) sobiv platvorm tilkade klassifitseerimiseks, iii) lihtsustatud tilkade massiivne andmeanalüüs ning iv) kasutajasõbralikud analüütilised töövood erinevate katsestsenaariumide jaoks tilkades. Nende eesmärkide saavutamiseks välditakse vajadust keeruliste analüüside järele ning võimaldatakse kasutajatel lihtsamalt tuvastada tilgad ja/või nendes leiduvad huvipakkuvad objektid.

Appendix 1

Publication I

Bartkova, S., Vendelin, M., Sanka, I., Pata, P., Scheler, O. (2020). Droplet image analysis with user-friendly freeware CellProfiler. *Analytical Methods*, 12(17), 2287–2294.



Cite this: *Anal. Methods*, 2020, 12, 2287

Received 6th January 2020
Accepted 25th March 2020

DOI: 10.1039/d0ay00031k

rsc.li/methods

Droplet image analysis with user-friendly freeware CellProfiler†

Simona Bartkova,^a Marko Vendelin,^b Immanuel Sanka,^a Pille Pata^a and Ott Scheler^{*a}

Droplet microfluidic assays are rapidly gaining popularity because they enable analysis of biochemical reactions, individual cells or small cell populations with high sensitivity, precision and accuracy in a high-throughput manner. Nonetheless, there is a demand for user-friendly and low-cost droplet analysis technology. In this article, we meet this demand by developing two droplet analysis pipelines via free open-source software CellProfiler (CP) and its companion (CellProfiler Analyst, CPA). To illustrate the competence of the pipelines as an independent, and freely accessible droplet analysis tool for any researcher without the need of programming skills, we show (i) droplet digital quantification of viable fluorescent bacteria using single-color images and (ii) analysis of a multi-color fluorescence image with droplets containing different chemical compositions.

Introduction

Droplet microfluidic assays are increasingly finding use in microbial and nucleic acid analysis assays. In microbiology, droplet microfluidics has opened up several new experimental possibilities.¹ Examples of microfluidic applications include antibiotic susceptibility studies at the population and the single cell level,² investigation of microbial interactions,³ and quantification of viable bacteria.⁴ Due to its high precision, sensitivity and robust automatic calibration, droplet digital PCR (ddPCR) has become a strong alternative to traditional quantitative PCR (qPCR) in quantifying nucleic acids in both diagnostics and laboratory research.^{5,6} Both, in microbiology and nucleic acid analyses, the detection of positive signals in droplets is usually achieved by measuring their fluorescence.^{1,7}

Analysis of detected droplets often comprises of analysis software coupled to specific droplet platforms. However, the availability outside designated microfluidics laboratories is

limited. Such a limitation commonly reflects the lack of transferability and user-friendliness of the platforms for non-specialists. Commercial droplet analysis platforms have overcome some of these limitations through the development of ddPCR⁵ e.g. from Bio-Rad,⁸ RainDance Technologies⁹ and Stilla Technologies.¹⁰ Their application is designed to be comprehensible by non-specialists. This can also be expanded beyond nucleic acids to include other fluorescent targets, like eukaryotic cells and microbes. Each of the companies provide their own combined solution for droplet handling fluidics, fluorescence imaging equipment and its data analysis tools. This experiment is either done by reading fluorescence droplets in a microfluidic channel one-by-one^{8,9} or imaging droplets in a monolayer 2D array format.¹⁰ The main disadvantage of these platforms is their limited accessibility to a wider audience due to high initial hardware and software acquisition costs.⁵

Many microfluidics laboratories have instead developed their own custom platforms for droplet detection. Similar to commercial platforms, they are based on either the detection of droplets in microfluidic channels^{11–13} or imaging a 2D monolayer array.^{14,15} In both cases, the droplet fluorescence analysis usually involves developing a custom script based on e.g. Labview,^{2,16} ImageJ,¹⁵ Matlab,^{17–19} FluoroCellTrack,²⁰ or Open Source Computer Vision Library (OpenCV).^{21,22} However, scripting such analytical tools and customizing them to meet the needs of specific experimental assays in the lab requires expertise in scripting and programming. This is not always sufficiently available in traditional biology and chemistry laboratories. Recently developed inexpensive and portable smartphone-based platforms solve this concern through custom made droplet analysis phone apps.^{23,24} Both platforms show great potential for point-of-care application; nonetheless, the

^aDepartment of Chemistry and Biotechnology, Tallinn University of Technology, Akadeemia tee 15, 12618 Tallinn, Estonia. E-mail: ott.scheler@taltech.ee

^bDepartment of Cybernetics, Tallinn University of Technology, Akadeemia tee 15, 12618 Tallinn, Estonia

† Electronic supplementary information (ESI) available: ESI Fig. S1 and S2 showing the cross-validation results of our constructed training data and model in CPA classifier. ESI Fig. S3 illustrating histograms based on data from the analyzed multicolor fluorescence image.³³ ESI Table S1 with all exported data from our dataset through “ExportToSpreadsheet” module. ESI Table S2 containing the comparison of the ability of CP identifying droplets and CPA identifying viable bacteria *versus* results obtained by manual counting. Finally, a detailed guide for droplet analysis using our two publicly available pipelines (<https://github.com/taltechmicrofluidics/CP-for-droplet-analysis>) with software CellProfiler (version 3.1.8) and CellProfiler Analyst (version 2.2.1) (pdf). See DOI: 10.1039/d0ay00031k

technology is still bound to (i) specific droplet platforms and (ii) the target of interest (*i.e.* nucleic acids²³ and viable bacteria²⁴). In addition, the applications are currently still inaccessible online.

Here, we argue that there is an unmet need for a ready-to-implement droplet analysis technology that is (i) easily adaptable, regardless of laboratory set-up, biological targets and personnel; and at the same time (ii) affordable for researchers in conventional chemistry and biology laboratories. For example, depositing droplets as a 2D array on modified microscope slides for imaging is a cost-effective and easy-to-learn approach to introduce high-throughput droplet-based screening for diverse research environments.¹⁷ Suitable software for the analysis of such arrays are ImageJ^{15,19} and Matlab,^{17,19} though both of them have a steep learning curve.

In this paper, we address this issue and describe a droplet analysis technology based on free open-source software CellProfiler (CP) and its companion CellProfiler Analyst (CPA). The software was created by the Broad Institute Imaging Platform and have enabled biologists without training in scripting or programming to quantitatively measure phenotypes from high-throughput fluorescence microscopy images.²⁵ CPA was designed to provide user-friendly tools for the interactive

exploration and analysis of the data created by CP.²⁶ This includes a supervised machine learning tool called "Classifier" that can be trained to recognize phenotypes and automatically score millions of cells.²⁶

Results and discussion

We demonstrate the applicability of our technology using droplet digital quantification of viable fluorescent bacteria. We generate water-in-oil droplets using a microfluidic chip with flow-focusing geometry.² Droplets contain diluted bacterial cells, growth media, and the fluorescent tracer dye fluorescein isothiocyanate (FITC) that generates a low intensity fluorescent background needed for subsequent identification of droplets (Fig. 1A). For visualization, we deposit a droplet monolayer on a modified microscope slide and image them with a fluorescence widefield microscope (Fig. 1B).

We introduce droplet images to our CP image analysis pipeline, by importing raw Tagged Image File (TIF) format images with droplets into the constructed CP pipeline. CP employs Bio-Formats to read input images and can currently read more than 100 available file formats such as BMP, GIF, JPG, PNG, and TIF.²⁷ Lossless formats, such as TIFF, are

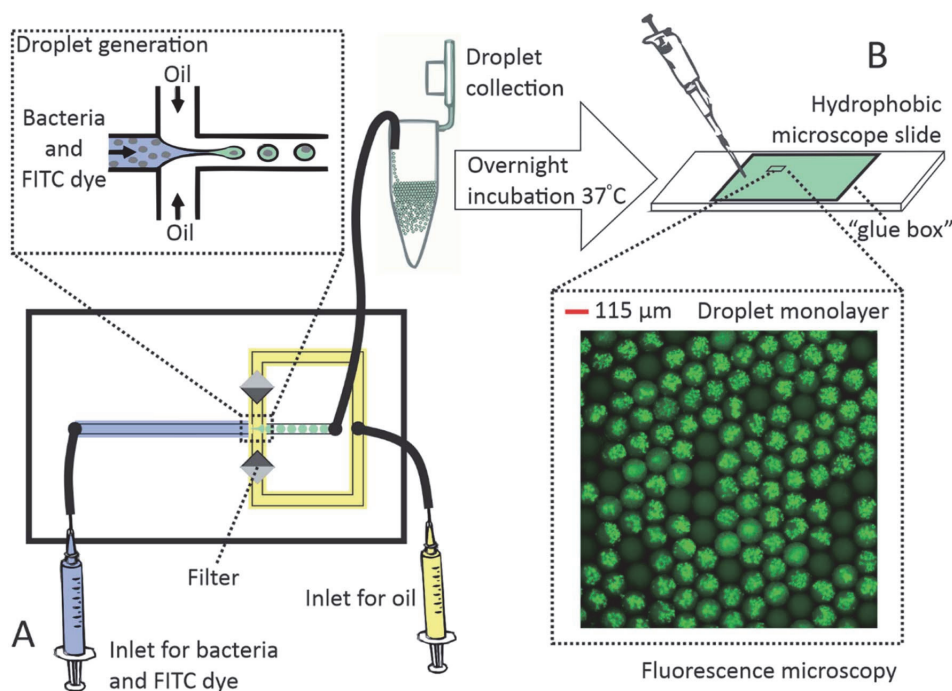


Fig. 1 Droplet generation and microscopy. (A) We generate water-in-oil droplets with an average diameter of $\sim 115\ \mu\text{m}$, containing bacterial cells, growth media and FITC tracer dye, followed by overnight incubation of droplets at $37\ ^\circ\text{C}$; (B) next, we pipette the overnight-incubated droplets onto a modified hydrophobic microscope slide where they form a monolayer. Then, we fence the droplets by a previously deposited rectangular elevation made of super glue, which we call a "glue box". Next, we loosely cover the "glue box" with a coverslip. Finally, we visualize the 2D array droplets using a fluorescence widefield microscope. In this experiment, we analyze ~ 6900 droplets on a single microscope slide.

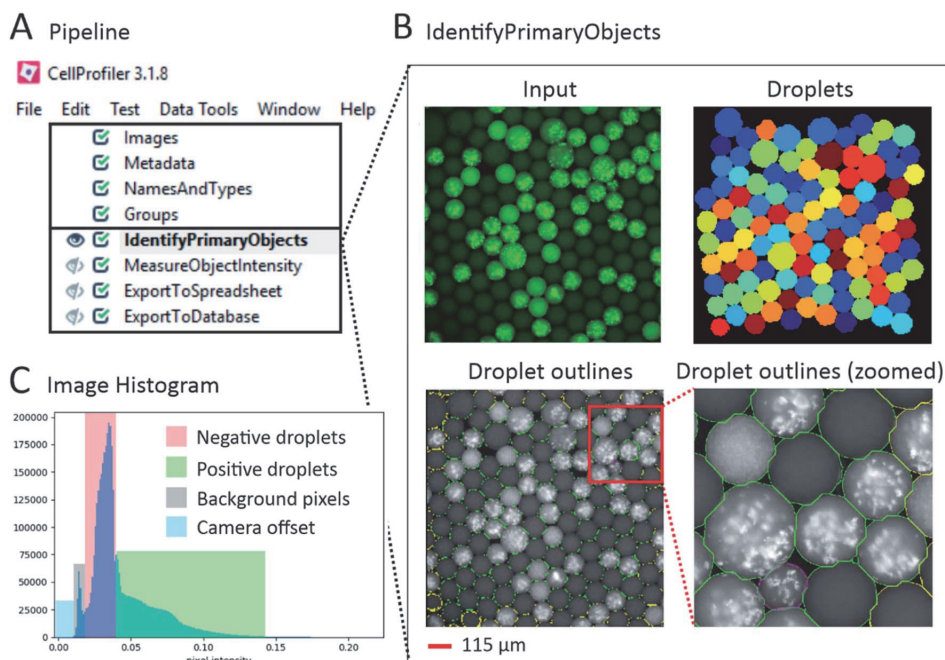


Fig. 2 Image analysis with CellProfiler. (A) Overview of our CellProfiler pipeline for analyzing droplet images; (B) we use the CellProfiler module “IdentifyPrimaryObjects” for identifying droplets in each image. The module identifies objects (in our case droplets) based on multiple parameter settings. Key parameters include: (i) the typical diameter of objects, (ii) threshold strategy (automated strategies or manual), (iii) method for distinguishing and drawing lines between clumped objects (shape or intensity), (iv) discarded objects outside of the diameter range, and (v) discarded objects touching the border. We color the “Input image” green for visualization purposes, but all images we use for analysis are in greyscale. In the “Droplets outlines” image, all the found droplets with fluorescence intensity units above our threshold have markings around them. Those that are within the correct diameter range are marked with a green outline. Droplets with a yellow marking are discarded because they are touching the image border and the small droplet with the purple marking is discarded due to having a diameter outside the designated range; (C) the image histogram from the module “Images” depicting four typical pixel intensity sections in our images: camera offset, background, negative droplets and positive droplets.

preferred²⁷ and pixels are kept stable, suitable for scientific image data.²⁸

The CP pipeline we present identifies droplets and measures their relative fluorescence. We created a pipeline to identify and measure the relative fluorescence intensity of droplets in our images (Fig. 2A). To identify droplets, we use the module “IdentifyPrimaryObjects” (Fig. 2B). This module has several features, including thresholding and object classification, which is usually used for the identification of objects in grayscale images. The conditions for these features are specified by the user. Thresholding turns image pixels into values, which are then classified into two types (black and white).²⁹ A specific value is set as the limit for each pixel type. This limit is applied for the segmentation of objects.³⁰

In CP, there are different types of automatic (*e.g.* Otsu and Robust Background) and manual thresholding methods, which are all explained in CP. We determined that the manual threshold method provides the most optimal droplet detection. By use of the module “Images” in CP (Fig. 2A), we visualized the image pixel intensity distribution (Fig. 2C). The dip between the

histogram peak depicting image background pixels and the peak depicting image droplet pixels (Fig. 2C), enables us to find a suitable threshold range. We test the different values within the range of 0.021–0.024 in CPs “Test Mode” option on 30 of our images to find the most optimal threshold value for our dataset. The selection of the manual threshold and sensitivity of this method is dependent on the user’s bacterial strain (or other particles of interest), droplet incubation time, microscope hardware, and used labeling techniques.^{4,31,32} In our experiment, droplets with viable bacteria and empty droplets have similar fluorescence intensities right after droplet formation so we use overnight incubation to obtain a clear threshold separation between the two groups. We also recommend using the same excitation light and readout sensitivity in practice to simplify analysis. We then apply the “MeasureObjectIntensity” module to measure the relative fluorescence of all droplets. Measurement data is automatically saved as either an .h5 file or a .mat file, based on the user’s preference. The module “ExportToSpreadsheet” (optional) exports all measurements as a .csv file, which can be opened in other programs such as

Analytical Methods

Microsoft Excel (see the ESI Table S1†), R, or Python. Finally, we apply the module “ExportToDatabase” to create a SQLite database with all data measurements for further analysis in CPA.

We use the data generated by the CP pipeline to calculate the number of viable fluorescent bacteria in the sample. We import the data from the CP pipeline stored in the SQLite database into CPA, where we (i) train and employ the CPA Classifier tool to calculate the number of viable fluorescent bacteria in the sample (Fig. 3A) and (ii) construct a histogram to display the frequency distribution of the mean of the relative fluorescence intensity (Fig. 3B). Although additional classes can be added to CPA Classifier, in our case we have a simple system where we

want to distinguish between droplets containing viable bacteria and empty droplets. We evaluate our constructed training dataset and model by 5 fold cross-validation displayed in two ways. For each fold, our training dataset is subdivided further into a training and testing set, where the algorithm is first trained on the training set and then evaluated on the test set. CPA displays accuracy of classification through a confusion matrix, a table that describes how well the cross-validation model for classifying droplets performs compared to the actual training dataset with the predefined two classes that we construct.²⁶ The results yield a 99.50% accuracy (see the ESI Fig. S1†). A classification report is also displayed by CPA to

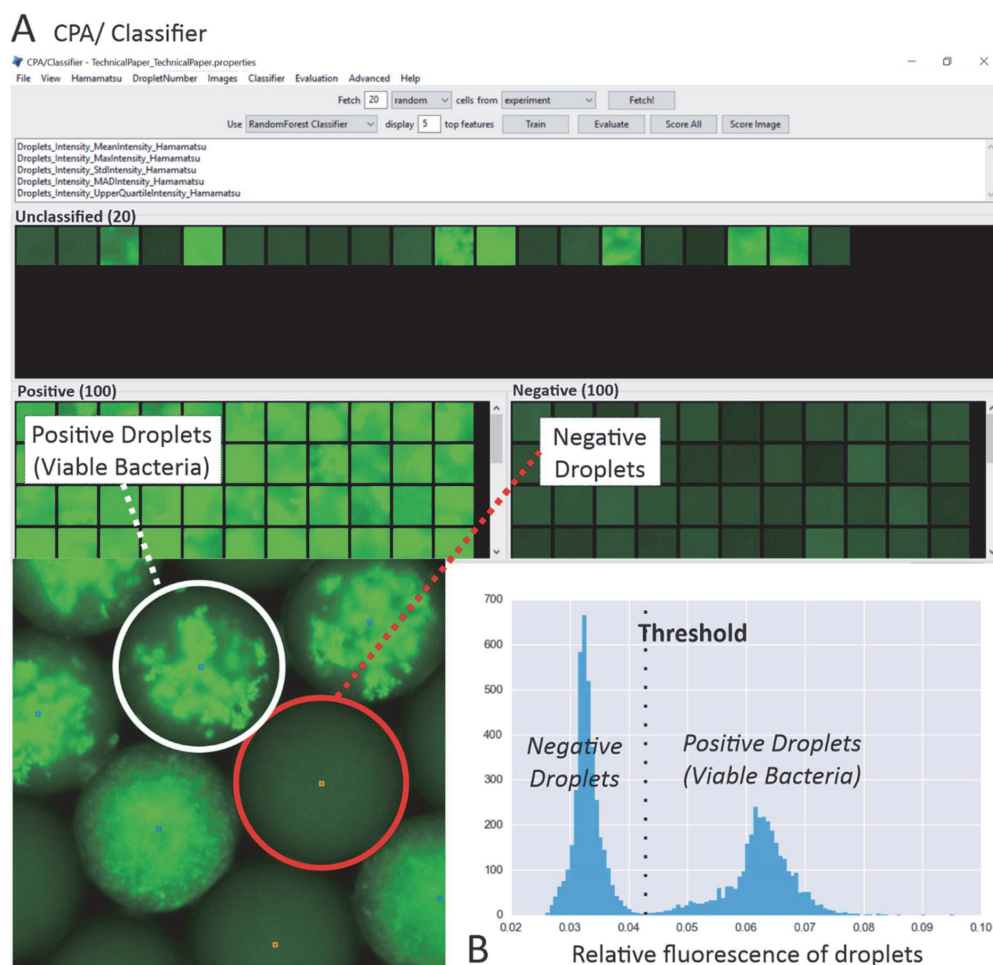


Fig. 3 Data visualization and classification with CellProfiler Analyst tools. (A) We train the supervised machine learning tool “Classifier” to automatically classify negative and positive droplets. The overall positive fraction of bacteria is $3153/6899 = 0.45$, that translates into the mean bacteria cells per droplet $\lambda = 0.61$ based on the Poisson distribution model. This correlates with our aim during droplet formation of having roughly the same amount of positive and negative droplets in our sample for illustration purposes. For single cell experiments the value is recommended to be around 0.1; (B) we visualize data via the “Histogram” tool, based on the mean fluorescence intensity units of droplets. For visualization purposes, we draw a manual threshold based on the depicted mean fluorescence between the fluorescence peak of negative and positive droplets, respectively.

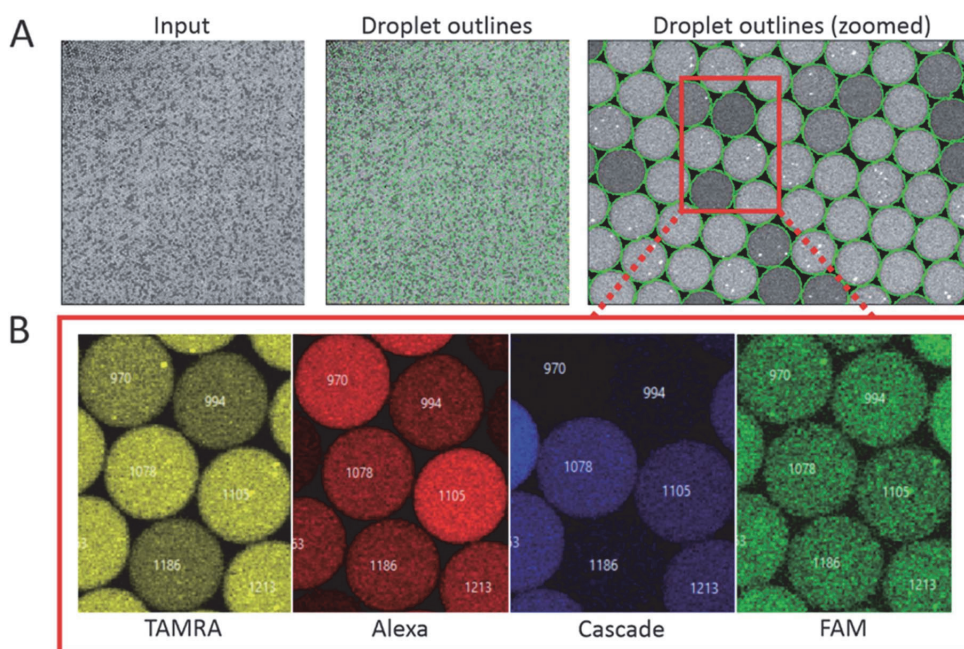


Fig. 4 Image analysis of the multicolor fluorescence image from the study by Genot *et al.* (2016)³³ with CellProfiler (CP). (A) We use module “IdentifyPrimaryObjects” for identifying droplets in the aligned image depicting the fluorescent dye 5-carboxytetramethylrhodamine (TAMRA). The key parameters for droplet identification are listed in Fig. 2. Two parameters were adjusted from our data set to this: (i) the manual threshold and (ii) typical diameter of objects. The “Droplets outlines” image depicts the ~10,000 identified droplets (same as reported in the original study³³), which had fluorescence intensity units above the manually set threshold; (B) we import the data generated by CP into CellProfiler Analyst to visualize the fluorescence intensities of the four dyes within five representative droplets. The dyes include: fluorescein (FAM), 5-carboxytetramethylrhodamine (TAMRA), dextran Alexa Fluor 647 (Alexa), and dextran Cascade Blue (Cascade).

evaluate precision and recall metrics and the F1-score for the two classes. This results in values of 0.99 and 1.00 for the negative and positive classes respectively (see the ESI Fig. S2†).

To further demonstrate that this method has broad application across many systems, we also apply our method to a multicolor fluorescence image from the study by Genot *et al.* (Nat. Chem., 2016).³³ By using the image of droplets containing various chemical compositions, the study investigated a bistable DNA switch reaction network. Among other ESI,† the study provides readers with four images, each depicting one of the four fluorescent dyes present in droplets.³³ To ensure droplets in all four images are aligned prior to droplet identification in CP, we add module “Align” before module “IdentifyPrimaryObjects” in our pipeline. We use the aligned image depicting the fluorescent dye TAMRA³³ to adjust the manual threshold and typical diameter range in “IdentifyPrimaryObjects” to identify droplets (Fig. 4), which are subsequently used for measuring the relative fluorescence intensity of each of the four dyes within the droplets by module “MeasureObjectIntensity”. To illustrate the accuracy of our method, we use the data generated by our CP pipeline to replicate results from the study shown in their Fig. S16 (ref. 33) (ESI Fig. S3†).

Finally, we compare the overall ability of CP to identify droplets in our fluorescence images and the external multicolor fluorescence image with results obtained by manual counting. In both cases CP identifies slightly fewer droplets (see the ESI Table S2†). This is caused by some droplet size variation and the lack of optimal image quality, which highlights the importance of obtaining uniform and good quality droplets, as well as high quality images for enabling the best possible analysis.

Based on our experience, we have found that CP has an intuitive user interface that is easy to learn for the general audience. Even though some of the features need to be adjusted to meet the specific criteria, this software can give some room for the user to find their expected result. Because of this user-friendly interface, no prior experience in programming is needed for using CP or its companion CPA.³⁴ Taken together with the available online examples, tutorials, manuals and discussion forums, CP is an easy-to-learn analysis software for biologists and other scientists without a background in bioinformatics.³⁴ The combination of ease-of-use and flexibility of this technology moreover extends its scope across many scientific fields where the use of droplet microfluidic has shown promising results *e.g.* generating microspheres and vesicles

Analytical Methods

with various components^{35,36} and clinical settings *e.g.* antimicrobial susceptibility testing.^{5,37}

Conclusion

In this article, we presented a droplet analysis technology based on the free open-source software CellProfiler and successfully analyzed thousands of droplets with viable fluorescent bacteria and a multicolor fluorescence image with droplets containing different chemical compositions. The droplet analysis technology we described here is easily adaptable, affordable and automated. This makes it an excellent software tool for assisting researchers in conventional laboratories around the world, particularly to extract and analyze quantitative data from digital droplet assays containing thousands of droplets and more. The advantages of this technology include high adaptability, speed, and accessibility at zero cost. The main disadvantage in our case was the inability to import certain file types directly into the software (*e.g.* Obtained Hierarchical Data Format 5 [HDF5] files). In the end, whether one decides to implement this or another droplet analysis technology, the most critical part is acquiring high-quality images prior to analysis, thereby taking advantage of less manual work, faster analysis and higher adaptability.

Materials and methods

Droplet generation and collection

We generate water-in-oil droplets with an average diameter of 115 μm using a poly(dimethylsiloxane) (PDMS) microfluidic chip with flow-focusing geometry² (Fig. 1A). Droplets contain an overnight culture of *Escherichia coli* JEK 1036 with a chromosome-incorporated gene encoding the green fluorescence protein (GFP). Bacteria are grown in Luria broth mixed with FITC dye, roughly diluted to a bacterial density yielding a 50 : 50 ratio of positive and negative droplets and 1 $\mu\text{g mL}^{-1}$ of FITC (Fig. 1A). We use Novec HFE 7500 fluorocarbon oil with 2% concentration of perfluoropolyether (PFPE)-poly(ethylene glycol) (PEG)-PFPE triblock surfactant for the continuous phase. The surfactant was a kind gift from Professor Piotr Garstecki from the Institute of Physical Chemistry, Polish Academy of Sciences. After collection in a 1.5 mL Eppendorf tube, we incubate the droplets overnight at 37 °C.

Droplet imaging

We construct a 1.5 \times 1 cm rectangular elevation with a height of 250 μm of four layers of super glue (Loctite Precision), which we call the “glue box”, on a microscope slide (Kalek, $\pm 76 \times 26$ mm). After drying, we silanize the “glue box” for 3 h using tri-decafluoro-1,1,2,2-tetrahydrooctyl-1-trichlorosilane vapors (United Chemical Technologies, USA).

We pipette the droplets onto a hydrophobic microscope slide where they form a monolayer inside the “glue box”, which we then loosely cover with a coverslip (Kalek, 18 \times 18 mm) (Fig. 1B). Then we acquire the fluorescence intensity of the bacteria (GFP) and FITC using a fluorescence widefield

microscope with the following settings: objective UPlanFL N 10 \times /0.3, excitation LED 470 nm (at 40%), excitation filter 461–483 nm, excitation dichronic 488 LPXR, and emission filter set to 550/88 HC.

CellProfiler pipeline

We use CP (version 3.1.8)^{25,34} to design and execute the pipeline. Prior to importing into CP, we convert all 64 acquired images in the HDF5 format to the grayscale TIF format, owing to CP using Bio-Formats to read input images.²⁷ We import the folder containing the TIF format images into the CP “Images” module. Next, we use the “IdentifyPrimaryObject” module to find objects of interest (droplets) in the analyzed images. To find all droplets based on manual examination of 30 randomly selected images from the experiment, we set the threshold manually to 0.023. We enhance the image contrast during inspection for visualization purposes, yet use only original grayscale images for analysis. Next, we set the typical diameter range of found droplets to 150–250 pixels. All droplets found outside of the diameter range or touching the border of the image are discarded. The separation of clumped droplets is based on the shape. Then we use the “MeasureObjectIntensity” module to measure the pixel intensity units of the objects, with the intensity range set to Image Metadata in the “NamesAndTypes”. A detailed explanation of all the available intensity ranges can be found in the CP online manual (<http://cellprofiler-manual.s3.amazonaws.com/CellProfiler-3.1.8/index.html>). We export the data as a .csv file to allow optional analysis through other software using the “ExportToSpreadsheet” module. We also export the data to the database SQLite and create a properties file using the “ExportToDatabase” module to enable further analysis of data in CPA. Finally, CP also automatically creates a data output file in the HDF5 format.

For analysis of the external multicolor fluorescence image,³³ we add an additional module “Align” to our constructed pipeline, which we place prior to module “IdentifyPrimaryObjects”. We use this module to align the droplets in all channels (*i.e.* in the four grayscale images, each depicting a respective dye Alexa, Cascade, FAM and TAMRA). The only differences in module “IdentifyPrimaryObject” are the manual threshold settings and typical diameter range of the found objects, which are adjusted to 0.12 and 40–70 pixels respectively, based on the aligned image depicting the fluorescent dye TAMRA. We then use these found objects (droplets) in the “MeasureObjectIntensity” module to measure the pixel intensity units of each of the four dyes (FAM, TAMRA, Alexa, and Cascade) within all the identified droplets using the same settings as before. We export all data the same way as explained above.

The two pipelines described here are publicly available at our repository on GitHub (<https://github.com/taltechmicrofluidics/CP-for-droplet-analysis>). An accompanying detailed guide for droplet analysis is available as online ESI.†

CellProfiler analyst

We use the properties file created by CP to facilitate the access of CPA (version 2.2.1)²⁶ to the SQLite database containing the

CP pipeline data. We visualize the mean fluorescence intensity units of droplets by using the CPA histogram tool (Fig. 3A). In the CPA supervised machine learning tool “Classifier”, we fetch 200 random droplets from the images and use them to manually create two classification categories: positive (with bacteria) and negative (without bacteria) (Fig. 3B). Each category contains 100 droplets that the classifier can use for training. Based on the random forest algorithm,³⁸ the classifier acquires the following five top features it utilizes for automatic classification: MeanIntensity (average pixel intensity within an object), UpperQuartileIntensity (intensity value of the pixel for which 75% of the pixels in the object have lower values), MaxIntensity (maximal pixel intensity within an object), StdIntensity (standard deviation of the pixel intensities within an object), and MADIntensity (median absolute deviation value, defined as the median $[|x_i - \text{median}(x)|]$, of the intensities within the object). We then evaluate and apply the training set and model to the 64 images to classify all the droplets. For future use, we also save the training set as a .csv file and our model as a .model file. Finally, based on the positive fraction of bacteria, we calculate the mean bacterial cells per droplet using Poisson distribution.²

For CPA analysis of the external dataset of a multicolor fluorescence image,³³ we produce the following graphs as done by Genot *et al.* (2016)³³ shown in their Fig. S16: (i) two density histograms through CPA's density plot tool depicting droplet mean fluorescence intensity units (Alexa, Cascade) and (FAM, TAMRA) (ESI Fig. S3A†) (ii) four histograms with droplet mean fluorescence intensity units of each of the four dyes (FAM, TAMRA, Alexa, and Cascade) by use of the CPA histogram tool (ESI Fig. S3B†).

For a more detailed guide on all droplet analysis in CPA, consult the detailed guide “CellProfiler and CellProfiler Analyst Guide” in the online available ESI.†

Author contributions

The manuscript was written through contributions of all authors. All authors have approved the final version of the manuscript.

Conflicts of interest

There are no conflicts to declare.

Acknowledgements

Most of the work was carried out in the laboratory set up with the support from the TTÜ development program 2016–2022”, project code 2014-2020.4.01.16-0032. We also acknowledge the Estonian Research Council grants MOBTP109 (OS), PRG620 (OS), IUT33-7 (MV) and MOBJD556 (SB). We are very grateful for Prof. Piotr Garstecki at the Institute of Physical Chemistry, Polish Academy of Sciences for providing us the surfactant and microfluidic chip mold.

References

- 1 T. S. Kaminski, O. Scheler and P. Garstecki, *Lab Chip*, 2016, **16**, 2168–2187.
- 2 O. Scheler, K. Makuch, P. R. Debski, M. Horka, A. Ruszczak, N. Pacocha, K. Sozanski, O.-P. Smolander, W. Postek and P. Garstecki, *bioRxiv*, 2019, 328393.
- 3 J. Park, A. Kerner, M. A. Burns and X. N. Lin, *PLoS One*, 2011, **6**, e17019.
- 4 O. Scheler, N. Pacocha, P. R. Debski, A. Ruszczak, T. S. Kaminski and P. Garstecki, *Lab Chip*, 2017, **17**, 1980–1987.
- 5 L. Cao, X. Cui, J. Hu, Z. Li, J. R. Choi, Q. Yang, M. Lin, L. Ying Hui and F. Xu, *Biosens. Bioelectron.*, 2017, **90**, 459–474.
- 6 L. Gorgannezhad, H. Stratton and N.-T. Nguyen, *Micromachines*, 2019, **10**, 408.
- 7 O. Scheler, W. Postek and P. Garstecki, *Curr. Opin. Biotechnol.*, 2019, **55**, 60–67.
- 8 C. B. Hughesman, D. X. J. Lu, K. Y. P. Liu, Y. Zhu, C. F. Poh and C. Haynes, *PLoS One*, 2016, **11**, e0161274.
- 9 S. Rutsaert, K. Bosman, W. Trypsteen, M. Nijhuis and L. Vandekerckhove, *Retrovirology*, 2018, **15**, 16.
- 10 J. Madic, A. Zocovic, V. Senlis, E. Fradet, B. Andre, S. Muller, R. Dangla and M. E. Droniou, *Biomol. Detect. Quantif.*, 2016, **10**, 34–46.
- 11 D. Pekin, Y. Skhiri, J.-C. Baret, D. Le Corre, L. Mazutis, C. Ben Salem, F. Millot, A. El Harrak, J. B. Hutchison, J. W. Larson, D. R. Link, P. Laurent-Puig, A. D. Griffiths and V. Taly, *Lab Chip*, 2011, **11**, 2156.
- 12 S. W. Lim, T. M. Tran and A. R. Abate, *PLoS One*, 2015, **10**, e0113549.
- 13 S. B. Berry, J. J. Lee, J. Berthier, E. Berthier and A. B. Theberge, *Anal. Methods*, 2019, **11**, 4528–4536.
- 14 X. Bian, F. Jing, G. Li, X. Fan, C. Jia, H. Zhou, Q. Jin and J. Zhao, *Biosens. Bioelectron.*, 2015, **74**, 770–777.
- 15 B. Demaree, D. Weisgerber, A. Dolatmoradi, M. Hatori and A. R. Abate, *Methods Cell Biol.*, 2018, **148**, 119–131.
- 16 D.-K. Kang, X. Gong, S. Cho, J. Kim, J. B. Edel, S.-I. Chang, J. Choo and A. J. DeMello, *Anal. Chem.*, 2015, **87**, 10770–10778.
- 17 A. Baccouche, S. Okumura, R. Sieskind, E. Henry, N. Aubert-Kato, N. Bredeche, J.-F. Bartolo, V. Taly, Y. Rondelez, T. Fujii and A. J. Genot, *Nat. Protoc.*, 2017, **12**, 1912–1932.
- 18 J. Tamminen, E. Lahdenperä, T. Koironen, T. Kuronen, T. Eerola, L. Lensu and H. Kälviäinen, *Chem. Eng. Sci.*, 2017, **167**, 54–65.
- 19 S. L. Pratt, G. K. Zath, T. Akiyama, K. S. Williamson, M. J. Franklin and C. B. Chang, *Front. Microbiol.*, 2019, **10**, 2112.
- 20 M. Vaithiyanathan, N. Safa and A. T. Melvin, *PLoS One*, 2019, **14**, e0215337.
- 21 Z. Hu, W. Fang, T. Gou, W. Wu, J. Hu, S. Zhou and Y. Mu, *Anal. Methods*, 2019, **11**, 3410–3418.
- 22 K. Gawryszewski, Z. A. Rana, K. W. Jenkins, P. Ioannou and D. Okonkwo, *Int. J. Comput. Appl.*, 2019, **41**, 329–342.
- 23 T. Gou, J. Hu, W. Wu, X. Ding, S. Zhou, W. Fang and Y. Mu, *Biosens. Bioelectron.*, 2018, **120**, 144–152.

- 24 X. Cui, L. Ren, Y. Shan, X. Wang, Z. Yang, C. Li, J. Xu and B. Ma, *Analyst*, 2018, **143**, 3309–3316.
- 25 A. E. Carpenter, T. R. Jones, M. R. Lamprecht, C. Clarke, I. Kang, O. Friman, D. A. Guertin, J. Chang, R. A. Lindquist, J. Moffat, P. Golland and D. M. Sabatini, *Genome Biol.*, 2006, **7**, R100.
- 26 D. Dao, A. N. Fraser, J. Hung, V. Ljosa, S. Singh and A. E. Carpenter, *Bioinformatics*, 2016, **32**, 3210–3212.
- 27 M.-A. Bray, M. S. Vokes and A. E. Carpenter, *Curr. Protoc. Mol. Biol.*, 2015, **109**, 14.17.1–14.17.13.
- 28 D. W. Crome, *Methods Mol. Biol.*, 2013, **931**, 1–27.
- 29 R. T. Singh, S. Roy, I. O. Singh, T. Sinam and M. K. Singh, *Int. J. Comput. Sci.*, 2012, **8**, 271–277.
- 30 K. K. Singh and A. Singh, *Int. J. Comput. Sci. Iss.*, 2010, **7**(5), 414–417.
- 31 M. Mattiazzi Usaj, E. B. Styles, A. J. Verster, H. Friesen, C. Boone and B. J. Andrews, *Trends Cell Biol.*, 2016, **26**, 598–611.
- 32 S. Shashkova and M. C. Leake, *Biosci. Rep.*, 2017, **37**, 1–19.
- 33 A. J. Genot, A. Baccouche, R. Sieskind, N. Aubert-Kato, N. Bredeche, J. F. Bartolo, V. Taly, T. Fujii and Y. Rondelez, *Nat. Chem.*, 2016, **8**, 760–767.
- 34 C. McQuin, A. Goodman, V. Chernyshev, L. Kamentsky, B. A. Cimini, K. W. Karhohs, M. Doan, L. Ding, S. M. Rafelski, D. Thirstrup, W. Wiegraebe, S. Singh, T. Becker, J. C. Caicedo and A. E. Carpenter, *PLoS Biol.*, 2018, **16**, e2005970.
- 35 H. Wang, Y. Liu, Z. Chen, L. Sun and Y. Zhao, *Sci. Adv.*, 2020, **6**, eaay1438.
- 36 Y. Yu, L. Shang, J. Guo, J. Wang and Y. Zhao, *Nat. Protoc.*, 2018, **13**, 2557–2579.
- 37 A. M. Kaushik, K. Hsieh, L. Chen, D. J. Shin, J. C. Liao and T. H. Wang, *Biosens. Bioelectron.*, 2017, **97**, 260–266.
- 38 L. Breiman, *Mach. Learn.*, 2001, **45**, 5–32.

Appendix 2

Publication II

Sanka, I., Bartkova, S., Pata, P., Smolander, O.-P., Scheler, O. (2021). Investigation of Different Free Image Analysis Software for High-Throughput Droplet Detection. *ACS Omega*, 6(35), 22625–22634.

Investigation of Different Free Image Analysis Software for High-Throughput Droplet Detection

Immanuel Sanka, Simona Bartkova, Pille Pata, Olli-Pekka Smolander, and Ott Scheler*



Cite This: *ACS Omega* 2021, 6, 22625–22634



Read Online

ACCESS |



Metrics & More



Article Recommendations



Supporting Information



ABSTRACT: Droplet microfluidics has revealed innovative strategies in biology and chemistry. This advancement has delivered novel quantification methods, such as droplet digital polymerase chain reaction (ddPCR) and an antibiotic heteroresistance analysis tool. For droplet analysis, researchers often use image-based detection techniques. Unfortunately, the analysis of images may require specific tools or programming skills to produce the expected results. In order to address the issue, we explore the potential use of standalone freely available software to perform image-based droplet detection. We select the four most popular software and classify them into rule-based and machine learning-based types after assessing the software's modules. We test and evaluate the software's (i) ability to detect droplets, (ii) accuracy and precision, and (iii) overall components and supporting material. In our experimental setting, we find that the rule-based type of software is better suited for image-based droplet detection. The rule-based type of software also has a simpler workflow or pipeline, especially aimed for non-experienced users. In our case, CellProfiler (CP) offers the most user-friendly experience for both single image and batch processing analyses.

■ INTRODUCTION

Droplet microfluidics has become a powerful tool for high-throughput analysis over the last few decades.¹ It allows compartmentalization of samples in massive parallelization.² This high-throughput technique is also compatible with different analytical technologies, e.g., mass spectrometry.³ Droplets are often applied for high sensitivity nucleic acid diagnostics⁴ or different microbiological studies.⁵ For instance, the tool has also been used to perform high-throughput screening for protein crystals,⁶ DNA quantification by digital droplet polymerase chain reaction (ddPCR),^{7,8} detecting viable bacteria and heteroresistance in antimicrobial experiments,^{9,10} or performing experiments with mammalian cells.¹¹

Image-based analysis has often been used in droplet microfluidic experiments.¹² The analysis has been implemented in different types of image data, from single static image up to real-time data, either by bright-field or fluorescence microscopy.¹³ This approach has been used for a wide range of experiments, such as bacterial surveillance of foodborne contamination,¹⁴ screening of specific substrates,¹⁵ single-cell analysis,¹⁶ and detecting viable bacteria or viruses (e.g., SARS-CoV-2).^{17,18} Image-based droplet analysis (IDA)

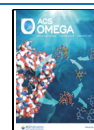
often requires specific skills in programming that are not widely available in non-specialist laboratories. Most of the published articles in droplet detection use scripted programs, such as Circular Hough Transform in Python programming language,¹⁹ Mathematica,^{20,21} Scikit-image in Python,²² Image Processing Toolbox from MATLAB,²³ OpenCV and Keras in Python,²⁴ and OpenCV in C++.²⁵ There are some user-friendly software that may be used for droplet microfluidic image analysis, such as the Zen imaging program²⁶ and NIS-Elements from NIKON.¹⁴ However, these kinds of programs are only commercially available.

There is a need for widely accessible and user-friendly IDA tools for image-based droplet analysis. Open-source software is available and can be used to detect and/or analyze droplets. For example, ImageJ software has been used to analyze image

Received: May 21, 2021

Accepted: August 13, 2021

Published: August 26, 2021



data in general²⁷ including droplets,²⁸ or CellProfiler (CP), which was developed to identify and measure various bioimage data.²⁹ Even though some published articles mention the use of the software, information regarding their workflow is limited (the data is often missing from publications). This would confuse early-stage researchers with little or no experience in image analysis, specifically for image-based droplet detection using no programming skills. However, novel workflows can be constructed by combining functions, modules, or pipelines from different software, like building a puzzle.³⁰

Here, we (i) demonstrate how to use different software for the analysis of droplet images in static 2D images and (ii) explore the differences and similarities of workflows in the different software from the perspective of detecting, counting, and measuring the properties (including but not limited to droplet number, diameter, fluorescence intensity of droplets, etc.) using four selected software (Table 1).

Table 1. General Characteristics of Selected Software^c

Requirement	CellProfiler	ImageJ	Ilastik	QuPath
Version	4.0.3	1.52p	1.3.3	0.2.3
Operating system	Win, Mac	Win, Mac	Win, Mac	Win, Mac
Bit machine	64 and 32	64 and 32	64	64
RAM and hard disk space	4 Gb & NA	NA & NA	8 Gb & NA	4 Gb & NA
Written in	Python	Java	Python	Java
Compatible file format	wide ^a	wide ^a	wide ^a	wide ^a
Output ^b	v	v	v	v
Available plugins	v	v	-	-
Documentation	v	v	v	v
Batch processing	v	v	v	v

^aWide is the general image file type, such as TIFF, JPEG, PNG, etc.

^bGenerates object size, pixel intensity, circularity, object position, etc.

^cv = available, - = not available

RESULTS AND DISCUSSION

Software Selection and Workflow Construction. The most popular software for image analysis are ImageJ (IJ), CellProfiler (CP), Ilastik (Ila), and QuPath (QP). Here, we use Twitter and Scopus repositories to find the popularity of the software in the field of image analysis. Twitter has been used for research purposes before.³² We found that social media also give researchers the opportunity to “push” their findings and correlate them to a greater citation.³³ To find the popularity, we executed Twint³⁴ Python script using each of the software’s name as the keyword. For finding the results from Scopus’ repository, we also used the same keyword. Both searches were performed to acquire data from January 1, 2010 to December 31, 2020. Based on the Scopus and Twitter search (obtained on February 11, 2021), we showed the sum of “tweets” or 160-character max of text from Twitter and the sum of Scopus search in scatter plot (Figure 1A). The most popular software are ImageJ,²⁷ CP,³⁵ Ilastik, and QuPath, in blue, red, cyan, and green color, respectively. Ilastik uses the concept of supervised machine learning in their workflow,³⁶ and QuPath has been used as a whole slide image analysis tool.³⁷ We continued with these four popular software tools and used them to detect droplets on the image dataset previously described by Bartkova et al.³¹ (Figure 1B). Then, we

took a deeper look into their workflow and assessed their performance with different key parameters (Figure 1C).

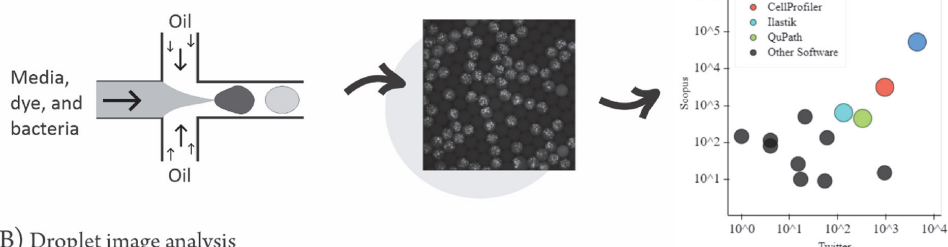
Rule-Based and Machine Learning-Based Software for Droplet Detection. We divided the selected software into two groups (rule-based and machine learning-based) according to their workflow. In the rule-based software group (CP and ImageJ), users have to manually provide settings for the program to select the pixels of interest with numeric or known parameter in order to detect droplets. In the machine learning-based group (Ilastik and QuPath), on the other hand, users may select the areas of the image (labeling) and manually annotate them as objects of interest (e.g., droplets or background) for pixel classification. Based on these characteristics, we described the abstraction of the process with three increasing levels and used it to direct the image-based droplet detection.

Pre-processing, Processing, and Post-processing Concepts. We used the terms (i) pre-processing, (ii) processing, and (iii) post-processing. (i) In pre-processing, we modified, adjusted, and prepared the image data for further use. For instance, we performed pre-processing to duplicate the image data, introduce features, and make annotation(s) on the image. In addition, we also include the image setup, such as image upload, metadata setting, and supporting option before processing the image data. For instance, we also included the macro record in IJ and the metadata setup in CP.

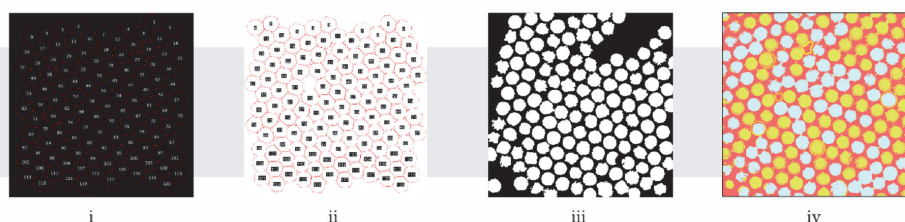
(ii) In processing, we conducted segmentation or pixel partitioning based on color, intensity, or texture along with droplet detection or counting process.³⁸ Usually, processing steps may help users obtain a specific type of data.³⁹ In our case, we introduced thresholding to distinguish between the background (dark) and the foreground (droplets). For the details, CP came in handy and only needed one module named “IdentifyPrimaryObject”, which contained some options to detect droplets. This included thresholding, smoothing, segmentation, and automatic selection. In ImageJ, processing steps had three options: “Thresholding”, “Watershed”, and “Analyze Particle”. Similar to CP, these three steps will provide selections to detect the droplets. In the processing part, Ilastik had to process “Thresholding”, “Object Feature Selection”, and “Object Classification” for selecting the droplets and discarding the background. In QuPath, we found all of these features in “Pixel Classifier”. The settings included a classifier from an artificial neural network with multilayer perception (ANN_MLP)⁴⁰ with high resolution, using four multiscale features (Gaussian, gradient magnitude, Hessian determinant, and Hessian max eigenvalue) with probability as an output.

(iii) For the last step, in post-processing, we prepared data extraction or generation for further use, for example, to generate a table of data or type of images for visualization. In CP, this last step was performed with “OverlayOutlines”, “OverlayObject”, “DisplayDataOnImage”, and “ExportToSpreadsheet”. These modules generated the images and results in CSV format. The order was similar in ImageJ and Ilastik, but the option was available in “ROI Manager” and “Export”, respectively. In QuPath, the results can be obtained by exporting annotations from detected objects or called as labeled images. We used the Groovy script to generate this result using commands in “Workflow” tab. Groovy is a compiled language that can be integrated seamlessly with Java. However, it has some semantic and practical differences, especially regarding syntax.⁴¹ For a brief workflow/pipeline, we provide the scheme of third level complexity in Figure 2.

A) Droplet generation and detection



B) Droplet image analysis



C) Three levels of abstraction in image-based droplet detection

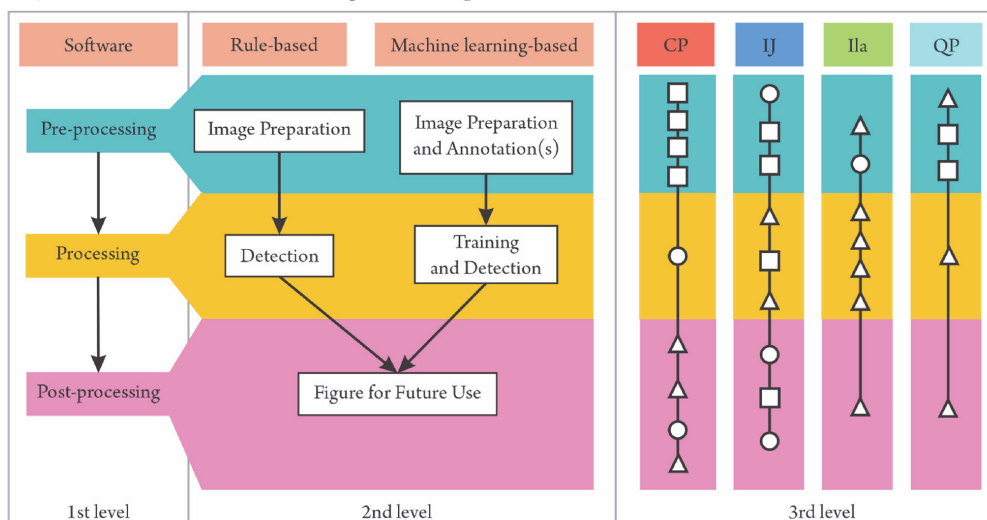


Figure 1. Schematic of droplet generation and image analysis of a single image. (A) We generated water-in-oil droplets using a flow-focusing microfluidic chip (left). We used a fluorescence microscope to obtain “raw images” of droplets that contained fluorescence producing bacteria (middle). For the analysis of droplet images, we used the four most popular image analysis software that were selected according to hits in social media (Twitter) and Scopus search (obtained on February 11, 2021) (right). (B) Droplet detection comparison among (i) ImageJ (IJ), (ii) CellProfiler (CP), (iii) Ilastik (Ila), and (iv) QuPath (QP). (C) We divided the image processing software into two groups (rule-based and machine learning-based) and explored their logic and working principle on three levels of abstraction. (1) The first level shows that used software are very similar in their basic image processing logic. They usually have three processing stages in their image analysis logic: pre-processing, processing, and post-processing. (2) The second level shows distinction between two groups of software in droplet detection: rule-based, where users define how to detect droplets by giving specific parameters (e.g., threshold or size), or machine learning-based, where users classify/annotate grouping of pixels on an image. (3) The third level shows a number of different steps and modules in processing stages. For the object on the left side of each workflow, we use a triangle to determine the module with only one option, rectangle for the module with two to eight options, and circle for the module with more than eight options.

CP Has the Highest Accuracy and Precision. By comparing the results with manually counted droplets

(7145), we investigated the ability of the analyzed software to detect droplets. We only counted the droplets that did not

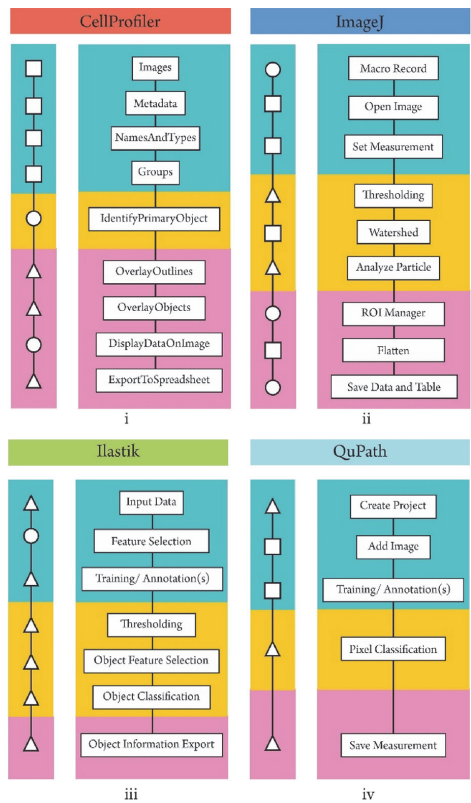


Figure 2. Detailed third level of abstraction for image-based droplet detection using (i) CellProfiler, (ii) ImageJ, (iii) Ilastik, and (iv) QuPath. The symbols represent how many options are within each module, referring to the previous figure where the triangle, rectangle, and circle represent one, two to eight, and more than eight options, respectively. The background colors correspond to pre-processing (cyan), processing (yellow), and post-processing (magenta).

touch the image border and did not make a bundle (joint droplets because of failed segmentation). We performed sensitivity and specificity tests using True Positive (TP), False Positive (FP), and False Negative (FN) values based on the comparison with manual counting.⁴² The TP confirms the positive droplet detection in the data. For FP, the value is obtained by finding false droplet detection or underestimation (type I error). In FN, the software does not detect the droplet or performs overestimation (type II error). We defined TN as the background (black = 0). After the calculation, we obtained the accuracy $((TP + TN)/(TP + TN + FP + FN))$ and precision $(TP/(TN + TP))$ from the detection. This accuracy explains the ratio between the correct droplet detection and total number of droplet detection. On the other hand, precision describes the probability to produce the correct droplet detection in total positive detection.^{43–45} The accuracy of each detection ranges from 74.7 to 96.2%. One of the software managed to generate a precision of up to 99.8% (Table 2).

Low-Image Quality Gives More False Detection. From Figure 3, we can see how each group shares similar errors in

Table 2. CP Gives the Highest Accuracy and Precision

Category	CellProfiler	ImageJ	Ilastik	QuPath
% Accuracy	96.2%	92.7%	74.7%	80.9%
% Precision	99.8%	96.3%	80.2%	83.1%



Figure 3. Droplet detection errors are higher in machine learning-based software in the diagnostic test. The figure shows the False Positive (FP, wrong detected droplet) and False Negative (FN, wrong undetected droplet) events per image. Each block represents event or image error detection. The scale shows the number of errors (dark = high and bright = low).

every event (detection per image). We compared the false detection results (both FP and FN) from each of the software. We found that the rule-based group (CP and ImageJ) have less false detection compared to the machine learning-based group (Ilastik and QuPath). However, Ilastik and QuPath received high error because they do not have filters to eliminate the droplets that touch the border, and some droplets are falsely detected as joint droplets (Figure S1). Figure 3 also shows images, which may have bad quality for droplet detection. For instance, image numbers 2, 19, and 64 depict the highest error values from all four software. Notwithstanding, CP outperforms the other software and has both high accuracy and precision.

Each Software Requires Different Workflows for Batch Processing. CP is the most suitable software for batch analysis or high throughput analysis. In CP, we can analyze a whole set of images with a press of a single button “Analyze Images” on the main menu. The software will process available images uploaded in the “Images” module (default module). We tested and used the batch processing option to analyze 64 images straight after we had our pipeline/workflow set. In ImageJ, we processed the batch analysis using a recorded macro by single image analysis. We also performed some macro script cleaning (e.g., closing unnecessary tabs during the process), which was written in the macro recorder. After cleaning, we selected the input and output folders and performed batch processing through the “Process” tab. For Ilastik, we executed batch processing after the last option of the pipeline. We just needed to upload the images and started the “Process all files”. QuPath demanded macroprogramming commands for executing batch analysis. However, this software provided an automated script generator that simplified the macro record to perform batch analysis. ImageJ and QuPath required a macro script for batch analysis. Even though this macro script was easy to do, creating a macro script for the first time could become an obstacle for researchers who are not familiar with any programming language or practices.⁴⁶ From

our viewpoint, CP and Ilastik had the most user-friendly interface for batch processing because they provide the option to scale up after single image pipeline construction and do not require any programming steps. Therefore, finding any additional button or tab to batch process the images was unnecessary. On the other hand, process and scripting were required in ImageJ and QuPath.

Modularity Gives More Flexibility in Developing Pipelines. Rule-based class software are flexible and have modular options in processing image(s). As rule-based tools, CP and ImageJ offered options that could be added and removed depending on the user's preferences, such as the type of thresholding algorithm, filters, and other modules. In machine learning-based software, the features were embedded in the pipeline and had limited availability for additional settings. For example, Ilastik had some pre-defined pipelines: one of them was object classification and pixel classification.³⁶ These two were fixed in the interface of Ilastik and may be rearranged only through Python programming. From Figure 1C, the third level of complexity also represents the modularity in which Ilastik and QuPath were more limited than CP and ImageJ. For instance, CP had the "IdentifyPrimaryObject" module that could be duplicated in one pipeline, while in Ilastik, "Thresholding" could be performed only once within the pre-defined workflow. This complication placed Ilastik as the least flexible tool followed by QuPath.

Batch Processing Time Is Shorter in Java-Based Software. Macro programming language affects the software processing time, particularly in batch analysis. CP or ImageJ expected less computational power for the use since they did not implement machine learning classification methods in our pipelines. The use of machine learning requires training and features implementation that requires more computational power.⁴⁷ The rule-based software used object logic classification⁴⁸ and did not require training set to test the defined parameters, e.g., size of the object or maximum length of the object. In QuPath and Ilastik, the classification depended on a supervised machine learning process.^{36,48} We used manual annotations (droplets and background) in making the classifier before processing whole pixels. We also previously compared the minimum hardware requirements for each of the tools (Table 1). Based on the comparison, ImageJ was the only one that did not put any minimum requirement on the random-access memory (RAM). We also expected that the machine learning-based software might take more time to process the whole set of images. Therefore, we also tried running the whole pipeline and comparing the performance from each of the tools. We tested each pipeline with the same computer having an Intel Core i3-9100F processor, 8GB RAM, NVIDIA GeForce GTX 1660 SUPER, 120Gb SSD PANTHER and running in a Windows operating system. In our setting (with the same environment and background setting), we found that QuPath and ImageJ perform faster than CP and Ilastik in batch processing (Figure 4). The experiment was conducted by running the same pipeline 10 times to find the deviation as well. Tool's batch processing language (macros) may cause this difference. At the beginning, we expected Ilastik and QuPath to have longer processing time than CP and ImageJ because of the machine learning-based processing. However, ImageJ and QuPath performed faster than others. In principle, there are two types of program that bioinformaticians use: compiled and interpreted.⁴⁹ ImageJ and QuPath use Java based (macros) code that is compiled once before the program processes the

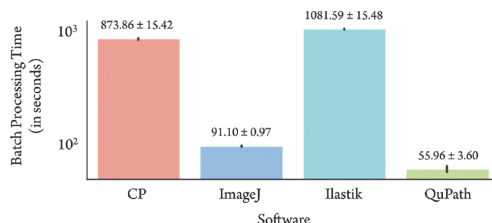


Figure 4. Java-based software has a shorter processing time in the droplet detection scenario. We ran the droplet detection pipelines in the same computer with the same dataset. Error bars show standard deviation between 10 replicate analysis runs with the same set of 64 images.

batch analysis. Presumably, this allows the program to run faster. On the other hand, CP and Ilastik use Python to process batch analysis. In Python, variables and functions will be run through an interpreter every time the program needs to process the task, in our case, to detect droplets in every image. Regardless, we do not have enough evidence to claim that the type of software may shorten the processing time. Nonetheless, a speed comparison of different types of language (including Python and Java) to run the same command showed that implementation in Java performs up to 20 times faster than in Python.⁵⁰ We also note that different hardware can alter the performance of software in different settings, but the relative ratios of needed computing resources should be similar.

Documentation Is Important in Pipeline Development. CP and ImageJ have sufficient examples and documentation for novice users. Each of the software provides documentation and examples for guiding their users. CP and ImageJ have been developed since 2005 and 1987, respectively.^{46,51} Therefore, these rule-based software have more users and examples, e.g., ImageJ has a distribution for compiling the biological image analysis plugins called Fiji.²⁷ CP also provides some tutorials, examples, and other documentation on their website, e.g., detecting different cell morphology and tracking objects (www.cellprofiler.org). On the contrary, Ilastik and QuPath have limited documentation for accompanying new users. However, these two software also have extensive documentation, including their manuals and tutorials for both novice and advanced users at their website (<https://ilastik.org/documentation> and <https://qupath.readthedocs.io>). Additionally, there are some forums such as image.sc forum (forum.image.sc) that are actively helping other bioimage researchers or software users.

Plugins May Ease Users to Perform Specific Image-Based Detection. Plugins in CP and ImageJ can be used as an extensible option in processing images. Plugins or add-on can be used to improve default options within the software. These may be utilized by other software developers. As an additional option, plugins may help the user implement specific cases of detection. Before Ilastik and QuPath were developed, ImageJ had plugins called Trainable WEKA Segmentation that, in principle, works similarly to machine learning-based software.⁵² In CP, plugins are also available. For instance, we found one plugin that analyzes mass cytometry (multiplexed images) called *ImcPluginsCP*.⁵³ Here, we did not add any plugins to detect droplets and we used similar settings to see the tool's ability to detect and count droplets. The extension software for CP, CellProfiler Analyst (CPA),⁵⁴ could

be an option to enhance droplet detection, which has been described briefly in our previous research.³¹ Based on our classification, CPA belongs to machine learning-based software because users need to supervise or train the data at the beginning. However, this software is not a standalone software and requires feature extraction or properties file that contain the observed data from CP.^{31,55}

Image Components Take Important Role in Processing. Rule-based software are more suitable for analyzing droplet microfluidic image data. Rule-based software provide more options, e.g., to disregard the object that touches the border/frame, which resulted in high accuracy and precision. On the other hand, the machine learning-based software required more optimization to train the classifier. We only used 12 lines (5 lines for determining droplets and 7 lines to define borders between droplets and background) to supervise each class (background and droplet). Each line represents the pixels for each group. This pixel manual selection works better if the image has similar properties in majority and represents the pixel distribution of an object, for example, borders between droplets and empty droplets. Even though the droplet's border looks the same across the image, the pixel distributions are varied. We picked more lines to define borders. We also needed extra time to train the classifier (in minutes) when setting the machine learning-based software to determine the 12 lines. However, this cannot represent all of the properties and may result in joint droplets. To overcome this, a larger training set and improvement of the classifier would presumably give a better result. As an image processing tool, the machine learning-based software like QuPath has a more specific purpose. Moreover, this software was created to accommodate whole slide image and large image data analyses, specifically for complex tissue images.³⁷ However, a comparison has been made between QuPath and CP coupled with CPA.⁴⁸ The comparison also shows the pros and cons between the rule-based and machine learning-based software in renal tissues. Furthermore, ImageJ, CP, Ilastik, and QuPath have shown their capability in detecting droplets and generating the results as standalone tools.

Data Acquisition Can Be Embedded in the Pipeline. Droplet detection is often used as a preliminary step in droplet microfluidic experiment. It is possible to expand the pipeline for further analysis, e.g., bacteria detection,³¹ enzyme reaction measurement,⁵⁶ chemical purification analysis,⁵⁷ and metal extraction.⁵⁸ This step is usually performed to extract the different aspects of a droplet (size, texture, volume, etc.) through pixel analysis. However, each software has its own option and feature to obtain the particular information, for example, "MeasureObjectSizeAndShape" and "MeasureObjectIntensity" in CP and "Set measurement" and "ROI Manager" in ImageJ. Nonetheless, this further analysis is not within the scope of this article. We try to focus on the principle of image-based droplet detection in different software and their components that may ease the user with no experience in image-based analysis.

■ CONCLUSIONS

This investigation gives insights into processing droplet microfluidic images using the four currently most popular software tools. We classified the types of open-source software into rule-based and machine learning-based groups. Both groups have three levels of complexity that cover pre-processing, processing, and post-processing steps. These

steps help users, specifically with no programming experience, to choose and perform their image analysis. In our experimental setup, we found that the rule-based type of software is better suited for image-based droplet detection. The rule-based type tools also have a simpler workflow or pipeline, especially aimed for non-experienced users. In our case, CP outperforms other software in terms of accuracy, precision, and user-friendliness (defined as usability for non-experienced users in building the pipeline and performing image-based droplet detection using available software modules). In terms of time processing, ImageJ and QuPath give faster processing time to detect droplets in 64 images. On the other hand, Ilastik gives a direct module that may ease early-stage researchers in image-based detection using the annotation principle. However, the optimal software choice may definitely be different for other users depending on their experimental conditions and acquired images. Our paper would serve as a starting point for them to compare available solutions and start with settings optimization, either using rule-based or machine learning-based software. In addition, published research, documentation, or forum discussions (such as www.image.sc) help in finding the most suitable software pipeline for image-based droplet detection and analysis.

■ METHODS

Software Search and Selection. We used selected software tools to detect droplets using the procedure explained by Bartkova et al.³¹ We found several available and accessible software tools online such as CP,³⁵ ImageJ,²⁷ Ilastik,³⁶ QuPath,³⁷ Icy,⁵⁹ BioFilmQ,⁶⁰ CellOrganizer,⁶¹ CellCognition,⁶² BioImageXD,⁶³ BacStalk,⁶⁴ Advanced CellClassifier,⁶⁵ Phenoripper,⁶⁶ and Cytomine.⁶⁷ We have tested every software mentioned previously to perform image-based droplet detection; however, not all of the software had a good documentation, workflow, reference, and user-friendly interface. Therefore, we tried to find the most preferred tools available online by using Twint-Twitter Intelligence Tool script³⁴ written in Python and Scopus search from their website (<https://www.scopus.com>). The search has the same filter, including the search time (01-01-2010 until 31-12-2020), and only receives the result in "English". Therefore, the search both in Twitter and Scopus will not consider any data outside the filter. Both Twitter and Scopus data were obtained on February 11, 2021. We used each software's name as the keyword for the search. For the Twitter search, the processing was executed in Jupyter Notebook (ver. 6.0.3)⁶⁸ within Anaconda Navigator.⁶⁹ We also imported datetime and Pandas as additional libraries. For the Scopus search, it was performed using the same keyword. Both results were visualized together using Bokeh and NumPy libraries in Python.^{70–72}

Droplet Generation and Image Acquisition. We repeated the method described in Bartkova et al. to generate droplets and their image data.³¹ We used a set of 64 images to test the most popular software to detect droplets. The images are 2D layers of droplets generated by fluorescence confocal microscopy. We used the same images to find a suitable workflow for each software and describe it thoroughly in the next paragraph. Using the data, we calculated the precision and accuracy of detecting the droplets by comparing the results with manual counting using the same batch processing results in the same attempt.

Image Analysis with the Most Popular Software. The image data were analyzed first as a single image using ImageJ

(ver. 1.52p), CP (ver. 4.0.3), Ilastik (ver. 1.3.3), and QuPath (ver. 0.2.3). For each of the software, we describe the pipeline construction in the following paragraphs. Pipelines can be found at <https://github.com/taltechmicrofluidics>.

Pipeline Construction in CP. We used our previous pipeline³¹ in CP as the basis for exploring other tools. We uploaded the image through a drag and drop feature in the Images module and set the Metadata, NamesAndTypes, and Groups according to our setting. We used the “IdentifyPrimaryObject” to detect droplets. We also used the same setting that is also provided in our GitHub repository (github.com/taltechmicrofluidics/CP-for-droplet-analysis). The “MeasureObjectIntensity” and “ExportToSpreadSheet” modules were also set as previously. The results were obtained automatically after pressing the “Analyze Image” button.

Pipeline Construction in ImageJ. For ImageJ, we recorded the workflow in the macro record option. This record was used to make scripts for batch processing. To upload the image, we use Open Image from the File tab in the main menu. The parameter was set within “Set Measurement” under “Analyze” tab, and we only ticked “Area” for obtaining the pixels’ area in one droplet. This was followed with processing workflow, which included segmentation using “Threshold” under “Adjust” option in the “Image” tab. The threshold was determined as 1507, corresponding to 0.023 scale, described in our previous article using CP. The thresholding was followed with “Watershed” to separate droplets from each other. The counting was performed using “Analyze Particle” under the “Analyze” tab. We set the size corresponding to the range we described in CP, 22,500 up to 62,500 pixels² with 0 circularity. Once we finished the processing step, we downloaded the image through the “Flatten” option in the “ROI Manager” menu. We obtained the results in the table, which appeared straight after we performed the analysis.

Pipeline Construction in Ilastik. In Ilastik, we used “Pixel Classification” and “Object Classification” pre-defined workflow. We loaded the image in the Input Data module and selected the features for the training set. Since we did not have any reference regarding this type of workflow, we used the recommendation from image.sc forum, starting by adding 0.30, 1.00, and 3.50 sigma or scale corresponding to the selected features, e.g., Gaussian Filter, for color/intensity, edge, and texture. We trained the program to distinguish between the background (dark) and droplets using manual annotations/labels. For thresholding, we used the default smoothing value (1.0 and 1.0) with a 0.70 threshold. For the size filter, we put values that correspond to the settings in ImageJ, 22,500 for the minimum size and 62,500 for the maximum size. This was followed by using the standard object selection feature option and selecting the detected droplets in object classification as a sample. After finishing the setup, we obtained the results by exporting both object predictions and measured features.

Pipeline Construction in QuPath. In QuPath, we started the workflow by creating a project (Create Project) and uploading the image (Add Image). Once the selected image was ready, we performed annotations similar to Ilastik. This process aimed to distinguish the background and foreground (droplets). After annotating the image, we performed “Pixel Classification” using the artificial neural network (ANN_MLP) classifier with high (downsample = 4.0) resolution. For the features, the scales were 1.0, 2.0, and 4.0 for Gaussian gradient magnitude, Hessian determinant, and Hessian max eigenvalue, respectively. We created object detection for droplets and

measured all detected droplets. We set a thick boundary class to make borders between each of the droplets. We saved the measurement data from the measurement menu.

Batch Processing from Each of the Software. In CP, we performed batch processing by loading the set of images in the Images module and run the “Analyze Images” button. For ImageJ, we executed batch processing using the “Batch Process” option under the “Process” tab. We used a recorded macro with some adjustments to execute the images in the Input folder. By processing the images through this option, we generated results directly to the Output folder. In Ilastik, we continued the batch processing straight after setting up the workflow. Similar to CP, we executed batch processing after uploading the images and only needed to press the “Process all images” button. In QuPath, we transformed the workflow from a single image into scripts to execute the batch processing. Since QuPath provides the script builder, we did not have to script by ourselves, and we could start batch processing by executing the script and ran it for the whole image set in the project. However, the image results from QuPath require additional script using Groovy. We managed to generate the results and you may find the script in our GitHub. We stored both single and batch processing pipelines from each of the software here: (github.com/taltechmicrofluidics/Software-Analysis).

Data Acquisition and Processing. We gathered all results and processed them in Microsoft Excel as follows. We tested the results with sensitivity and specificity tests and used manual counting as the reference.^{42,73,74} We used these formulas for the test:

$$\text{FP Rate} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

$$\text{TP Rate} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

Where TP is the correct droplet Detection compared to ground truth, FP is the wrong detection (detecting background), FN is the wrong detection (software cannot recognize existed droplet), TN is the background (0), accuracy is the quality of correctness, and precision is the similarity upon repeatable counting.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acsomega.1c02664>.

Software settings, detailed diagnostic test results, image quality description, and step-by-step guideline and utilization from the tested software (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Ott Scheler – Department of Chemistry and Biotechnology, Tallinn University of Technology, 12618 Tallinn, Estonia; orcid.org/0000-0002-8428-1350; Email: ott.scheler@taltech.ee

Authors

Immanuel Sanka – Department of Chemistry and Biotechnology, Tallinn University of Technology, 12618 Tallinn, Estonia

Simona Bartkova – Department of Chemistry and Biotechnology, Tallinn University of Technology, 12618 Tallinn, Estonia

Pille Pata – Department of Chemistry and Biotechnology, Tallinn University of Technology, 12618 Tallinn, Estonia

Olli-Pekka Smolander – Department of Chemistry and Biotechnology, Tallinn University of Technology, 12618 Tallinn, Estonia; orcid.org/0000-0002-6795-7734

Complete contact information is available at:

<https://pubs.acs.org/10.1021/acsomega.1c02664>

Author Contributions

The manuscript was written through contributions from all authors. I.S., O.S., and O.-P.S. conceived the study. I.S. conducted the research and wrote the article with support from all the other authors. S.B. and P.P. were responsible for the microbiology part, droplet experiments, and microscopy imaging. All authors have given their approval to the final version of the manuscript.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The research was performed partially in the laboratory setup with the support from the TTU Development Program 2016–2022 (project no. 2014–2020.4.01.16.0032). We also acknowledge the Estonian Research Council grants MOBTP109, PRG620, and MOBJD556. Authors received help for early software screening from Matin Nuhamunada, Afif Pranaya Jati, and Ahmad Ardi in Universitas Gadjah Mada (Indonesia).

REFERENCES

- (1) Ding, Y.; Howes, P. D.; Demello, A. J. Recent Advances in Droplet Microfluidics. *Anal. Chem.* **2020**, *92*, 132–149.
- (2) Scheler, O.; Postek, W.; Garstecki, P. Recent Developments of Microfluidics as a Tool for Biotechnology and Microbiology. *Curr. Opin. Biotechnol.* **2019**, *55*, 60–67.
- (3) Diefenbach, X. W.; Farasat, I.; Guetschow, E. D.; Welch, C. J.; Kennedy, R. T.; Sun, S.; Moore, J. C. Enabling Biocatalysis by High-Throughput Protein Engineering Using Droplet Microfluidics Coupled to Mass Spectrometry. *ACS Omega* **2018**, *3*, 1498–1508.
- (4) Vasudevan, H. N.; Xu, P.; Servellita, V.; Miller, S.; Liu, L.; Gopez, A.; Chiu, C. Y.; Abate, A. R. Digital Droplet PCR Accurately Quantifies SARS-CoV-2 Viral Load from Crude Lysate without Nucleic Acid Purification. *Sci. Rep.* **2021**, *11*, 780.
- (5) Kaminski, T. S.; Scheler, O.; Garstecki, P. Droplet Microfluidics for Microbiology: Techniques, Applications and Challenges. *Lab Chip* **2016**, *16*, 2168–2187.
- (6) Du, W.-B.; Sun, M.; Gu, S.-Q.; Zhu, Y.; Fang, Q. Automated Microfluidic Screening Assay Platform Based on DropLab. *Anal. Chem.* **2010**, *82*, 9941–9947.
- (7) Hindson, B. J.; Ness, K. D.; Masquelier, D. A.; Belgrader, P.; Heredia, N. J.; Makarewicz, A. J.; Bright, I. J.; Lucero, M. Y.; Hiddessen, A. L.; Legler, T. C.; Kitano, T. K.; Hodel, M. R.; Petersen, J. F.; Wyatt, P. W.; Steenblock, E. R.; Shah, P. H.; Bousse, L. J.; Troup, C. B.; Mellen, J. C.; Wittmann, D. K.; Erndt, N. G.; Cauley, T. H.; Koehler, R. T.; So, A. P.; Dube, S.; Rose, K. A.; Montesclaros, L.; Wang, S.; Stumbo, D. P.; Hodges, S. P.; Romine, S.; Milanovich, F. P.; White, H. E.; Regan, J. F.; Karlin-Neumann, G. A.; Hindson, C. M.; Saxonov, S.; Colston, B. W. High-Throughput Droplet Digital PCR System for Absolute Quantitation of DNA Copy Number. *Anal. Chem.* **2011**, *83*, 8604–8610.
- (8) Hindson, C. M.; Chevillet, J. R.; Briggs, H. A.; Gallichotte, E. N.; Ruf, I. K.; Hindson, B. J.; Vessella, R. L.; Tewari, M. Absolute Quantification by Droplet Digital PCR versus Analog Real-Time PCR. *Nat. Methods* **2013**, *10*, 1003–1005.
- (9) Scheler, O.; Pacocha, N.; Debski, P. R.; Ruzsaczak, A.; Kaminski, T. S.; Garstecki, P. Optimized Droplet Digital CFU Assay (DdCFU) Provides Precise Quantification of Bacteria over a Dynamic Range of 6 Logs and Beyond. *Lab Chip* **2017**, *17*, 1980–1987.
- (10) Scheler, O.; Makuch, K.; Debski, P. R.; Horka, M.; Ruzsaczak, A.; Pacocha, N.; Sozański, K.; Smolander, O. P.; Postek, W.; Garstecki, P. Droplet-Based Digital Antibiotic Susceptibility Screen Reveals Single-Cell Clonal Heteroresistance in an Isogenic Bacterial Population. *Sci. Rep.* **2020**, *10*, 1–8.
- (11) De Cesare, I.; Zamora-Chimal, C. G.; Postiglione, L.; Khazim, M.; Pedone, E.; Shannon, B.; Fiore, G.; Perrino, G.; Napolitano, S.; Di Bernardo, D.; Savery, N. J.; Grierson, C.; Di Bernardo, M.; Marucci, L. ChipSeg: An Automatic Tool to Segment Bacterial and Mammalian Cells Cultured in Microfluidic Devices. *ACS Omega* **2021**, *6*, 2473–2476.
- (12) Zantow, M.; Dendere, R.; Douglas, T. S. Image-Based Analysis of Droplets in Microfluidics. In *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*; Institute of Electrical and Electronics Engineers, 2013; pp. 1776–1779.
- (13) Zhu, Y.; Fang, Q. Analytical Detection Techniques for Droplet Microfluidics-A Review. *Anal. Chim. Acta* **2013**, *787*, 24–35.
- (14) Harmon, J. B.; Gray, H. K.; Young, C. C.; Schwab, K. J. Microfluidic Droplet Application for Bacterial Surveillance in Fresh-Cut Produce Wash Waters. *PLoS One* **2020**, *15*, No. e0233239.
- (15) Najah, M.; Mayot, E.; Mahendra-Wijaya, I. P.; Griffiths, A. D.; Ladame, S.; Drevelle, A. New Glycosidase Substrates for Droplet-Based Microfluidic Screening. *Anal. Chem.* **2013**, *85*, 9807–9814.
- (16) Chen, P.; Chen, D.; Li, S.; Ou, X.; Liu, B.-F. Microfluidics towards Single Cell Resolution Protein Analysis. *TrAC, Trends Anal. Chem.* **2019**, 2–12.
- (17) Cui, X.; Ren, L.; Shan, Y.; Wang, X.; Yang, Z.; Li, C.; Xu, J.; Ma, B. Smartphone-Based Rapid Quantification of Viable Bacteria by Single-Cell Microdroplet Turbidity Imaging. *Analyst* **2018**, *143*, 3309–3316.
- (18) Samacoits, A.; Nimsamer, P.; Mayuramart, O.; Chantaravisoot, N.; Sitti-Amorn, P.; Nakhakes, C.; Luangkamchorn, L.; Tongcham, P.; Zahm, U.; Suphanpayak, S.; Padungwattanachoke, N.; Leelarthaphin, N.; Huayhongthong, H.; Pisitkun, T.; Payungporn, S.; Hannanta-Anan, P. Machine Learning-Driven and Smartphone-Based Fluorescence Detection for CRISPR Diagnostic of SARS-CoV-2. *ACS Omega* **2021**, *6*, 2727–2733.
- (19) Vaithyanathan, M.; Safa, N.; Melvin, A. T. FluoroCellTrack: An Algorithm for Automated Analysis of High-Throughput Droplet Microfluidic Data. *PLoS One* **2019**, *14*, No. e0215337.
- (20) Genot, A. J.; Baccouche, A.; Sieskind, R.; Aubert-Kato, N.; Bredeche, N.; Bartolo, J. F.; Taly, V.; Fujii, T.; Rondelez, Y. High-Resolution Mapping of Bifurcations in Nonlinear Biochemical Circuits. *Nat. Chem.* **2016**, *8*, 760–767.
- (21) Baccouche, A.; Okumura, S.; Sieskind, R.; Henry, E.; Aubert-Kato, N.; Bredeche, N.; Bartolo, J.-F.; Taly, V.; Rondelez, Y.; Fujii, T.; Genot, A. J. Massively Parallel and Multiparameter Titration of Biochemical Assays with Droplet Microfluidics. *Nat. Protoc.* **2017**, *12*, 1912–1932.
- (22) Svensson, C.-M.; Shvydkiv, O.; Dietrich, S.; Mahler, L.; Weber, T.; Choudhary, M.; Tovar, M.; Figge, M. T.; Roth, M. Coding of Experimental Conditions in Microfluidic Droplet Assays Using Colored Beads and Machine Learning Supported Image Analysis. *Small* **2018**, *15*, 1802384.
- (23) Byrnes, S. A.; Phillips, E. A.; Huynh, T.; Weigl, B. H.; Nichols, K. P. Polydisperse Emulsion Digital Assay to Enhance Time to Detection and Extend Dynamic Range in Bacterial Cultures Enabled by a Statistical Framework. *Analyst* **2018**, *143*, 2828–2836.

- (24) Dressler, O. J.; Howes, P. D.; Choo, J.; Demello, A. J. Reinforcement Learning for Dynamic Microfluidic Control. *ACS Omega* **2018**, *3*, 10084–10091.
- (25) Zang, E.; Brandes, S.; Tovar, M.; Martin, K.; Mech, F.; Horbert, P.; Henkel, T.; Figge, M. T.; Roth, M. Real-Time Image Processing for Label-Free Enrichment of Actinobacteria Cultivated in Picolitre Droplets. *Lab Chip* **2013**, *13*, 3707–3713.
- (26) Sarkar, S.; Cohen, N.; Sabhachandani, P.; Konry, T. Phenotypic Drug Profiling in Droplet Microfluidics for Better Targeting of Drug-Resistant Tumors. *Lab Chip* **2015**, *15*, 4441–4450.
- (27) Schindelin, J.; Arganda-Carreras, I.; Frise, E.; Kaynig, V.; Longair, M.; Pietzsch, T.; Preibisch, S.; Rueden, C.; Saalfeld, S.; Schmid, B.; Tinevez, J. Y.; White, D. J.; Hartenstein, V.; Eliceiri, K.; Tomancak, P.; Cardona, A. Fiji: An Open-Source Platform for Biological-Image Analysis. *Nat. Methods* **2012**, *9*, 676–682.
- (28) Najah, M.; Griffiths, A. D.; Ryckelynck, M. Teaching Single-Cell Digital Analysis Using Droplet-Based Microfluidics. *Anal. Chem.* **2012**, *84*, 1202–1209.
- (29) Lamprecht, M. R.; Sabatini, D. M.; Carpenter, A. E. CellProfilerTM: Free, Versatile Software for Automated Biological Image Analysis. *BioTechniques* **2007**, *42*, 71–75.
- (30) Miura, K.; Paul-Gilloteaux, P.; Tosi, S.; Colombelli, J. Workflows and Components of Bioimage Analysis. In *Bioimage Data Analysis Workflows*; Springer: Cham, 2020; pp. 1–7.
- (31) Bartkova, S.; Vendelin, M.; Sanka, I.; Pata, P.; Scheler, O. Droplet Image Analysis with User-Friendly Freeware CellProfiler. *Anal. Methods* **2020**, *12*, 2287–2294.
- (32) Malik, A.; Heyman-Schrum, C.; Johri, A. Use of Twitter across Educational Settings: A Review of the Literature. *International Journal of Educational Technology in Higher Education*; Springer: Netherlands, 2019; pp. 1–22.
- (33) Klar, S.; Krupnikov, Y.; Ryan, J. B.; Searles, K.; Shmargad, Y. Using Social Media to Promote Academic Research: Identifying the Benefits of Twitter for Sharing Academic Work. *PLoS One* **2020**, *15*, No. e0229446.
- (34) OSIN team - Twint Project. *GitHub - Twintproject/Twint: An Advanced Twitter Scraping & OSINT Tool Written in Python That Doesn't Use Twitter's API, Allowing You to Scrape a User's Followers, Following, Tweets and More While Evading Most API Limitations*. 2020, p <https://github.com/twintproject/twint>.
- (35) McQuinn, C.; Goodman, A.; Chernyshev, V.; Kamentsky, L.; Cimini, B. A.; Karhohs, K. W.; Doan, M.; Ding, L.; Rafelski, S. M.; Thirstrup, D.; Wiegand, W.; Singh, S.; Becker, T.; Caicedo, J. C.; Carpenter, A. E. CellProfiler 3.0: Next-Generation Image Processing for Biology. *PLoS Biol.* **2018**, *16*, No. e2005970.
- (36) Berg, S.; Kutra, D.; Kroeger, T.; Straehle, C. N.; Kausler, B. X.; Haubold, C.; Schiegg, M.; Ales, J.; Beier, T.; Rudy, M.; Eren, K.; Cervantes, J. I.; Xu, B.; Beuttenmüller, F.; Wolny, A.; Zhang, C.; Koethe, U.; Hamprecht, F. A.; Kreshuk, A. Ilastik: Interactive Machine Learning for (Bio)Image Analysis. *Nat. Methods* **2019**, *16*, 1226–1232.
- (37) Bankhead, P.; Loughrey, M. B.; Fernández, J. A.; Dombrowski, Y.; McArt, D. G.; Dunne, P. D.; McQuaid, S.; Gray, R. T.; Murray, L. J.; Coleman, H. G.; James, J. A.; Salto-Tellez, M.; Hamilton, P. W. QuPath: Open Source Software for Digital Pathology Image Analysis. *Sci. Rep.* **2017**, *7*, 16878.
- (38) Liang, Y.; Zhang, M.; Browne, W. N. Image Segmentation: A Survey of Methods Based on Evolutionary Computation. In *Simulated Evolution and Learning. SEAL 2014. Lecture Notes in Computer Science*; Springer: Verlag, 2014; Vol. 8886, pp. 847–859.
- (39) Uchida, S. Image Processing and Recognition for Biological Images. *Dev., Growth Differ.* **2013**, *55*, 523–549.
- (40) Darvishan, A.; Bakhshi, H.; Madadkhani, M.; Mir, M.; Bemani, A. Application of MLP-ANN as a Novel Predictive Method for Prediction of the Higher Heating Value of Biomass in Terms of Ultimate Analysis. *Energy Sources, Part A* **2018**, *40*, 2960–2966.
- (41) Abdul-Jawad, B. *Groovy and Grails Recipes*; Apress Media: LLC, 2009.
- (42) Gaddis, G. M.; Gaddis, M. L. Introduction to Biostatistics: Part 3, Sensitivity, Specificity, Predictive Value, and Hypothesis Testing. *Ann. Emerg. Med.* **1990**, *19*, 591–597.
- (43) Cecconi, M.; Rhodes, A.; Poloniecki, J.; Della Rocca, G.; Grounds, R. M. Bench-to-Bedside Review: The Importance of the Precision of the Reference Technique in Method Comparison Studies—with Specific Reference to the Measurement of Cardiac Output. *Crit. Care* **2009**, *13*, 201.
- (44) Bland, J. M.; Altman, D. G. Comparing Methods of Measurement: Why Plotting Difference against Standard Method Is Misleading. *Lancet* **1995**, *346*, 1085–1087.
- (45) Bland, J. M.; Altman, D. G. Measuring Agreement in Method Comparison Studies. *Stat. Methods Med. Res.* **1999**, *8*, 135–160.
- (46) Carpenter, A. E.; Jones, T. R.; Lamprecht, M. R.; Clarke, C.; Kang, I.; Friman, O.; Guertin, D. A.; Chang, J.; Lindquist, R. A.; Moffat, J.; Golland, P.; Sabatini, D. M. CellProfiler: Image Analysis Software for Identifying and Quantifying Cell Phenotypes. *Genome Biol.* **2006**, *7*, R100.
- (47) Frank, M.; Drikakis, D.; Charissis, V. Machine-Learning Methods for Computational Science and Engineering. *Computation* **2020**, *8*, 15.
- (48) Ribeiro, G. P.; Endringer, D. C.; De Andrade, T. U.; Lenz, D. Comparison between Two Programs for Image Analysis, Machine Learning and Subsequent Classification. *Tissue Cell* **2019**, *58*, 12–16.
- (49) Bonnal, R. J. P.; Yates, A.; Goto, N.; Gautier, L.; Willis, S.; Fields, C.; Katayama, T.; Prins, P. Sharing Programming Resources between Bio* Projects. In *Methods in Molecular Biology*; Humana Press Inc., 2019; Vol. 1910, pp. 747–766.
- (50) Fourment, M.; Gillings, M. R. A Comparison of Common Programming Languages Used in Bioinformatics. *BMC Bioinformatics* **2008**, *9*, 82.
- (51) Schneider, C. A.; Rasband, W. S.; Eliceiri, K. W. NIH Image to ImageJ: 25 Years of Image Analysis. *Nat. Methods* **2012**, *9*, 671–675.
- (52) Arganda-Carreras, I.; Kaynig, V.; Rueden, C.; Eliceiri, K. W.; Schindelin, J.; Cardona, A.; Sebastian Seung, H. Trainable Weka Segmentation: A Machine Learning Tool for Microscopy Pixel Classification. *Bioinformatics* **2017**, *33*, 2424–2426.
- (53) Zanotelli, V. R.; Leutenegger, M.; Lun, X.; Georgi, F.; de Souza, N.; Bodenmiller, B. A Quantitative Analysis of the Interplay of Environment, Neighborhood, and Cell State in 3D Spheroids. *Mol. Syst. Biol.* **2020**, *16* (), DOI: 10.15252/msb.202009798.
- (54) Jones, T. R.; Kang, I. H.; Wheeler, D. B.; Lindquist, R. A.; Papallo, A.; Sabatini, D. M.; Golland, P.; Carpenter, A. E. CellProfiler Analyst: Data Exploration and Analysis Software for Complex Image-Based Screens. *BMC Bioinformatics* **2008**, *9*, 482.
- (55) Dao, D.; Fraser, A. N.; Hung, J.; Ljosa, V.; Singh, S.; Carpenter, A. E. CellProfiler Analyst: Interactive Data Exploration, Analysis and Classification of Large Biological Image Sets. *Bioinformatics* **2016**, *32*, 3210–3212.
- (56) Ochoa, A.; Álvarez-Bohórquez, E.; Castellero, E.; Olguin, L. F. Detection of Enzyme Inhibitors in Crude Natural Extracts Using Droplet-Based Microfluidics Coupled to HPLC. *Anal. Chem.* **2017**, *89*, 4889–4896.
- (57) Mary, P.; Studer, V.; Tabeling, P. Microfluidic Droplet-Based Liquid-Liquid Extraction. *Anal. Chem.* **2008**, *80*, 2680–2687.
- (58) Tamminen, J.; Lahdenperä, E.; Koiranen, T.; Kuronen, T.; Eerola, T.; Lensu, L.; Kälviäinen, H. Determination of Single Droplet Sizes, Velocities and Concentrations with Image Analysis for Reactive Extraction of Copper. *Chem. Eng. Sci.* **2017**, *167*, 54–65.
- (59) De Chaumont, F.; Dallongeville, S.; Olivo-Marin, J. C. ICY: A New Open-Source Community Image Processing Software. In *International Symposium on Biomedical Imaging*; IEEE Computer Society, 2011; pp. 234–237.
- (60) Hartmann, R.; Jeckel, H.; Jelli, E.; Singh, P. K.; Vaidya, S.; Bayer, M.; Rode, D. K. H.; Vidakovic, L.; Diaz-Pascual, F.; Fong, J. C. N.; Dragoš, A.; Lamprecht, O.; Thöming, J. G.; Netter, N.; Häussler, S.; Nadell, C. D.; Sourjik, V.; Kovács, A. T.; Yildiz, F. H.; Drescher, K. Quantitative Image Analysis of Microbial Communities with BiofilmQ. *Nat. Microbiol.* **2021**, *6*, 151–156.

- (61) Majarian, T. D.; Cao-Berg, I.; Ruan, X.; Murphy, R. F. CellOrganizer: Learning and Using Cell Geometries for Spatial Cell Simulations. In *Methods in Molecular Biology*; Humana Press Inc., 2019; Vol. 1945, pp. 251–264.
- (62) Held, M.; Schmitz, M. H. A.; Fischer, B.; Walter, T.; Neumann, B.; Olma, M. H.; Peter, M.; Ellenberg, J.; Gerlich, D. W. CellCognition: Time-Resolved Phenotype Annotation in High-Throughput Live Cell Imaging. *Nat. Methods* **2010**, *7*, 747–754.
- (63) Kankaanpää, P.; Paavolainen, L.; Tiitta, S.; Karjalainen, M.; Päivärinne, J.; Nieminen, J.; Marjomäki, V.; Heino, J.; White, D. J. BioImageXD: An Open, General-Purpose and High-Throughput Image-Processing Platform. *Nat. Methods* **2012**, *9*, 683–689.
- (64) Hartmann, R.; Teeseling, M. C. F.; Thanbichler, M.; Drescher, K. BacStalk: A Comprehensive and Interactive Image Analysis Software Tool for Bacterial Cell Biology. *Mol. Microbiol.* **2020**, *114*, 140–150.
- (65) Piccinini, F.; Balassa, T.; Szkalisity, A.; Molnar, C.; Paavolainen, L.; Kujala, K.; Buzas, K.; Sarazova, M.; Pietiainen, V.; Kutay, U.; Smith, K.; Horvath, P. Advanced Cell Classifier: User-Friendly Machine-Learning-Based Software for Discovering Phenotypes in High-Content Imaging Data. *Cell Syst.* **2017**, *4*, 651–655.e5.
- (66) Rajaram, S.; Pavie, B.; Wu, L. F.; Altschuler, S. J. PhenoRipper: Software for Rapidly Profiling Microscopy Images. *Nat. Methods* **2012**, *9*, 635–637.
- (67) Marée, R.; Rollus, L.; Stévens, B.; Hoyoux, R.; Louppe, G.; Vandaele, R.; Begon, J. M.; Kainz, P.; Geurts, P.; Wehenkel, L. Collaborative Analysis of Multi-Gigapixel Imaging Data Using Cytomine. *Bioinformatics* **2016**, *32*, 1395–1401.
- (68) Kluyver, T.; Ragan-Kelley, B.; Pérez, F.; Granger, B.; Bussonnier, M.; Frederic, J.; Kelley, K.; Hamrick, J.; Grout, J.; Corlay, S.; Ivanov, P.; Avila, D.; Abdalla, S.; Willing, C. Jupyter Notebooks—a Publishing Format for Reproducible Computational Workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas - Proceedings of the 20th International Conference on Electronic Publishing, ELPUB 2016*; IOS Press: BV, 2016; pp. 87–90.
- (69) Anaconda Software Distribution. Anaconda Software Distribution. Anaconda 2016, p <https://www.anaconda.com/>.
- (70) The Pandas Development Team. Pandas-Dev/Pandas: Pandas 1.2.3. Zenodo 2020, p <https://zenodo.org/record/4572994>, DOI: 10.5281/ZENODO.4572994.
- (71) Bokeh Development Team. Bokeh: Python Library for Interactive Visualization. 2018, p <http://www.bokeh.pydata.org>.
- (72) Harris, C. R.; Millman, K. J.; van der Walt, S. J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N. J.; Kern, R.; Picus, M.; Hoyer, S.; van Kerkwijk, M. H.; Brett, M.; Haldane, A.; del Río, J. F.; Wiebe, M.; Peterson, P.; Gérard-Marchant, P.; Sheppard, K.; Reddy, T.; Weckesser, W.; Abbasi, H.; Gohlke, C.; Oliphant, T. E. Array Programming with NumPy. *Nature* **2020**, *585*, 357–362.
- (73) Soleymani, R.; Granger, E.; Fumera, G. F-Measure Curves: A Tool to Visualize Classifier Performance under Imbalance. *Pattern Recognit.* **2020**, *100*, 107146.
- (74) Landgrebe, T. C. W.; Paclik, P.; Duin, R. P. W.; Bradley, A. P. Precision-Recall Operating Characteristic (P-ROC) Curves in Imprecise Environments. In *Proceedings - International Conference on Pattern Recognition*; Institute of Electrical and Electronics Engineers, 2006; Vol. 4, pp. 123–127.

Appendix 3

Publication III

Sanka, I., Bartkova, S., Pata, P., Ernits, M., Meinberg, M. M., Agu, N., Aruoja, V., Smolander, O.-P., Scheler, O. (2023). User-friendly analysis of droplet array images. *Analytica Chimica Acta*, 1272, 341397.



Contents lists available at ScienceDirect

Analytica Chimica Acta

journal homepage: www.elsevier.com/locate/aca

User-friendly analysis of droplet array images

Immanuel Sanka^{a,1}, Simona Bartkova^{a,1}, Pille Pata^a, Mart Ernits^b, Monika Merje Meinberg^c, Natali Agu^c, Villem Aruoja^d, Olli-Pekka Smolander^{a,1,*}, Ott Scheler^{a,1,**}

^a Department of Chemistry and Biotechnology, School of Science, Tallinn University of Technology, Akadeemia tee 15, 12618, Tallinn, Estonia

^b MATTER, Institute of Technology, University of Tartu, Nooruse 1, 50411, Tartu, Estonia

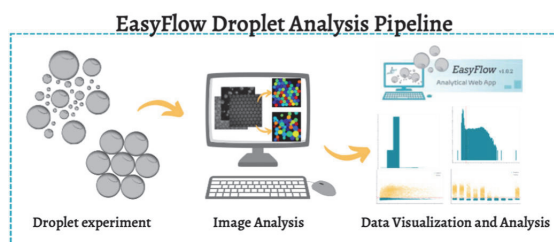
^c Rapla Gymnasium, Kooli 8, 79513, Rapla, Estonia

^d Laboratory of Environmental Toxicology, National Institute of Chemical Physics and Biophysics, Akadeemia tee 23, 12618, Tallinn, Estonia

HIGHLIGHTS

- Image-analysis software are combined with data visualization tool EasyFlow for user-friendly analysis of droplet arrays.
- EasyFlow simplifies building analysis and visualization pipelines for droplet arrays.
- EasyFlow analyses your droplet data based on their size, signal and labels.

GRAPHICAL ABSTRACT



ARTICLE INFO

Keywords:

EasyFlow
User-friendly
Droplet analysis
High-throughput analysis

ABSTRACT

Water-in-oil droplets allow performing massive experimental parallelization and high-throughput studies, such as single-cell experiments. However, analyzing such vast arrays of droplets usually requires advanced expertise and sophisticated workflow tools, which limits accessibility for a wider user base in the fields of chemistry and biology. Thus, there is a need for more user-friendly tools for droplet analysis. In this article, we deliver a set of analytical pipelines for user-friendly analysis of typical scenarios in droplet experiments. We built pipelines that combine various open-source image-analysis software with a custom-developed data processing tool called “EasyFlow”. Our pipelines are applicable to the typical experimental scenarios that users encounter when working with droplets: i) mono- and polydisperse droplets, ii) brightfield and fluorescent images, iii) droplet and object detection, iv) signal profile of droplets and objects (e.g., fluorescence).

1. Introduction

Droplet technologies enable massive parallelization of experiments

in chemical and biological laboratories. The method utilizes micro- or nanoscale water-in-oil droplets that are generated by mixing immiscible liquids (water and oil) [1,2]. Various tools can be used for droplet

* Corresponding author.

** Corresponding author.

E-mail addresses: olli-pekka.smolander@taltech.ee (O.-P. Smolander), ott.scheler@taltech.ee (O. Scheler).

¹ Equal contribution.

generation, e.g., microfluidics, vortexing, and manual shaking [3,4]. Droplets are widely used for research purposes e.g., in high-throughput sequencing and digital droplet PCR [5–7]. Droplet-based platforms have been increasingly used in microbial studies [8]: e.g., in microbiome, individual microbe or microbial community studies [9–11] and anti-microbial heteroresistance studies [12]. The technologies are also used in other fields, e.g., in microalgae's lipid production [13], in immunoassay of glycoprotein treatment and profiling [14], and metabolomics analysis [15].

Imaging is one of the most used methods in droplet experiments [16]. Different imaging approaches are used for the analysis of droplets: light and fluorescent microscopy [17], high resolution microscopy (e.g., scanning electron microscopy) [18], or a built-in smartphone device [19]. Droplet imaging has been applied to analyze droplet encapsulation rate, quantify objects (e.g., metal particles and bacteria), isolate protein crystal, amplify nucleic acid, and for susceptibility tests [3,20–23]. Droplet imaging often generates vast amount of raw data that needs to be processed further for proper interpretation of results.

Image analysis often demands sophisticated workflow and programming skills or image analysis experts for the analysis. For example, the image data needs to be processed to find the objects of interest and extract information from them in readable formats, such as a tabular data in comma separated values (.csv). There are different image analysis software which available online, e.g., CellProfiler™, ImageJ, Ilastik, QuPath, Icy, etc. [24–28]. These software tools are user-friendly, share the same principles and are often supported by tutorials for new users [29]. Recently, we built a user-friendly detection pipeline for droplet microfluidics using CellProfiler™ and CellProfiler Analyst™ [30]. Thus far, there are very few such published full analytical pipelines for detecting droplets and to process the high-throughput data. Moreover, the further data processing often needs programming proficiency or advanced data analysis tools, e.g., Python/C++/MATLAB/R [23, 31–35]. This limitation leads to low reproducibility and a steep learning curve in applying droplet tools for a wider user base.

Here, we address this gap by introducing a set of user-friendly analytical pipelines that combine open-source image-analysis software with a custom-developed data processing tool called “EasyFlow”. We

demonstrate the wide applicability of developed pipelines on different experimental droplet array image datasets. In our paper we use a term “droplet array” to describe a situation where droplets are positioned for the analysis as a layer (e.g. for imaging).

2. Results and discussion

We build our pipelines combining various open-source image-analysis software (Ilastik, CellProfiler™) with a custom-developed data processing tool called “EasyFlow”. These pipelines are user-friendly and designed to be immediately useable by common researchers. These pipelines require no previous experience in programming or image analysis software packages developed using Matlab, R, C++ or any other environment.

EasyFlow is a web application written in Python [36] that performs calculations, data grouping, and visualization for droplet data. EasyFlow uses the Pandas [37] and NumPy [38] libraries to process the output data from image analysis software, perform basic statistics and binning to match with required data for visualization.

EasyFlow utilizes the Bokeh [39] library for generating plots and is bundled in the Streamlit [40] library to present them in a form of web application. EasyFlow can be used and accessed at <https://easyflow.taltech.ee> (Fig. 1A). It can process image data acquired from droplets with varying content and labels. We tested EasyFlow's capabilities by conducting four experiments using brightfield and fluorescent image data that represents both mono- and polydisperse droplet settings (Fig. 1B). In these experiments, we encapsulated bacteria, microplastic beads (or microbeads), and microalgae as our objects of interest. The detailed settings are described in the methods section (pipeline construction and detection modules). We generated monodisperse droplets as described in our research in Bartkova et al. [30] and polydisperse droplets as shown in Byrnes et al. [41]. We performed brightfield imaging for droplets with microplastic beads and fluorescent microscopy for bacteria and microalgae imaging.

Every image analysis scenario requires an individual pipeline constructed using image analysis software. For this, the user compiles and constructs software modules according to their need [42]. In some cases,

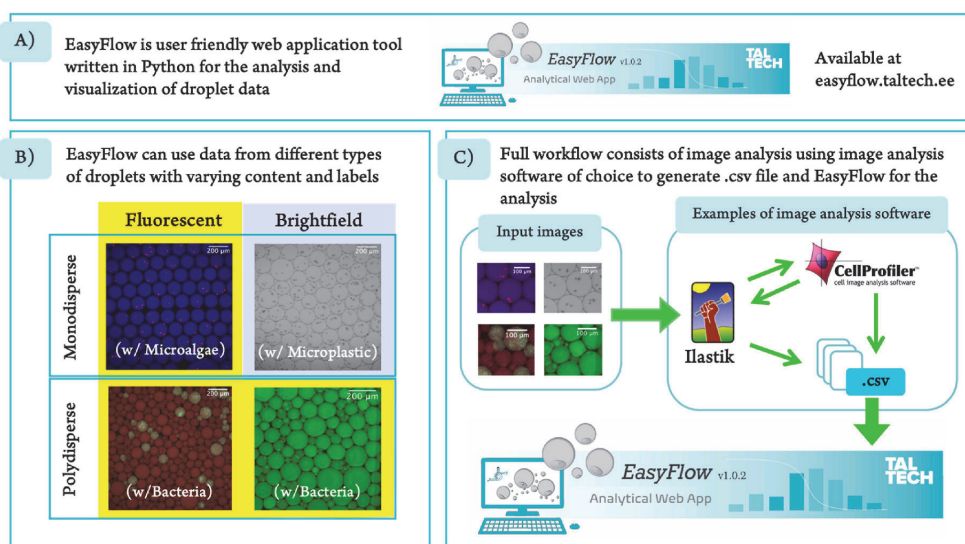


Fig. 1. (A) EasyFlow (easyflow.taltech.ee) simplifies building pipelines for droplet-based data analysis and visualization. (B) Pipelines with EasyFlow are suitable for mono-andpolydisperse droplets and for both brightfield and fluorescent images. For pipeline development we used freely available software (CellProfiler™ and Ilastik) that generate results in.csv file. (C) EasyFlow can host or process data which is generated by any image processing software and conforms with the required format.

a pipeline can also be assembled using different software combinations [43,44]. For instance, we used both CellProfiler™ [24] and Ilastik [26] to detect droplets and our objects of interest (Fig. 1C). We used the software because they have high accuracy and precision when it comes to droplet detection [29]. The detailed pipeline will be discussed in the section where we discuss the combination of software, e.g., Ilastik and CellProfiler™ for detecting microplastic beads. In our examples, we generated all.csv data using ExportToSpreadsheet module from CellProfiler™. From image analysis software we transferred the data (as.csv) into EasyFlow web-application that generates quick analysis and visualizations of experimental data. Even though we used CellProfiler™ to generate the.csv file, EasyFlow can also process any.csv or.xlsx file that is generated from other image analysis software.

Easyflow uses.csv or.xlsx files as input to automatically visualize droplet data in four different graphs. With EasyFlow, the user can obtain i) a droplet size distribution, ii) droplet signal distribution, iii) the relationship between droplet size and signal, and iv) a comparison of experimental conditions (label) (Fig. 2).

- i) By using size distribution result, we were able to determine whether the droplet's sizes were homogenous or heterogeneous, in which monodisperse or polydisperse droplets were used, respectively.
- ii) For the signal histogram, it shows pixel intensities from detected droplets. This histogram can distinguish two types of objects, or, in our example case, empty droplets and droplets with an

encapsulated object. We built EasyFlow in Python and used the Bokeh library for the visualization. Therefore, this signal histogram in EasyFlow can show result in either logarithmic or linear scale which is provided in a ready-to-show tab.

- iii) The relationship between size and signal data provides signal distribution against the volume which can be used to indicate whether the droplets are clumped only in specific volumes or distributed evenly.
- iv) The comparison of experimental conditions shows the signal distribution with its designated label.

All of the graph types are widely used in droplet-based experiments, e.g., droplet size comparison [45], pixel distribution [46], pixel intensity in different experimental condition(s) [12]. This shows that EasyFlow includes relevant analysis options and can simplify data processing and results generation. Furthermore, it is adaptable to any experimental setting. EasyFlow does not have any significant minimum hardware requirements, software dependencies and only needs an internet connection and an internet browser (e.g., Google Chrome [47], Mozilla Firefox [48], Safari [49], or Edge from Microsoft [50]) to process the data. EasyFlow can also be accessed using a smartphone (both Android or iPhone) or any other device (e.g., tablet) which has access to internet browser.

For user comfort, EasyFlow provides tunable thresholding and data binning (Supp. Figure 2). EasyFlow has a thresholding feature which facilitates user to distinguish two types of signal data. The histograms

EasyFlow provides full profile of droplet-based experiment in four graphs

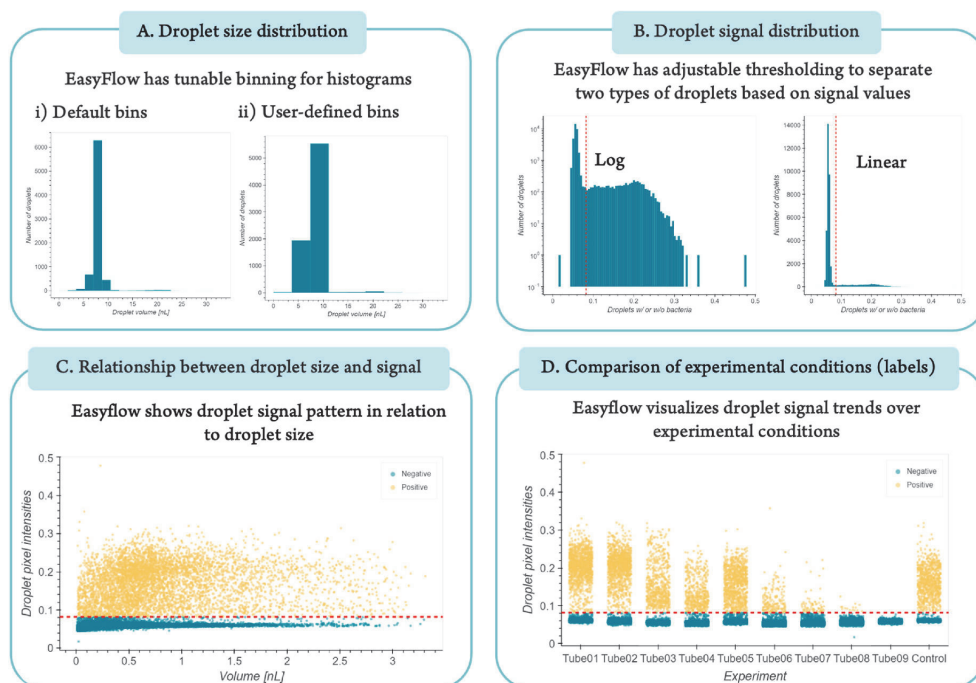


Fig. 2. EasyFlow provides four features and capabilities in processing image analysis software output. EasyFlow generates essential graphs, including (A) signal histogram, (B) distribution of droplet signals, (C) relationship between droplet size and signal data, and (D) comparison of experimental condition (label) graphs. These graphs are commonly used to find a quick analysis for droplet-based data. EasyFlow also provides flexibility for adjusting and generating a threshold value and binning options for histograms (Supp. Fig. 1). The threshold value will help user to classify two types of signal data and binning options provides tunable grouping for signal and size data. In addition, we also add binning table and basic statistical data, e.g., mean, standard deviation, and coefficient of variation in each graph (Supp. Fig. 2). The signal histogram binning can be adjusted depending on user's preferences, e.g. A-i) default binning from EasyFlow based on input data and A-ii) adjusted binning to have less data grouping.

also have interactive pop-out information panels to show the detailed information regarding data binning. Using this feature, user is able to select the threshold by browsing the data or finding the right binning for the data. As a default, EasyFlow provides thresholding which is set to detect minimum value after first maxima of the signal data. EasyFlow utilizes this thresholding to define the types of droplets in the

relationship between size and signal data and condition-based experiment data, in which later followed by basic statistics calculation (e.g., standard deviation and percentage of coefficient variation). The binning options allow EasyFlow to generate histograms based on the range of defined groups or the number of defined bins. EasyFlow has the flexibility to set the binning using number of bins or break points (bin edges).

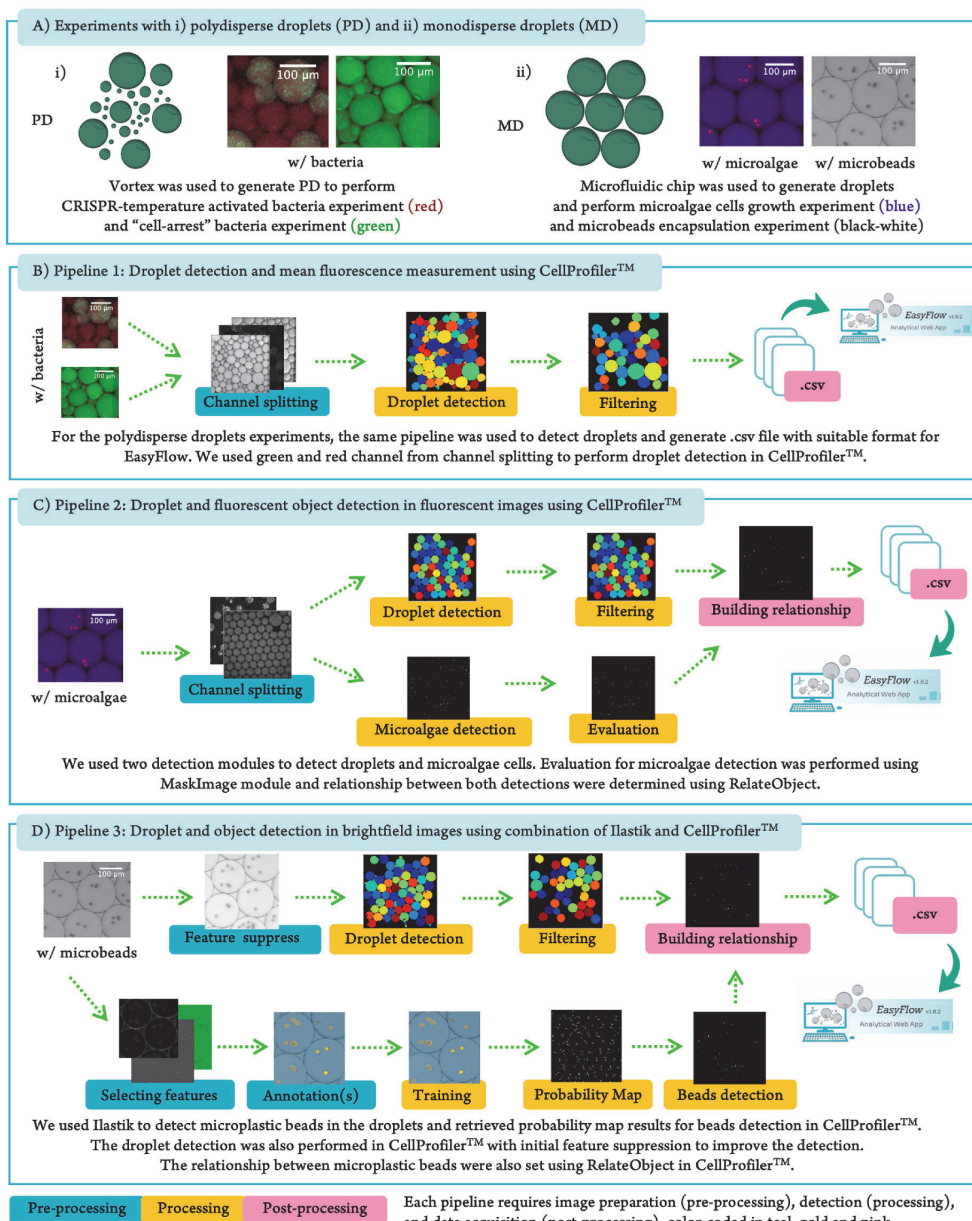


Fig. 3. Three image analysis pipelines accommodate typical droplet-based experiment analysis. (A) We used different type of droplets, i) polydisperse droplets which were generated using vortex and ii) monodisperse droplets which were produced using microfluidics setup. We developed three pipelines to perform the analysis that cover (B) droplet detection and fluorescence measurement, (C) droplet and object detection using fluorescent images, and (D) droplet and object detection using brightfield images. We built these pipelines using user-friendly image analysis software CellProfiler™ and Ilastik. All of the results generated from these pipelines were exported using ExportToSpreadsheet module from CellProfiler™ and ready for Ilastik.

EasyFlow also provides an option for users to insert custom plot titles and toggle off the thresholding line. For now, EasyFlow can host 200 megabytes.csv or.xlsx file and able to process up to 1 million data point where it should be enough for most droplet-based experiments.

2.1. Image analysis pipelines for droplet-based experiments

We provide three different pipeline examples for common experimental scenarios that use either mono- or polydisperse droplets (Fig. 3A). In our pipelines, we use open-source CellProfiler™ and Ilastik because of their easy-to-use, reproducible, and user-friendly characteristics. We tested these software to detect droplets in our previous research [30] and now, we combined them in our detection pipelines. In brief, we used CellProfiler™ to detect droplets (both mono- and poly-disperse) and microalgae cells and Ilastik to detect microplastic beads.

We prepared three pipelines using three stages (pre-processing, processing, and post-processing), which we have described previously in Sanka et al. [29]. In pre-processing, we adjusted and prepared the image data to suit the processing part, including channel splitting, greyscale converting, feature selection, and annotation. After the images were ready for processing, we proceeded with processing each image. For that we segmented pixels into partitioned image data and used them to detect droplets or our object of interest. Usually, processing considers color, intensity, or texture to retrieve specific object(s) or type of data [51,52]. In processing, we utilized modules that are available in the software to detect droplets and objects in the droplet, including thresholding, training, masking, filtering, measuring, and making relationships between detected objects (droplets and objects inside the droplets). After the processing step, we only needed the.csv file from detected droplets and objects in the post-processing step. Furthermore, we exported them into a single file which is ready to use in EasyFlow.

Pipeline 1: Droplet detection and mean fluorescence measurement using CellProfiler™. In this experiment, we had multi-channel fluorescent images [53]. The pipeline started with pre-processing using ColorToGray module in CellProfiler™. This module splits the image data as which were stacked from different channels. We use false coloring of red, green, and blue only for visualization. In CellProfiler™, we used IdentifyPrimaryObject module to detect droplets. This module can perform pixels classification using available thresholding strategies. From available thresholding strategies, we tested the algorithms that worked best for the image data. We initially tested Adaptive and Global thresholding strategies and concluded that Adaptive Otsu works best for our droplet detections (Supp. Table 1 and Supp. Figure 3). Using this strategy, we detected droplets and measured each droplet using MeasureSizeShape and MeasureObjectIntensity modules that are also available in CellProfiler™. These measurements are beneficial to perform detection evaluation. For instance, we evaluated the droplet detection using Filtering module. We evaluated droplet detection by applying rules in the measurement results. This filtering improves droplet detection and disregards droplets that have a distorted shape. These droplets have imperfect segmentation that makes droplet's shape not circular. The droplets which we aim to get have specific range of eccentricity (conic section), solidity (overall concavity), and form factor (ratio between object's area and circumscribed circle) (Supp. Figure 4 and 5). These parameters are commonly used to eliminate non-circular or non-ball-like objects [54,55]. For the post-processing, we only need to export the results as a.csv file. In this case, we used ExportToSpreadsheet module in CellProfiler™ and retrieved the.csv file which is a suitable format for EasyFlow.

Pipeline 2: droplet and fluorescent object detection in fluorescent images using CellProfiler™. This pipeline uses CellProfiler™ software and starts with color splitting using ColorToGrey module for pre-processing. The processing starts on droplet detection and object detection (microalgae). We used blue channel for the droplet and red channel for our object detection. We used the same IdentifyPrimaryObject module as described previously in the Pipeline 1. For detected

droplets, we also measured the pixels and size using MeasureSizeShape and MeasureObjectIntensity and evaluated the detection using Filtering module. Since we needed to enhance the pixels to detect our object of interest, we used EnhanceOrSuppressFeatures before implementing another IdentifyPrimaryObject. This helps to detect the objects, especially when there are more than one objects in droplet, or they are close to each other. We used MaskImage module to remove detected objects which were not in the detected droplets. To build a relationship between detected droplets and our object of interest, we added RelateObject module in the pipeline. From these results, we exported the results using ExportToSpreadsheet module to obtain the.csv file for EasyFlow.

Pipeline 3: droplet and object detection in brightfield images using combination of Ilastik and CellProfiler™. We used CellProfiler™ to detect droplets and Ilastik to detect objects (microplastic beads). Ilastik is a supervised machine learning image processing software [26]. This software has a built-in pipeline called *Pixel Classification and Object Identification* that helps detecting objects in our brightfield images. For detecting the objects, we started with features introduction and annotation(s). We set three sigma values for each feature (color/-intensity, edge, texture) which is mandatory before annotating image data. For the annotation(s), we selected and labeled the objects and background of an image. The background refers to everything but objects. After determining the labels, we set the thresholding and selected another feature from detected objects using Object Feature Selection as Standard Object Features. For the last step of processing, we distinguished detected objects into "beads" and "false detection of beads". This step was completed with a post-processing step, a probability map generation, where all steps were repeated for all images using Batch Processing module. In CellProfiler™ we prepared modules that utilize both the original image and the probability map image. For detecting droplets, we used EnhanceOrSurpress module to suppress image features that interrupt droplet detection. The features can be any objects but droplets. These preliminary steps are common to improve the detection [56,57]. After the pre-processing step, we detected droplets using IdentifyPrimaryObject followed by MeasureSizeShape, MeasureObjectIntensity, and Filtering modules. In CellProfiler, we performed object detection using probability map image data which were generated in Ilastik and followed by MeasureObjectSizeShape module. Using both droplets and objects detection results, we built a relationship between them using RelateObject module. To generate the.csv file, we used the same module as in other pipelines: ExportToSpreadsheet.

2.2. Demonstration of EasyFlow pipelines in experimental scenarios

Each experiment utilizes droplets which were generated by using either microfluidic chip or vortexing. Reagents used are Novec HFE 7500 fluorocarbon oil with 2% concentration of perfluoropolyether (PFPE)-poly(ethyleneglycol) (PEG)-PFPE triblock surfactant for the continuous phase and water based medium for the dispersed phase. Droplets can remain stable for more than 72 h.

1a Controlled activation of bacterial growth (Fig. 4A). Bacteria growth in droplets can be turned ON/OFF using a CRISPR-based system. We encapsulated and incubated bacteria containing a CRISPR system overnight in polydisperse droplets. The CRISPR system has an anhydrotetracycline (aTc) "toggle switch" introduced by Gardner [58], whereby it is turned ON in the presence of aTc and OFF when aTc is absent. When the system is ON, the bacteria growth is inhibited and if the system is turned OFF the bacteria can grow. The CRISPR system is also temperature sensitive. During incubation at 37 °C or lower, the CRISPR system functions as it should. However, incubation at high temperature such as 42 °C renders it inactive even when the "ON switch" aTc is present. This corresponds to previous work by Wiktor et al. [59] where the cells can restart their replication and proliferation after switching the temperature at 42 °C. Therefore, we start seeing

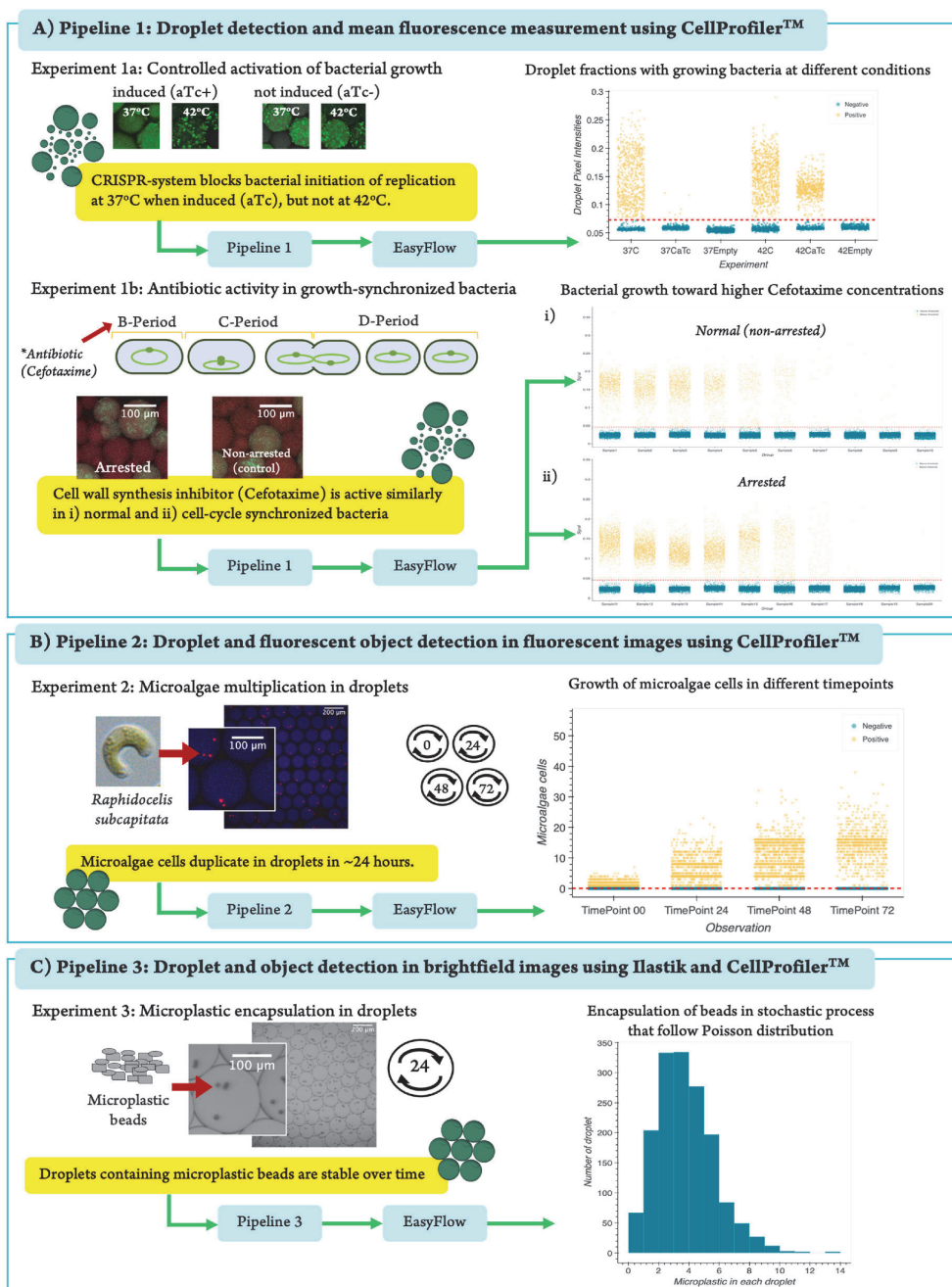


Fig. 4. Demonstration of EasyFlow pipelines in experimental scenarios. A) Here, we show that bacteria growth can be controlled via temperature sensitive CRISPR system (Exp. 1a) and bacteria respond to antibiotics similarly regardless of their synchronization in cell cycle (Exp. 1b). B) Microalgae multiply in droplets stably over 72 h incubation period (Exp. 2). C) Microplastic beads do not affect the stability of droplets during 24 h incubation (Exp. 3). In this figure, we show only one representative graph for each experiment. Full info for each experiment is added in the Supp. Fig. 10.

bacterial growth in the label-based graphs at 42°C label in the figure.

We used the signal distribution to find a threshold which can be applied to classify droplets, both empty droplets and droplets with growing bacteria. This threshold was set using the method which we have mentioned previously in Fig. 2B. In this experiment, we had 5119 droplets in total using 198 images.

1b Antibiotic activity in growth-synchronized bacteria (Fig. 4A).

Here we encapsulated bacteria to investigate if antibiotic response of bacteria differs depending on the stage of their cell cycle. Bacteria (in our experiment *E. coli*) have a cell cycle consisting of three distinguished periods: i) the B-period where DNA replication is initiated, ii) the C-period which starts from replication through termination and iii) the D-period when the termination ends, and the division of bacteria starts [60]. We used serine hydroxamate (SHX) to arrest the DNA replication [61] and thus synchronized the cell cycle of bacteria in their B-period. We then added different concentrations of antibiotic and removed SHX to let bacteria replicate again in the presence of the antibiotic, all starting from their B-period. The response was compared to non-arrested bacteria (control group) where same concentrations of antibiotics were added, but SHX was absent. Based on our experiment, there was no difference in viability between the arrested and non-arrested (control) bacteria groups. This indicates that being in the B-period of the cell cycle does not affect antibiotic response of the bacteria. Nevertheless, we were able to show EasyFlow's capability in giving quick analysis in this "cell-cycle arrest" experiment. In this experiment we had 108,129 droplets from 900 images.

2. **Microalgae multiplication in droplets (Fig. 4B).** Multiplication of microalgae *Raphidocelis subcapitata* cells in droplets can easily be monitored over time. The microalgae are widely used for toxicology assays [62,63]. In our droplet environment, the algae multiplied stably throughout the whole 72 h incubation period. This is important because it has been shown in traditional incubation settings that microalgae growth can slow down after a certain density is reached [64]. This analysis is based on 7607 droplets which we detected from 146 images.
3. **Microplastic encapsulation in droplets (Fig. 4C).** Droplets are also a suitable and stable platform for the analysis of microplastics. Microplastic pollution has developed into a serious environmental concern. It appears that microplastic has an increasingly detrimental impact on all life forms and especially in the marine and remote environments [65]. Therefore, there is a need to provide a user-friendly high-throughput platform to assess the effect of microplastic to microorganisms. Here, we show that droplets are a suitable for this as they are stable over time in microplastic presence. We obtained droplets containing different number of microplastic objects, ranging from 1 to 14. Droplet encapsulation followed theoretical Poisson distribution (Supp. Figure 6 and Supp. Table 2). We also observed that the droplets remained stable with additional oil during incubation. In this experiment, we had 45 images with detected 1591 droplets after 24 h of incubation

2.3. Pipeline limitations in image-based droplet analysis

Droplet type and image acquisition techniques determine the quality of image data, which affects subsequent analysis. For instance, the use of monodisperse droplets give higher detection accuracy than using polydisperse droplets (from 80% to 97% for monodisperse and 53%–64% for polydisperse) (Supp. Figure 7). The accuracy of object detection inside the droplets was 72% and 96% with microbeads and microalgae, respectively (Supp. Figure 8). Another issue with polydisperse droplets

is the overlapping of droplets with different sizes. This can cause segmentation failure and lower the detection quality (Supp. Figure 4 and Fig. 9). In this case the failure of segmentation is filtered out and is not counted in our detection pipelines (Supp. Figure 9).

3. Conclusion

We provide user-friendly droplet analysis pipelines for high-throughput studies in chemistry and biotechnology. Our pipelines can process different experimental scenarios: e.g., droplet or droplet-object analysis in either brightfield or in fluorescent images. Our pipelines provide user-friendly analysis tools with low-learning curve for any researcher who has need to analyze droplet arrays. Our pipelines have a user-friendly web-based analysis tool EasyFlow, which can process and visualize image analysis data from droplet experiments. EasyFlow has shallow learning curve and thus allows user, especially a non-programmer specialist, to easily analyze image-based droplet data and interpret the results.

By introducing these pipelines to the community, we wish to democratize droplet-based techniques and enable the easy analysis of the results. In EasyFlow, we use.csv (comma separated value) or.xlsx (Excel) filetype as an input. This allows the use of almost any image analysis software to produce the input data. In addition, EasyFlow is written in Python and is open source, and therefore a further development to include additional calculations or features is simple. Moreover, EasyFlow is not limited to droplet-based data. In principle, any data that is stored as.csv or.xlsx files and contains labels, sizes, and associated signals can be used in EasyFlow-based pipelines.

4. Methods

EasyFlow software development. EasyFlow is written in Python [36] using the following libraries, such as NumPy [38] for working with arrays, Pandas [37] to serve the data as a table or data frame, Math [66] and Statistics [67] to perform the calculations, Regex [68] to execute regular expression command, and Matplotlib [69] and Bokeh [39] to visualize the results and to provide users with necessary interactivity. These scripts are utilized by a user-friendly interface built with Streamlit [40] environment and library to give user-friendly interface. EasyFlow is hosted and deployed in a local server (in virtual machine running in Ubuntu Server 20.04 LTS) in Tallinn University of Technology (Taltech) accessible at <https://easyflow.taltech.ee> domain and the most recent version of EasyFlow's source code is available in our GitHub repository <https://github.com/taltechloc/sw-easyflow-v1>.

Droplet generation. Droplets are generated using the following materials: oil, surfactant, object of interest and its medium. In brief, we used surfactant (perfluoropolyether (PFPE)–poly (ethylene glycol) (PEG)–PFPE triblock surfactant) and Novex HFE 7500 fluorocarbon oil to make the immiscible liquid/layer in the droplet formation. In experiments 1a and 1b, we generated polydisperse droplets by adding object of interest with medium and the oil/surfactant phase in a 1:1 ratio in 5 mL Eppendorf tubes and vortexed for 5 s. For experiments two and three, microfluidics setup is used for droplet generation to produce monodisperse droplets, which is described in our previous work [30].

Experimental design. For experiment 1a, our object of interest was *Escherichia coli* JEK 1036 with a chromosome-incorporated gene encoding the green fluorescence protein (GFP) and plasmid pdCas9deg3 containing the CRISPR activated system and antibiotic resistance gene for chloramphenicol. We used Luria broth mixed with Chloramphenicol and Dextran, Alexa Fluor™ 647 (Invitrogen, Life Technologies Corporation) for the medium. There were three different batches of droplets containing: i) bacteria and medium, ii) only medium, and iii) bacteria in medium with added anhydrotetracycline (aTc). Each batch was split into two tubes that were incubated at 37 °C and 42 °C respectively. In experiment 1b where the cell-cycle was investigated, our object of interest was *Escherichia coli* JEK 1036 without plasmid pdCas9deg3. We

used Luria broth mixed with Alexa Fluor™ 647 (Invitrogen, Life Technologies Corporation) and nine different concentrations of Cefotaxime for the medium, wherein we incubated either serine hydroxamate (SHX) treated *E. coli* that were all starting growth from the B-period, or non-treated (control) *E. coli*. We incubated all 20 samples for 24 h at 37 °C and did an imaging after the incubation. In the third experiment (microalgae growth), our objects of interest were microalgae *Raphidocelis subcapitata* cells. We used OECD 201 medium containing nitrogen, phosphorus, calcium, potassium, magnesium, microelements, and vitamins mixed with Dextran, Cascade Blue™ (Invitrogen, Life Technologies Corporation) as the medium and kept the microalgae at room temperature illuminated by a LED (Light Emitting Diode) table lamp, and imaged at 0 h, 24 h, 48 h, and 72 h to observe the microalgae growth. The 0-h time point was our baseline of growth that we compared to the other timepoints. For the fourth experiment, we prepared droplets containing 10µm polyethylene microplastic beads in sterile water, incubated the sample for 24 h 37° followed by imaging.

Imaging setup. In the first experiment (CRISPR activated system), LSM 510 Laser Scanning Microscope (Zeiss, Germany) was used and set on Zen 2009 software with the following settings: Plan-Apochromat 10X/0.45 objective, Argon/2 and HeNe633 lasers, Transmission light (Bright Field), pinhole size 452 µm. For the rest of the experiments, Zeiss LSM 900 confocal laser scanning microscope (Zeiss, Germany) running on Zen 3.3 software (blue edition) was used with the following settings: Plan-Apochromat 10X/0.45 objective, diode lasers 640, 488 and 405 nm, DIC light, pinhole size 460 µm.

Pipeline construction. In this experiments, three pipelines were constructed to analyze image data from our experiments, i) to detect droplets and measure the fluorescent of bacteria, ii) to detect droplets and microalgae cells in monodisperse droplet, and iii) to detect droplets and microplastic beads in brightfield images.

For Pipeline 1, CellProfiler™ (version 4.2.1) was used to analyze the images and constructed the available modules into a pipeline. The modules which were used involved: ColorToGray, IdentifyPrimaryObjects, MeasureSizeShape, MeasureObjectIntensity, FilterObjects, and ExportToSpreadsheet. Each module hosts variables that were mandatory for droplet detection, for instance, the range of diameter in IdentifyPrimaryObjects module which were set from 20 to 400-pixel units or 12.5–250 µm. Selected thresholding algorithm, Adaptive with Otsu algorithm, was used and 350 of adaptive window was set for droplet detection. Different correction factors and smoothing scale were tested to find the right setting for this pipeline. In the MeasureSizeShape and MeasureObjectIntensity, droplets profile was measured (including their mean intensity and size in pixel size) from the images. For the FilterObject, different settings were tested including eccentricity, solidity, and form factor. In this pipeline, eccentricity was used with the range of 0–0.5 and solidity with 0.93–1.00 for the filtering. Results were retrieved in comma separated value (.csv) format using ExportToSpreadsheet. Each experiment was performed in batch using 198 images for CRISPR activated system experiment and 900 images for cell-cycle experiment. The step-by-step to construct a pipeline in CellProfiler can be found in our previous work [30].

Pipeline 2 is also performed only in CellProfiler™ (version 4.2.1). However, additional modules compared to Pipeline 1 were needed to detect microalgae cells. The pipeline starts with ColorToGray module to split image channels. There were two channels which were used, blue channel to detect droplets and red channel to detect the microalgae cells. The droplet detection had similar setting compared to Pipeline 1 with some adjustments in the adaptive window size, thresholding smoothing scale and correction factor. For the microalgae detection, EnhanceOrSuppressFeatures was added to enhance the microalgae cells images using “Speckles” feature type with the size of 50 in “Fast” mode. MaskImage module was also added to filter the microalgae cells which are present in the detected droplets. The IdentifyPrimaryObjects module was set with 10-50-pixel unit using Adaptive Robust Background algorithm in 50 window size to detect microalgae cells. The “Mean” was

selected as the averaging method with “Standard Deviation” as the variance method in the setting. This microalgae cells detection was followed by MeasureObjectIntensity and MeasureSizeAndShape to retrieve the measurement data. After performing both detections, RelateObjects module was added to build the relationship between droplet and microalgae cells detections. The results were exported as.csv file format using ExportToSpreadsheet module. For this experiment, 146 images were used in a batch analysis.

In Pipeline 3, two software, CellProfiler™ (version 4.2.1) and Ilastik (version 1.3.3), were used to detect droplets and microplastic beads and to build the relationship between both detected objects. Ilastik was used to detect the microplastic beads and to provide the probability map that represents the beads in each image. There was a pre-defined workflow, Pixel Classification and Object Classification, from Ilastik which used to detect the beads. Detailed guideline for this workflow usage can be found in our previous research methods published in Sanka et al. [29] The sigma or scale value of 0.30, 1.00, and 3.50 were selected for this experiment. Correspond to our previous method, the microplastic beads were annotated as our object of interest and labeled everything but the beads as a background. This annotation was used to train the program and determine the object of interest, in our case, it is between the beads and background. In thresholding, core and final values were determined as 0.65 with filter size ranging from 85 to 500 in pixels to enhance the detection. After the thresholding, second classification was performed to enhance microplastic beads detection and disregarded the wrong detected object. After second training, a probability map was retrieved in Object Information Export module. For this step, critical step needed to be performed to transpose the axis order into “cyx” and changed the filetype output as TIFF format. These settings can be found in “Choose Export Image Settings” option in the module. The probability map then can be imported into CellProfiler™ to perform re-detection of microplastic beads using IdentifyPrimaryObject module. Probability map simplifies the detection of microplastic beads, compared to direct detection from brightfield images. In this re-detection, 10 to 45-pixel units were set with Adaptive-Otsu thresholding strategy and method. The default settings were set to support the detection. To detect the droplets, microplastic beads were suppressed using EnhanceOrSuppressFeature module with “Suppress” operation and feature size of 20. For droplet detection, the same modules which explained in the previous pipelines were used using “Suppressed” images. FilterObjects module was also added in this pipeline and was set with specific range of eccentricity (0–0.5) and form factor of 0.8–1.0. After detecting both droplets, the relationship between detected droplets and microplastic particles were built using RelateObjects module. This module will give the number of microplastic beads in each droplet. Using ExportToSpreadsheet, the results are exported and retrieved as a.csv file. For this experiment, 45 images were used in a batch.

EasyFlow settings step-by-step. EasyFlow only requires data (either in.csv or.xlsx format) to be uploaded in the online platform. The uploading process can be performed either by drag-and-drop feature or by clicking the upload section which available on the homepage. Once the file is uploaded, all results will be shown in the platform and four figures will be presented directly. Additional setting to define threshold, size and signals bins was performed in Easyflow for our experiments. For detailed information, the value of 0.0702 was used for in CRISPR activated system experiment, 0.0459 for cell-cycle experiment, 0.9 for microalgae cells growth experiment and 0.9 for microplastic beads encapsulation rate experiment. The value of 0.9 for microalgae and microplastic were used since value less than 1.0 for threshold will give the value of the number of empty droplets. Since EasyFlow is a static web application, it does not save the data which is uploaded by user. Therefore, all images need to be saved once users are satisfied with the results or visualizations.

Notes

No additional information is present.

CRediT authorship contribution statement

Immanuel Sanka: Conceptualization, Methodology, Software, Validation, Data curation, Formal analysis, Visualization, Supervision, Writing – original draft. **Simona Bartkova:** Conceptualization, Investigation, Supervision, Formal analysis, Writing – original draft. **Pille Pata:** Investigation, Resources, Writing – review & editing. **Mart Ernits:** Software. **Monika Merje Meinberg:** Investigation. **Natali Agu:** Investigation. **Villem Aruoja:** Investigation, Supervision, Writing – review & editing. **Olli-Pekka Smolander:** Conceptualization, Funding acquisition, Supervision, Writing – review & editing. **Ott Scheler:** Conceptualization, Funding acquisition, Supervision, Writing – review & editing, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The research was performed partially in the laboratory established with the support from the TTU Development Program 2016–2022 (project no. 2014–2020.4.01.16.0032). We also acknowledge the Estonian Research Council grants MOBTP109, PRG620, MOBJD556, PRGI427, European Union's Horizon 2020 research and innovation program, Grant Agreement No. 856705 (ERA Chair "MATTER") and European Regional Development Fund in University of Tartu, Estonia. We also thank Prof. Piotr Garstecki (Institute of Physical Chemistry, Polish Academy of Sciences) for giving us the surfactant and microfluidic chip mold. In addition, we are grateful to Tamas Pardy, Ph.D. to improve this article via discussions.

Appendix A. Supplementary data

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.aca.2023.341397>.

References

- H. Zhang, G. Jenkins, Y. Zou, Z. Zhu, C.J. Yang, Massively parallel single-molecule and single-cell emulsion reverse transcription polymerase chain reaction using agarose droplet microfluidics, *Anal. Chem.* 84 (2012) 3599–3606, https://doi.org/10.1021/AC2033084.SUPPL_FILE/AC2033084_SI_001.PDF.
- Á. Ríos, M. Zougagh, M. Avila, Miniaturization through lab-on-a-chip: utopia or reality for routine laboratories? A review, *Anal. Chim. Acta* 740 (2012) 1–11, <https://doi.org/10.1016/j.aca.2012.06.024>.
- S.A. Byrnes, E.A. Phillips, T. Huynh, B.H. Weigl, K.P. Nichols, Polydisperse emulsion digital assay to enhance time to detection and extend dynamic range in bacterial cultures enabled by a statistical framework, *Analyst* 143 (2018) 2828–2836, <https://doi.org/10.1039/c8an00029h>.
- M. Najah, A.D. Griffiths, M. Ryckelynck, Teaching single-cell digital analysis using droplet-based microfluidics, *Anal. Chem.* 84 (2012) 1202–1209, <https://doi.org/10.1021/ac202645m>.
- K. Matula, F. Rivello, W.T.S. Huck, Single-cell analysis using droplet microfluidics, *Adv Biosyst* 4 (2020), 1900188, <https://doi.org/10.1002/adbi.201900188>.
- C.M. Hindson, J.R. Chevillet, H.A. Briggs, E.N. Gallichotte, I.K. Ruf, B.J. Hindson, R.L. Vessella, M. Tewari, Absolute quantification by droplet digital PCR versus analog real-time PCR, *Nat. Methods* 10 (2013) 1003–1005, <https://doi.org/10.1038/nmeth.2633>.
- B.J. Hindson, K.D. Ness, D.A. Masquelier, P. Belgrader, N.J. Heredia, A. J. Makarewicz, I.J. Bright, M.Y. Lucero, A.L. Hiddessen, T.C. Legler, T.K. Kitano, M. R. Hodel, J.F. Petersen, P.W. Wyatt, E.R. Steenblock, P.H. Shah, L.J. Bousse, C. B. Troup, J.C. Mellen, D.K. Wittmann, N.G. Erndt, T.H. Cauley, R.T. Koehler, A. P. So, S. Dube, A.A. Rose, L. Montesclaros, S. Wang, D.P. Stumbo, S.P. Hodges, S. Romine, F.P. Milanovich, H.E. White, J.F. Regan, G.A. Karlin-Neumann, C. M. Hindson, S. Saxonov, B.W. Colston, High-throughput droplet digital PCR system for absolute quantitation of DNA copy number, *Anal. Chem.* 83 (2011) 8604–8610, <https://doi.org/10.1021/ac202028g>.
- T.S. Kaminski, O. Scheler, P. Garstecki, Droplet microfluidics for microbiology: techniques, applications and challenges, *Lab Chip* 16 (2016) 2168–2187, <https://doi.org/10.1039/C6LC00367B>.
- J. Kehe, A. Kulesa, A. Ortiz, C.M. Ackerman, S.G. Thakku, D. Sellers, S. Kuehn, J. Gore, J. Friedman, P.C. Blainey, Massively parallel screening of synthetic microbial communities, *Proc. Natl. Acad. Sci. USA* 116 (2019) 12804–12809, <https://doi.org/10.1073/PNAS.1900102116>.
- L. Mahler, S.P. Niehs, K. Martin, T. Weber, K. Scherlach, C. Hertweck, M. Roth, M. A. Rosenbaum, Highly parallelized droplet cultivation and prioritization on antibiotic producers from natural microbial communities, *Elife* 10 (2021), <https://doi.org/10.7554/ELIFE.64774>.
- W.J. Watterson, M. Tanyeri, A.R. Watson, C.M. Cham, Y. Shan, E.B. Chang, A. M. Eren, S. Tay, Droplet-based high-throughput cultivation for accurate screening of antibiotic resistant gut microbes, *Elife* 9 (2020) 1–22, <https://doi.org/10.7554/ELIFE.56998>.
- O. Scheler, K. Makuch, P.R. Debski, M. Horka, A. Ruszczak, N. Pacocha, K. Sozański, O.P. Smolander, W. Postek, P. Garstecki, Droplet-based digital antibiotic susceptibility screen reveals single-cell clonal heteroresistance in an isogenic bacterial population, *Sci. Rep.* 10 (2020) 1–8, <https://doi.org/10.1038/s41598-020-60381-z>.
- G. Zheng, F. Gu, Y. Cui, L. Lu, X. Hu, L. Wang, Y. Wang, A microfluidic droplet array demonstrating high-throughput screening in individual lipid-producing microalgae, *Anal. Chim. Acta* 1227 (2022), 340322, <https://doi.org/10.1016/j.ACA.2022.340322>.
- T. Liénard-Mayor, C. Bricteux, A. Bendali, N.T. Tran, A. Bruneel, M. Taverna, T. D. Mai, Lab-in-droplet: from glycan sample treatment toward diagnostic screening of congenital disorders of glycosylation, *Anal. Chim. Acta* 1221 (2022), 340150, <https://doi.org/10.1016/j.ACA.2022.340150>.
- D. Feng, H. Li, T. Xu, F. Zheng, C. Hu, X. Shi, G. Xu, High-throughput single cell metabolomics and cellular heterogeneity exploration by inertial microfluidics coupled with pulsed electric field-induced electrospray ionization-high resolution mass spectrometry, *Anal. Chim. Acta* 1221 (2022), 340116, <https://doi.org/10.1016/j.ACA.2022.340116>.
- W. wen Liu, Y. Zhu, "Development and application of analytical detection techniques for droplet-based microfluidics"—A review, *Anal. Chim. Acta* 1113 (2020) 66–84, <https://doi.org/10.1016/j.ACA.2020.03.011>.
- A. Saateh, A. Kalantarifar, O.T. Celik, M. Asghari, M. Serhatilolu, C. Elbuken, Real-time impedimetric droplet measurement (iDM), *Lab Chip* 19 (2019) 3815–3824, <https://doi.org/10.1039/C9LC00641A>.
- Y. Zhu, Q. Fang, Analytical detection techniques for droplet microfluidics—A review, *Anal. Chim. Acta* 787 (2013) 24–35, <https://doi.org/10.1016/j.aca.2013.04.064>.
- N.A. Szydłowski, H. Jing, M. Alqashmi, Y.S. Hu, Cell phone digital microscopy using an oil droplet, *Biomed. Opt. Express* 11 (2020) 2328, <https://doi.org/10.1364/BOE.389345>.
- M. Vaithyanathan, N. Safa, A.T. Melvin, FluorocellTrack: an algorithm for automated analysis of high-throughput droplet microfluidic data, *PLoS One* 14 (2019), <https://doi.org/10.1371/journal.pone.0215337>.
- J. Tamminen, E. Lahdenperä, T. Koiranen, T. Kuronen, T. Eerola, L. Lensu, H. Kälviäinen, Determination of single droplet sizes, velocities and concentrations with image analysis for reactive extraction of copper, *Chem. Eng. Sci.* 167 (2017) 54–65, <https://doi.org/10.1016/j.CES.2017.03.048>.
- P. Vallotton, C. Sun, D. Lovell, V.J. Fazio, J. Newman, DroplIT, an improved image analysis method for droplet identification in high-throughput crystallization trials, *J. Appl. Crystallogr.* 43 (2010) 1548–1552, <https://doi.org/10.1107/S0021889810040963/CG5163SUP3.JPG>.
- C. Svensson, O. Shvydkiv, S. Dietrich, L. Mahler, T. Weber, M. Choudhary, M. Tovar, M.T. Figge, M. Roth, Coding of experimental conditions in microfluidic droplet assays using colored beads and machine learning supported image analysis, *Small* 1802384 (2018), 1802384, <https://doi.org/10.1002/smll.201802384>.
- A.E. Carpenter, T.R. Jones, M.R. Lamprecht, C. Clarke, I. Kang, O. Friman, D. A. Guertin, J. Chang, R.A. Lindquist, J. Moffat, P. Golland, D.M. Sabatini, CellProfiler: image analysis software for identifying and quantifying cell phenotypes, *Genome Biol.* 7 (2006) R100, <https://doi.org/10.1186/gb-2006-7-10-r100>.
- J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, B. Schmid, J.Y. Tinevez, D.J. White, V. Hartenstein, K. Elceiri, P. Tomancak, A. Cardona, Fiji: an open-source platform for biological-image analysis, *Nat. Methods* 9 (2012) 676–682, <https://doi.org/10.1038/nmeth.2019>.
- S. Berg, D. Kutra, T. Kroeger, C.N. Straehle, B.X. Kausler, C. Haubold, M. Schiegg, J. Ales, T. Beier, M. Rudy, K. Eren, J.I. Cervantes, B. Xu, F. Beuttenmueller, A. Wolny, C. Zhang, U. Koethe, F.A. Hamprecht, A. Kreshuk, Ilastik: interactive machine learning for (bio)image analysis, *Nat. Methods* 16 (2019) 1226–1232, <https://doi.org/10.1038/s41592-019-0582-9>.
- P. Bankhead, M.B. Loughrey, J.A. Fernández, Y. Dombrowski, D.G. McArt, P. D. Dunne, S. McQuaid, R.T. Gray, L.J. Murray, H.G. Coleman, J.A. James, M. Salto-Tellez, P.W. Hamilton, QuPath, Open-source software for digital pathology image analysis, *Sci. Rep.* 7 (2017) 1–7, <https://doi.org/10.1038/s41598-017-17204-5>.

- [28] F. de Chaumont, S. Dallongeville, N. Chenouard, N. Hervé, S. Pop, T. Provoost, V. Meas-Yedid, P. Pankajakshan, T. Lecomte, Y. le Montagner, T. Lagache, A. Dufour, J.C. Olivo-Marin, Icy: an open bioimage informatics platform for extended reproducible research, *9* (2012) 690–696, *Nat. Methods* 9 (2012) 7, [10.1038/nmeth.2075](https://doi.org/10.1038/nmeth.2075).
- [29] I. Sanka, S. Bartkova, P. Pata, O.-P. Smolander, O. Scheler, Investigation of different free image analysis software for high-throughput droplet detection, *ACS Omega* 6 (2021) 22625–22634, <https://doi.org/10.1021/ACSEOMEGA.1C02664>.
- [30] S. Bartkova, M. Vendelin, I. Sanka, P. Pata, O. Scheler, Droplet image analysis with user-friendly freeware CellProfiler, *Anal. Methods* 12 (2020) 2287–2294, <https://doi.org/10.1039/d0ay00031k>.
- [31] O.J. Dressler, P.D. Howes, J. Choo, A.J. Demello, Reinforcement learning for dynamic microfluidic control, *ACS Omega* 3 (2018) 10084–10091, <https://doi.org/10.1021/acsomega.8b01485>.
- [32] E. Zang, S. Brandes, M. Tovar, K. Martin, F. Mech, P. Horbert, T. Henkel, M. T. Figue, M. Roth, Real-time image processing for label-free enrichment of Actinobacteria cultivated in picolitre droplets, *Lab Chip* 13 (2013) 3707–3713, <https://doi.org/10.1039/c3lc50572c>.
- [33] S.S. Kumar, C. Li, C.E. Christen, C.J. Hogan, S.A. Fredericks, J. Hong, Automated droplet size distribution measurements using digital inline holography, *J. Aerosol Sci.* 137 (2019), 105442, <https://doi.org/10.1016/j.jaerosci.2019.105442>.
- [34] A. Chiu, M. Ayub, C. Dive, G. Brady, C.J. Miller, twodpcc: an R/Bioconductor package and Shiny app for Droplet Digital PCR analysis, *Bioinformatics* 33 (2017) 2743–2745, <https://doi.org/10.1093/bioinformatics/btx308>.
- [35] D. Attali, R. Bidshahri, C. Haynes, J. Bryan, Dpccr: an R Package and Web Application for Analysis of Droplet Digital PCR Data, 2016, p. 5, <https://doi.org/10.12688/F1000RESEARCH.9022.1>. F1000Res.
- [36] Welcome to Python.org (n.d.), <https://www.python.org/>. (Accessed 16 January 2021).
- [37] The Pandas Development Team, pandas-dev/pandas 3 (2020), <https://doi.org/10.5281/ZENODO.4572994>, Pandas 1.2, <https://zenodo.org/record/4572994>.
- [38] C.R. Harris, K.J. Millman, S.J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N.J. Smith, R. Kern, M. Picus, S. Hoyer, M.H. van Kerkwijk, M. Brett, A. Haldane, J.F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, T.E. Oliphant, Array programming with NumPy, *Nature* 585 (2020) 357–362, <https://doi.org/10.1038/s41586-020-2649-2>.
- [39] Bokeh Development Team, Bokeh: Python Library for Interactive Visualization, 2018, <http://www.bokeh.pydata.org>.
- [40] (n.d. Streamlit, <https://streamlit.io/>. (Accessed 3 November 2021).
- [41] S.A. Byrnes, T.C. Chang, T. Huynh, A. Astashkina, B.H. Weigl, K.P. Nichols, Simple polydisperse droplet emulsion polymerase chain reaction with statistical volumetric correction compared with microfluidic droplet digital polymerase chain reaction, *Anal. Chem.* 90 (2018) 9374–9380, <https://doi.org/10.1021/acs.analchem.8b01988>.
- [42] K. Miura, Bioimage Data Analysis Workflows, 2020.
- [43] X. Xiao, Y. Qiao, Y. Jiao, N. Fu, W. Yang, L. Wang, R. Yu, J. Han, Dice-Xmbd, Deep learning-based cell segmentation for imaging mass cytometry, *Front. Genet.* 12 (2021), 721229, <https://doi.org/10.3389/fgene.2021.721229/FULL>.
- [44] D.J. Logan, J. Shan, S.N. Bhatia, A.E. Carpenter, Quantifying co-cultured cell phenotypes in high-throughput using pixel-based classification, *Methods* 96 (2016) 6, <https://doi.org/10.1016/j.ymeth.2015.12.002>.
- [45] M. Sun, Z. Li, S. Wang, G. Maryu, Q. Yang, Building dynamic cellular machineries in droplet-based artificial cells with single-droplet tracking and analysis, *Anal. Chem.* (2019), <https://doi.org/10.1021/acs.analchem.9b01481>.
- [46] G.P. Rutkowski, I. Azizov, E. Unmann, M. Dudek, B.A. Grimes, Microfluidic droplet detection via region-based and single-pass convolutional neural networks with comparison to conventional image analysis methodologies, *Machine Learning with Applications* 7 (2022), 100222, <https://doi.org/10.1016/j.mlwa.2021.100222>.
- [47] Google Chrome - Download the Fast, Secure Browser from Google, (n.d.), https://www.google.com/chrome/?brand=YTUHF&gclid=Cj0KCQjwvjaYBhDIARisA08PKE16EZSJ5Jona4qbVXEkh7wAjhvFlxXZ8j3HZ2Gzcng3t0aKalz3waAm6ZEA1w_wcB&gclid=aw.ds (Accessed September 11, 2022).
- [48] Download Firefox Browser — Fast, Private & free — from Mozilla, n.d. <https://www.mozilla.org/en-US/firefox/new/>. (Accessed 11 September 2022).
- [49] Safari, Apple (n.d.), <https://www.apple.com/safari/>. (Accessed 11 September 2022).
- [50] Download Microsoft Edge Web Browser, Microsoft (n.d.), <https://www.microsoft.com/en-us/edge>. (Accessed 11 September 2022).
- [51] Y. Liang, M. Zhang, W.N. Browne, Image segmentation: a survey of methods based on evolutionary computation, *Lecture Notes in Computer Science*, in: *Simulated Evolution and Learning*, SEAL, Springer Verlag, Dick, G., 2014, pp. 847–859, https://doi.org/10.1007/978-3-319-13563-2_71, 2014.
- [52] S. Uchida, Image processing and recognition for biological images, *Dev. Growth Differ.* 55 (2013) 523–549, <https://doi.org/10.1111/dgd.12054>.
- [53] J. Bahlmann, N. Madrahimov, F. Daniel, D. Theidel, D.E. DeTemple, M. Buettner, A. Bleich, A. Haverich, A. Heisterkamp, S. Kalies, Establishment of a guided, in vivo, multi-channel, abdominal, tissue imaging approach, *Sci. Rep.* 10 (2020) 1–10, <https://doi.org/10.1038/s41598-020-65950-w>, 2020) 1–9.
- [54] B.M. Tiemeijer, M.W.D. Sweep, J.J.F. Sleetboom, K.J. Steps, J.F. van Sprang, P. de Almeida, R. Hammink, P.H.J. Krouwel, A.L.P.M. Smits, J. Tel, Probing single-cell macrophage polarization and heterogeneity using thermo-reversible hydrogels in droplet-based microfluidics, *Front. Bioeng. Biotechnol.* 9 (2021) 953, <https://doi.org/10.3389/fbioe.2021.715408>, BIFTEX.
- [55] D.I. Andersson, H. Nicoloff, K. Hjort, Mechanisms and clinical relevance of bacterial heteroresistance, *Nat. Rev. Microbiol.* 17 (2019) 8, <https://doi.org/10.1038/s41579-019-0218-1>, 17 (2019) 479–496.
- [56] F. Buggenthin, C. Marr, M. Schwarzfischer, P.S. Hoppe, O. Hilsenbeck, T. Schroeder, F.J. Theis, An automatic method for robust and fast cell detection in bright field images from high-throughput microscopy, *BMC Bioinf.* 14 (2013) 1–12, <https://doi.org/10.1186/1471-2105-14-297>, FIGURES/4.
- [57] M. Schwarzfischer, C. Marr, J. Krumsiek, P. Hoppe, T. Schroeder, F. Theis, Efficient fluorescence image normalization for time lapse movies, 5–5, in: *Proc. Microscopic Image Analysis with Applications in Biology*, 2011. maaab-2011-heidelberg-papers.html. (Accessed 11 September 2022).
- [58] T.S. Gardner, C.R. Cantor, J.J. Collins, Construction of a genetic toggle switch in *Escherichia coli*, *Nature* 403 (2000) 6767, <https://doi.org/10.1038/35002131>, 403 (2000) 339–342.
- [59] J. Wiktor, C. Lesterlin, D.J. Sherratt, C. Dekker, CRISPR-mediated control of the bacterial initiation of replication, *Nucleic Acids Res.* 44 (2016) 3801, <https://doi.org/10.1093/NAR/GKW214>.
- [60] J.D. Wang, P.A. Levin, Metabolism, cell growth and the bacterial cell cycle, *Nat. Rev. Microbiol.* 7 (2009) 822, <https://doi.org/10.1038/NRMICRO2202>.
- [61] B. Das, R.K. Bhadra, (p)ppGpp metabolism and antimicrobial resistance in bacterial pathogens, *Front. Microbiol.* 11 (2020) 2415, <https://doi.org/10.3389/fmicb.2020.563944>, BIFTEX.
- [62] S. Suzuki, H. Yamaguchi, N. Nakajima, M. Kawachi, *Raphidocelis subcapitata* (=Pseudokirchneriella subcapitata) provides an insight into genome evolution and environmental adaptations in the Sphaeropleales, *Sci. Rep.* 8 (2018) 1–8, <https://doi.org/10.1038/s41598-018-26331-6>, 2018) 1–13.
- [63] J. Guo, Z. Ma, J. Peng, J. Mo, Q. Li, J. Guo, F. Yang, Transcriptomic analysis of *Raphidocelis subcapitata* exposed to erythromycin: the role of DNA replication in hormesis and growth inhibition, *J. Hazard Mater.* 402 (2021), 123512, <https://doi.org/10.1016/j.jhazmat.2020.123512>.
- [64] F. Lananan, A. Jusoh, N. Ali, S.S. Lam, A. Endut, Effect of Conway Medium and f/2 Medium on the growth of six genera of South China Sea marine microalgae, *Bioresour. Technol.* 141 (2013) 75–82, <https://doi.org/10.1016/j.biortech.2013.03.006>.
- [65] A.A. Horton, D.K.A. Barnes, Microplastic pollution in a rapidly changing world: implications for remote and vulnerable marine ecosystems, *Sci. Total Environ.* 738 (2020), 140349, <https://doi.org/10.1016/j.scitotenv.2020.140349>.
- [66] math, Mathematical functions — Python 3.10.0 documentation (n.d.), <https://docs.python.org/3/library/math.html>. (Accessed 3 November 2021).
- [67] statistics, Mathematical statistics functions — Python 3.10.0 documentation (n.d.), <https://docs.python.org/3/library/statistics.html>. (Accessed 3 November 2021).
- [68] re — Regular expression operations, Python 3.10.0 documentation (n.d.), <https://docs.python.org/3/library/re.html>. (Accessed 3 November 2021).
- [69] Matplotlib, Python plotting — Matplotlib 3.4.3 documentation (n.d.), <https://matplotlib.org/stable/index.html>. (Accessed 3 November 2021).

Curriculum vitae

Personal data

Name: Immanuel Sanka
Date of birth: August 3, 1992
Place of birth: Magelang, Indonesia
Citizenship: Indonesia

Contact data

E-mail: Immanuel.sanka@gmail.com

Education

2019–current Tallinn University of Technology – PhD
2016–2018 Uppsala University – M.Sc.
2010–2015 Universitas Gadjah Mada – S.Si.
2007–2010 Magelang State High School 3 – High School

Language competence

English Fluent
Bahasa Fluent

Professional employment

2019–now Co-founder of Synbio.id
2023–now iGEM consultant for iGEM Indonesian League
2018–2019 Chief Operating Officer (JALA Tech Pte. Ltd.)

Other Publications

1. Sanka I., et al. 2023. Synthetic biology in Indonesia: Potential and projection in a country with mega biodiversity. *Biotechnology Notes* (New Journal from Elsevier) (<https://doi.org/10.1016/j.biotno.2023.02.002>)
2. Azizi T., L. C. D. Araujo, Z. Catecioglu, A. J. Clancy, M. L. Feger, O. Liran, C. O'Byrne, I. Sanka, et al. A COST Action on microbial responses to low pH: Developing links and sharing resources across the academic-industrial divide. 2022. *New Biotechnology* (IF 6.49)
3. Sanka, I., et al. Carapace surface architecture facilitates camouflage of the decorator crab *Tiarinia cornigera*. *Acta biomaterialia*. (IF 10.63)
4. Sanka, I., et al. 2017. The effects of diatom pore-size on the structures and extensibilities of single mucilage molecules. *Carbohydrate research*. (IF 2.975)
5. Zusron, M., Sanka I., et al. 2015. Glass coating natural fibres by diatomisation: a bright future for biofouling technology. *Materials Today Communications*. (IF 3.383)
6. Nailil Husna, I. Sanka, et al. 2017. Prevalence and distribution of thalassemia trait screening. *Journal of Medical Science*. 49(3): 106–113. <http://dx.doi.org/10.19106/JMedSci004903201702>

Achievements

2023 COST Action CA18113 Virtual Mobility grant for EuroMicroPH dashboard improvement 2022.
2023 DORA Grant for research visit in diagnostics startup for 6 months (Clemedi AG – Switzerland and Germany).
2022 COST Action CA18113 Virtual Mobility grant for EuroMicroPH dashboard development

- 2020 Training school grant from COST Action CA15124 (France)
- 2019 Best startup in BEKUP, BEKRAF for Pre-Startup (Yogyakarta)
- 2017 iGEM - Best Manufacturing and Gold Medal in (Giant Jamboree in Boston, US)
- 2017 Ambassador of Uppsala University in Science and Technology for Indonesia

Conferences and courses

- 2023 Swedish Microfluidics in Life Science - Oral Presentation
- 2023 NEUBIAS Training School for Defragmentation in Cloud – Workshop and Oral Presentation
- 2022 iGEM Giant Jamboree – Judge for Software track
- 2022 MicroTAS Conference – Poster Presentation
- 2022 EMBL Microfluidics Conference – Poster Presentation
- 2021 NanoBioTech-Montreux Conference (2021) – Poster Presentation
- 2021 Next Generation Sequencing analysis, including de novo bacterial genome assembly, quality control, and annotation (2021) – Course
- 2021 scRNAseq data analysis – Course
- 2021 Swedish Microfluidics in Life Science – Oral Presentation
- 2021 MicroTAS Single-Cell Data Analytics – Poster Presentation
- 2020 NEUBIAS Training School for Bioimage Analysts (2020) – Workshop and Poster Presentation

Elulookirjeldus

Isikuandmed

Nimi: Immanuel Sanka
Sünniaeg: 3. August, 1992
Sünnikoht: Magelang, Indonesia
Kodakondsus: Indonesia

Kontaktandmed

E-post: Immanuel.sanka@gmail.com

Hariduskäik

2019– praegu Tallinna Tehnikaülikool – PhD
2016–2018 Uppsala University – MSC
2010–2015 Universitas Gadjah Mada – BSC
2007–2010 Magelang State High School 3 – Keskkharidus

Keelteoskus

Inglise keel Kõrgtase
Bahasa Kõrgtase

Teenistuskäik

2019– praegu kaasasutaja Synbio.id
2023– praegu iGEM konsultant iGEM Indonesian League
2018–2019 Tegevjuht (JALA Tech Pte. Ltd.)

Muud publikatsioonid

1. Sanka I., et al. 2023. Synthetic biology in Indonesia: Potential and projection in a country with mega biodiversity. *Biotechnology Notes* (New Journal from Elsevier) (<https://doi.org/10.1016/j.biotno.2023.02.002>)
2. Azizi T., L. C. D. Araujo, Z. Catecioglu, A. J. Clancy, M. L. Feger, O. Liran, C. O'Byrne, I. Sanka, et al. A COST Action on microbial responses to low pH: Developing links and sharing resources across the academic-industrial divide. 2022. *New Biotechnology* (IF 6.49)
3. Sanka, I., et al. Carapace surface architecture facilitates camouflage of the decorator crab *Tiarinia cornigera*. *Acta biomaterialia*. (IF 10.63)
4. Sanka, I., et al. 2017. The effects of diatom pore-size on the structures and extensibilities of single mucilage molecules. *Carbohydrate research*. (IF 2.975)
5. Zusron, M., Sanka I., et al. 2015. Glass coating natural fibres by diatomisation: a bright future for biofouling technology. *Materials Today Communications*. (IF 3.383)
6. Nailil Husna, I. Sanka, et al. 2017. Prevalence and distribution of thalassemia trait screening. *Journal of Medical Science*. 49(3): 106-113. <http://dx.doi.org/10.19106/JMedSci004903201702>

Saavutused

- 2023 COST Action CA18113 Virtuaalmobiilsuse grant “for EuroMicropH dashboard improvement 2022”.
- 2023 DORA semester välismaal grant viibimaks 6 kuud teadustööd tegeva startup-i juures (Clemedi AG – Switzerland and Germany).
- 2022 COST Action CA18113 Virtuaalmobiilsuse grant “for EuroMicropH dashboard development”

- 2020 Training school grant from COST Action CA15124 (France)
- 2019 Parim startup "BEKUP, BEKRAF for Pre-Startup" (Yogyakarta)
- 2017 iGEM - Best Manufacturing and Gold Medal in (Giant Jamboree in Boston, US)
- 2017 Uppsala Ülikooli tudengisaadik Indoneesias teaduse ja tehnoloogia valdkonnas

Konverentsid ja kursused

- 2023 Swedish Microfluidics in Life Science – suuline ettekanne
- 2023 NEUBIAS Training School for Defragmentation in Cloud – Töötuba ja suuline ettekanne
- 2022 iGEM Giant Jamboree – tarkvara kohtunik
- 2022 MicroTAS Conference – poster ettekanne
- 2022 EMBL Microfluidics Conference – poster ettekanne
- 2021 NanoBioTech-Montreux Conference (2021) – poster ettekanne
- 2021 Next Generation Sequencing analysis, including de novo bacterial genome assembly, quality control, and annotation (2021) – kursus
- 2021 scRNAseq data analysis – kursus
- 2021 Swedish Microfluidics in Life Science – suuline ettekanne
- 2021 MicroTAS Single-Cell Data Analytics – poster ettekanne
- 2020 NEUBIAS Training School for Bioimage Analysts (2020) – Töötuba ja suuline ettekanne

ISSN 2585-6901 (PDF)
ISBN 978-9916-80-050-8 (PDF)