THE UNIVERSITY
of EDINBURGH

This thesis has been submitted in fulfilment of the requirements for a postgraduate degree (e.g. PhD, MPhil, DClinPsychol) at the University of Edinburgh. Please note the following terms and conditions of use:

Optimisation of Microfluidic Experiments for Model Calibration of a Synthetic Promoter in S. cerevisiae

Zhaozheng Hou

Thesis submitted for the degree of
Doctor of Philosophy

School of Engineering
The University of Edinburgh

2021

Declaration

I hereby declare that:

1. The thesis has been composed by myself;
2. Except where states otherwise by reference or acknowledgment, the work presented is entirely my own;
3. The work has not been submitted for any other degree or professional qualification.

Zhaozheng Hou

April 2021

Abstract

This thesis explores, implements, and examines the methods to improve the efficiency of model calibration experiments for synthetic biological circuits in three aspects: experimental technique, optimal experimental design (OED), and automatic experiment abnormality screening (AEAS). Moreover, to obtain a specific benchmark that provides clear-cut evidence of the utility, an integrated synthetic orthogonal promoter in yeast (*S. cerevisiae*) and a corresponded model is selected as the experiment object.

This work first focuses on the "wet-lab" part of the experiment. It verifies the theoretical benefit of adopting microfluidic technique by carrying out a series of in-vivo experiments on a developed automatic microfluidic experimental platform. Statistical analysis shows that compared to the models calibrated with flow-cytometry data (a representative traditional experimental technique), the models based on microfluidic data of the same experiment time give significantly more accurate behaviour predictions of never-encountered stimuli patterns. In other words, compare to flow-cytometry experiments, microfluidics can obtain models of the required prediction accuracy within less experiment time.

The next aspect is to optimise the "dry-lab" part, i.e., the design of experiments and data processing. Previous works have proven that the informativeness of experiments can be improved by optimising the input design (OID). However, the amount of work and the time cost of the current OID approach rise dramatically with large and complex synthetic networks and mathematical models. To address this problem, this thesis introduces the parameter clustering analysis and visualisation (PCAV) to speed up the OID by narrowing down the parameters of interest. For the first time, this thesis proposes a parameter clustering algorithm based on the Fisher information matrix (FIMPC). Practices with in-silico experiments on the benchmarking promoter

show that PCAV reduces the complexity of OID and provides a new way to explore the connections between parameters. Moreover, the analysis shows that experiments with FIMPC-based OID lead to significantly more accurate parameter estimations than the current OID approach.

Automatic abnormality screening is the third aspect. For microfluidic experiments, the current identification of invalid microfluidic experiments is carried out by visual checks of the microscope images by experts after the experiments. To improve the automation level and robustness of this quality control process, this work develops an automatic experiment abnormality screening (AEAS) system supported by convolutional neural networks (CNNs). The system learns the features of six abnormal experiment conditions from images taken in actual microfluidic experiments and achieves identification within seconds in the application. The training and validation of six representative CNNs of different network depths and design strategies show that some shallow CNNs can already diagnose abnormal conditions with the desired accuracy. Moreover, to improve the training convergence of deep CNNs with small data sets, this thesis proposes a levelled-training method and improves the chance of convergence from 30% to 90%.

With a benchmark of a synthetic promoter model in yeast, this thesis optimises model calibration experiments in three aspects to achieve a more efficient procedure: experimental technique, optimal experimental design (OED), and automatic experiment abnormality screening (AEAS). In this study, the efficiency of model calibration experiments for the benchmarking model can be improved by: adopting microfluidics technology, applying CAVP parameter analysis and FIMPC-based OID, and setting up an AEAS system supported by CNN. These contributions have the potential to be exploited for designing more efficient in-vivo experiments for model calibration in similar studies.

Keywords: synthetic biology; model calibration; microfluidics; parameter clustering; convolutional neural network (CNN)

Lay Summary

This PhD thesis involves optimising the efficiency for a type of experiment in developing synthetic biological circuits. *Synthetic biology* is a field that aims to implement new functions in cells with increased reliability and efficiency by designing and integrating synthetic genetic circuits in these cells. However, the synthetic biology community still has a limited understanding of the highly entangled regulatory mechanisms in cells. As a result, the development of synthetic biological circuits is far from the "plug and play" paradigm but closer to a cycle of "design-build-test-design" (the DBT cycle). Nowadays, in most cases, it still takes years to turn a conceptual design into a commercial synthetic biological product. In practice, the "test" steps, i.e., the experiments on these synthetic prototypes to quantitively analyse their performances under the given stimuli, takes a considerable share of the development time. Quantitative analysis is usually achieved by first establishing one or more mathematical models for the interested synthetic parts, and then tuning the model parameters to minimise the difference between the model predictions and the observations in the experiments. This procedure is called model calibration.

This thesis explores several methods to improve the efficiency of these experiments, so that with the same or even less experiment duration and resources, the performances of cells could be quantitatively analysed with higher accuracy. The work is divided into three aspects: improving the experimental technique, optimising the stimuli design of these experiments (OED), and adopting an automatic experiment abnormality screening (AEAS). In this study, a model of a synthetic promoter (a type of DNA sequences that regulates the expression of proteins) in yeast (S. cerevisiae) is used as a benchmark.

This study statistically proves that the efficiency of model calibration experiments for the benchmarking model can be improved by:

1. Replacing the classic flow-cytometry experiments with microfluidics experiments (upgrading the experiment technique);
2. Applying clustering analysis of the model parameters to find the parameters having similar effects to the model when their values change and running optimal experimental design (OED) based on the clustering results. Parameters having similar effects to the observables are most likely to cause identifiability problems and lead to inaccurate calibrations, so clustering these parameters together can achieve a more efficient OED process by narrowing down the parameters to focus on;
3. Setting up an automatic experiment abnormality screening system supported by a convolutional neural network. This system can identify the abnormal experimental condition during the experiment to reduce the time wasted on invalid experiments and also free the experts from the drudgery of manual checking after experiments.

These contributions have the potential to be exploited for designing more efficient experiment platforms for model calibration in similar studies.

Acknowledgements

Firstly, I would like to thank my principal supervisor Dr. Filippo Menolascina for his patient and responsible guidance throughout this research. His rigorous and professional work style has set an excellent example for my research and work in the future. Also, gratitude to my assistant supervisor Dr. Gregory Francois, for his encouragement and advice full of wisdom and flexibility when I face difficulties during the study. Thank you to my annual review examiner and lab safety officer, Dr. Leonardo Rios Solis, for his kind support and inspirations of my research. Thanks to the management team of the IBioE lab at the University of Edinburgh, this project would not have been possible without their enthusiasm and participation: school biosafety officer Dr. Andy Downes, lab manager Prof. Alistair Elfick, support technician Katalin Kis, as well as former support technician Dr. Helen Cope.

I want to thank Dr. Lucia Bandiera for her great help throughout my studies, from carrying out the experiments to writing the journal paper. Also, Dr. Rose Rae, Dr. Alex McVey, and David Gomez Cabeza for their assistance and supports of the wet-lab operations; Dr. Ivan Clark, Ge Hao, and Juozas Pazera for their kind help on developing the automatic experimental platform; Dr. Varun Bhaskar Kothamachu for his cooperation in the in-silico experiments and thesis writing; Frederick Starkey for the kind help and responsible guidance in the image processing with convolutional neural networks. Many thanks to Dr. Trevor Ho and Dr. Jamie Gilman to share the knowledge and experiences of the experimental design, and provide constructive feedbacks for this thesis. And also, Dr. Anthony Callanan and Thomas Bate for their help in tutoring related courses. It has been a great honour and privilege to work with you all. Thank you to other academicians who kindly shared their knowledge during my thesis writing: Dr. Peng Xu, Dr. Ying Ma, and Bovie Xiong. I would also thank my viva examiners: Dr. Giovanni Stracquadanio and Prof. Jesús Picó. I have learned a lot from them during the viva and thesis modification.

Finally, I would like to express my deepest gratitude and best wishes to my family and my friends who provided supports and encouragements throughout my PhD.

Contents

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Background

1.1.1 Background of Synthetic Biology

Synthetic biology is a field of bioengineering that aims to implement new functions in cells with increased ease and efficiency by designing new genetic circuits and studying their behaviour in cells [1, 2]. In the 2012 Synthetic Biology Roadmap for the UK, the potential contributions of this field are summarised under three headings: well-being (diseases prediction/prevention, healthcare and employment), security (food, water and energy supply), and sustainability (natural resources management) [3, 4]. Although synthetic biology is a relatively recent discipline that inherits (but not merely a direct extension of) the genetic engineering technologies [5], it has developed as a key distinction for the Bioeconomy growth [6, 7]. Despite the severe impact of a global pandemic, the global investments of this field broke the historical records in 2020 [8-10] (Figure 1-1).
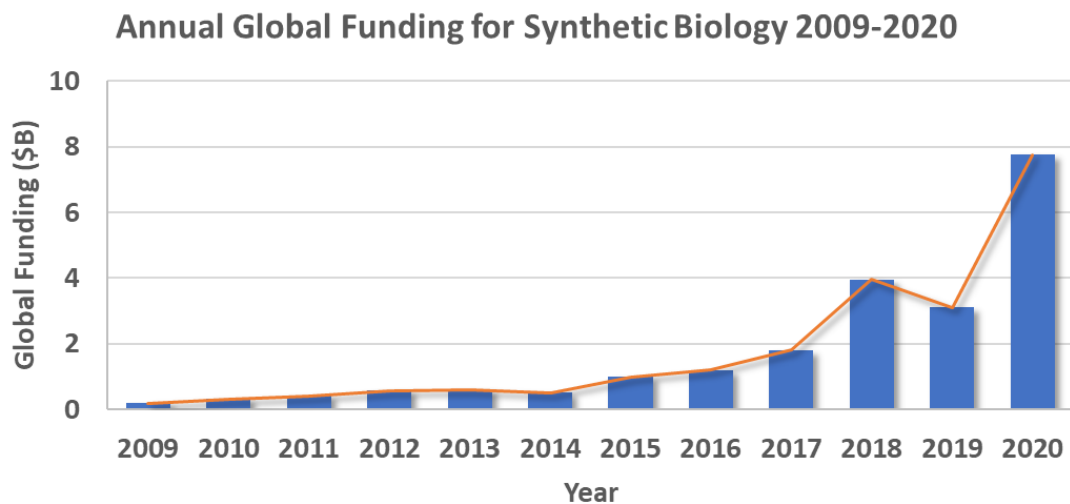
Figure 1-1. Global Funding for Synthetic Biology since 2009 [9].
This summary includes private, public, and non-dilutive government-grant funding.

Although the genetic-level manipulations of cells have presented before the conceptualisation of synthetic biology, many scholars hold the view that these designed biological systems are mostly *ad hoc,* limited in complexity, and are not always predictable when implemented in different biological contexts [11, 12]. To address this challenge, synthetic biology involves interdisciplinary projects that bring together the experts of different realms from cell biology to engineering and introduces a framework that is more "engineering-ready" [11, 13, 14] (Figure 1-2). Researchers implement engineering concepts such as modularisation and standardisation since the emergence of this field [15-17].
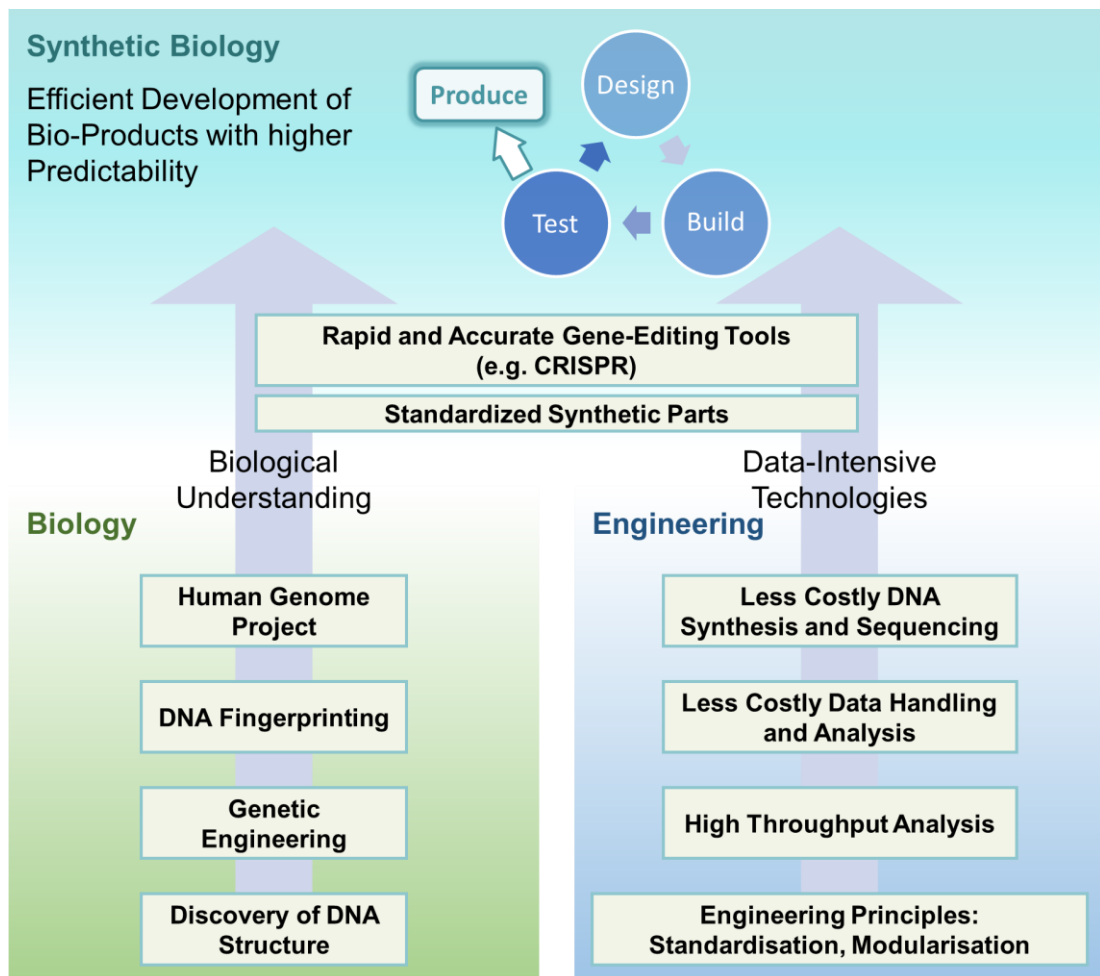
Figure 1-2. A confluence of advanced biological technologies and engineering principles leads to the emergence of synthetic biology [14].

A concrete practice of the engineering ideology in synthetic biology is the separation of the biological circuits design, genetic fabrication, and the experiments for testing these circuits [12, 15]. With the advent of precise gene-

editing techniques such as CRISPR-Cas9, the speed, costs and robustness of gene editing have been continuously improved [7, 18]. The sprung-up DNA foundries around the world provide genetic toolkits services with increased efficiency and affordability [3, 7, 19].

1.1.2 Design-Build-Test (DBT) Cycle and Model Calibration

With the guidance of engineering principles, synthetic biology projects involve a paradigm of directly assembling the functioning systems with standardised genetic parts. In practices, this construction procedure cannot be simplified as a "plug-and-play" pattern [5, 20]. In evolution, species change different features at the genetic level concurrently, which results in highly entangled and delicate gene regulation networks. Because of this, the complex crosstalk and resource sharing between synthetic and natural networks could lead to unwanted behaviours [13]. It is also described as interfacing engineering problems by Prof. Kitney and Prof. Freemont [5]. Therefore, the development of the prototypes and commercial products that work as expected requires decent knowledge not only of every synthetic part but also of the chassis organisms (the cells being modified) [12].

Prof. Clarke (Co-chair of the UK Synthetic Biology Leadership Council) and many scholars emphasises the necessity of applying a cycle pattern to overcome the interfacing problem during the development of synthetic biological products [5, 20]: as shown at the top of Figure 1-2, the pattern stands for the iterative procedures of design-build-test-(analysis/learn) (DBT/DBTA/DBTL [3, 14, 20-22]). The principle is to design and build chassis organisms containing synthetic parts, and then carry out experiments to test the performance of the product and to calibrate the corresponding mathematical models. The calibrated models would be used to refine the design to achieve better performance and reliability [13, 17].

On the one hand, about the building procedure, developments in DNA synthesis and sequencing in the last decade have dramatically improved its affordability and efficiency [23, 24]. On the other hand, about the testing procedure, although advances in measurement technologies (e.g. flow-cytometry [25] and microfluidics [26]) enabled high-throughput quantifications of fluorescent reporter proteins and reduced the reagent costs of experiments, the informativeness of wet-lab experiments is still in general limited by sparse (in time) and noisy observations and sub-optimal stimuli design because of the traditional experimental platforms [27, 28]. Till now, the test phase is still a costly and time-consuming step of the DBT cycle, which slows down the progress of product development [29, 30]. Moreover, with the increasing complexity of discovered and engineered genetic pathways, the demand for efficient testing experiments is more urgent than ever [22, 31].

1.1.3 Developments and Challenges of the Testing Stage

As introduced in the previous section 1.1.2, the testing stage is the stage that carries out series of experiments to obtain more knowledge of the biology projects (such as synthetic promoters, enzymes, or networks) and check whether the designed parts perform as expected. Figure 1-3 shows the four considerations of the techniques used at this stage, and also the main directions of improvements in recent years [32-34].

Efficiency is one of the most focused aspects of the techniques. A reason is that to further explore the potential of synthetic bioproducts, researchers are considering more complex models with more variables, which leads to an exponential increase in the requirement of experimental data [33, 35]. Benefiting from the developments in molecular engineering and automation systems, some high-throughput methods (such as flow-cytometry [25, 36] and microdroplet technology [37-39]) have become widely adopted and well-established experimental techniques. These methods allow quick observation

of cells in multiple channels [40, 41], so that researchers could gain information of the entire cell population within few seconds. Furthermore, automated microplate readers [42, 43] and microfluidic devices [44-46] allow parallel experiments, which also increases the efficiency of the testing stage. Moreover, many newly developed experimental devices enable measurements with higher sampling frequency and more complex, precise, and dynamic control of extracellular media [47-49]. Correspondingly, how to find the optimal experimental design in the enlarged feasible design space is also an interesting topic in this field [50-52].
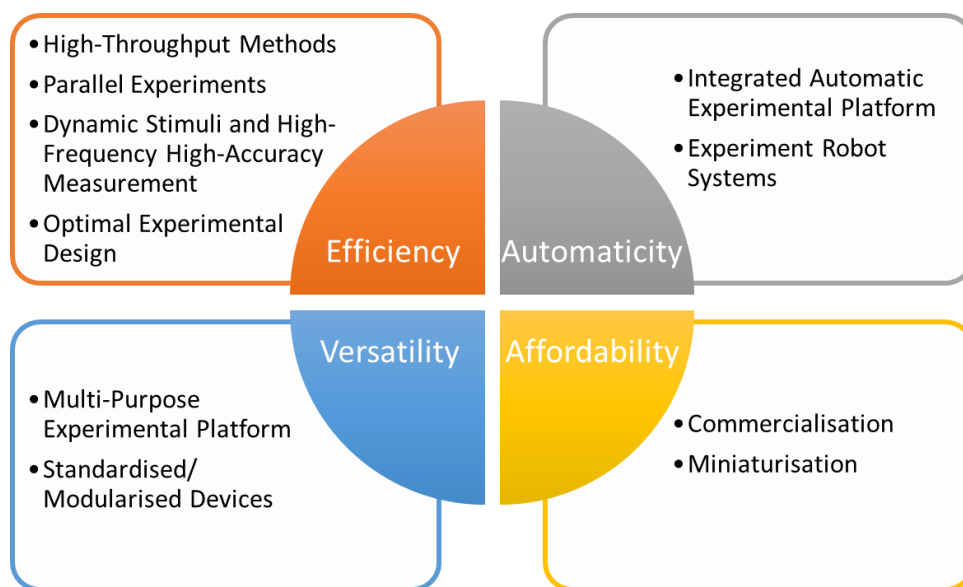
Figure 1-3. Main considerations and recent directions of improvements for the techniques used at the testing stage [32-34].

As shown in Figure 1-3, there are also other considerations except for the efficiency of the testing experiments. For example, the automation of experiments is a very popular direction in both the academic field and the related market [53-56]. The versatility is another practical aspect. For end-users, the design of experimental systems needs to balance the forward-looking and the compatibility with current equipment, and ideally also suitable for future projects [57-59]. The last aspect is affordability. As mentioned previously, the cost of testing has been generally decreased in the last decade [3, 7, 19]. However, the multi-purpose automatic experimental platforms are still generally considered too expensive to be widely available [32, 60].

Considering the majority of this aspect is beyond the scope of this thesis [61-64], it will not be expanded in detail here.

Overall, the recent developments in the techniques used in the testing stage have dramatically increased the experiment efficiency. These developments also increased the importance of optimal experimental design and experiment automation to a new level. However, to what extent would end users expect to benefit from adopting particular new experimental techniques (which could be costly for purchasing and training) is rarely investigated in detail [13, 25]. It is one of the concerns researchers have when deciding the experimental plan for the testing stage.

1.1.4 Model Calibration in Systems Biology

Although this thesis aims to optimise the efficiency and robustness of experiments for calibrating the models of synthetic networks, most of the conclusions can also be applied to the modelling of endogenous networks in cells without synthetic parts, which is a common procedure of researches in systems biology [65].

In a highly-cited viewpoint article [13], experts of synthetic biology, including Prof. Weiss and Prof. Elowitz, elaborated the complementarity and considerable overlap between this research and systems biology. Mathematical modelling is an effective tool in both fields, but the objectives are different. For synthetic biology, the models are calibrated for guiding the design of synthetic networks; for systems biology, the models are helpful to understand the biological mechanisms of the networks and make predictions of their behaviour [2, 13]. Because of this, successful attempts for optimising the model calibration experiments may provide references for both of the research fields. This view has been proven by many previous cases [66-70].

1.2 Previous Attempts to Improve the Efficiency of Test Phase

As introduced in the section above, the test phase aims to carry out experiments on the synthetic cells and obtain calibrated models to describe the behaviours of these organisms. Many researchers hold a view that the procedure can be split into the wet-lab part (experiments on cells) and dry-lab part (experimental design and processing of experimental data) [71-77]. Correspondingly, previous works on achieving a more efficient test phase can also be generally classified into two groups: improving the experimental techniques (wet-lab) [77], and improving the experimental design and model calibration (dry-lab) [72].

1.2.1 Attempts in Experimental Techniques

Experimental techniques decide the types of stimuli can be given to cells, and so as the type and accuracy of data that can be obtained from experiments. The two commonly adopted devices for observing synthetic cells' behaviours at the cellular level are flow cytometers and fluorescence microscopes [78].

In the last few decades, flow cytometry has been one of the most potent and broadly-adopted techniques to extract cells' information in biological and medical experiments [25, 36, 79]. In a flow-cytometer (Figure 1-4), cells are suspended in media and flow though the measuring position (commonly referred to as the interrogation point or laser intercept) one by one. laser light beam illuminates each cell and the light scatter and excited fluorescence emissions are quantitively measured by corresponded detectors. The signal intensity can be used to derive the features of cells, including the size, the complexity of inner structures, and the quantity of fluorescence as reporter of certain molecules [78]. With the developments in hardware and reagents, flow cytometry experienced an "explosion" in the number of colour channels [79].

In some recent experiments, flow cytometry can measure up to 28 fluorescent channels in one experiment and obtain single-cell-level data [40, 41].
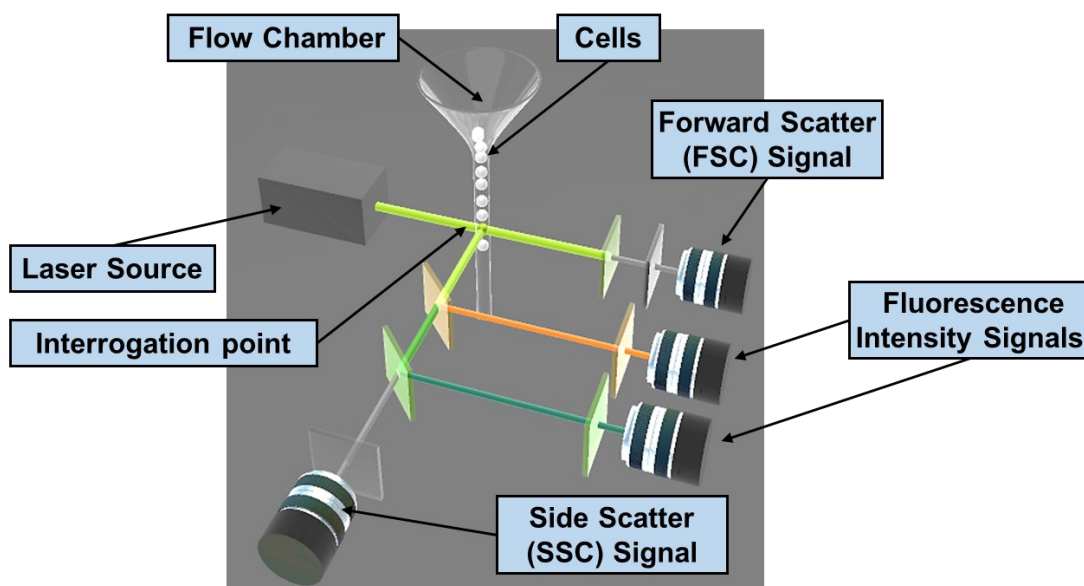
Figure 1-4. Illustration of the mechanism of a flow cytometer.

An illustration of fluorescence microscopy is shown in Figure 1-5. Unlike flow-cytometers that measure the signal intensities, the output of fluorescence microscopes are images, which leads to relatively lower throughput (the number of processed cells for each sample) [78]. However, since it does not need to resuspend the cells, microscopy has the potential of continuous observation of certain cells. Another emerging experimental technique known as microfluidics furtherly extends the advantage of microscopy. By manipulating media flows on microfluidic chips with sizes of a few cm^2, this technique allows continuous monitoring of the cells under more complex and dynamic stimuli [45, 80-82]. (Details of these experimental techniques would be discussed in details in chapter 2).

The different mechanisms of traditional flow-cytometry and microscopy with microfluidics affect the experiments in many aspects, including the accuracy of observation and the stress applied to cells during the cell growth and measurement [83]. Therefore, although microfluidics experiments have a theoretical advantage by allowing more complex and dynamic stimuli, it remains as a question that whether its advantages could significantly improve

the efficiency for model calibration in practices. In the last decade, there have been many previous studies for different purposes that involved experiments with both flow-cytometry and microscopy (without microfluidics) [25, 83-86]. These cases generally support the view that which of the experimental techniques is appropriate depends on the types of application, and the difference in calibrated models can be due to the choice of the technique. As emphasised by Prof. Barteneva and Prof. Smolke, comparing these techniques and the corresponding calibrated models is valuable to understand the result differences between experiment platforms [13, 25].
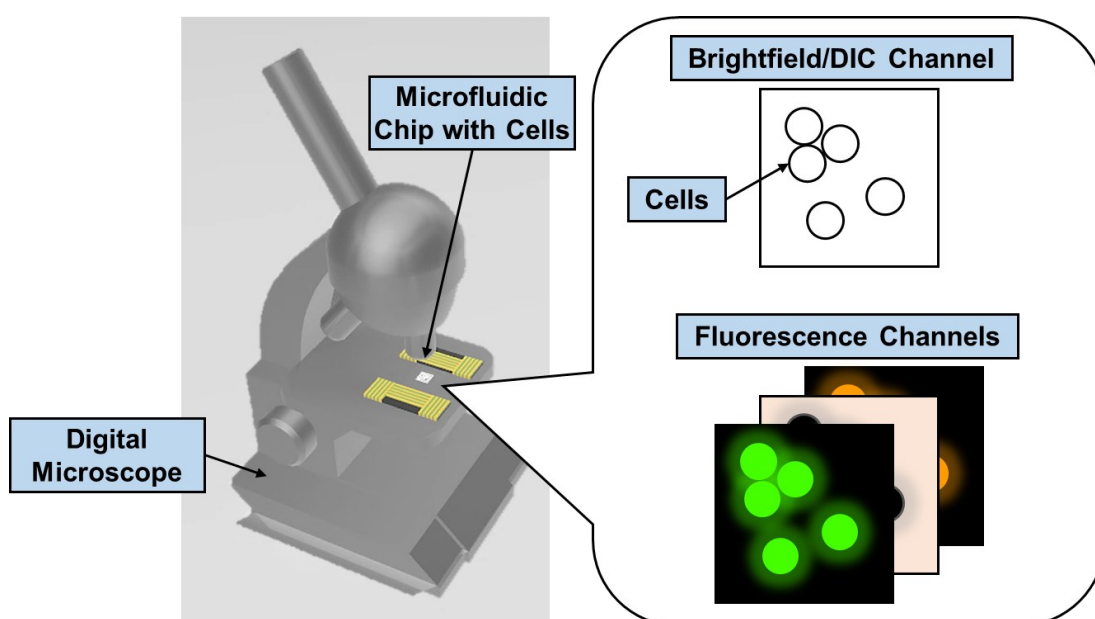
Figure 1-5. Illustration of the mechanism of fluorescence microscopy with microfluidics.

1.2.2 Attempts in Model Calibration and Experimental Design

Algorithms of model calibration and experimental design is another aspect that affects the efficiency of experiments. Model calibration refers to the process that looks for the model parameter set that minimises the difference between model predictions and experimental data [51, 52]. For non-linear models (a typical case in biological and chemical researches [51, 87, 88]), the calibration is generally considered as non-linear programming (NLP) problems [89, 90].

There are a few representative algorithms to solve these NLP problems, such as simulated annealing [91, 92], particle swarm [93, 94], evolutionary algorithms [95, 96] and scatter-search-based algorithms [89, 90, 97]. Improving the efficiency and robustness of solvers for these problems is a typical topic for experts in applied mathematics rather than bioengineers [98-100]. It is also important to point out that theoretically, no algorithm can guarantee to solve general NLP problems within a finite time [101]. Moreover, which of these algorithms can find the globally optimal solution with the fastest convergence vary with the NLP problems [52].

The optimisation of experimental designs (OED) is to increase the model accuracy and parameter identifiability without increasing the cost of experiments (e.g. experiment time and resources) [51, 52]. As shown in Table 1-1, the related algorithms kept developing with the growth in computing power and the complexity of the models to calibrate. In this table, the optimal experimental designs based on the Fisher information matrix (FIM) are proven to be suitable for guiding the experiment calibration of non-linear models [87, 102]. However, the cost of FIM-based OED in the computing power/time is considerable with large models with increasing numbers of parameters to estimate [89, 102, 103]. How to reduce the computational cost for the FIM-based OED while maintaining the estimation accuracies of these parameters is an interesting and valuable question that remains to investigate.

Table 1-1. Common algorithms to design the experiments for model calibration. (The algorithms are generally ordered from simple to complex of the calculation)

Algorithm Name	Related References
one factor at a time (OFAT)	Czitrom (1999) [104]
central composite design (CCD)	Agar et al. (2014) [105]
box-behnken design (BBD)	Ferreira et al. (2007) [106]
fractional factorial design	Jaynes et al. (2013) [107]
Taguchi's design of experiment	Agar et al. (2014) [105]
Fisher information matrix-based OED	Rodriguez et al. (2006) [51]
Bayesian OED	Huan et al. (2013) [108]

1.3 AIM and Approaches

1.3.1 Overall Aim and Chapter Division

As introduced in previous sections, the costly and time-consuming experiments at the testing phase have become a bottleneck that slows down the DBT cycle of synthetic biology production [29, 30]. To address this problem, this thesis explores the approaches to improve the efficiency of experiments for model calibration and examine these solutions with a developed automatic experimental platform and a model of a synthetic inducible promoter in yeast. Moreover, the calibration of a mathematical model of a synthetic inducible promoter in S. cerevisiae (yeast) is carried out as the benchmarking. The work can be divided into three aspects, and chapter 2-4 provide the corresponded details: the experimental technique, the optimisation of experimental design, and automatic experiment abnormality screening.

1.3.2 Microfluidic Experiments on Developed Automatic Platform

The work on the experimental technique aspect is validating the benefit of microfluidic experiments compared to the classic flow-cytometry experiments for model calibration. As introduced in section 1.2, although microfluidic devices allow more dynamic stimulus, to what extent this can benefit the calibration accuracy has not been thoroughly investigated. In chapter 2, a series of in-vivo experiments on the yeast cells with an embedded inducible promoter is carried out on a developed microfluidic experimental platform. The experimental data and data from flow-cytometry experiments in a previous research project by Gnügge et al. [109] is used for model calibration and cross-validation. The statistical analysis of the comparison will show whether microfluidic experiments lead to more accurate calibrated models with the same experimental time.

1.3.3 Optimal Experimental Design with Parameter Clustering

For the aspect of optimal experimental design (OED), chapter 3 explores an approach that carries out the OED based on parameter clustering analysis and visualisation (PCAV) to achieve more efficient experimental designs. These methods cut the experiment into a series of sub-experiments. Only a subset of parameters is focused in each OED, so the computing complexity of the OED for each sub-experiment is reduced.

Moreover, this thesis proposes a parameter clustering algorithm based on the Fisher information matrix (FIMPC) for the first time. The accuracy of model calibration base on the experiments optimised with parameter clustering and the previous OED approaches and compared and analysed.

1.3.4 Automatic Experiment Abnormality Screening System

The work in chapter 4 is about developing an automatic experiment abnormality screening (AEAS) system supported by convolutional neural networks (CNNs). It does not come from the previous division of wet-lab and dry-lab, but from the practice of wet-lab experiments of the previous aspects. The works in chapter 2 came across a few different abnormal experimental conditions, which causes invalid experimental data and slows down the research progress. In the past, the checking of these conditions is carried out with a manual screening of all the experimental images. However, it would not be practical for the potential mass parallel microfluidic experiments.

An AEAS system with convolutional neural networks (CNNs) is developed and validated. Moreover, some image augmentation methods and training strategies are researched. The experiences can be a reference for the development of image-based AEAS systems for other experimental platforms and engineering practices.

Chapter 2: Improving the Accuracy and Automation of Model Calibration Experiments with Microfluidic Technique

Microfluidics (MF) as an emerging experimental technique allows higher sampling frequencies and more dynamic stimuli compared to flow-cytometry (FC), a classic and broadly adopted experimental method. On the other hand, FC as a mature experiment technique can observe 10+ times more cells at each sampling point with more robust quantitation. Although a quantitative comparison between these two techniques is very helpful for understanding the differences in results and also for guiding the choice of technique for future studies, this point has not been investigated in many studies. In this chapter, a series of MF experiments are carried out on a developed automated experimental platform. By calibrating the model with both MF experimental data and FC data, the work in this chapter quantitively validates the advantage of MF experiments for model calibration with an orthogonal promoter model as a benchmark. Results suggest that compared to traditional flow-cytometry experiments, microfluidic experiments lead to models that give better predictions in stimulus different from the experimental designs.

2.1 Introduction

2.1.1 Flow Cytometry and Fluorescence Microscopy in Biology

In the last 30 years, flow cytometry has been one of the most powerful and broadly adopted technologies to extract cells' information in biological and medical experiments [25, 36, 79]. It is an experimental technique that allows rapid and simultaneous acquisition of multiple fluorescence channel over thousands of cells. As shown in Figure 1-1, suspended sample cells flow

through the device one by one and irradiated with a beam of light (for most cases, laser). The obtained light-scattering features contain information of size, granularity, and very importantly, the concentration of fluorescent-labelled molecules [110]. With the developments in hardware, reagents, and data processes theories, flow cytometry experienced an "explosion" in the number of colour channels [79]. In some recent experiments, flow cytometry can measure up to 28 colours [40, 41, 111, 112].
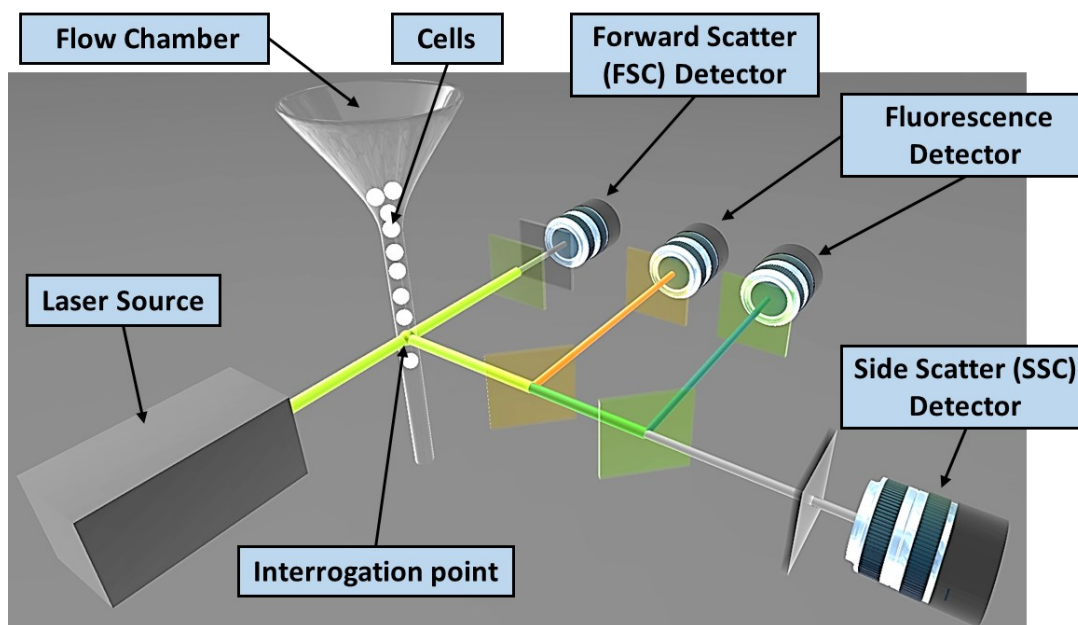
Figure 2-1. Illustration of a traditional flow cytometer.
The intensity of forward scatter (FSC) light is proportional to cells' size, side scatters (SSC) is proportional to cell's internal structural complexity, photomultiplier tubes (PMTs) measure the intensity of fluorescence lights of different channels.

In some more recent studies, experimental technologies based on microscope images instead of intensity data of light beams, such as high-throughput microscopy [113, 114] and imaging flow cytometry [25], also achieve the single-cell-level data acquisition. Depending on the hardware settings, experimental data is based on images of ten to thousands of cells. One of these methods' advantages is that they allow the time-lapse analysis of individual cells(observation with cell tracking) as they remain in multi-well plates or on slides during the experiments [113].

2.1.2 Microfluidic Experimental Technique

Microfluidics is an emerging experimental technique that shifted from phenomenological studies to medical and biotechnological applications in recent years [45]. In the early stages of biopharmaceutical research, traditional experimental equipment such as shake flasks and bench-scale bioreactors are commonly adopted [115, 116]. These approaches lead to laborious and expensive experiments and cannot support accurate control of complex dynamic stimuli [80-82]. In contrast, typical microfluidic devices are of a few cm^3 in size, which processes or manipulates cell media in network channels with microliters or femtoliters diameters in volume (Figure 2-2) [117, 118]. This technology not only saves the resources for experiments, but also allows more highly complex and rapid control of the cellular microenvironment [119, 120].

For decades, many previous researchers developed microfluidic experiment method and demonstrated its potential [45, 119, 121-124], such as the early attempts of miniaturising the bioreactors [62, 116] and microfluidic input multiplexor systems [125]. Nowadays, microfluidic devices can guide individual cells into desired locations, and then apply accurate and dynamic stimulus while continuously monitoring the cells' behaviours [126, 127].
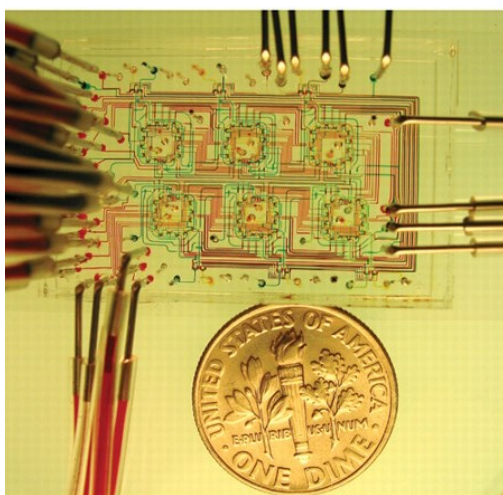
Figure 2-2. A microfluidic chemostat for experiments [128].
Cells and the media are loaded from the ports which connect to the media sources via tubing.

In 2011, Ferry et al. introduced a microfluidic chip design that allows 24-72-hour-long experiments on yeast cells [124]. This chip is also the design used in the work of this thesis. The chip (Figure 2-3), which is called MFD005a in the original paper, is designed to grow yeast cells in a monolayer in the chamber (the grey area). The extracellular environment is controlled by the mixing ratio between two media resources. Ferry et al. improved the mixing junction and proposed the dial-a-wave (DAW) junction (the green cashed circle in Figure 2-3). Compared to the traditional design (called T-junction [121, 124]), the DAW junction has two extra channels to the waste port, allowing a more reliable mixture for mixing ratios close to 0% or 100%.More details about how this chip design works are given in section 0.

Furthermore, Ferry et al. have also shown a chip design that is theoretically capable of carrying out parallel experiments (Figure 2-4). However, it is challenging to balance the pressures in the channels for these parallel experiments. Moreover, as mentioned in their paper, lens movement between chambers during the experiment may generate bubbles in the microscopy oil, resulting in focusing problems and blurry images [124]. This problem will also be discussed in chapter 4 about the experiment abnormality screening system. In other words, microfluidic technology has the potential to achieve massive parallel experiments. However, there are some issues to solve in quality control and other aspects to fully develop this potential.
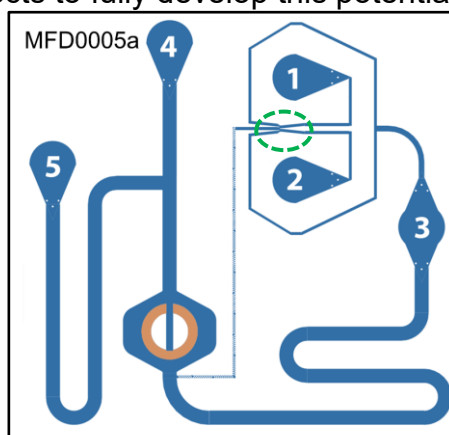
Figure 2-3. Microfluidic chip MFD005a's architecture [124].
Blue lines are the channels of the chip; the expanded nodes with numbers are the ports connecting to the media supply; the orange area is the cell chamber; green dashed region is the DAW junction.
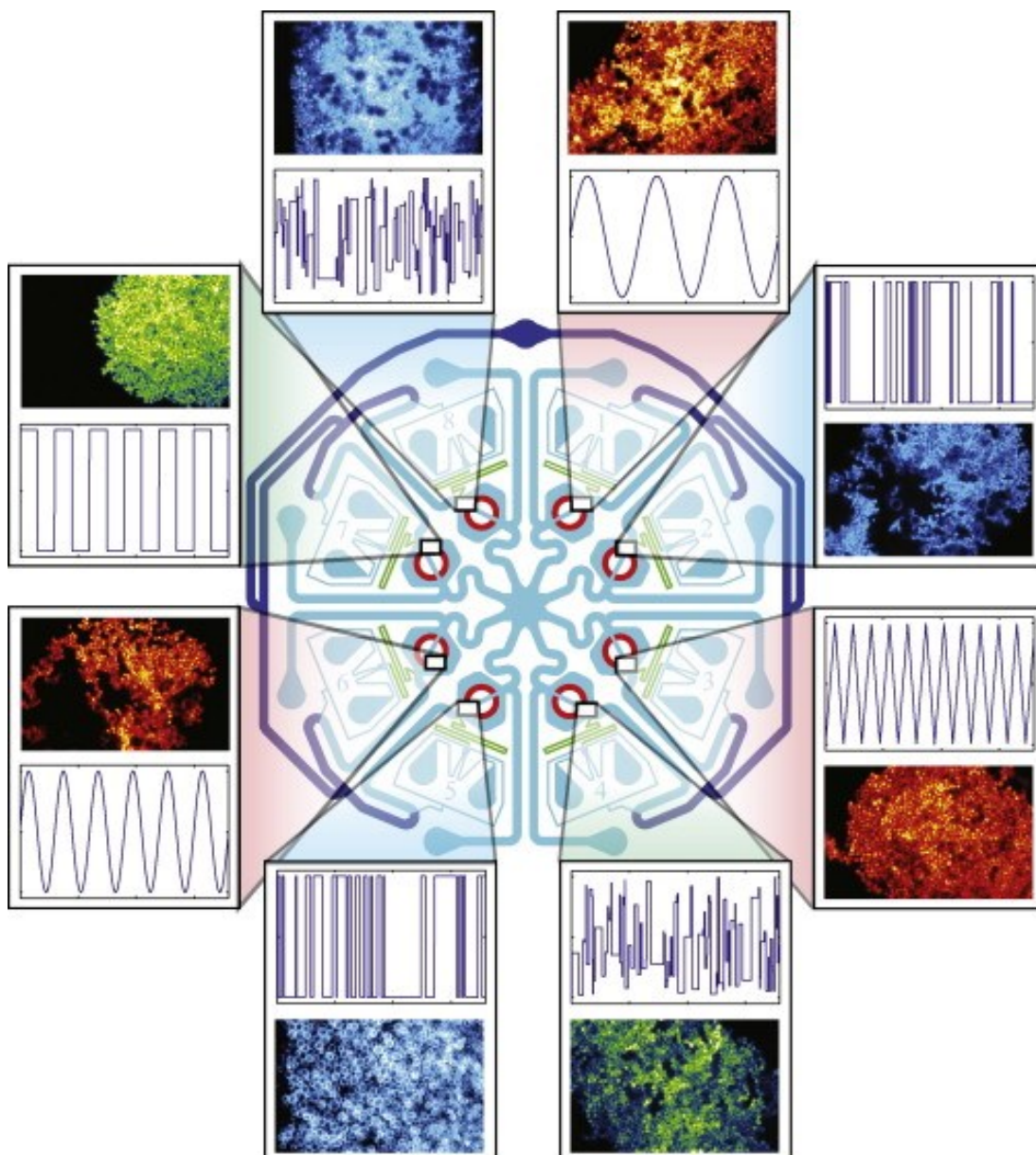
Figure 2-4. Microfluidic chip for eight parallel experiments [124].
The light blue parts are similar to the design in Figure 2-3, and the
corresponded port 4 and 3 for each experiment are connected to
reduce the number of required media sources.

Although microfluidics is an emerging technique that has its unique advantage, there are also aspects that can be improved, such as image-based cell segmentation [129]. Cell segmentation is one of the key steps that decide the accuracy of the quantification. But different from flow-cytometry which has well-proven and adopted method for similar requirement [36, 41, 78], the generic algorithms or tools for image-based cell segmentation is still a

developing project, and most of the current tools (such as WEKA [130], and CDeep3M [131]) work with hand-tailored features for each study case [129-134]. As shown in

Table 2-1, there is no direct answer to the question "Flow-cytometry and microfluidic microscopy, which of them works better for model calibration", which is important for the researchers considering upgrading experiment systems or building an experiment platform from scratch.

Table 2-1. Comparison of the advantages and disadvantages of the two experimental techniques to investigate.

Aspects	Flow-Cytometry	Microscopy+ Microfluidics
Throughput	**Higher**	Lower
Channels/ Parameters	**More**	Less
Quantitation	**Easier**	More Difficult
Sampling Frequency	Lower	**Higher**
Cell Tracking	Not Available	**Possible**
Stimuli	Simpler	**More Complex**

2.1.3 Attempts of Automated Experiments

Apart from the data acquisition and bioreactor designs, another critical aspect of bioengineering experiments' efficiency is the level of automation [135]. Recent attempts in related fields also reflect the demand for automated and integrated experimental technology in both industry and academic field [53, 135]. Moreover, the epidemic of COVID-19 also highlights the benefit of automated experiments under social distancing constraints and lockdowns.

The general vision of an automated experiment platform is a system that minimises human labours and only involves people at the beginning and end of experiments. However, as the famous saying, "Rome was not built in one day," the automated platform is achieved by automating the experiment procedures parts by parts. Although almost all the current automated platforms are closely integrated with different experimental hardware such as flow cytometer [135-137], microfluidic devices [116, 138, 139], or other mini-bioreactor systems like Chi.Bio [140-142], there is a common pattern that can be found within these systems as shown in Figure 2-5. This pattern also guided the development of the platform used in this thesis (section 2.2.4). Table 2-2 compares the automation level of some recent studies in related fields and this work. It can be seen that more and more experimental procedures are automated with the development of tools for automatic control. Although previous studies have shown that the potential of automated microfluidic experiments [118, 124], the automation level of microfluidic experiments in practice is generally lower than flow-cytometry experiments. It is likely because flow-cytometry presents for a longer time and is more familied by researchers. Compared to related studies in Table 2-2, the work in this thesis rises the automation level of microfluidic experiments.
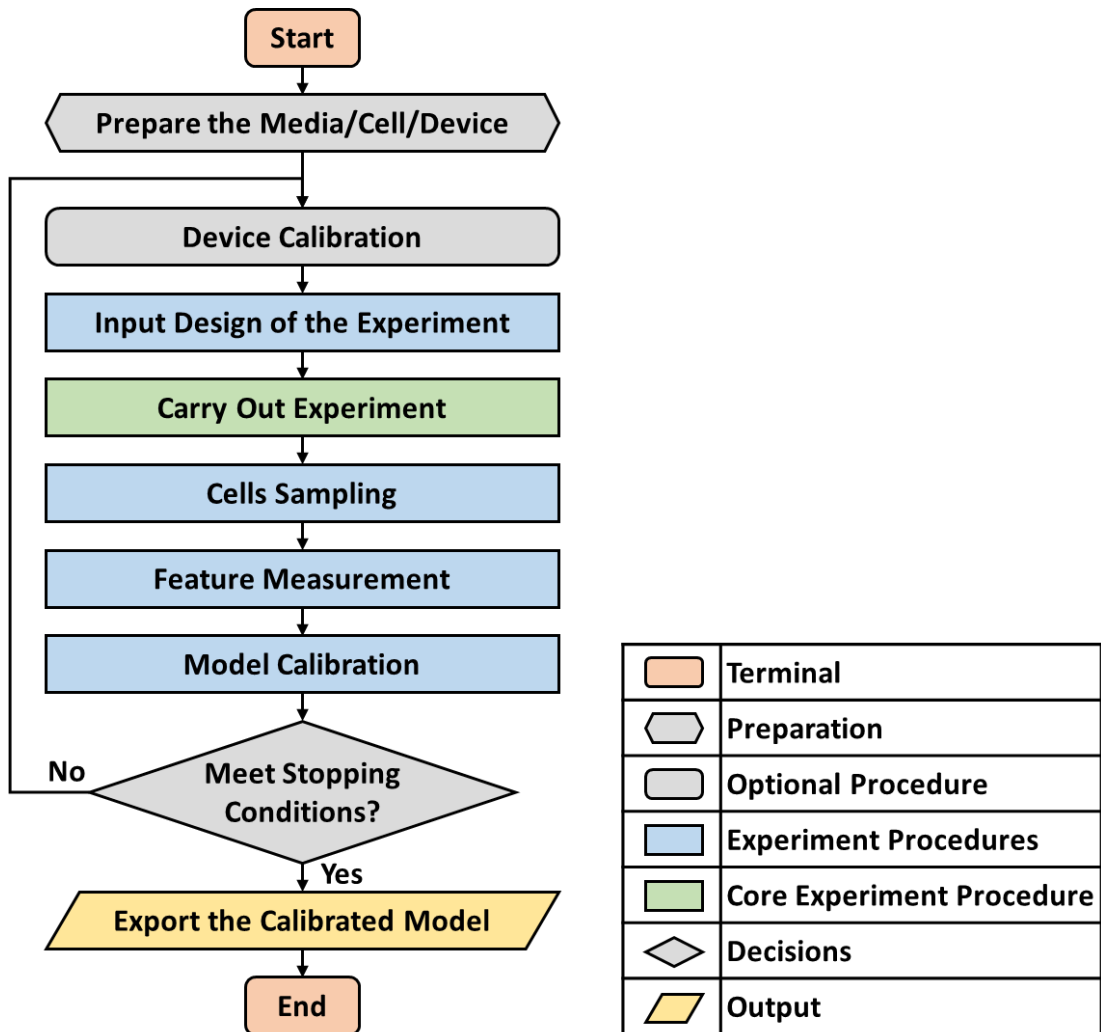
Figure 2-5. The common pattern of model calibration experiments.

Table 2-2. Automation level of some recent highly-cited model calibration studies in related fields and the work in this thesis.

Researcher (Year) [ref]	Experiment Object	Main Experiment Device	Automated Procedures				
			Device Calibration*	Input Design for Model Calibration	Sampling	Measuring	Model Calibration
Takahashi et al. (2012) [143]	mouse cells	plate-reader	○	○	○	●	○
Liu et al. (2012) [144]	Zebrafish Embryo	fluorescence imaging	○	○	○	●	●
Huang et al. (2015) [145]	*S. cerevisiae*	microfluidics	○	○	○	○	○
Cornaglia et al. (2016) [146]	*C. elegans*	microfluidics	○	○	●	●	●
Borujeni et al. (2016) [147]	*E. coli*	flow-cytometry	○	○	○	●	○
Zhang et al. (2017) [148]	Human cells	microfluidics	●	○	N/A	●	●
Nandania et al. (2018) [149]	*E. coli*	flow-cytometry	●	○	○	●	●
Harrigan et al. (2018) [136]	*S. cerevisiae*	flow-cytometry	○	●	●	●	●
Beal et al. (2019) [150]	multi-purpose	flow-cytometry	●	○	●	●	●
Fedorec et al. (2020) [151]	*E. coli*	flow-cytometry	○	○	●	●	○
Bertaux et al. (2020) [135]	*S. cerevisiae*	flow-cytometry	○	●	●	●	●
Burger et al. (2020) [53]	Chemicals	gas chromatography	●	●	●	●	●
Soffer et al. (2021) [152]	*E. coli*	microfluidics	○	○	N/A	●	○
This thesis	*S. cerevisiae*	microfluidics	●	●	N/A	●	●

The studies are sorted from old to new. Green and Blue correspond to the two experimental techniques to compare in this thesis. Solid/hollow dots are automated/manual procedures, and N/A means this study does not need this procedure. The device calibration procedure in this table only refers to some related calibration work of the main device, not all the calibrations of the device.

2.1.4 Experimental Subject: orthogonal induced promoter

The comparison between flow cytometry and microfluidic experiment requires comparable data that corresponds to both of the experimental techniques. Selecting a benchmarking subject that has been investigated with one of these techniques would save almost half of the time for in-vivo experiments. An ideal selection should also try to control the uncertainties in related aspects which may interfere or even bias the comparison results. For example, if the model structure for the subject is inaccurate, or the model parameters depend on other factors which are not included in the model, the credibility of the comparison of model calibrations would be undermined.

For the considerations above, a synthetic orthogonal promoter (Figure 2-6) in a yeast (*Saccharomyces cerevisiae*) strain is selected as the benchmark to address the benefit of microfluidic experiments for model calibration. This yeast strain is constructed by Gnügge et al. which is labelled as LacI-NLS (yRG500) in their original paper [109] and developed based on the strain BY4741. Gnügge and colleagues engineered an endogenous promoter sequence with heterologous DNA from *Escherichia coli* and established a network that regulates the expression of Citrine reporter with the concentration of isopropyl-β-D-1-thiogalactopyranoside (IPTG). The regulation path and mechanisms with the Lac operator in *E. coli* has been well studied since the 1990s [109, 153, 154]. Moreover, this promoter established an orthogonal regulation of Citrine expression, compared to networks synthesised with homologous DNA sequences, this network avoids the unwanted potential regulation from and onto other genes [109, 155, 156].
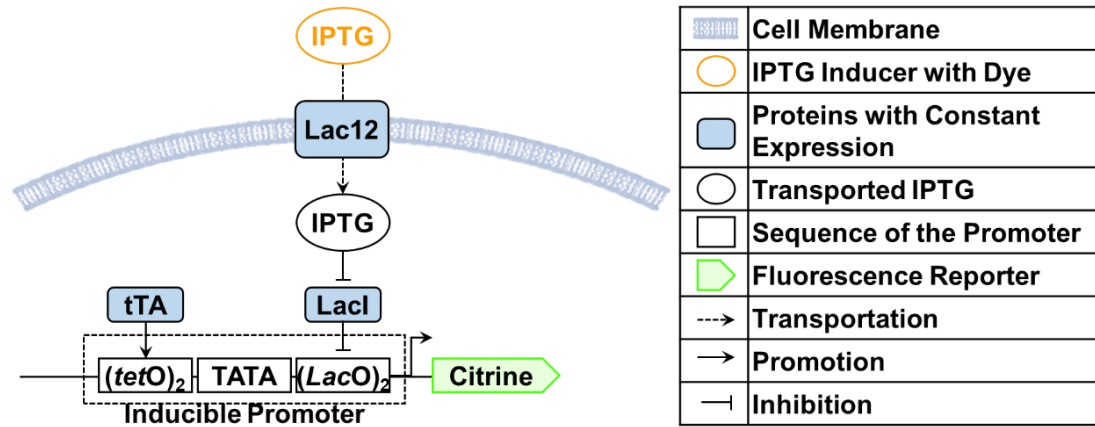
Figure 2-6. The orthogonal promoter designed by Gnügge et al. [109].

As shown in Figure 2-6, the promoter is constructed by placing operators (*tetO*)$_2$ and (*lacO*)$_2$ up-and-down-stream of the TATA box of homologous promoter CYC1 [157]. Proteins tTA, LacI, and Lac12 are expressed at a constant level with ACT1 promoter in yeast, which is not shown in the figure. These DNA sequences are integrated into the yeast genome. The corresponding mathematical model used in this study is introduced in section 2.2.1.

2.1.5 Novelty of this Study

For the first time, this work carries out in-vivo experiments to compare the model calibration accuracy from flow-cytometry and microfluidic experiments with an orthogonal inducible promoter. Traditional flow-cytometry is a well-proven and effective experiment technique for calibrating biological models. In comparison, Microfluidics is a relatively new and emerging technology that has the potential of less resource usage, dynamic stimuli, and continuous observation. This study performs microfluidic experiments to calibrate an orthogonal promoter model and verify if the experimental platform can lead to comparable model prediction accuracies as the ones calibrated with flow-cytometry experiments. Barteneva et al. commented that this type of

verification work helps overcome ambiguities in model calibration associated with different experimental techniques [25].

This work raises the microfluidic experiments' automation level by developing a control platform that integrates most of the experiment procedures, including the optimal input design and model calibration procedures. Moreover, the platform also supports remote control of the experimental device.

2.2 Methods

2.2.1 Mathematical Model to Calibrate

As introduced in section 2.1.4, the experiments aim to calibrate a model describing the behaviour of an orthogonal inducible promoter in yeast cells (Figure 2-6). The model selected in this work is an ordinary differential equation (ODE) model from previous work [27]. As shown in Eq. 2-1, the model has three states which correspond to the concentrations of three different molecules: Citrine mRNA, immature folded Citrine protein, and matured Citrine protein which is fluorescent. The $[IPTG]$ in brown is the input; the $[Cit_{fluo}]$ in green is the observable; the characters in red are model parameters.

$$\frac{d}{dt}[Cit_{mRNA}] = \alpha_1 + Vm_1 \frac{[IPTG]^{h_1}}{[IPTG]^{h_1} + (Km_1)^{h_1}} - d_1[Cit_{mRNA}]$$

$$\frac{d}{dt}[Cit_{foldedP}] = \alpha_2[Cit_{mRNA}] - (d_2 + K_f)[Cit_{foldedP}] \qquad \text{Eq. 2-1}$$

$$\frac{d}{dt}[Cit_{fluo}] = K_f[Cit_{foldedP}] - d_2[Cit_{fluo}]$$

In Eq. 2-1, the induction mechanism is modelled with a Hill function. α_1 is the basal transcription factor; Vm_1 is the maximal induced transcriptional rate; h_1 is the Hill coefficient; Km_1 is the Michaelis Menten coefficient (the concentration corresponds to half of the maximal induction level); α_2 is the translation rate; K_f is the maturation rate; d_1 and d_2 are the degradation rate

for mRNA and protein. Figure 2-7 illustrates the reactions involved in this model. The parameter units, calibrated values based on Gnügge et al.'s experiments and the feasible ranges for calibrations are given in Table 2-3.
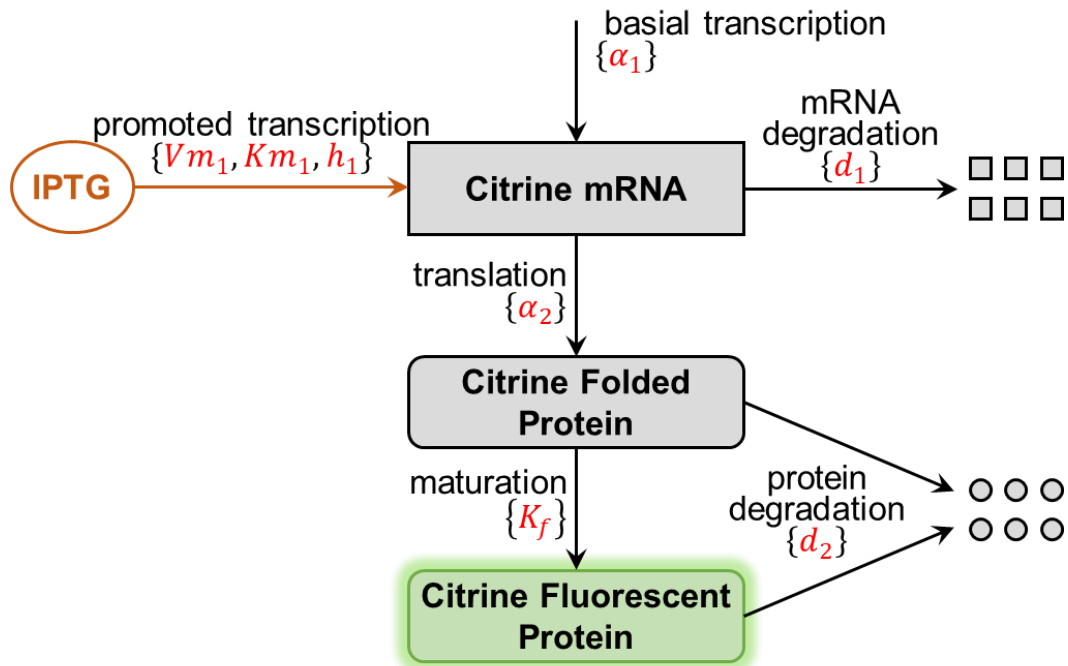
Figure 2-7. Illustration of the reactions involved in the model, and how the input IPTG affects the observed fluorescence signal.

Table 2-3. Feasible ranges and best fits (based on flow-cytometry experiments) of model parameters.

Symbol	Physical Meanings (unit)	Best Fit [27, 109]	Feasible Range	Ref.
α_1	basal transcription factor (A.U./min)	0.0164	3.88e-5 to 0.495	[158]
α_2	translation rate of Citrine mRNA (min^{-1})	6.64	0.243 to 6.81	[27]
Vm_1	maximal induced transcriptional rate (A.U./min)	0.292	0.0388 to 0.495	[159]
Km_1	Michaelis-Menten coefficient of inducible promoter (μM)	5.14	2.00 to 10.0	[109]
h_1	Hill coefficient of inducible promoter (1)	1.72	0.500 to 4.90	[27]
d_1	mRNA degradation rate (min^{-1})	0.220	7.70e-3 to 0.230	[160]
d_2	Citrine protein degradation rate (min^{-1})	5.75e-3	5.98e-5 to 0.245	[161]
K_f	maturation rate of Citrine (min^{-1})	0.0216	0.0120 to 0.0217	[162]

In practice, the observation data is derived based on the signal intensity in the corresponding fluorescence channel. With different experimental equipment, the same amount of Citrine proteins may lead to different signal intensities. In other words, the microfluidic experiments carried out in this study and the flow-cytometry experiments would have different observation units. Therefore, the observed signal intensity can be expressed as Eq. 2-2:

$$y_{FC} = scal_{FC} * [Cit_{fluo}]$$
$$y_{MF} = scal_{MF} * [Cit_{fluo}]$$

Eq. 2-2

where y_{FC} and y_{MF} are the intensities of observed signals in flow-cytometry and microfluidic experiments, $scal_{FC}$ and $scal_{MF}$ are the corresponding scaling factors.

Including the scaling factors, there are nine parameters in the model. However, not all these parameters can be calibrated at the same time because some of the parameter groups are structurally unidentifiable. In other words, the model predicted signal intensity y would be identical with some different value sets of these parameters. Figure 2-8 gives details of the two transformations that would generate exactly the same model predictions. It means that in the calibration, two parameters from two out of three groups must be fixed: $\{\alpha_1, Vm_1\}$, $\{\alpha_2\}$, $\{scal\}$.

$$
\begin{array}{ccc}
\begin{matrix} \alpha_1 \\ Vm_1 \\ \alpha_2 \\ [Cit_{mRNA}] \end{matrix}
\Rightarrow
\begin{matrix} \alpha_1 * C \\ Vm_1 * C \\ \alpha_2 / C \\ [Cit_{mRNA}] * C \end{matrix}
\qquad
\begin{matrix} \alpha_2 \\ [Cit_{foldedP}] \\ [Cit_{fluo}] \\ scal \end{matrix}
\Rightarrow
\begin{matrix} \alpha_2 / C \\ [Cit_{foldedP}] / C \\ [Cit_{fluo}] / C \\ scal * C \end{matrix}
\end{array}
$$

Figure 2-8. The two transformations that remain the model-predicted signal intensity y unchanged.

It can be seen that if the concentration of mRNA ($[Cit_{mRNA}]$) can be measured with a fixed scaling factor [163, 164], the identifiability problem could be solved. At the moment, to address this problem, the scaling factors and α_2 are fixed during the calibration. The scaling factors are fixed as 1, i.e., $y = [Cit_{fluo}]$, and $[Cit_{fluo}]$ shall be directly noted as the measured output value in different

arbitrary units for flow-cytometry and microfluidic experiments. The α_2 is fixed to a value derived from a model by Gnügge et al. where this value is identifiable [109]. The reason why α_2 is fixed rather than α_1 or Vm_1 is because α_2 describes the translation of the mRNA of Citrine, and this procedure is unlikely to be significantly affected by the IPTG input and gene modification of the promoter (which is the main difference in this network compared to previous related studies).

The model-predicted dose-response curve (Figure 2-9) and dynamic observations (Figure 2-10) are plotted as to provide a more intuitive understanding of how these parameters affect the model predictions. The dynamic simulation is when cells start from the steady-state, which corresponds to $0\ \mu M$ IPTG and then switch to an IPTG concentration randomly chosen between $0.1 - 1000\ \mu M$ in the log-scale every 3 hours. In the figures, θ^* is the best-fitted parameter value according to Gnügge et al.'s experiments [109]. The parameter values are increased/decreased by 10% and doubled/halved to see the difference in observation.

For the dose-response curves, it can be seen that h_1 and Km_1 affect the shape and horizontal shifts, and the other parameters contribute to the magnitudes of the observation. Dynamic curves show that parameters giving similar effects on the dose-response curve may have different effects on cells' dynamic behaviour (such as d_1 and d_2). The patterns of how parameters contribute to model predictions will also be reflected in the parameter clustering in chapter 3.
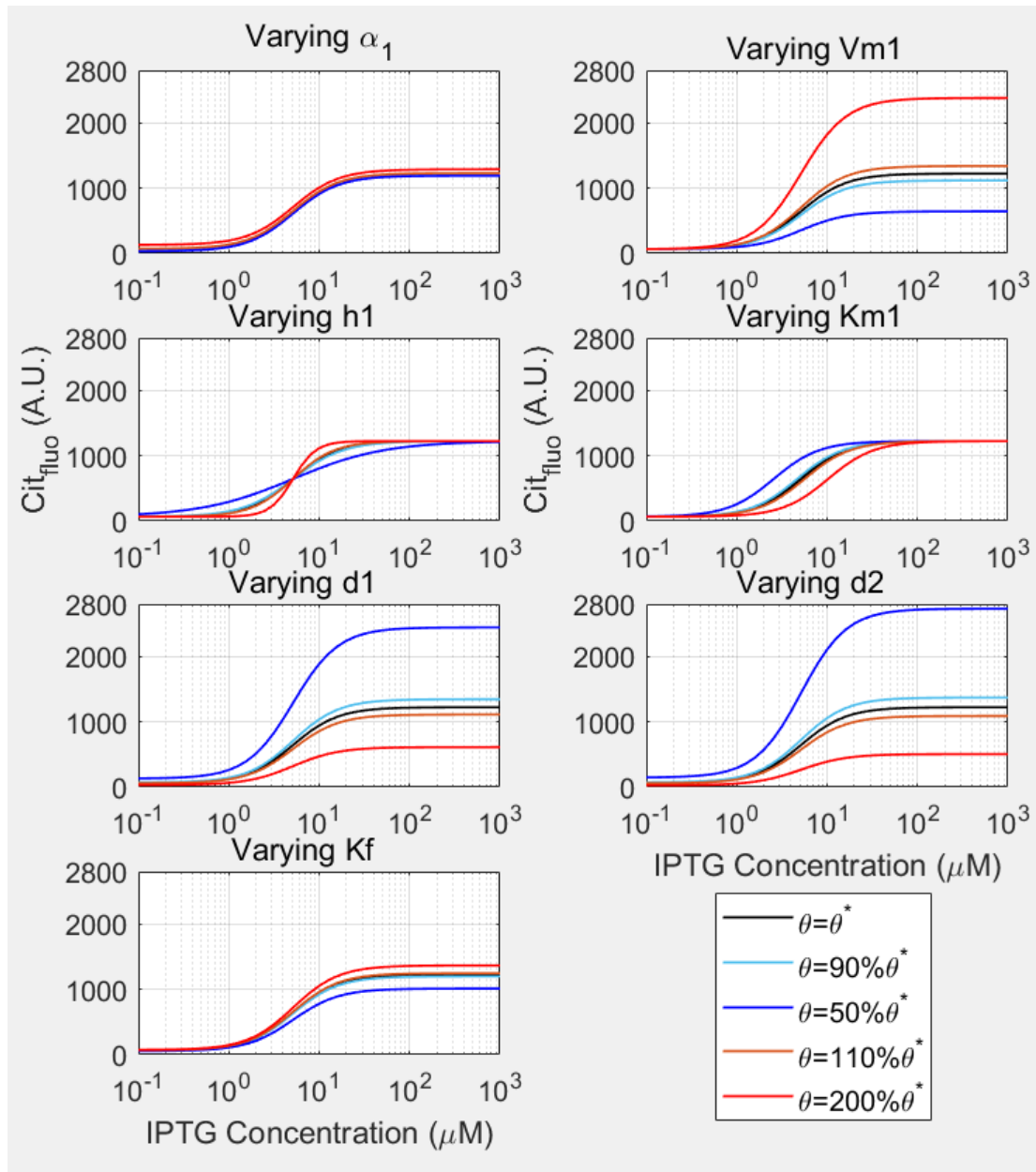
Figure 2-9. Model-predicted steady observation with varying one parameter at once.
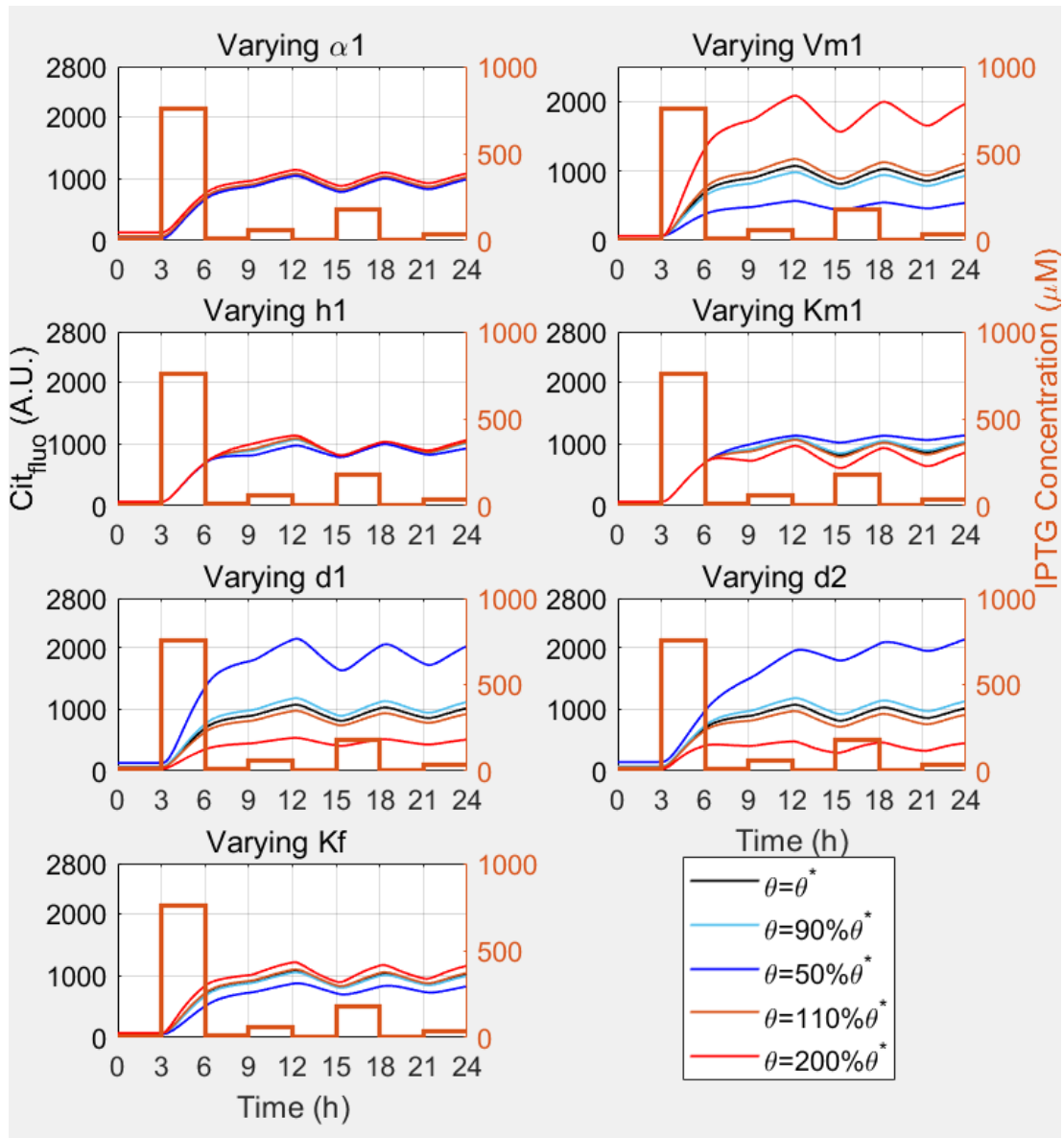
Figure 2-10. Model-predicted dynamic observation under a randomized stimulus, with varying one parameter at once.

2.2.2 Structural Identifiability Analysis of Model Parameters

As mentioned in the previous section 2.2.1, there would be structural identifiability issues when fitting all the model parameters at the same time, in other words, there are multiple parameter value sets that could provide the same model predictions. A very important point to mention is that one of the

identifiability problems involves tree parameters (α_1, Vm_1, and $\alpha2$), and this problem is difficult to find with many analysis approaches.

The analysis of parameter structural identifiability is often necessary for mathematical modelling, and the development in algorithms is a widely discussed aspect in related fields because most of the current approaches are typically computationally intensive or have limited applicability [165, 166]. Most of these approaches can be classified into three groups: sensitivity-based approach [167, 168], objective-function-based approach [165, 169, 170], and parameter-fitting-based approach [171-173]. The discussion of these three approaches are provided as follows:

Sensitivity-based approaches mainly reflect two considerations: a. if the observation does not change with a model parameter (the corresponded sensitivity is zero or negligible), then it is impossible to estimate that parameter according to the observation; b. the sensitivity matrix of the observation corresponding to variances in different parameters could provide information about the identifiability with its rank [167, 168, 174]. In this case, 30 in-silico experiments with randomised stimuli (same as the experiment setting in section 3.2.5) are used to analyse the identifiability as randomised stimuli are unlikely to introduce extra identifiability problem in this case. As shown in Figure 2-11, the three unidentifiable parameters are not the least sensitive parameters, and the sensitivity vectors for every experiment are with full rank (eight as the number of parameters). At this stage, sensitivity-based analysis (at least without further process) can hardly find the identifiability problem with the parameter set α_1, Vm_1, and $\alpha2$. Moreover, the Fisher information matrix (a mathematical tool for quantifying the informativeness of experimental designs, as introduced in section 3.1.2, [175, 176]) based on sensitivity matrix can find the rank lost and point out the unidentifiable parameter set.

Objective-function-based approaches are the methods that work with the objective function for model fitting. The mechanism is that if the objective

function achieves the optimal value with multiple parameter sets, there would be a problem with the identifiability [170]. A typical method is that in each trial, select a subset of parameters as "free" parameters and fix the others, and then analyse the objective function's behaviour as a function of these free parameters. If the objective function achieves its optimum at different points, the identifiability problem can be found. Although theoretically this approach can find the unidentifiable parameter set in this case, the corresponded search is costly in computing. With more free parameters, both the possible combinations and the complexity of each trial increase exponentially [165, 169], not to mention the number of model parameters is also increasing nowadays. Therefore in practice, if there is no concern of the identifiability for particular parameter sets, most of the researchers would not expand the search to three free parameters or more [165, 177].
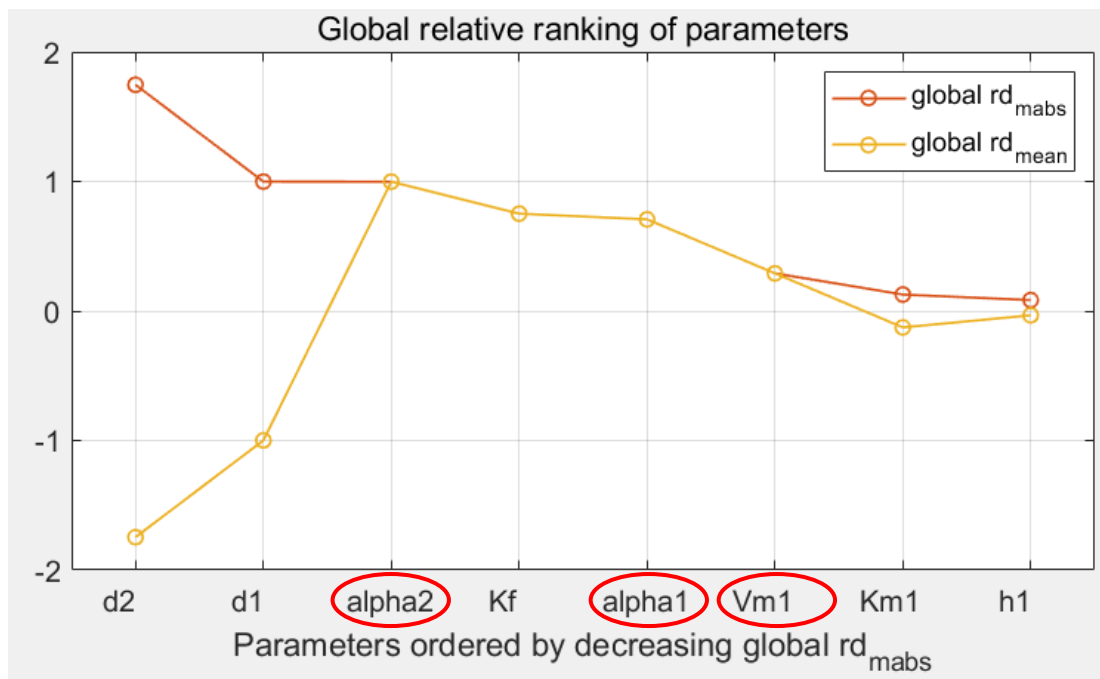
Figure 2-11. Sensitivity ranking of the parameters.
rd_{mabs} is the averaged absolute relative sensitivity of all the observations corresponding to parameter changes, rd_{mean} is the averaged relative sensitivity, red circles labels the unidentifiable parameters.

The parameter-fitting-based approach is the third category for identifiability analysis. This approach runs parameter estimation for several (usually hundreds) times with stochastic solvers. Because of the stochasticity, the estimated results would not be exactly the same and will form a distribution. If this distribution separates along a particular line, plane, or hyperplane, it suggests a problem of identifiability [171-173]. In this case, 125 trials of PE are used for identifiability analysis in this study (Figure 2-12). Similar to a previous study [166], this approach does not always provide the correct result. For example, this approach suggests the identifiability problems with α_1 vs Vm_1 and Kf vs d_1, which can be rejected by manual double-checking of the model The correlation between α_1 and Vm_1 is because the ratio between these parameters can be more accurately estimated compared to their overall scale, but it is not the case that these two parameters are not identifiable. But the problem with the three truly unidentifiable parameters is not highlighted with this analysis.
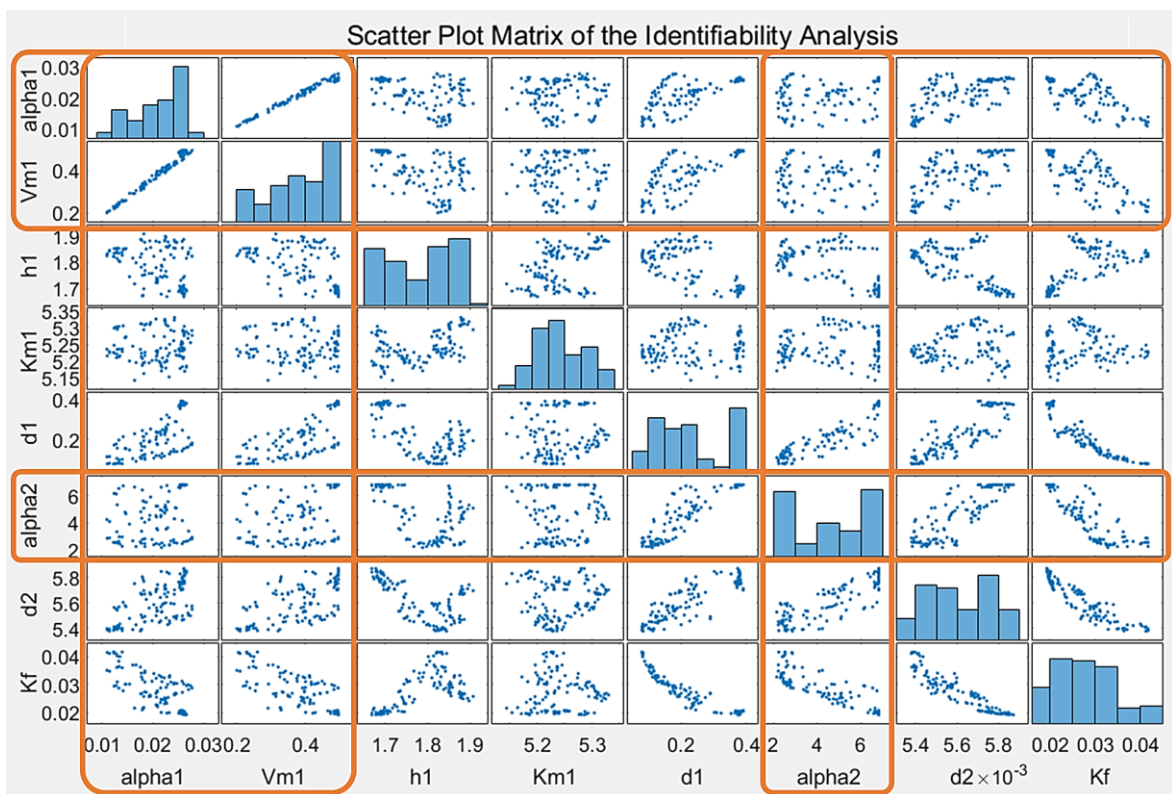
Figure 2-12. Scatter plot matrix of the parameter estimation trials. Unidentifiable parameters are marked with orange rectangles.

The parameter clustering algorithm introduced in the next chapter (section 3.2) is found to be an approach with great potential to find the unidentifiable parameters. It groups the parameters according to the informativeness for estimating each parameter corresponds to different experimental designs. Parameters having identifiability issues will remain poorly identifiable with all the designs, which is different from other parameters. In the FIM-based clustering approach, the three unidentifiable parameters are always clustered in a common group. The related results are given in section 3.3.3.

2.2.3 Comparison between Flow-Cytometry and Microfluidics

The work presented in this chapter aims to validate the microfluidic experiments and compare its model calibration accuracy with a classic approach (flow cytometer in this case).

Before comparing the microfluidic and flow-cytometry experiments, it is necessary to ensure the microfluidic device does not introduce too much stress to cells [178-180], and the cell conditions are similar in these experiments. As one of the commonly adopted indications of the stress and toxicity [181-184], the growth rates of yeast cells in the microfluidic device and the environment for flow-cytometry experiments are statistically analysed and compard. Moreover, it is necessary to examine the repeatability of microfluidic experiments, in other words, whether the experiments could obtain stable results in replicates.

About the comparison of prediction accuracy, the most direct method could be as follows:

step 1: Carry out model calibration base on all the experimental data available and obtain the corresponded parameter estimation (PE) result (the best estimation of the actual parameter set);

step 2: Calibrate the model with only microfluidic or flow-cytometer-based experimental data of the same experimental time;

step 3: Carry out the validation by comparing the experimental data from one experimental data set to the model prediction based on the other data set. The more efficient experiment set should be able to predict the other data set with acceptable accuracy.

However, there is a problem with step 1. Although for both microfluidic and flow-cytometer based experiments, there is data from 400+ hour experiment times (Table 2-4), the microfluidic experiments allow significantly higher sampling frequency, so the numbers of samples of the same experiment time are different. As a result, if the parameter values are estimated with all the experimental data, the cost function would favour one experimental method with a larger data size (which side to favour depends on the weighting based on experiment numbers, total durations, or the number of sample points). It may not be a severe problem in some studies that assume the informativeness difference between experiment systems is negligible. However, it is a problem in this thesis. Because whether the informativeness difference is negligible for microfluidics and flow-cytometry experiments is one of the main questions to answer in this chapter.

Table 2-4. Data size of the microfluidic and flow-cytometry experiments.

	microfluidics	flow-cytometry	Data size Ratio
No. Experiments	18	24	3:4
Total Duration	432 hours	1152 hours	3:8
No. Sample Times	5184	120	216:5

Therefore, the final methodology is adopted as follows:

step 1: Calibrate the model with only microfluidic or flow-cytometer-based experimental data corresponds to the same total experimental duration (288 hours);

step 2: Use the calibrated models to predict the experiments that are not used for calibration (144 hours).

step 3: Validate the models by comparing the experimental data to model predictions. A good network model should provide accurate model predictions for the experiments on the same experimental device and also for the data on the other device.

step 4: Repeat step 1-3 for several times (60 times in the practice of this thesis) with random selections of experiments for calibration and validation, and apply statistical analysis of the accuracy difference.

2.2.4 Microfluidic Experimental Platform
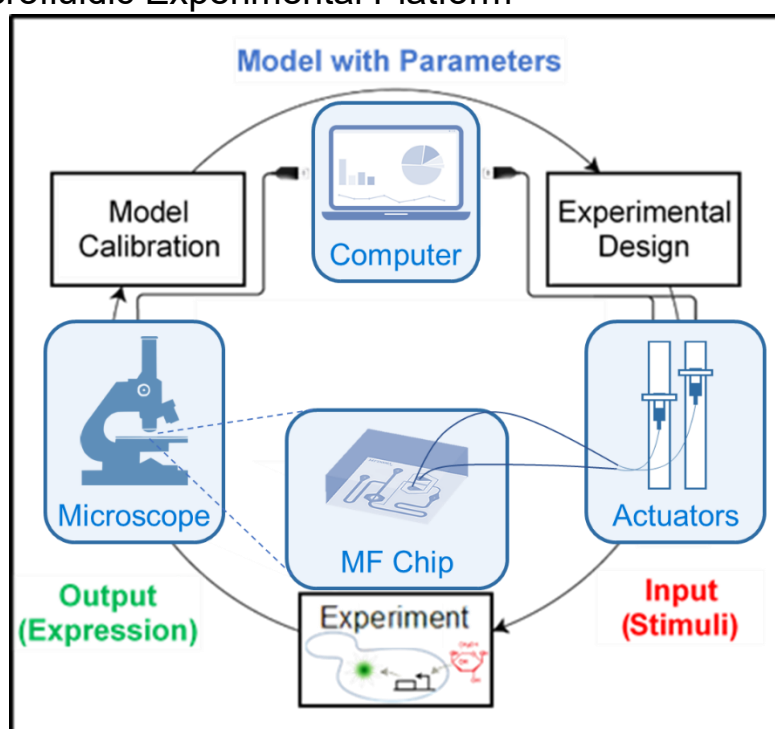
Figure 2-13. The workflow of the microfluidic experimental platform.

The microfluidic experiments are carried out on our automated experimental platform. Figure 2-13 shows that the experimental platform is constructed with four hardware parts: the computer, hydrostatic pressure actuation system (for controlling the input concentration), the microfluidic chip, and the digital microscope. Figure 2-14 are two photos of the equipment. The details of the hardware in given in Table 2-5.
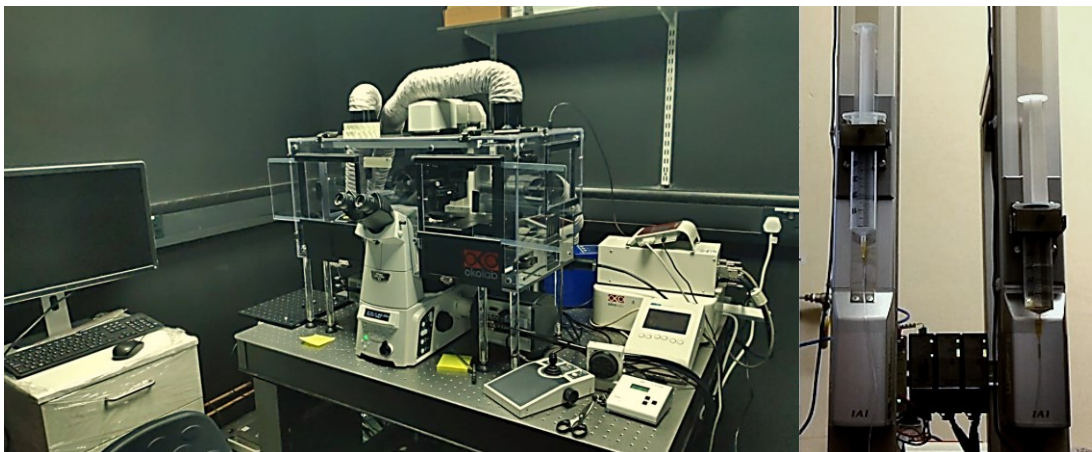
Figure 2-14. Photo of hardware devices of the platform.
Because of the limited space in the lab room, the actuators cannot
be included together with the other equipment.

Table 2-5. Details of Experiment Platform Devices.

Device	Details
Computer	OS: Windows 7 Involved Software: MATLAB, MicroMnager
Microscope	Nikon TI2
Camera	Andor iXonUltra 888
Objective Lens	Nikon 40x oil lens
Spectra	Lumencor SPECTRA light engine Sulforhodamine channel: Green Citrine channel: Teal
Filter Block	Sulforhodamine channel: TxRed Citrine channel: 69308
Actuation System	2* IAI Linear actuator: fast speed, 800mm travel length

Figure 2-15 shows the architecture of the platform. Modelling and calculations are mainly supported by the AMIGO2 toolbox [174], the control of actuation hardware are completed with the MODBUS interface supported by MATLAB Instrument Control Toolbox™, the micromanager supports control and data exchange with the microscope [185]. Moreover, two graphical user interfaces for controlling this system are developed for both MATLAB and Python.
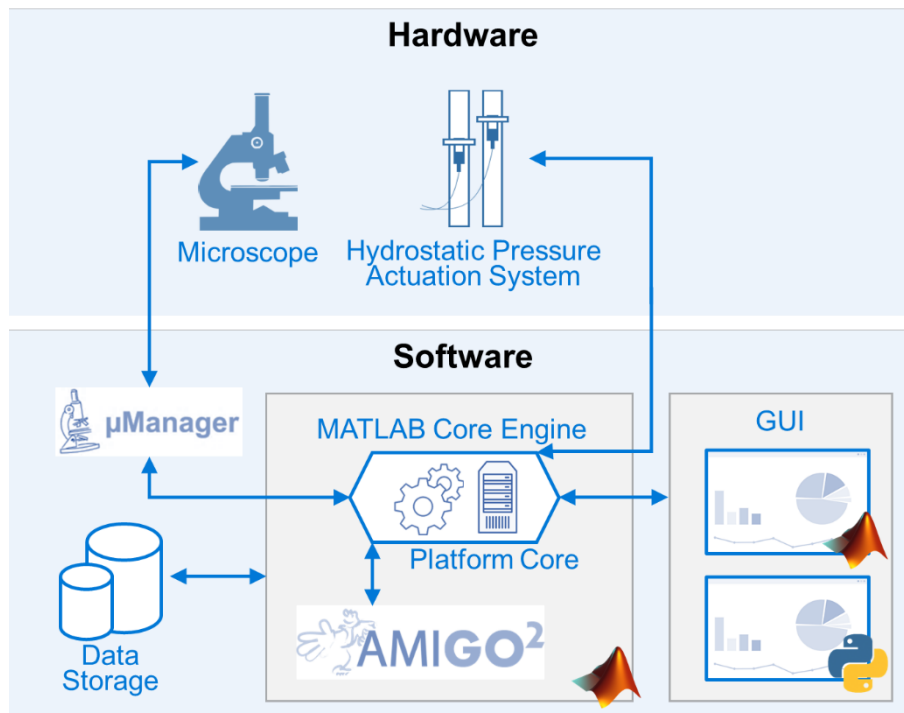
Figure 2-15. Architecture of the Experimental Platform.

The protocol for carrying out the experiment (Appendix A) and the design of the graphical user interface (Appendix B) are appended at the end of thesis.

2.2.5 Design of the Microfluidic Device

As introduced in section 2.1.2, the microfluidic device used in is project is the chip MFD0005a designed by Ferry et al. [124]. Figure 2-16 shows the detailed design of this chip. The media for each port is given in Table 2-6. To make hundreds of microfluidic chips for this study, a master board is purchased and

so as the IPTG powder and glass slides. The protocol of making the microfluidic chips in the lab is given in Appendix C.
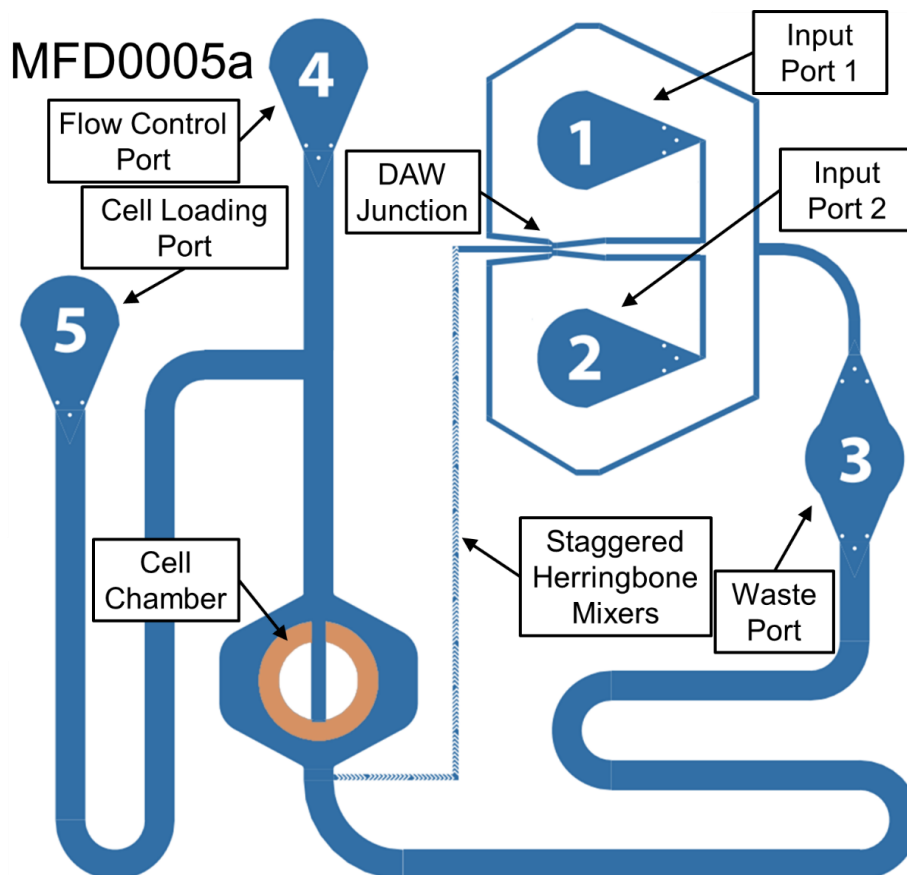
Figure 2-16. Detailed design of microfluidic chip MFD0005a [124].

In Figure 2-16, lines in blue are channels with a height of $10\mu m$ for the media flow. The orange area is the cell chamber which has a lower height of $3.5\mu m$. This area is only tall enough for one single yeast cell to "squeeze-in" so that the cells would grow in monolayer. There are five ports on the chip. Ports 1 and 2 are the two input media ports; port 3 is the waste port; port 4 is used to adjust the flow from port 5 where cells are loaded to the chip.

Table 2-6. Details of the media for each port.

Port ID	Content
1	Input 1: SC media + $100\mu M$ IPTG + $1\mu g/ml$ Sulforhodamine (dye)
2	Input 2: SC media
3	Waste: SC media
4	Flow Control: SC media
5	Cell Loading: SC media + Cells (loading) / SC media (experiment)

During the experiment, the port 5 is connected to a media source without cells and the media flow is from ports 1&2 to ports 3,4, and 5. When loading the cells, port 5 will be connected to a syringe with cells suspended in media, and the flow is from ports 4,5,1, and 2 to port 3, so that the cells will be loaded to the chamber.

The dial-a-wave (DAW) junction is a joint that convert hydrostatic pressure different of port 1&2 into a corresponding mixing ratio. The port with higher pressure (by changing the heights of the media sources bound to the linear actuators) will provide more media flow into the chip, which would eventually increase/decrease the concentration of IPTG and florescent dye in the chamber. The design if DAW junction is proposed by Ferry et al. [124]. Different from the traditional T junction, a DAW junction has two extra output channels which link to the waste port. This design solves the back-flow problem between the two input ports and allows more accurate and stable control of the mixing ratio as the input to the cells.

Staggered Herringbone Mixers is a region that has a specially designed channel ceil. This design introduces rotational force to the media when the media flows in it. With these mixers, the media from port 1 and port 2 will be mixed much more evenly by reaching the cell chamber, so that the concentration of the stimuli (IPTG in this study) will be the same for all the cells.

2.2.6 Processes of Microfluidic Experiment

Similar to flow-cytometry [186] and other biological experiments [126], microfluidic experiments involve multiple processes to carry out. Figure 2-17 is the flow chart of the experiment processes. Compared to the guidance provided by Ferry et al. [124], the modified platform in this study achieves a higher automation level by integrating the control of multiple experiment procedures to the uniform control panel: such as the DAW junction calibration, cell loading, experimental control. Moreover, from practices in wetting, it is

found that using two syringes for wetting instead of one can speed up the process. The protocol is provided in Appendix A.
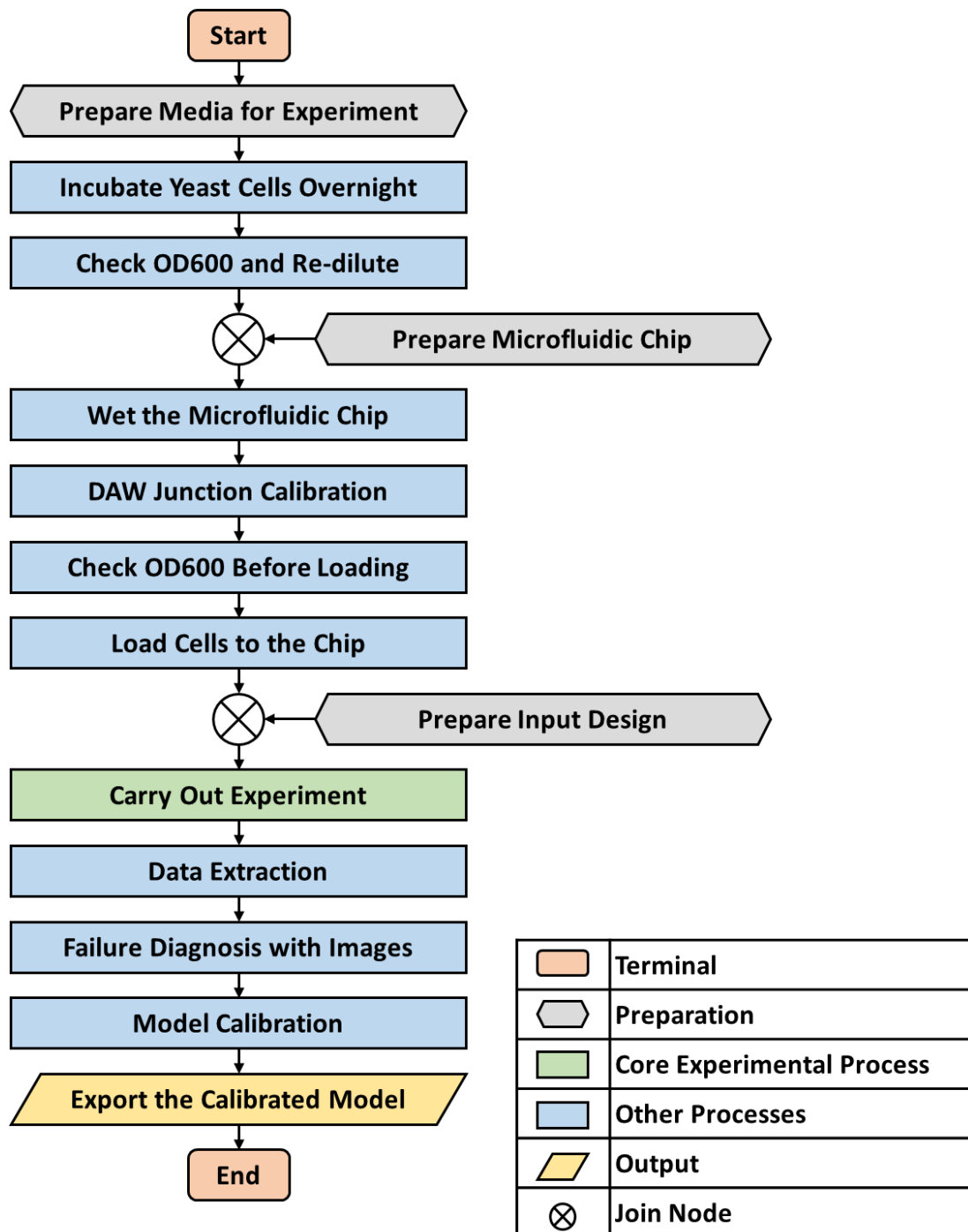
Figure 2-17. Flow chart of the processes for microfluidic experiments in this study.

2.2.7 Processing of Experimental Images

During the experiment, the digital microscope will take three images corresponds to the three channels: Differential Interference Contrast (DIC), Sulforhodamine, and Citrine channel. The DIC channel is used to distinguish and identify the cells. The Sulforhodamine channel is used to confirm the mixing ratio of the two input media sources because Sulforhodamine is added to one source. The Citrine channel can tell the output signal intensity. The processing of these images mainly involves three steps: cell segmentation, signal compensation, and signal readout.

Cell segmentation is only based on the DIC image and relies on the CellStar toolbox (MATLAB version) [187]. Although this toolbox supports both segmentation and tracking (S&T), the downstream algorithm for parameter estimation is not designed for single-cell tracking observations, so the cell-tracking is not used in this project. Figure 2-18 is a brief graphical explanation of the CellStar algorithm given by the developer team.
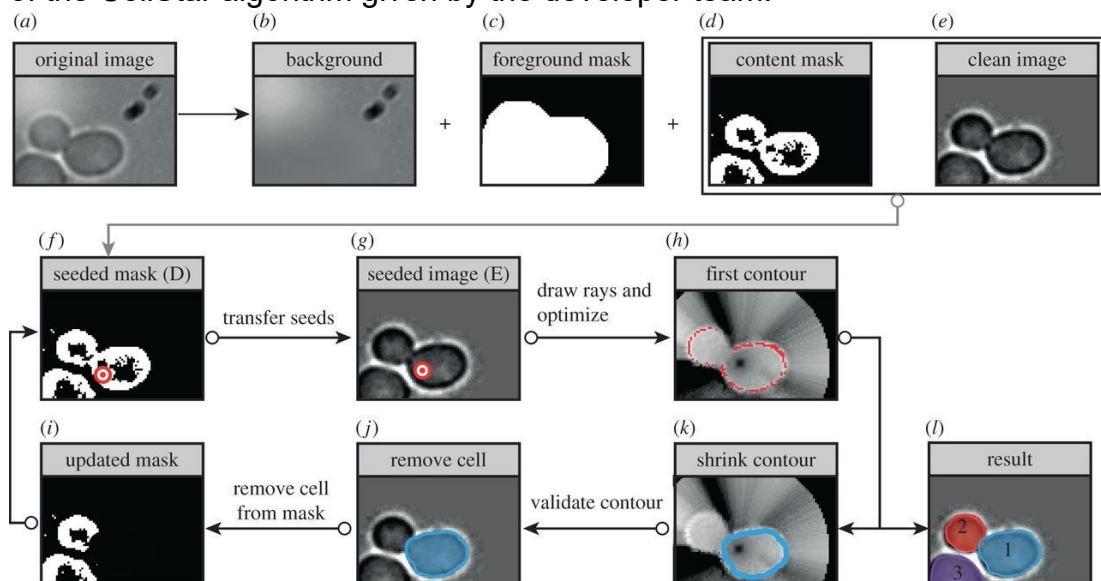
Figure 2-18. Segmentation pipeline of the algorithm [187]. a-e. generate the background/foreground/cell content masks and clean image as the preparation of segmentation; f-h. find the seed of a cell (redpoint) and then compute the edge (red line) along with the concentric rays, there is a constrain parameter to improve the accuracy of the edge detection; k-i. remove the identified cell from the image and repeat step 2 until there are no more cells on the image.

In this project, the background image is the average of the first ten DIC images with some manual modifications to remove the cell pixels. It is also why the device movement causes problems and needs to be identified as early as possible to avoid additional time and resource loss (see chapter 4 for details of the fault diagnosis). The first ten images' cells are also manually marked as the ground truth for calibrating the constrain parameter for the automated cell segmentation on later images.

Signal compensation is involved in the data processing because the signal crossover between Sulforhodamine and Citrine is not negligible from the raw data. Although the selection of Sulforhodamine as the input signal already consider the separation of the excitation/emission wavelengths with Citrine, the overlapping of the signals still above 5% in practice. To solve this problem, the signal crossover is quantified and extracted from the raw data. Figure 2-19 shows a diagrammatic representation of the fluorescence signal crossover. The compensation that corresponds to this representation is also known as hardware compensation [188].
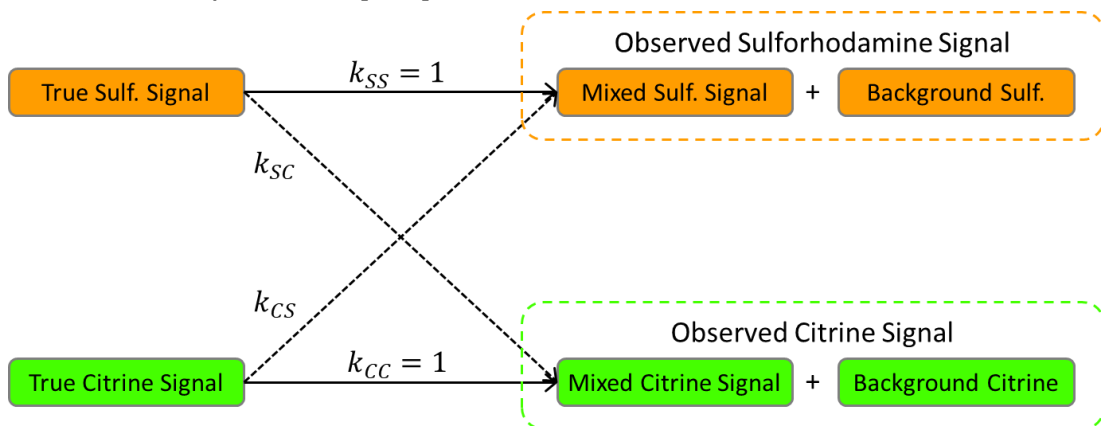
Figure 2-19. Diagram of the fluorescence signal crossover.
For example, the Sulforhodamine causing 1 unit signal intensity in the Sulforhodamine channel, would cause k_{SC} unit of signal increase in Citrine channel.

The corresponded mathematical expression of Figure 2-19 is as follows:

$$\begin{bmatrix} O_{Citrine} \\ O_{Sulf} \end{bmatrix} = \begin{bmatrix} 1 & k_{SC} \\ k_{CS} & 1 \end{bmatrix} \times \begin{bmatrix} S_{Citrine} \\ S_{Sulf} \end{bmatrix} + \begin{bmatrix} O_{Citrine}^0 \\ O_{Sulf}^0 \end{bmatrix} \qquad (Eq.\,2-3)$$

$$\begin{bmatrix} S_{Citrine} \\ S_{Sulf} \end{bmatrix} = \begin{bmatrix} 1 & k_{SC} \\ k_{CS} & 1 \end{bmatrix}^{-1} \times \begin{bmatrix} O_{Citrine} - O_{Citrine}^0 \\ O_{Sulf} - O_{Sulf}^0 \end{bmatrix} \qquad (Eq.\,2-4)$$

where S represents the true fluorescence signal, O represents the observed signal, O^0 represents the background signal.

The crossover coefficients k's values are possible to estimate according to the fluorescence spectra, the excitation/emission filter settings, and exposure times [189]. However, this method is complicated and not very accurate because it involves many uncontrollable factors of the light source, observed sample, lenses and other devices. Meanwhile, the experiments in this study offer an alternative and more accurate way to calculate these values. For the pixels away from yeast cells, it is reasonable to consider that there is no true Citrine signal; for the cell pixels at the beginning of experiments, it is reasonable to consider there is no true Sulforhodamine signal. In other words, these two groups of pixels should form two straight lines, and their slopes and intersection could be used to evaluate the compensation parameters (Figure 2-20). It is supported by images from experiments (Figure 2-21). Figure 2-22 shows an example of compensation for a step experiment. After compensation, the Citrine signal intensity remains around 0 and does not change with the IPTG Concentration (Sulforhodamine concentration) anymore.
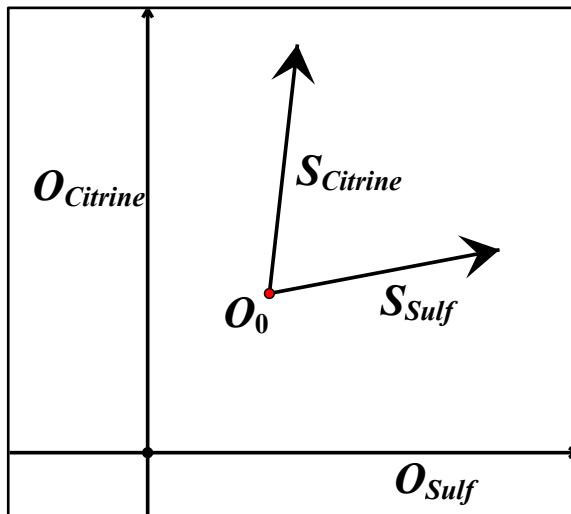
Figure 2-20. Illustration of the expected distributions of the two pixel groups.
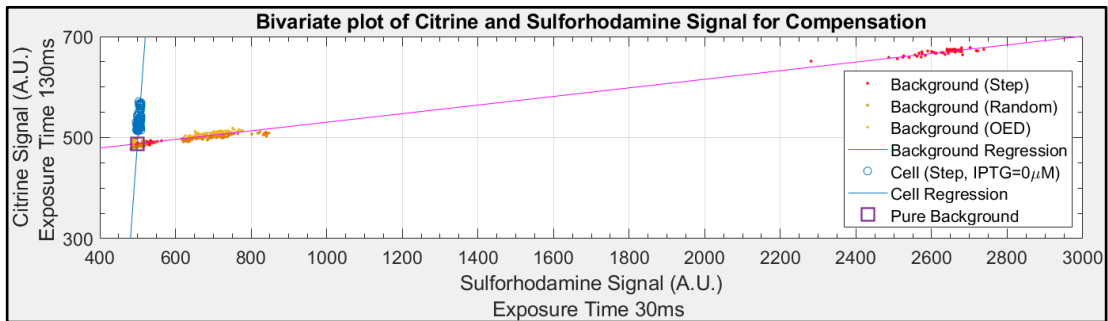
Figure 2-21. Two pixel groups from the experiment images shown in circles and dots.
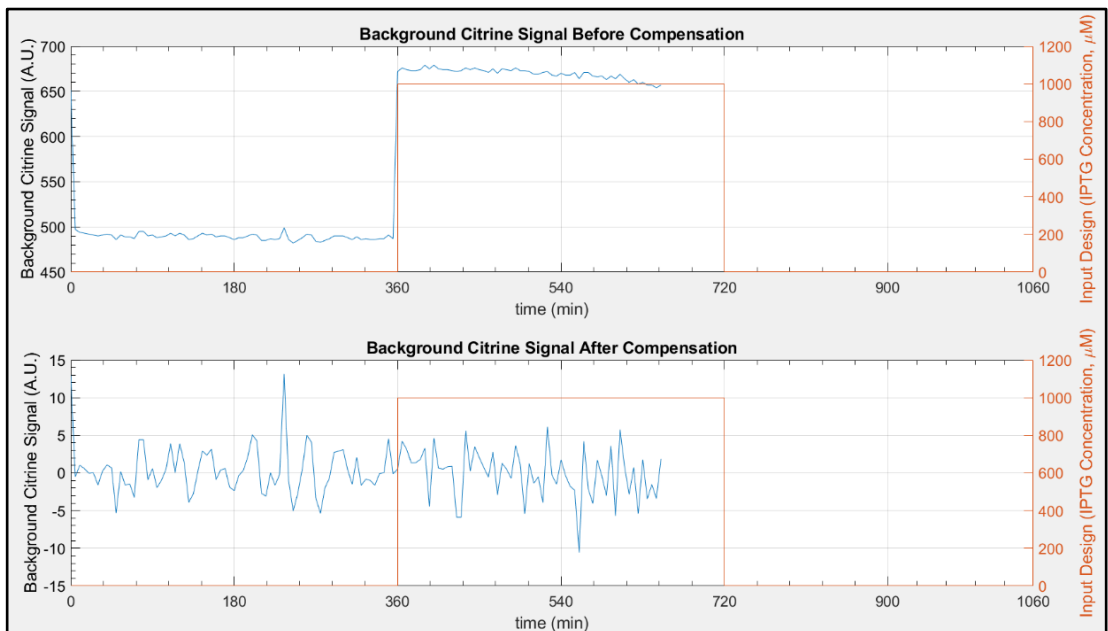
Figure 2-22. Comparing the Citrine in the background in a step experiment.

The signal readout is the last step that quantifies the signal intensity. The chamber cells are stuck between the floor and ceil, so they share the same "height". Therefore, each cell's Citrine concentration is proportional to the Citrine intensity increase of the cell compared to the background.

2.2.8 ANOVA analysis

To investigate microfluidic experiments' repeatability, each experimental design is carried out at least three times. It is reasonable to expect slightly different measurements in these replicates, but a mathematical tool is required to tell the acceptable limits of this difference. Suppose the difference between replicates is beyond this limit. In that case, it suggests a repeatability problem that the experimental results might be significantly biased by some of the non-repeatable experimental aspects (such as the selection of cell colonies, background light intensities, and operational uncertainties in media preparations).

The one-way analysis of variance (one-way ANOVA) F-test is adopted in this study to quantitively judge whether the measurements from replicates differ from each other. The T-test is not used because of the familywise error [190, 191].

The principle behind the F-test in this research is that for the sampled images at the same time point in all the replicates, the observed intensity distributions of cells should be the same. The corresponding F-score is the ratio of the between-replicate variability and the within-replicate variability, which is expressed as follows:

$$F^t = \frac{MS_M^t}{MS_R^t} \qquad\qquad (Eq.\,2-5)$$

where MS_M^t is the model mean of squares at sampling time t, and MS_R^t is the residual mean of squares at sampling time t. They are defined as:

$$MS_M^t = \frac{\sum_{i=1}^{k} n_i^t (\bar{y}_i^t - \bar{y}_{all}^t)}{k-1} \qquad (Eq.\,2-6)$$

$$MS_R^t = \frac{\sum_{i=1}^{k} (n_i^t - 1){\sigma_i^t}^2}{\left(\sum_{i=1}^{k} n_i^t\right) - k} \qquad (Eq.\,2-7)$$

where i is the index of the replicate, n_i^t is the number of observed cells in the ith replicate at time t, k is the number of replicates (3 in this study).

The calculated F-scores are compared to the critical values given by the F-distribution:

$$QF_{0.95}^t = F_{0.95}\left(k-1, \left(\sum_{i=1}^{k} n_i^t\right) - k\right) \qquad (Eq.\,2-8)$$

If the F-score F^t is smaller than $QF_{0.95}$, it means the intensity distributions from three replicates at the sampling time point are statistically different, suggesting a problem of repeatability.

2.2.9 Parameter Estimation

The parameter estimation (PE) is a procedure that searches for the feasible parameter value set that minimises the difference between the model prediction and experimental observation. The corresponded cost function to minimise is as follows:

$$cost_u(\theta) = \sqrt{\frac{1}{N} \sum \left(\frac{y_i - \hat{y}_{u,i}(\theta)}{\sigma_i}\right)^2} \qquad (Eq.\,2-9)$$

where u is the stimuli pattern, N is the number of observations, y_i and σ_i are the mean and standard deviation of the observation value at the ith sampling time, $\hat{y}$ is the model prediction.

Similar to many other biological models, the ODE model used in this project is non-linear and does not guarantee its convexity. Therefore, a global solver is required for both the PE and OED procedures. As shown in Table 2-7, the solvers for this type of problem are commonly divided into deterministic and stochastic [101, 192-195]. In recent years, there have been advances in both of the algorithm groups [194, 196], and at the moment there is a widely held view that deterministic approaches are generally more computationally intensive and may require certain assumptions of the mathematical problems to solve (which may not be guaranteed in general OED models) [90, 197-199].

Table 2-7. Table of typical global searching algorithms for solving nonlinear nonconvex optimisation problems.

Deterministic	Stochastic
αBB **[200-202]**, BARON **[203]**, LindoGLOBAL **[204, 205]**, Convex Solvers+Relaxation **[196, 206]**.	Genetic Algorithms [207, 208], Differential Evolution [209, 210], Particle Swarm Optimization [211, 212], Scatter Search [90, 213].

Moreover, some recent studies obtained outstanding results by adopting hybrid approaches of stochastic global solver+deterministic local solver [214-216]. These methods benefit from the advantages of both the solvers: stochastic global solver leads to a higher chance of finding the global optimum with limited evaluations, and deterministic local solver is more efficient for improving the accuracy of the final answer.

In this task, an enhanced version of scatter search (eSS [192]) is adopted as the global solver, and the Nelder-Mead simplex algorithm [217] is used as the local solver to furtherly increase the accuracy. The eSS algorithm shows outstanding robustness and efficiency in a previous benchmarking study by

Egea et al. [192], and this approach has shown to be efficient in a previous study on the same model [27].

2.2.10 Experimental design

Flow-cytometry experiments were performed by Gnügge et al. [109]. Yeast cells were grown overnight either without IPTG or of $1000\mu M$ IPTG, and then moved to another IPTG concentration media once the experiment starts. Cells were sampled and loaded to the flow-cytometer to read the Citrine channel intensity every 12 hours till the end of 48-hour-long experiments. The IPTG concentrations used in flow-cytometry experiments are given in Table 2-8.

Table 2-8. Table of concentrations used in the flow-cytometry experiments

Stage	IPTG concentrations (μM)											
Cell Preparation	0/1000											
During Experiment	0	2.5	5	7.5	10	15	20	25	35	50	100	1000

The microfluidic platform allows for more frequent sampling and more complex stimuli. However, the duration would be shorter because the cells would fill the chamber after 24 hours and cells in the middle will have limited access to nutrient (get unhealthy). The DIC images are taken every 2.5 mins, and the fluorescence images are taken every 5 mins.

The experimental designs are given in Figure 2-23. All the experimental designs start with a 3-hour-long period without IPTG to confirm cells are at a steady initial state. For random stimuli, the IPTG concentration is randomly selected from 0.1 to 1000 μM on a log scale.

The experimental designs are referred from classical experimental designs from a previous study on the same model [27]. Random, Step-7.5 and PulsesX-1000 are the experimental designs included in the original paper. However, practices show that the pulsing frequency in PulsesX-1000 is a bit high to observe the corresponding oscillated expression level for lower IPTG concentrations (the expression level would remain at the level of averaged IPTG concentration with minor changes to the pulses). Therefore, the pulsing frequency for the pulse experiments of lower IPTG concentrations is reduced from $3h^{-1}$ to $5h^{-1}$.
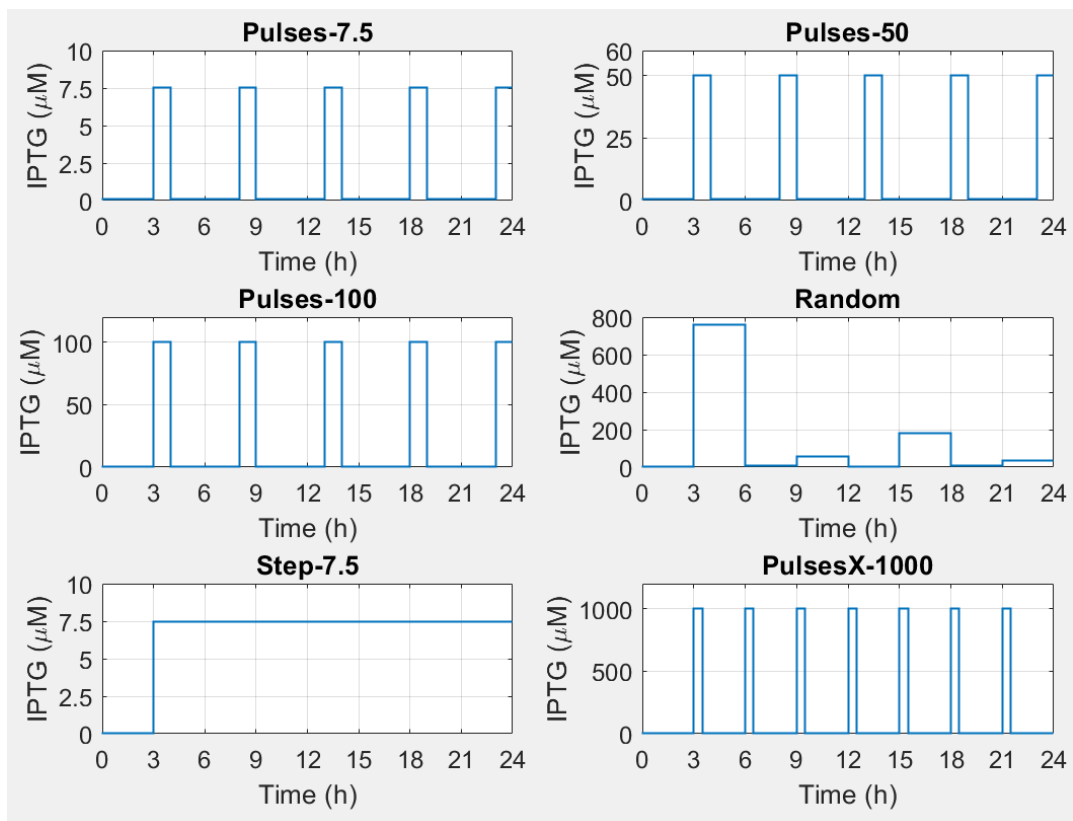
Figure 2-23. Experimental designs for the microfluidic experiments.

2.3 Results

2.3.1 Cell growth condition

As mentioned in section 2.2.3, it is vital to ensure that the microfluidic device does not introduce too much stress to cells and is comparable with flow-cytometry experiments. As previous studies have shown [181-184], when cells are under stress, the reactions involved in this model would be affected and so as the growth rate of cells. In this study, the growth rate in the cell chamber in microfluidic devices is compared to the growth rate in the incubator which is very similar to the environment in flow-cytometry experiments from Gnügge et al. [109]. The main difference is the container they used were glass tubes and plastic vial tubes were used in this study, which should not introduce a significant difference.
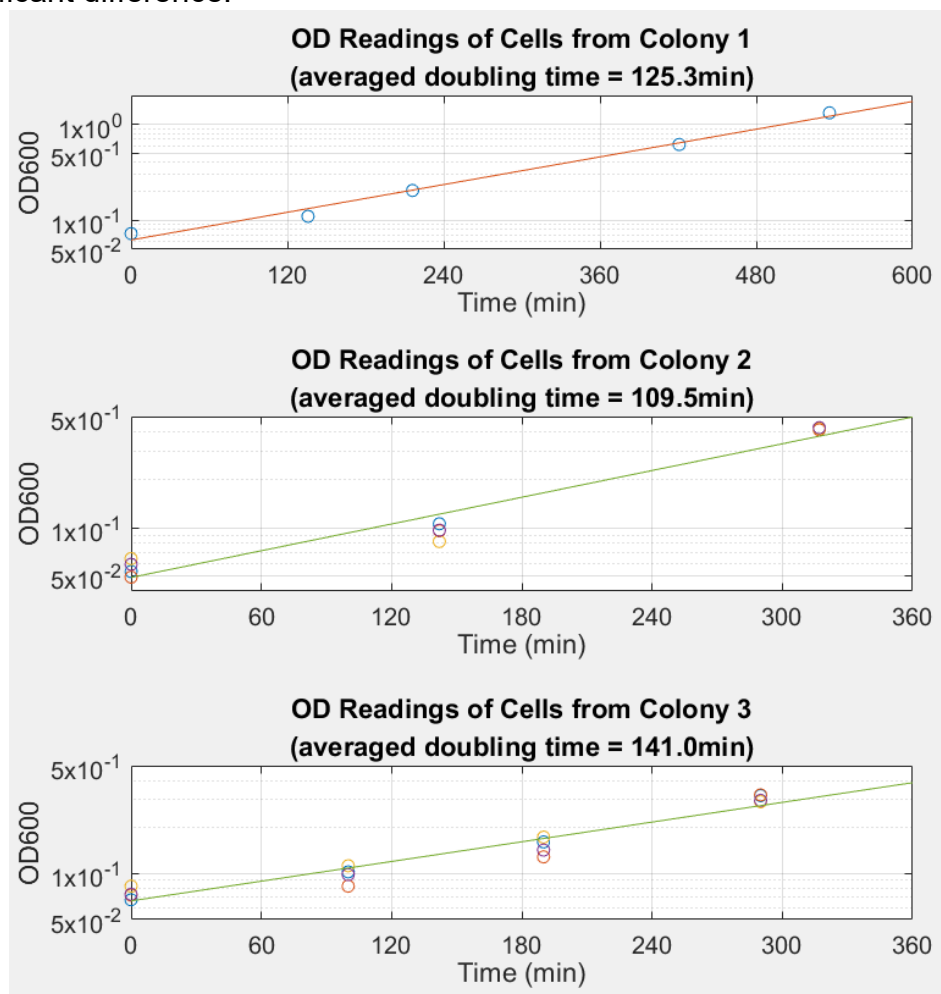
Figure 2-24. OD600 readings of incubated cells from three colonies.

According to the OD600 measurements (proportional to the number of cells in a unit volume,), the cells' doubling time in the incubator at 30 degrees temperature without IPTG is 126.0 min on average with a standard deviation of 17.2 min.
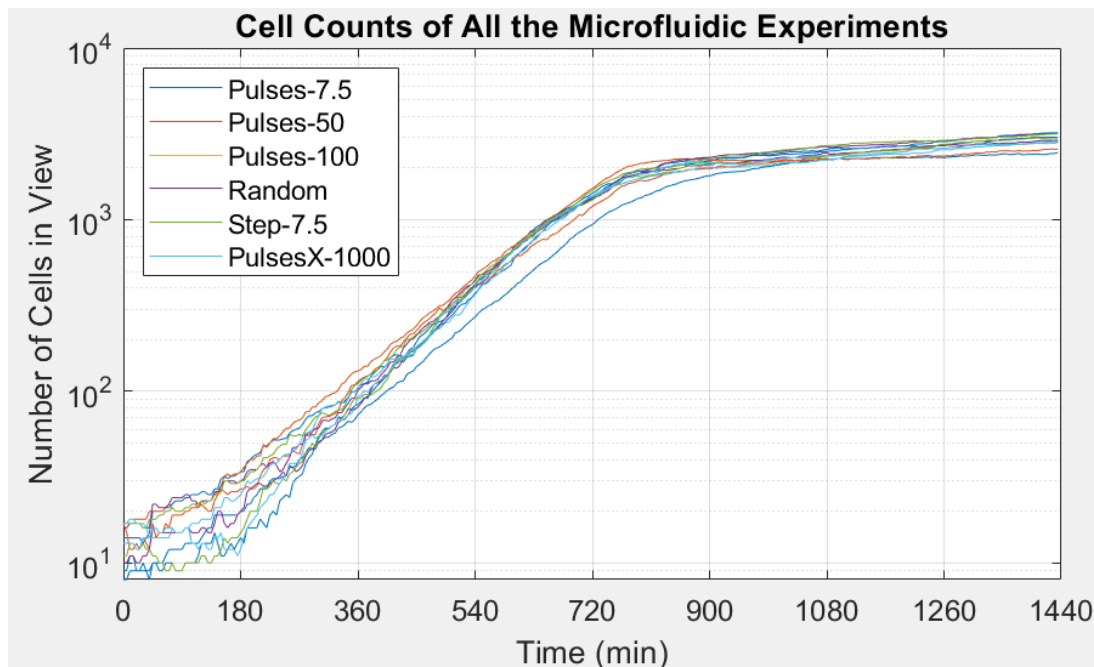
Figure 2-25. Cell counts of all the microfluidic experiments.

Figure 2-25 shows the count of cells in view during the microfluidic experiments. After 12 hours (720 min), the growth in number significantly drops, and cell count remains constant. The reason is not that the cells stop growing, but because the cells fill the sampling field, the newly budded cells push some of the cells out of the camera view.

Bootstrapping algorithm is adopted to calculate the doubling time in microfluidic devices. For every experiment, the doubling time is estimated base on cell counts at 15 pairs of time points randomly selected from 180 min to 600 min. After statistical analysis, the doubling time in microfluidic chips is 111.21 min on average with a standard deviation of 34.4 min. As shown in Figure 2-26, the t-test shows that the average doubling time in microfluidic chips is not statistically different from the doubling time in the incubator.
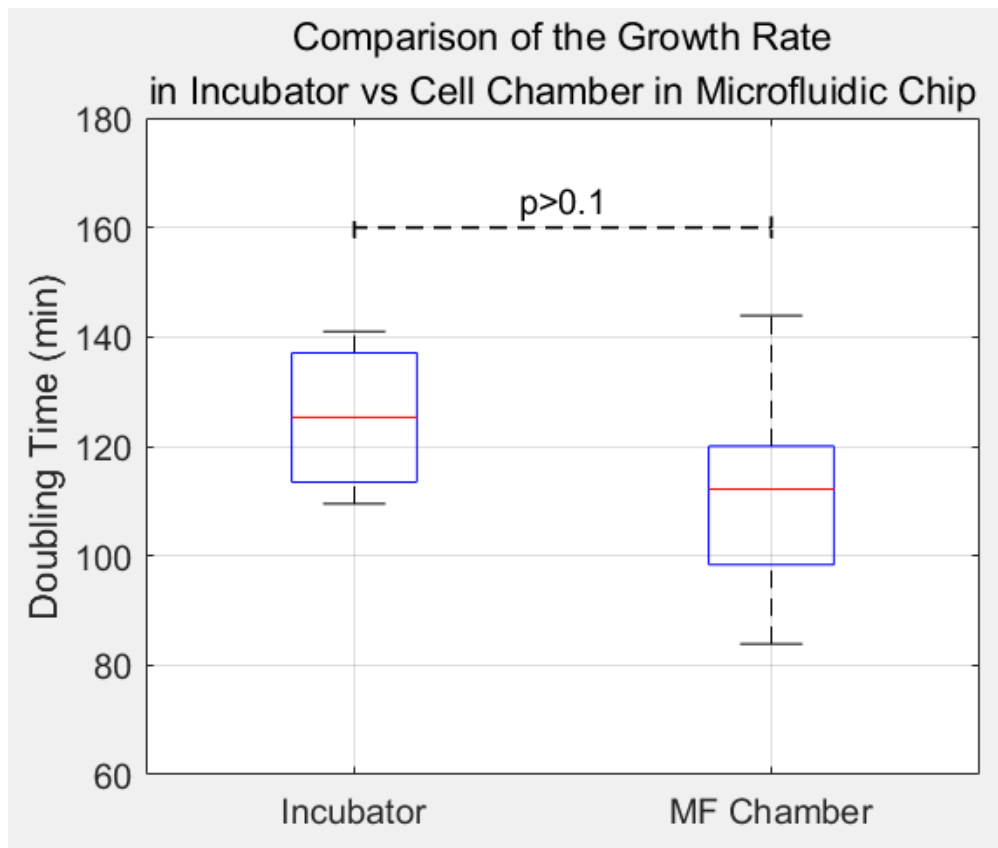
Figure 2-26. Comparison of the growth rate in incubator vs microfluidic chip. Both of the sampled distributions pass the Kolmogorov-Smirnov normality test.

2.3.2 Experimental Data and ANOVA of Replicates

The experimental data and analysis of variance (ANOVA) results are shown in Figure 2-27. The F-scores from the ANOVA are less than 30% of the 0.95 critical values. It means that the readouts in three replicates are not statistically different; the within-replicate variability between cells can explain more than 95% of the variance in the reading. Figure 2-28 shows the mean and standard deviation of observed experimental data at each observation time point.
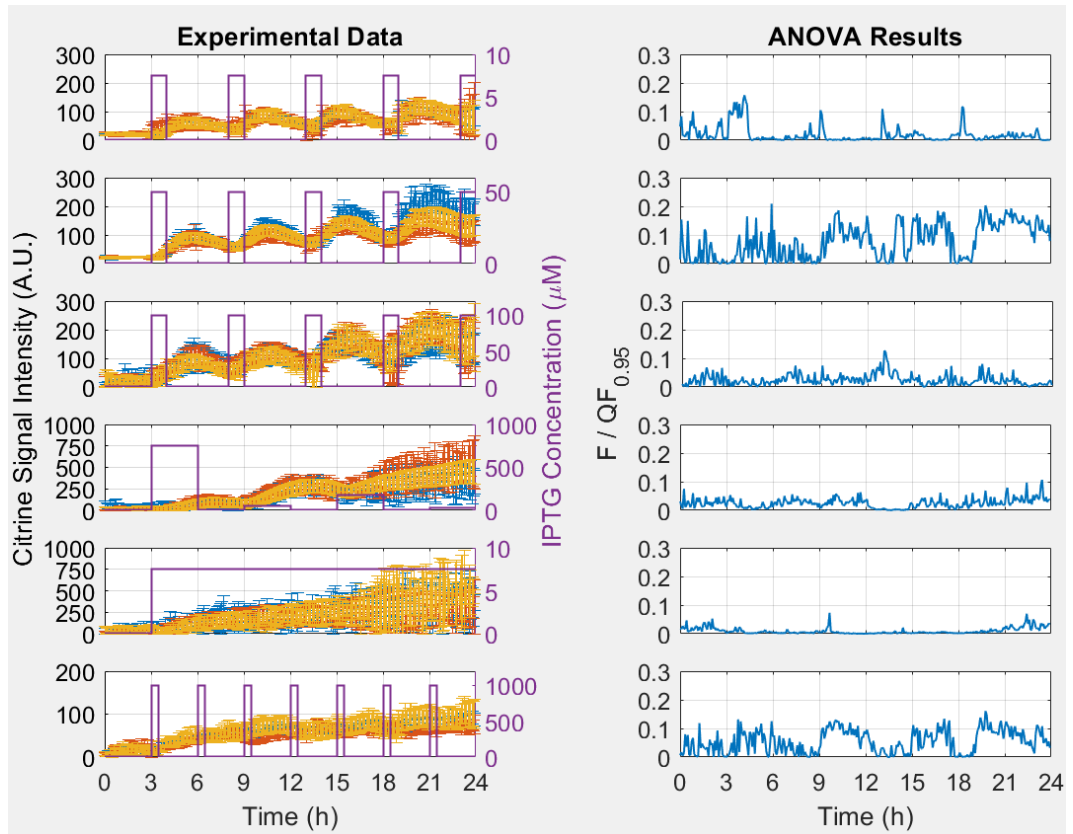
Figure 2-27. The experimental data (left) and ANOVA results (right).
The data of three replicates are shown in blue, red, and yellow with
mean and std. of signal intensity of all the observed cells.
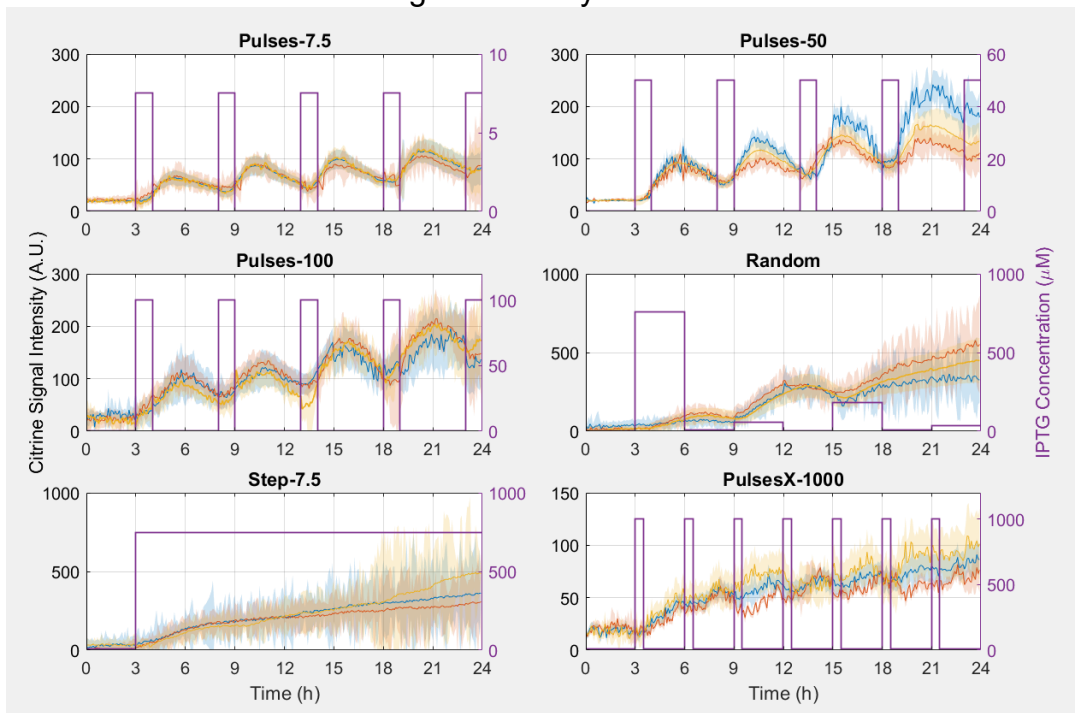
Figure 2-28. The experimental data.
Solid lines are the mean signal intensity and the coloured areas
shows the standard deviation.

53

2.3.3 Cross-Validation of the Calibrated Model

As discussed in section 2.2.3, for both flow-cytometry and microfluidic experiments, data of 288 experiment hours are used for model calibration. Data from 144 experiment hours are used for validation (three-fold validation). Because there are six experimental designs for microfluidic experiments, there are 15 different ways to separate the experimental data. For flow-cytometry experiments, 50 out of 41,783,280 possible separation ways are randomly selected for cross-validation.

The scores of the fitting are evaluated in the same way as the cost function for parameter estimation (Eq. 2-9), which is introduced in section 2.2.9. This value is always non-negative, and a smaller value means the model prediction is closer to the observation, in other words, more accurate:

$$cost_u(\theta) = \sqrt{\frac{1}{N} \sum \left(\frac{y_i - \hat{y}_{u,i}(\theta)}{\sigma_i} \right)^2} \qquad (Eq.\,2-9)$$

The scores for cross-validation are shown in Figure 2-29. The notations give the calibration set first and then the validation set. For example, MF-FC means calibrate the model with microfluidic fluidic experiments and validating the model with flow-cytometry experiments.

Figure 2-29 shows that the models calibrated with flow-cytometry (FC) data perform well on other FC experiments. However, they perform much worse in the validation with microfluidic (MF) experiments (the median of score increases to 14.89 times higher with $p < 10^{-23}$). On the other hand, the scores do not increase for models calibrated with MF experiments. Moreover, the MF-based models do not perform significantly worse than FC-based models in the validations with FC-experiments ($p > 0.1$). The average scores for FC based validation are generally smaller, which could be due to a simpler experimental

design in FC experiments (one step experiments). Thus, the response of the model states and observations are simpler and easier to be predicted.
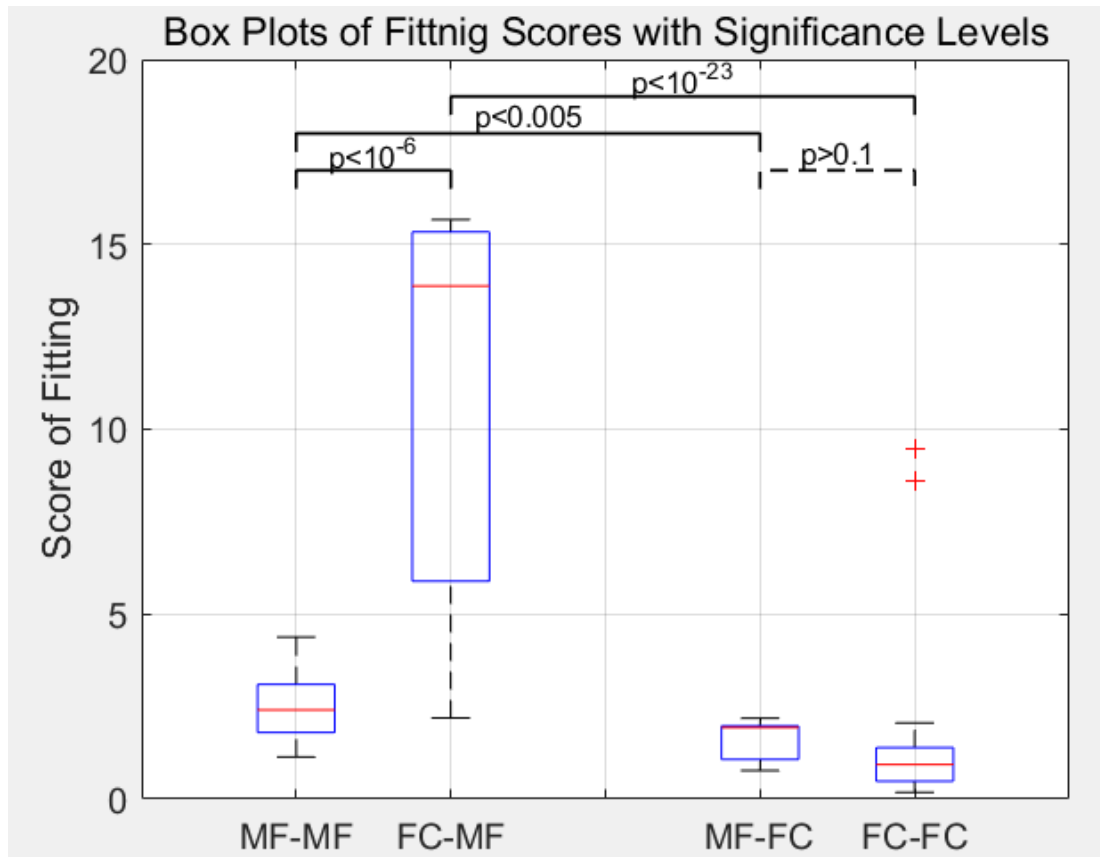
Figure 2-29. The comparison of validation scores for all the trials. Smaller scores stand for more accurate model predictions. The significance levels (p values) come from one-sided t-tests, and all of the four score groups pass the Kolmogorov-Smirnov normality test.

From this benchmarking experiment, microfluidic experiments lead to models with better universality for predicting cell behaviours. Comparing between models calibrated by FC and models calibrated by MF, the later are more likely to give more accurate predictions of the yeast strains' Citrine expressions when the circuit is subjected to a new experimental condition.

2.4 Discussion

2.4.1 Experimental Results

Analysis of the cell growth from section 2.3.1 shows that the cells' growth rates in microfluidic chip MFD005a with all the experimental designs are not significantly different from the rate during the incubation. It suggests that the IPTG stimuli (change in chemical environment) and microfluidic chip (change in the physical environment) do not introduce too much stress that disturbs yeast cells' growth. Moreover, since the incubation environment is very similar to the environment in flow-cytometry experiments, it is reasonable to expect cells would have comparable experimental data in both types of experiments.

The ANOVA analysis in section 2.3.2 shows that more than 95% of the expression level differences among the replicates of experiments can be explained by the with-in replicate variance, i.e., the variance between cells from the same trial. In other words, for all the experimental designs, the observations in the three microfluidic replicates are not statistically different. This analysis shows that the developed microfluidic experimental platform can achieve experiments on this yeast strain with satisfying repeatability.

The cross-validations in section 2.3.3 show that with the data from the same experimental time (288 hours), the models calibrated with microfluidic data perform better in the validation with microfluidic experiments with different experimental designs, and perform as well as flow-cytometry-based models in the validation on flow-cytometry experiments. Moreover, the opposite is not true: The models calibrated with flow-cytometry experiments give significantly worse scores in the validation with microfluidic experiments.

The work of this chapter shows that although microfluidic microscopy is yet a relatively new experiment technique, compared to flow-cytometry which is more sophisticated with higher throughput, microfluidic experiments achieve higher informativeness for calibrating the benchmarking inducible promoter.

As they lead to models that give better predictions in stimulus different from the experimental designs. The advantage with more complex dynamic stimuli and more frequent cell-level observations of microfluidics could be the cause of this difference.

2.4.2 Aspects Could be Improved in Future Research

There are aspects that can be improved. For microfluidic experiments in this study:

- The selections of fluorescence channels and filters can be improved. Although the signal separation of the input reporter and expressed fluorescent protein is considered in the early stage, recent studies introduced better options for the input reporter (such as Cy5 as shown in Figure 2-30), which can furtherly reduce the signal cross-over.
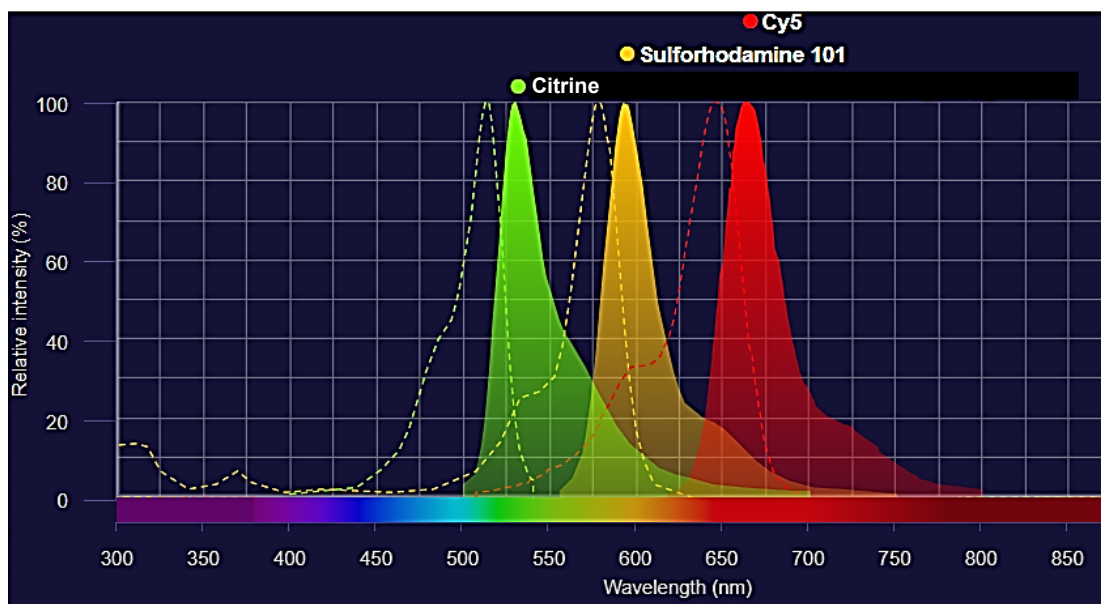
Figure 2-30. Comparing the excitation (dashed) and emission (solid) spectra of the involved fluorescence.
It can be seen that compared to Sulforhodamine, Cy5 has a significantly less cross-over with Citrine.

- The cell chamber design in the microfluidic chip can be improved. The current design of the cell chamber allows cells grown in monolayer, but because of this design, when cells fill the chamber and experiment last longer than 48 hours, some of the cells in the chamber will start dying (it can tell as the dying cells have significantly more complex observable structures in them). Some better designs (Figure 2-31 shows an example) which separate the cells and guarantee better nutrition supply may solve this problem and allow longer experiment durations [218].

- The input mixing junction can be improved. Although the DAW-junction generates more stable flow around 0% and 100% mixing ratios compared to the T-junction, it is still difficult to control the mixing ratio with an accuracy better than 3% in practice (it is not bad considering the 5% accuracy in Ferry's work [219]). Considering that many bioreactions interested in the dose-response with inputs vary in log scale [220, 221], this accuracy may limit the potential application. The input multiplexor system designed by Thorsen et al. [125] shows an option that may improve concentration control accuracy (Figure 2-32).
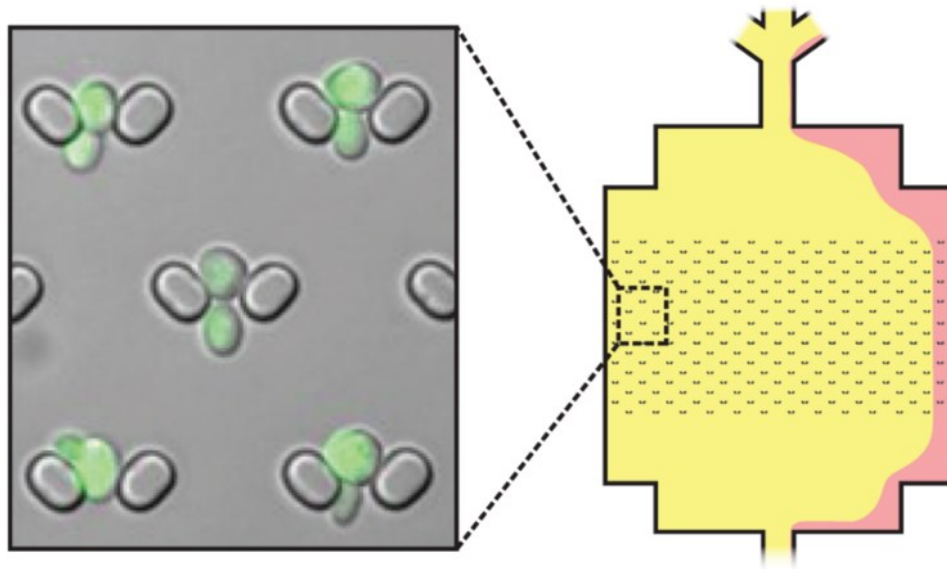
Figure 2-31. A microfluidic design that allows longer experiment durations than the current experiment [218].
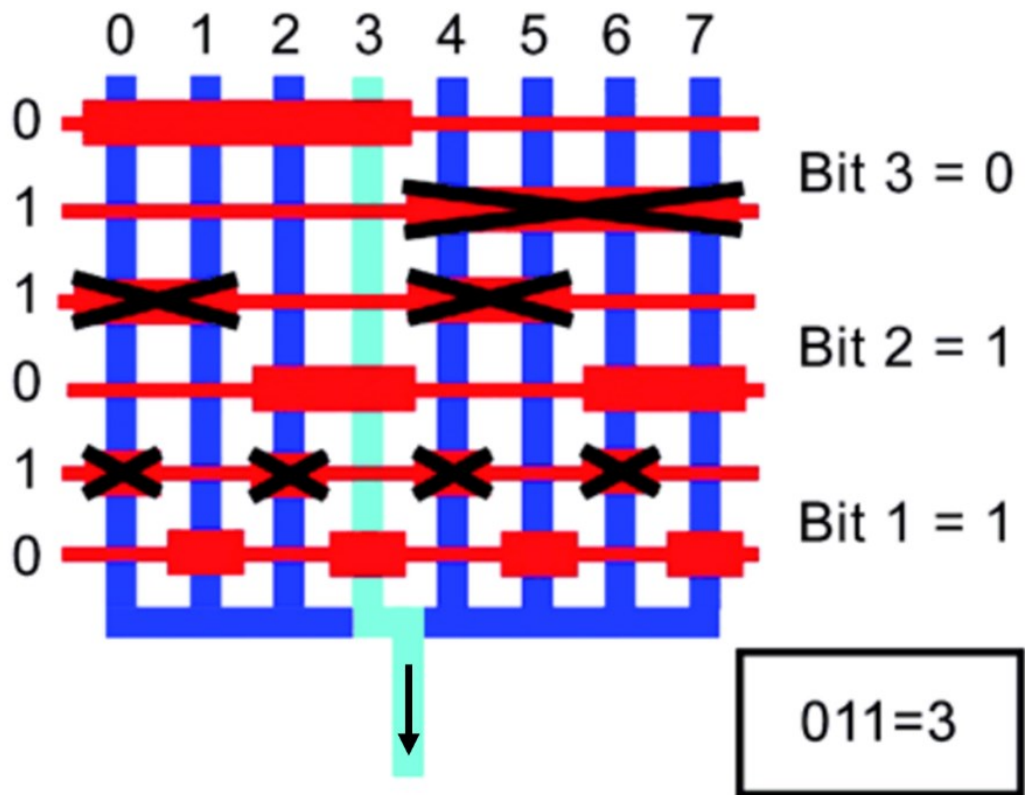
Figure 2-32. A design of microfluidic multiplexor [125].
Blue and cyan lines are channels for input media, red lines are pressure driven control lines with valves (wide parts) formed at the intersection with media channels. By controlling the pressure in the control lines (red), valves will block some of the channels and only allow media from certain source flow into the cell chamber.

Chapter 3:   Model-Based Optimal Experimental Design with Parameter Clustering

Previous case studies show that model-based optimal experimental design (MBOED) guided by the Fisher information matrix could optimise the biological experiments to increase the accuracy of model calibration. However, the time cost of the optimisation procedure rises dramatically with the increasing sizes and complexities of the models that need to calibrate. This chapter addresses this problem by introducing the automatic parameter clustering analysis and visualisation (PCAV) and narrowing down the focused parameters in each sub-optimisation task. Two clustering-based MBOED approaches are compared to previous methods and validated with in-silico benchmark experiments on the orthogonal promoter model. The analysis shows that both of the proposed clustering-based OED le to significantly more accurate parameter estimations than the traditional off-line OED method.

3.1 Introduction

3.1.1 Experimental Design as an Optimisation Problem

As introduced in chapter 1, model calibration is a procedure that updates the parameter values of a model based on experimental data so that the error of model predictions caused by parameter uncertainty is minimised [222]. Suppose the experimental data purely reflects the output of a model with a unique and identifiable parameter set, and there is no noise. In that case, the model calibration should capture this parameter set and give the model predictions exactly equal to observations. However, it is impossible to achieve such ideal results in practice, and the noisy data leads to estimations with less accuracy.

In practice, the effect of noise is usually corrected by carrying out multiple experiments that follow the law of large numbers [223]. Generally speaking, with more experimental data, the estimation of the identifiable parameters would be more accurate (narrower confidence interval). However, practically it is impossible to expect an infinite amount of data for calibrating a given model. Moreover, it has been previously shown that not all the experiments are equally informative for model calibration [224-227]. Therefore, optimising the experimental designs before carrying them out in wet-lab is a procedure that could improve the accuracy of parameter estimation with the same amount of experimental data [50-52].

The design of an experiment involves details of multiple aspects, such as the selection of stimuli and observations, stimulation patterns, the detailed settings of experimental equipment, sampling time, and the data processing methods. Among all of these aspects, the optimisation of the stimulation patterns, known as the optimal input design (OID), is one of the most widely-applied OED branches [228-230]. OID is commonly adopted because it does not involve any significant modifications to the experimental equipment, which could be costly; the only change is the input values of the experiment. Therefore, this chapter aims at improving the detailed design of experiments by implementing OID, with the MPLac model from the previous chapter used as a benchmark.

With a differential equation model, the optimal input design problem can be formulated as follows, in an abstract expression as follows:

$$\arg \max_{U} \phi\left(U, \hat{y}_0, \hat{\theta}\right)$$

Subject to:
$$y(t_0) = \hat{y}_0$$
$$\frac{dy}{dt}(y, t) = f_{Model}\left(y, \hat{\theta}, U(t)\right)$$
$$U_{min} \leq U(t) \leq U_{max}$$

where ϕ is a scaler (single-objective optimisation) or vector (multi-objective optimisation) that quantifies the estimation accuracy corresponding to an input

design U; $\hat{y}_0$ is the estimation of the initial states of the network; $\hat{\theta}$ is the best estimation of the parameter values before carrying out more experiments; f_{Model} is the mathematical model that gives the prediction of time derivatives of the states according to the current network states, parameter values, and input values; U_{min} and U_{max} are the boundaries for the input value.
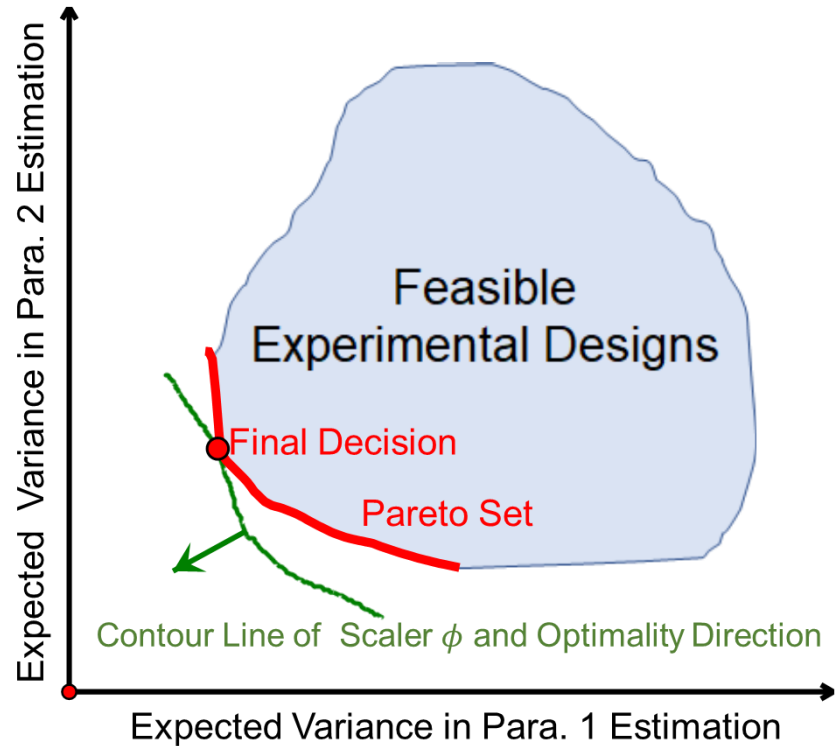
Figure 3-1. Optimal experimental design for two model parameters.

The choice of optimality ϕ is an interesting and complex topic, particularly when there are multiple parameters to fit. If there is only one parameter, there is a rather direct option: the inversion of expected variance of the estimation. However, for the cases with multiple parameters to fit, because of the physical meanings of the parameters, the input design U that minimise the variance of one parameter is usually different to the best design for another parameter. In other words, there would be trade-offs between the accuracies of parameter estimations. This situation leads to a Pareto set of experiments (Figure 1-1) [175], that for every experiment in this set, there is no experiment outcompeting it in the accuracy of every parameter. Because wet-lab experiments cannot carry out every Pareto-optimal design in parallels like a quantum computer and only one or a few experiments can be carried out. Usually, the chosen

optimality ϕ is a scaler criterion that quantifies the overall precision so that one design of the Pareto set would be considered the "optimal" one. The selections of this criterion ϕ would be discussed in the following section.

3.1.2 Current OED Optimality and the Limits

When introducing the optimality for model calibration, it is necessary to mention the Fisher information matrix (FIM). It is one of the most commonly adopted mathematical tools for defining informativeness metrics [176]. This matrix is the Hessian matrix of the log-likelihood function of the observable, with respect to variances in parameter values (Eq. 3-1) [231]:

$$FIM(\theta, U) \equiv E\left(\frac{\partial logl(\theta, U)}{\partial \theta} \cdot \frac{\partial logl(\theta, U)}{\partial \theta^T}\bigg|\theta\right)$$
Eq. 3-1

where, θ is the parameter vector, and $log\ell$ is the log-likelihood function of the observable.

This matrix is widely adopted because of the Cramér-Rao inequality (Eq. 3-2), that the inverse of FIM gives the lower variance bound of the estimation of parameters [176]:

$$Var_\theta(U) \geq FIM(\theta, U)^{-1}$$
Eq. 3-2

The evaluation of the FIM directly from its definition can be a time-consuming task because it involves calculating the estimations corresponds to the *a-priori* estimation distribution of all the parameters. In practice, if the distribution of observable follows a multivariate normal distribution (which is a commonly-adopted assumption), the FIM can be formulated into another form that is easier to calculate (Eq. 3-3) [176]:

$$FIM_{i,j}(\theta, U) = \left(\frac{\partial y(\theta, U)}{\partial \theta_i}\right)^T \sigma^{-1} \left(\frac{\partial y(\theta, U)}{\partial \theta_j}\right)$$
Eq. 3-3

where, σ is the covariance matrix of the observable.

In practice, there are multiple parameters to calibrate most of the time. The endogenous correlations between these parameters affect the estimation accuracy, and this effect will reflect on the off-diagonal elements and the Eigenvalues of the FIM. Figure 3-2 illustrates how the FIM's Eigenvalues describe the estimation accuracy.
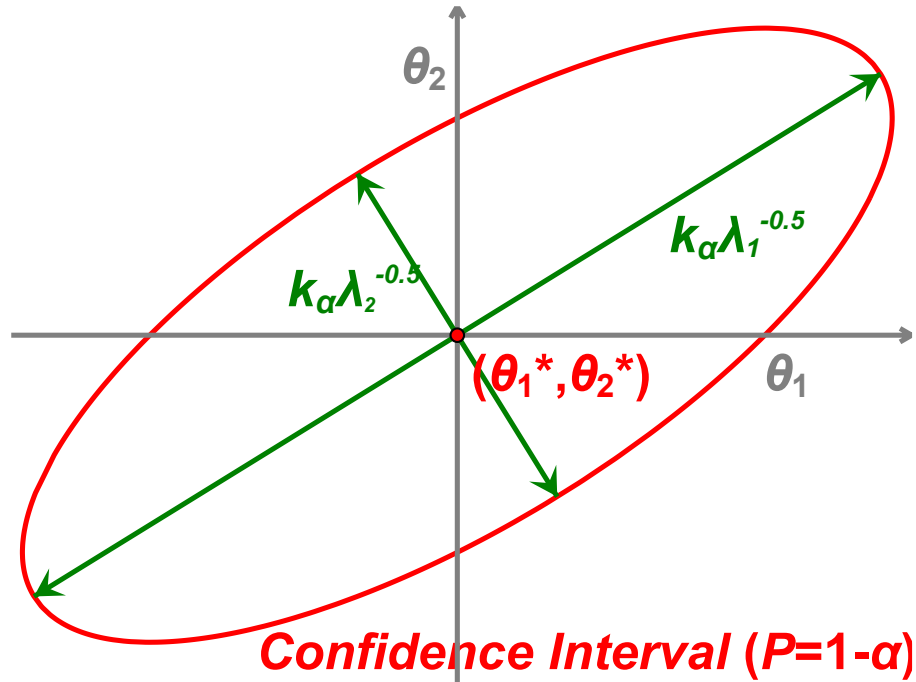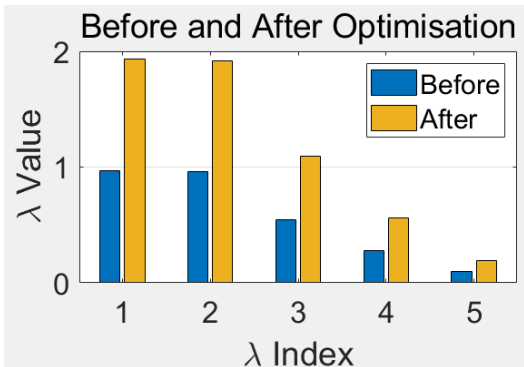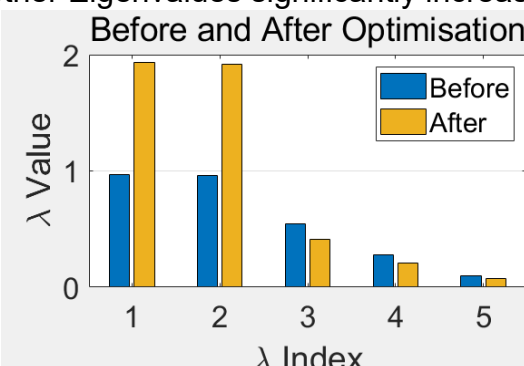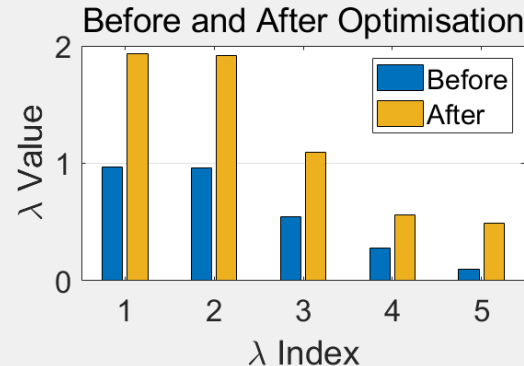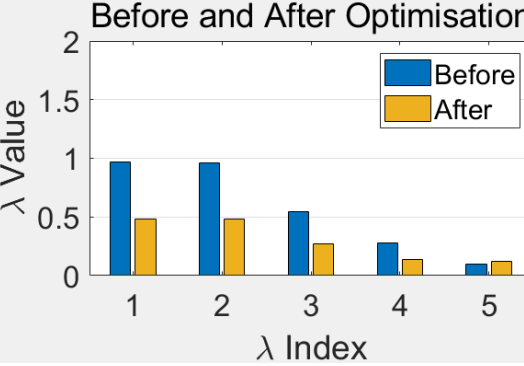
Figure 3-2. Illustration of how Eigenvalues λ reflect PE accuracy. This figure shows an example of estimating two parameters θ_1 and θ_2. According to the current best knowledge about the experimental noise and the model to calibrate, the parameter estimation results base on experiments is expected to fall in the red ellipse (confidence interval) with confidence level α, k_α is a value depends on the confidence level, (θ_1^*, θ_2^*) is the current best estimate of the parameter values.

Minimising the estimation variances for all the parameters (mathematically equivalent to maximising all the Eigenvalues of the FIM), as mentioned in section 3.1.1, will lead to a Pareto set. The most common solution to achieve one unique solution instead of a solution set, is to define an optimality criterion (scaler function ϕ in section 3.1.1) that quantifies the overall accuracy for all the parameters for fitting. A few criteria were adopted in previous studies for FIM-based OED [176, 232, 233]. However, as shown in Table 3-1, these criteria have their limits under non-ideal scenarios. There are also some

variants of these criteria that do not change the judgements about sub-ideal scenarios [234], such as Q-optimality, which introduces the quadratic terms of the FIM [235, 236], and P-optimality, which apply different weights or probabilities to the sample points [233, 237]. With the increasing size and complexity of synthetic biological networks, this limit becomes more significant and inevitable [238].

Table 3-1. OED criterion definitions and their ideal/non-ideal scenarios with illustrative examples.

Criteria	Definition	Ideal Scenario	Non-Ideal Scenario
D-Optimality	$\arg\max \det(FIM)$ $\equiv \arg\max \prod_i \lambda_i$	All the Eigenvalues increase after optimisation.	Some of Eigenvalues are still small or even smaller while the criteria gets better because the other Eigenvalues significantly increase.
A-Optimality	$\arg\max \operatorname{tr}(FIM)$ $\equiv \arg\max \sum_i \lambda_i$		
E-Optimality	$\arg\max \lambda_{min}$	Difference between the minimum and maximum Eigenvalues reduces. "Hopefully", the Eigenvalues generally increase.	The difference between the minimum and maximum Eigenvalues reduces, but the Eigenvalues generally reduce.
Modified E-Optimality	$\arg\max \dfrac{\lambda_{min}}{\lambda_{max}}$		

3.1.3 Off/On-Line and Clustered OED

Depending on how the FIM-based OED is used in experimental procedures, there are two typical OED approaches in previous studies: off-line and on-line OED (left and middle of Figure 3-3).

Off-line OED is one of the most classic approaches to adopting the optimal experimental design. In this approach, all the experimental designs are optimised and finalised before carrying them out.

The disadvantage of off-line OED is that when the initial parameter estimations are not close enough to the true parameter set, the optimised designs could be significantly different from the design corresponding to the model with the true parameter values. Thus, the informativeness will also be less than the expectation. To address this problem, on-line OED separates the experiment into several sub-experiments. Between these sub-experiments, the model would be calibrated and the experimental design of the next experiment will be optimised based on the model with updated parameter values. With this approach, since the second sub-experiment, OED would infer based on a model that is closer to the actual experiment object compared to the off-line OED, so the optimised experiments shall be more informative. The benefit of on-line OED has been proved with in-silico experiments on the same benchmark (the orthogonal inducible promoter) in previous works [27].

In the current off-line and on-line OED approaches, all the sub-experiments are designed to optimise the accuracy of all the interested parameters. It will lead to the problem of the accuracy trade-off between these parameters, which is introduced in section 3.1.2. In this thesis, a new approach (cluster-based OED) is introduced to address this problem. As shown in Figure 3-3 right, before optimising and starting the sub-experiments, a process of parameter clustering would be carried out. It can base on in-silico experiments with either random stimuli (most informative stimuli for linear models) or optimised

experiments with small evaluation numbers (shallow-search OED). The aim is to find suitable parameters to be grouped in the same cluster and be optimised in one of the sub-experiments. Then a deep-search OED would be carried out for each sub-experiment to finalise the experimental design. The detailed algorithms for cluster-based OED would be introduced in the method section of this chapter. The comparison of these three approaches for implementing OED in model calibration experiments is shown in Table 3-2. In the clustered OED approach, the concept of the epoch is introduced as inspired by the term in machine learning [239, 240]. An epoch refers to a set of sub-experiments that all the parameters are focused on once and once in these experiments.

Table 3-2. Comparison of off/on-line OED and cluster-based OED.

Types of OED	Highlights	Applicable Situations
Off-Line OED	1. "Once for all" OED procedure. No need to wait between experiments; 2. Easier to arrange the experiments with fixed designs; 3. Support parallel experiments.	1. Cannot run OED during the experiment (not allowed by the equipment or time schedule); 2. Relatively accurate initial guess with negligible accuracy trade-off.
On-Line OED	1. Improve the experimental designs after every experiment based on observed data; 2. Support iterative experiments.	1. Can run OED during the experiments; 2. Models with inaccurate initial parameter guesses; 3. The accuracy trade-off between parameters is not very significant.
Clustered OED	1. Improve the experimental designs after every epoch based on observed data (an epoch corresponds to a set of sub-experiments that each parameter are focused in one and only one of these sub-experiments); 2. Support iterative and parallel experiments; 3. Provide parameter clustering information; 4. Balance the experimental cost and the trade-off between parameter accuracies.	1. Models with inaccurate initial parameter guesses; 2. Models with a large number of parameters to calibrate; 3. The accuracy trade-off between parameters is significant.
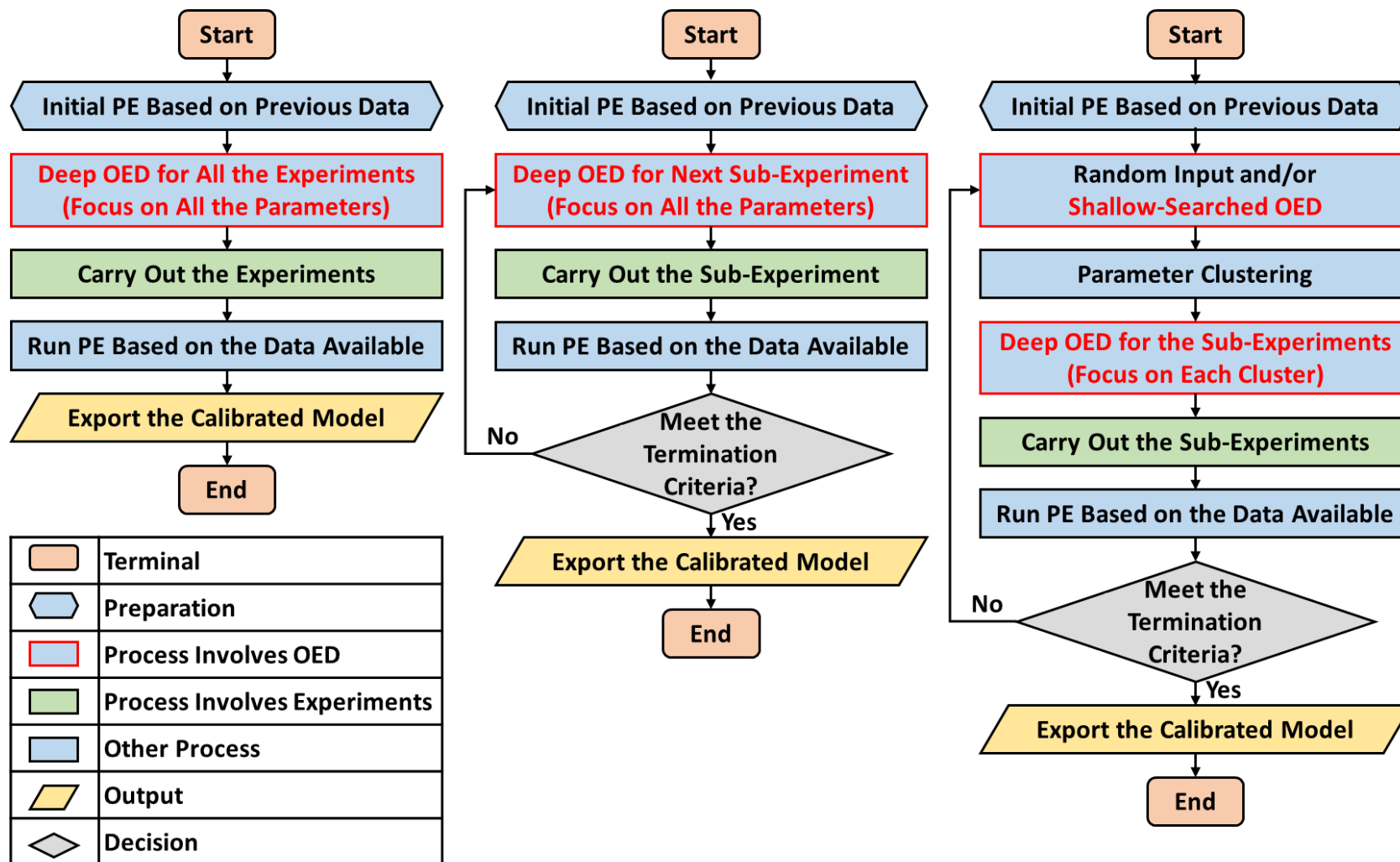
Figure 3-3. Flow charts of the off-line OED (left), on-line OED (middle), and cluster-based OED (right).

It may worth mentioning that there have been many studies adopted parameter clustering to guide the model calibration [167, 241-244]. However, these clustering approaches are designed to simplify the OED by reducing the number of parameters to fit accordingly, which are not suitable for the case when the interested parameters are already decided because of the aim of researches.

3.1.4 Application of Parameter Clustering in Previous Study

There are two methods for parameter clustering in this chapter: sensitivity-based clustering and FIM-based clustering. Some previous studies adopted the sensitivity-based approach, but few studies investigated the FIM-based clustering.

A very representative example showing the power of parameter clustering is a model calibration study by Chu et al. [238]. In this study, a signal transduction pathway for hepatocytes stimulated by Interleukin-6 is modelled and calibrated (Figure 3-4). It is a model established based on three sub-models [245-247] (shown in different colours) and involves 66 nonlinear ordinary differential equations, and 115 parameters (including 70 fixed parameters).

In Chu's study, the experimental data from previous flow-cytometry experiments [245, 248] is informative to calibration 8 parameters (the rank of FIM is 8). It means that they need to choose the 8 parameters from 45 (215,553,195 possible combinations) to calibrate to achieve the best fitting result. By applying the sensitivity-based clustering (Figure 3-5), they find that the 45 parameters form 11 clusters, and the parameters from the same cluster have a similar sensitivity pattern. It means that if two calibrated parameters are chosen from the same cluster, the calibration will lose a rank and leads to wildly inaccurate parameter estimations. Base on the clustering, the parameter selection is simplified to selecting 8 parameters from 11 (combination number drops from 215,553,195 to 165), which is much easier to solve.
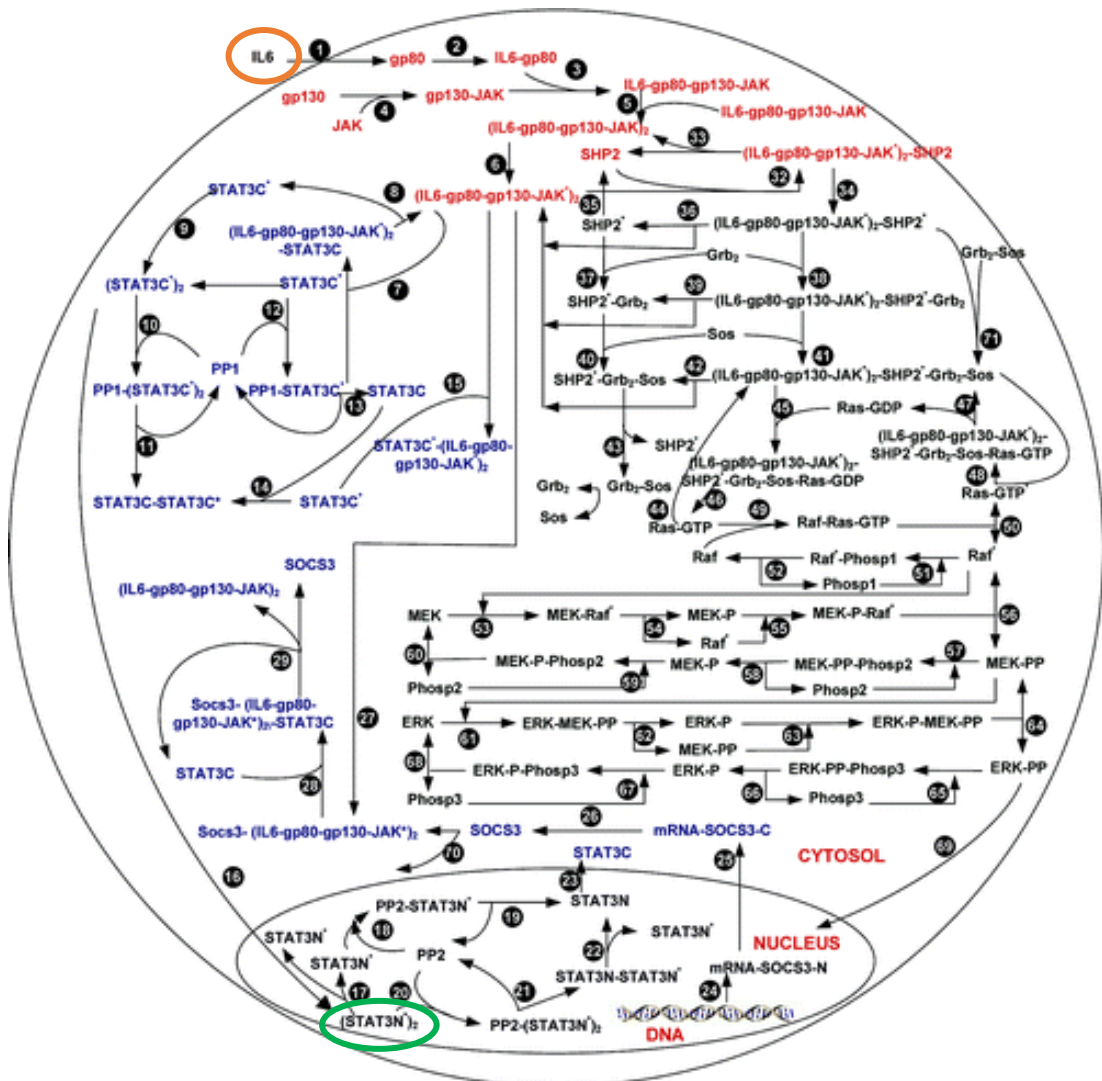
Figure 3-4. Pathway modelled in Chu et al.'s study [238].
The input Interleukin-6 ins shown in orange, and the observable (the molecule that is fluorescence-labelled and measured in the experiments) is shown in green.
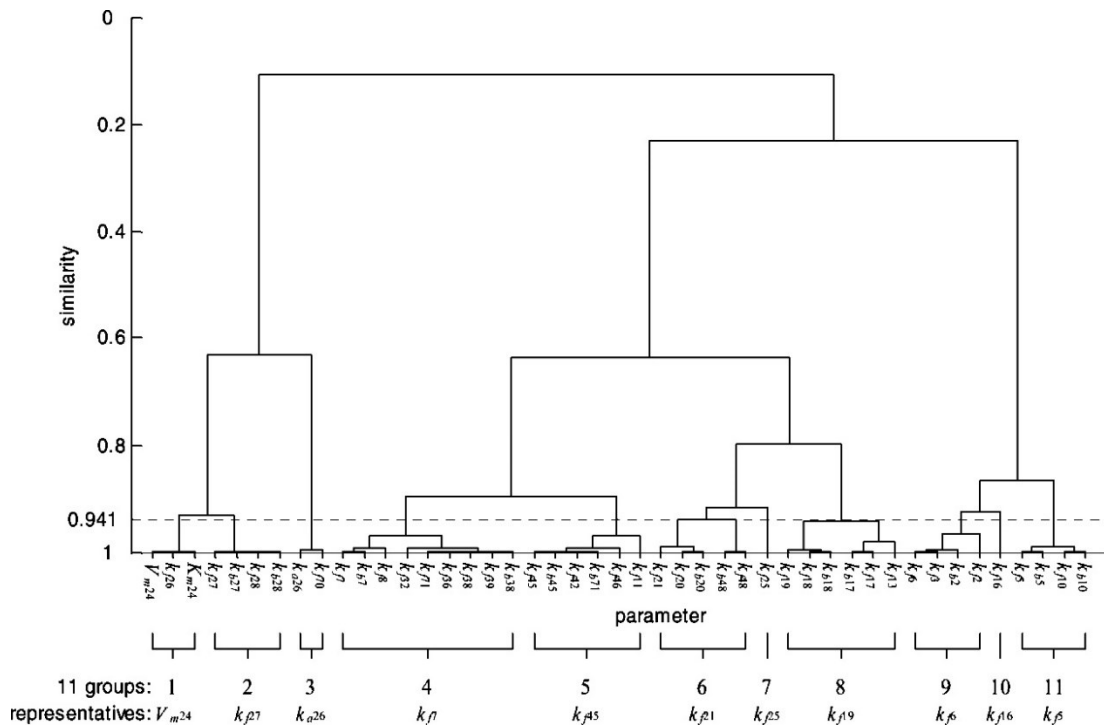
Figure 3-5. Hierarchical clustering base on cosine distances between the sensitivity vectors of 45 parameters [238].

3.1.5 Novelty of the Chapter

Previous works in optimal experimental design (OED) and parameter clustering has shown impressive achievements in related biological studies [238] that OED can help the researcher find the most efficient stimuli patterns for model calibration. Moreover, parameter clustering can find the parameters sharing similar identifiability patterns, leading to identifiability problems. These achievements point to a question: can the model calibration experiments be improved by applying the OEDs based on parameter clusters?

The work of this chapter is carried out to answer this question. Two clustered-OED approaches are proposed and compared to traditional experimental design (random stimuli), and two typical OED approaches. Comparing the PE results based on these experiments will show how significant clustered-OED can improve the experiment's informativeness.

3.2 Methods

3.2.1 Deep/Shallow-Search OED

Almost all the OED problems in biology are non-linear [249-251] and could not guarantee to find the globally optimal solution within a limited number of evaluations [101, 252-254]. The most apparent difference between deep/shallow search OED, is on the search depth, i.e., the number of evaluations before finalising the design. As mentioned previously, in most of the MBOED cases, it is theoretically impossible to guarantee the global-optimality of the best solution within a limited time. Meanwhile, it is evident that for most global searching algorithms such as eSS and DE, the longer time or more evaluations are involved, the more likely it is to obtain better solutions or the globally optimal solution.

Moreover, the deep/shallow search OED is also different in the purpose of the search: The deep-search OED is searching for a final experimental design that can be directly used in an experiment. In contrast, the shallow-search OED reveals the OED task's inner properties, i.e., which subsets of the parameters are suitable to be optimised together for the informativeness in one experiment. Therefore, the deep-search OED needs to be deep enough to find the optimal solution with relatively high accuracy. In contrast, shallow-search OED does not need to provide accurate solutions but need to find which of the parameter subsets can be optimised together within a small number of evaluations.

It is also important to point out that the convergence speed of OED depends on the initial guess and the first few searching iterations in practice [52, 89, 255]. Suppose some of the early samples in a searching trial reach the region around the globally optimal solution. In that case, the trial is much more likely to find the globally optimal solution than the other trials. Oppositely, suppose a trial misses the region around globally optimal solution at the early search stage. In that case, it is more likely to get trapped in a locally optimal point and spend the rest of the calculations searching for the exact values of this locally

optimal solution. From this point of view, shallow-search OEDs also provide information about "the OED of which parameter cluster is a simpler problem and is more likely to be solved with-in limited number of evaluations" (Figure 3-6).

In summary, the differences between deep/shallow-search OED can be expressed in Table 3-3.

Table 3-3. Comparison of deep/shallow-search OED.

Aspects	Deep-Search OED	Shallow-Search OED
Purpose	Find an exact experimental design that can be finally used in an experiment	Obtain evidence for anslysing the inner properties of the OED task
Maximum Calculating Time	Longer (4 h in practice)	Shorter (2 min in practice)
Maximum Number Of Evaluations	More (50,000 in practice)	Less (500 in practice)
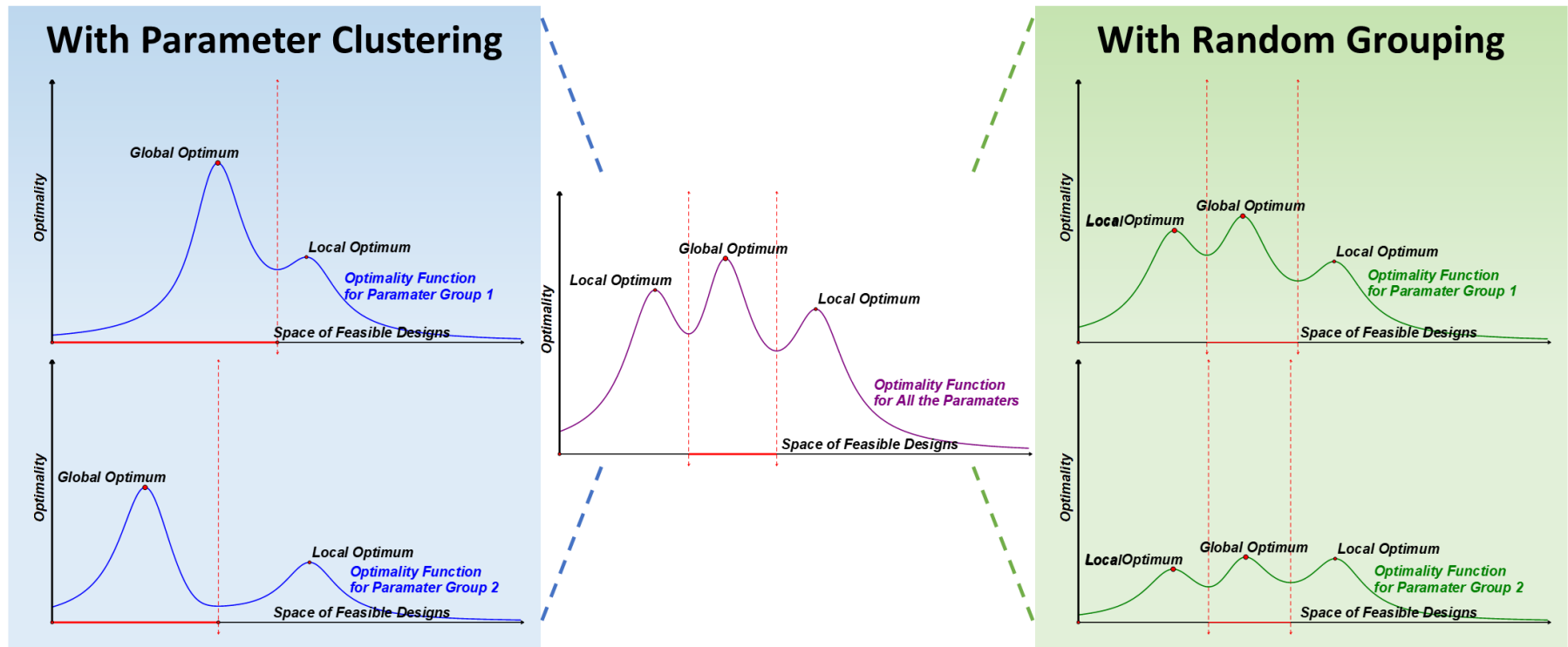
Figure 3-6. Illustration of the benefit of parameter clustering for optimal experimental design (OED).
In this plot, the space of feasible experimental design is simplified to show in one dimension (horizontal axis). The OED solver must have sample points in the red regions to approach the global optimum. In other words, a larger red region means the solver is more likely to find the globally optimal solution (faster with higher robustness). Compared to the OED for all the parameters (middle), parameter clustering (left) can find the parameters suitable to be optimised together and disentangle the optimality trade-offs between clusters, which cannot be achieved by random parameter grouping. As suggested by the red regions, parameter clustering also leads to more robust OED.

3.2.2 Parameter Clustering based on Sensitivity Information

● Sensitivity Information

This approach is based on the sensitivity of the model-predicted observable values corresponding to changes in each parameter value. The aim is to find the parameters which share similar stimuli-sensitivity function.

There are three commonly adopted metrics for describing the sensitivity information in experiments [174]: mean squared sensitivity (d_{msqr}), mean absolute sensitivity (d_{mabs}), and mean sensitivity (d_{mean}). Their definitions are shown in Eq. 3-4:

$$
\begin{aligned}
d_{msqr,i}^{j} &= \sqrt{\frac{1}{N_s^j} \sum_{s=1}^{N_s^j} \left(\frac{dy_s^j}{d\theta_i}\right)^2} \\
d_{mabs,i}^{j} &= \frac{1}{N_s^j} \sum_{s=1}^{N_s^j} \left|\frac{dy_s^j}{d\theta_i}\right| \\
d_{mean,i}^{j} &= \frac{1}{N_s^j} \sum_{s=1}^{N_s^j} \frac{dy_s^j}{d\theta_i}
\end{aligned}
\qquad (3-4)
$$

In Eq. 3-3, j is the experiment index, i is the parameter index, N_s^j is the number of sampling times, s is the index of sampling time, and $\frac{dy_s^j}{d\theta_i}$ is the derivative of the model prediction of the observable y at the specified sampling index s corresponds to a small value change in the corresponded parameter θ_i.

Both d_{msqr} and d_{mabs} always give non-negative values, and higher magnitudes represent the more sensitive observable corresponding to the changes in the model parameter. The difference is that d_{msqr} gives more weight to the more sensitive observations, whereas d_{mabs} gives even weight to all the sampled observations. Unlike the other two options, d_{mean} tells the

magnitude of sensitivity by its absolute value and shows whether the observable changes in the same direction with the parameter by its sign (position means the observable increases with a higher parameter value, and *vice versa.*).

d_{msqr} is chosen in this project because the direction of change does not matter the informativeness, and d_{msqr} gives more weights to the sample points with higher sensitivity levels.

● Metric of Distances between Sensitivity Vectors

Thus, the vectors of sensitivity for model parameter θ_i in N_j number of experiments can be obtained in the form as $V_i = \left[d^1_{msqr,i}, d^2_{msqr,i}, ..., d^{N_j}_{msqr,i} \right]$. Considering two parameters θ_α & θ_β and corresponding sensitivity vectors V_α and V_β, the level of dissimilarity can be quantified by cosine distance (Eq. 3-5) [256]:

$$d_{cosine}(V_\alpha, V_\beta) = 1 - \frac{V_\alpha V_\beta^T}{\sqrt{V_\alpha V_\alpha^T V_\beta V_\beta^T}} \qquad (3-5)$$

The relationship between the cosine distance and correlation can be expressed as Eq.3-6:

$$d_{cosine}(V_\alpha, V_\beta) = 1 - corr([V_\alpha, -V_\alpha], [V_\beta, -V_\beta]) \qquad (3-6)$$

Eq.3-6 also shows that the scaling of the sensitivity (or the units for the parameters) does not change the cosine distance. Most of the other distance metrics (such as Euclidean distance, city block distance, and Minkowski distance) do not have such property. They will give non-zero distances for two vectors in the same direction but with different magnitudes.

● Selection of Clustering Algorithm

There are multiple clustering algorithms for the parameter clustering task in this thesis [257, 258]. The three groups of algorithms that are most commonly adopted are k-means [259, 260], DBSCAN [261], and hierarchical [262]. Table 3-4 compares some of the related features of these methods.

Table 3-4. Comparison of three clustering methods [257, 263].

Algorithms	Specified Cluster Number	Outlier Detection	Selected
K-Means	Required	No	No
DBSCAN	Not required	Yes	No
Hierarchical	Can work with or without a specified cluster number	No	Yes

For this project, hierarchical clustering is in use because the other two are not ideal considering the purpose of the clustering and the physical meaning of sensitivity vectors. K-means is not chosen because practice in this task founds that the clustering results depends on the initial state and not as robust as the hierarchical algorithm [257, 263]. This is also supported by a previous parameter clustering case study with a biological model by Chu et al. [238]. DBSCAN is not chosen either because it considers that some of the vectors are misleading outliers that should not be included in any cluster. However, in this project, each of the objects for clustering corresponds to one parameter. It means that none of the objects should be considered outliers, which is against the primary hypothesis for DBSCAN clustering [261]. Moreover, a significant advantage of the hierarchical algorithm is that it also provides the connections between parameters within each cluster, and this information does not change with the cluster number. The corresponding dendrogram for visualising the results of hierarchical algorithm Figure 2-13(Figure 3-11) can show this information, which cannot be obtained in the other two algorithm families.

Hierarchical clustering has many variations in the methods calculating the distances between sub-clusters [257, 264]. As shown in Table 3-5, there are seven methods available in the build-in function in MATLAB, and they are also the most commonly used ones. In two previous researches [257, 265], UPGMA is selected because of its robustness and compatibility with non-Euclidian distances. The other methods are also computed with the PLac model based on in-silico experiments of both random and shallow-search OED. All the algorithms except the "single" method give the same clustering result.

Table 3-5. Seven MATLAB build-in methods for computing cluster distances.

Method	Description	Euclidean distances only	Selected
Average (UPGMA)	Unweighted average distance between cluster elements	No	**Yes**
Weighted (WPGMA)	Weighted average distance with a recursive distance definition	No	**No**
Centroid (UPGMC)	Distance between centroids of clusters	Yes	**No**
Median (WPGMC)	Distance between weighted centres of mass for clusters	Yes	**No**
Ward	Inner squared distance (minimize the variance in clusters)	Yes	**No**
Complete	Farthest distance between cluster elements	No	**No**
Single	Shortest distance between cluster elements	No	**No**

● Optimal Clustering Number

The hierarchical clustering algorithm generates a tree of clustered objects. By applying different distance thresholds, this tree can easily generate the cluster results corresponding to any feasible clustering numbers (Figure 3-5 is an example). However, for the study in this chapter, the physical meaning of this clustering results decides that there is an optimal cluster number. When the clustering number is too low, there would be too many parameters in one cluster, which cannot solve the key problem of the trade-off between estimation accuracies of parameters. When the clustering number is too high,

the algorithm will focus on tiny and detailed differences between parameters. This will lead to more sub-experiments and ignore some of the critical connections between related parameters, which leads to poor-identifiability.

Four commonly adopted criteria can be used to find the optimal clustering number (Table 3-6): gap criterion, silhouette values, Calinski-Harabasz index, and Davies-Bouldin index. Only the Gap and Silhouette measures are used in this study because only these two can work with non-Euclidean distances (e.g., the cosine distance in this project).

Table 3-6. Four commonly adopted criteria to evaluate the clustering number.

Method	Description	Euclidean distances only
Gap	Search for the "elbow" location of the cluster number-error measurement function.	**No**
Silhouette	Measure of how similar a point is to the other points in the same cluster, compared to points in other clusters.	**No**
Calinski-Harabasz	The ratio of between-cluster variance vs. within-cluster variance, also known as variance ratio criterion (VRC)	**Yes**
Davies-Bouldin	Ratio of within-cluster and between-cluster distances.	**Yes**

3.2.3 Parameter Clustering base on Fisher Information Matrices

This approach is based on the Fisher information matrices (FIMs) of the experiments. Compared to sensitivity vectors, FIMs provide more information about the accuracy for estimating multiple parameters [87, 232, 234]. The clustering procedure is equivalent to solving a non-linear optimisation problem. The task is looking for the optimal clustering result so that the patterns of informativeness for fitting parameter clusters for different experimental designs have maximally differed. As illustrated in Figure 3-6, if one parameter element is included in a wrong cluster, it will "contaminate" the informativeness pattern

of this cluster and make it more similar to the pattern for the cluster that the element should belong to.

From the description above, the clustering procedure seems to involve complex and repeated evaluations of the FIM for estimating different parameter clusters, but in fact, it can be obtained with a simple calculation that does not need to repeat. For each experiment, a "full" FIM can be calculated for the case of fitting all the parameters. As introduced in section 3.1.2, FIM is an $n_\theta \times n_\theta$ matrix where n_θ is the number of parameters to fit. FIM has an important property that for the case of fitting a subset of parameters with the same experimental design, the new FIM is exactly the corresponding part of the "Full" FIM (Figure 3-7). Therefore, the FIM for all the parameters contains
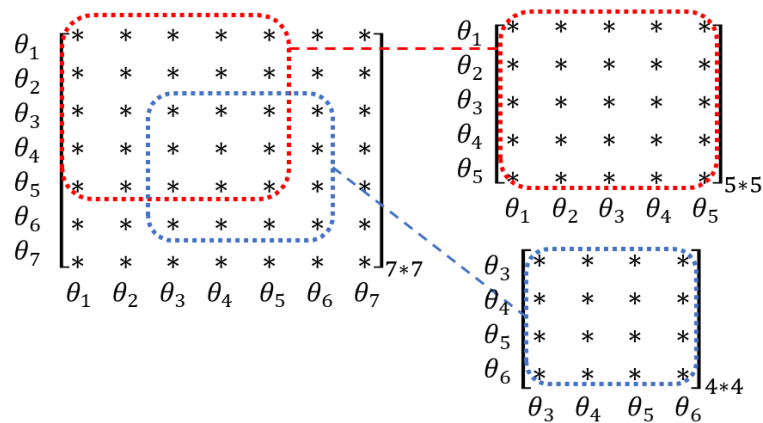
the estimation accuracy information to fit any subgroups of the parameters.

Figure 3-7. The "Full" FIM (left) can easily generate the FIMs for fitting subsets of parameters (right).

For a specific parameter clustering result, the determinant of the FIM (i.e., D-optimality) can be calculated for each parameter subset in every experimental design. This defines the vectors of the informativeness in a similar form as the sensitivity vectors in section 3.2.2: $V_i = \left[D_i^1, D_i^2, \dots, D_i^{N_j} \right]$, where i is the index of parameter cluster (not the index of a parameter), D_i^1 is the D-optimality for fitting the ith parameter cluster with the 1st experimental design, N_j is the total number of experimental designs. The task is to maximise a criterion that quantifies the difference between the informativeness vectors for different parameter clusters by adjusting the parameter clustering.

To the best of my understanding, there are no previous researches that cluster the parameters according to the FIM metrics. The previous section introduced a few commonly used clustering evaluation criteria. However, they are all based on the within-to-between cluster distances, which does not work for the FIM based clustering. There is only one vector for one cluster, so there is no "within-cluster distance" or information about the variance within the cluster. Therefore, in this task, the smallest cosine distance between informativeness vectors is chosen as the criterion, and this value should be as large as possible. A few other options have also been tried: determination coefficients (instead of cosine distance) and averaged between-cluster distance (instead of the smallest distance). The results show that the selected method is more robust and better balances informativeness and clustering complexity.

3.2.4 Comparison of Sensitivity-based and FIM-based Clustering

As introduced in the two previous sections, both of the approaches are designed for finding the best clustering results to guide the deep-search OED according to in-silico experimental data with random stimuli or shallow-search OEDs (the definitions of deep/shallow-search OED are given in Table 3-3). Table 3-7 is a table listing the main differences between these two methods. The computational complexity of these two approaches is not significantly different. In the practice of this case study, there are no fundamental differences in the clustering results, and the difference may be clearer to see with larger and more complex models.

Table 3-7. Comparison of sensitivity/FIM-based clustering.

Aspects	Sensitivity-Based clustering	FIM-Based clustering
Data references	Vectors of output-parameter sensitivities correspond to every experimental design	Fisher information matrix based on all the experimental designs
Results	Hierarchical clustering	Only clustering results

	+ Dendrogram	
Advantages	Also provide connection info between with-in parameters	More detailed consideration for estimating multiple parameters

3.2.5 In-Silico Experiments for Model Inference

The analysis of this chapter is based on in-silico experiments. It is because comparing the informativeness of different experimental designs involves massive experimental data, and it is not practical to carry out such many experiments in the lab in 1-2 years (the in-silico experiments in the study of this chapter corresponds to a total experimental time long than 900 days).

Before carrying out the in-silico experiments, a set of 30 randomised initial parameter estimations are generated. The initial values are within the parameter feasible regions, and follow even distributions in log scale.

The generated estimation set would be used in each OED approach and develop the corresponded experimental design. The cluster number will decide the number of sub-experiments, and the random stimuli and off/on-line OED would carry out with the same number of sub-experiments. Similar to the in-vivo experiments' schedule with random stimuli in chapter 2, each sub-experiment lasts 24 hours with 8 steps (3 hours for each step). The input (IPTG concentration) varies from 0 to 100 μM. The model with the best parameter set from chapter 2 flow-cytometer experiments is used to create the in-silico experiments to make the "true parameter values" known for the comparison.

Finally, the parameter estimation (PE) procedure would be carried out for all the trials.

3.3 Results

3.3.1 Results of Parameter Clustering

As mentioned in the section 3.2.2 and 3.2.3, there are two approaches for parameter clustering: sensitivity-based and FIM-based. Every parameter clustering involves 30 trials with random stimuli and 30 trials with shallow-search OED.

Figure 3-8 compares the observable mean squared sensitivity in experiments with both random stimuli and shallow-search OED with the true parameter value set. Moreover, Wilcoxon rank-sum tests (equivalent to Mann-Whitney U-tests) show that the shallow-search OEDs lead to significantly higher medians of averaged sensitivities in both parameter cluster 1 ($p = 3.16 \times 10^{-5}$) and cluster 2 ($p = 1.70 \times 10^{-3}$).
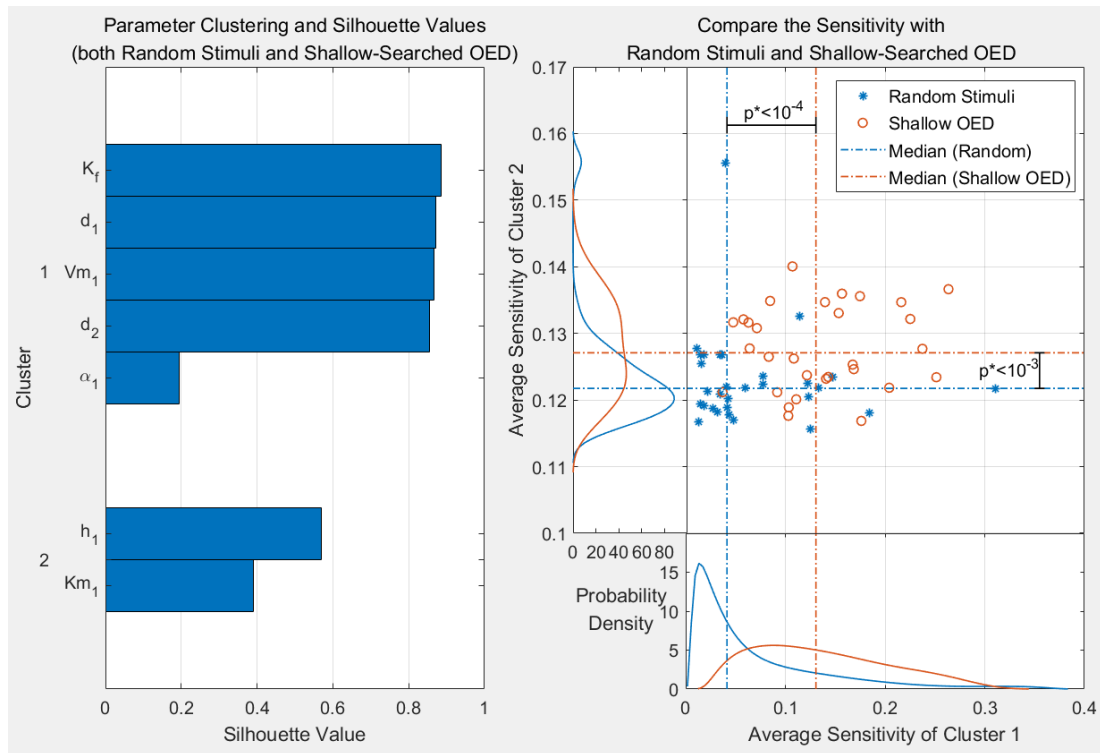
Figure 3-8. Comparing the observable mean squared sensitivity in experiments with both random stimuli and shallow-search OED. The significance levels come from one-sided Mann-Whitney U-tests because the distribution for random-stimuli experiments do not pass the Kolmogorov-Smirnov normality test.

The clustering results from random stimuli + shallow-search OED are shown as follows (cluster 1 in blue, cluster 2 in red, inputs in brown, and observable in green):

$$\frac{d}{dt}[Cit_{mRNA}] = \alpha_1 + Vm_1 \frac{[IPTG]^{h_1}}{[IPTG]^{h_1} + (Km_1)^{h_1}} - d_1[Cit_{mRNA}]$$

$$\frac{d}{dt}[Cit_{foldedP}] = \alpha_2[Cit_{mRNA}] - (d_2 + K_f)[Cit_{foldedP}]$$

$$\frac{d}{dt}[Cit_{fluo}] = K_f[Cit_{mRNA}] - d_2[Cit_{fluo}]$$
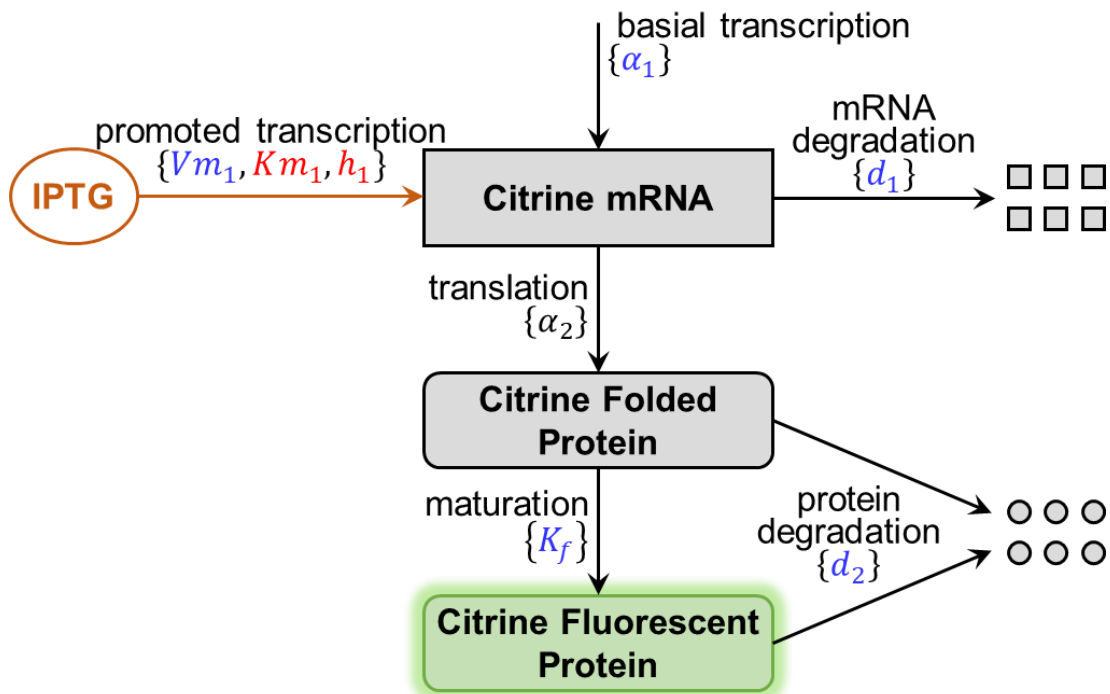
Figure 3-9. Parameter clustering results of random stimuli vs OED. Parameters are coloured by clusters.

From Figure 3-8 and Figure 3-9 it can be seen that:

1. Although there are some overlapping, the shallow-search OEDs generally achieve higher sensitivity levels than the random stimulus.

2. The magnitude of improvement in parameter cluster 1 (the median is almost 200% higher) is significantly larger than in cluster 2 (the median is less than 10% higher).

3. Notably, the parameters from cluster 2 corresponds to the promoted transcription, and have direct and unique contributions to the dose-response curve of this model (Figure 2-9). And random stimuli have already achieved relatively high sensitivity for cluster 2 parameters.

The results suggest that with a relatively small cost in time and calculation (shallow-search OED), the experimental designs can be optimised to achieve significantly higher sensitivity levels for the model parameters.

Since the shallow-search OEDs have significantly higher sensitivity level compared to the designs with random stimuli, the parameter clustering for deep-search OED will only use the data from shallow-search OED as the reference for further parameter clustering. This decision is to avoid the distraction from the informativeness differences between the random stimulus and shallow-search OEDs, which is not directly helpful to investigate the endogenous correlations between parameters.
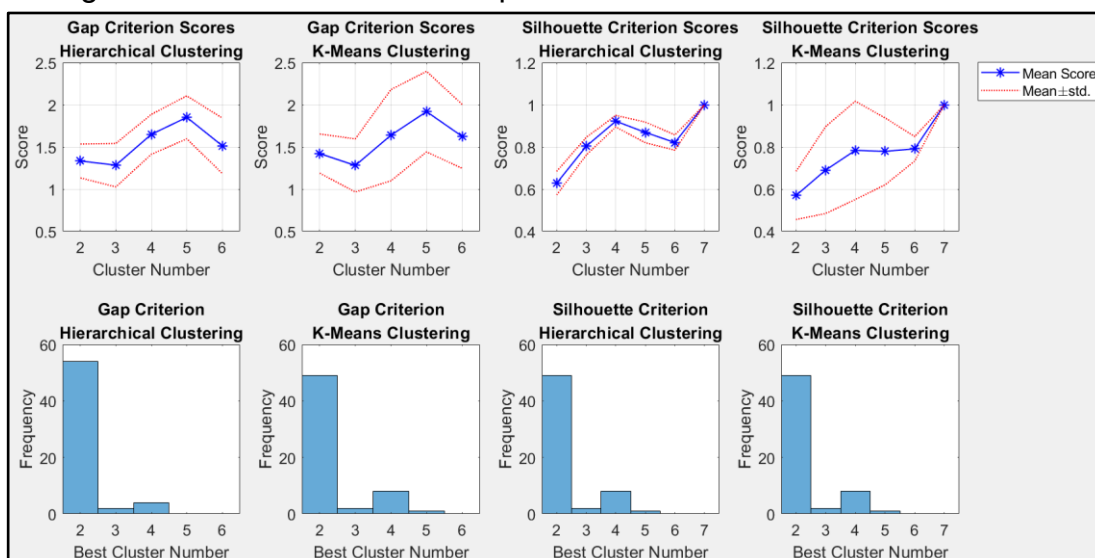
Figure 3-10. Criteria evaluations for the optimal clustering numbers based on shallow-search OEDs with 60 trials.
A number is better if the score is statistically lower than the others, for two cluster number not statistically different, the smaller number is better.

As introduced in the method section, two criterion and two clustering methods are considered to find the best clustering number (Figure 3-10). For robustness, the evaluations repeated for 60 times with bootstrap that chooses 15 experimental designs from 30 shallow-search OEDs. All the four combinations suggest the best clustering number is 2. The pattern of scores seems to mainly depend on the criterion (Gap/Silhouette scores). Moreover, it is not surprising to see that K-means introduces more variance to the evaluation. Unlike the Hierarchical clustering, which gives an affirmatory result for a given data set, the results of K-means clustering also depend on the initial randomised guess (although the used algorithm has already been improved in the robustness against this problem [260]). Therefore, the cluster number is decided to be two as suggested.
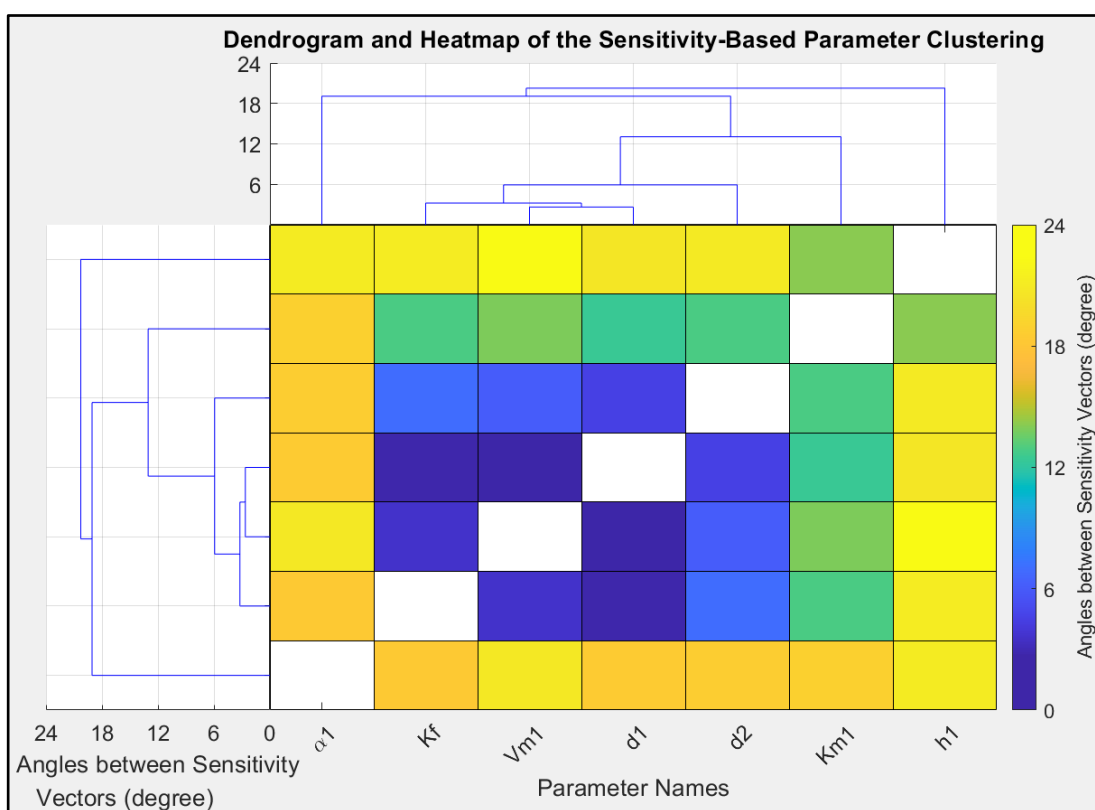
Figure 3-11. Dendrogram and heatmap of the clustering results based on shallow-search OED.

The clustering results from the shallow-search OED only (Figure 3-11). Brighter colours in the heatmap mean that the parameter vector directions, i.e., the sensitivity-stimuli pattern are more different. The clustering result is the

same as the clustering among random and shallow-search OED (cluster 1 in blue, cluster 2 in red, inputs in brown, and observable in green):

$$\frac{d}{dt}[Cit_{mRNA}] = \alpha_1 + Vm_1 \frac{[IPTG]^{h_1}}{[IPTG]^{h_1} + (Km_1)^{h_1}} - d_1[Cit_{mRNA}]$$

$$\frac{d}{dt}[Cit_{foldedP}] = \alpha_2[Cit_{mRNA}] - (d_2 + K_f)[Cit_{foldedP}]$$

$$\frac{d}{dt}[Cit_{fluo}] = K_f[Cit_{mRNA}] - d_2[Cit_{fluo}]$$

The FIM based clustering results are different from the sensitivity-based one. h_1 and Km_1 are separated into two individual clusters, the other parameters are in one common cluster as shown below: (cluster 1 in blue, cluster 2 in red, cluster 3 in grey, inputs in brown, and observable in green):

$$\frac{d}{dt}[Cit_{mRNA}] = \alpha_1 + Vm_1 \frac{[IPTG]^{h_1}}{[IPTG]^{h_1} + (Km_1)^{h_1}} - d_1[Cit_{mRNA}]$$

$$\frac{d}{dt}[Cit_{foldedP}] = \alpha_2[Cit_{mRNA}] - (d_2 + K_f)[Cit_{foldedP}]$$

$$\frac{d}{dt}[Cit_{fluo}] = K_f[Cit_{mRNA}] - d_2[Cit_{fluo}]$$
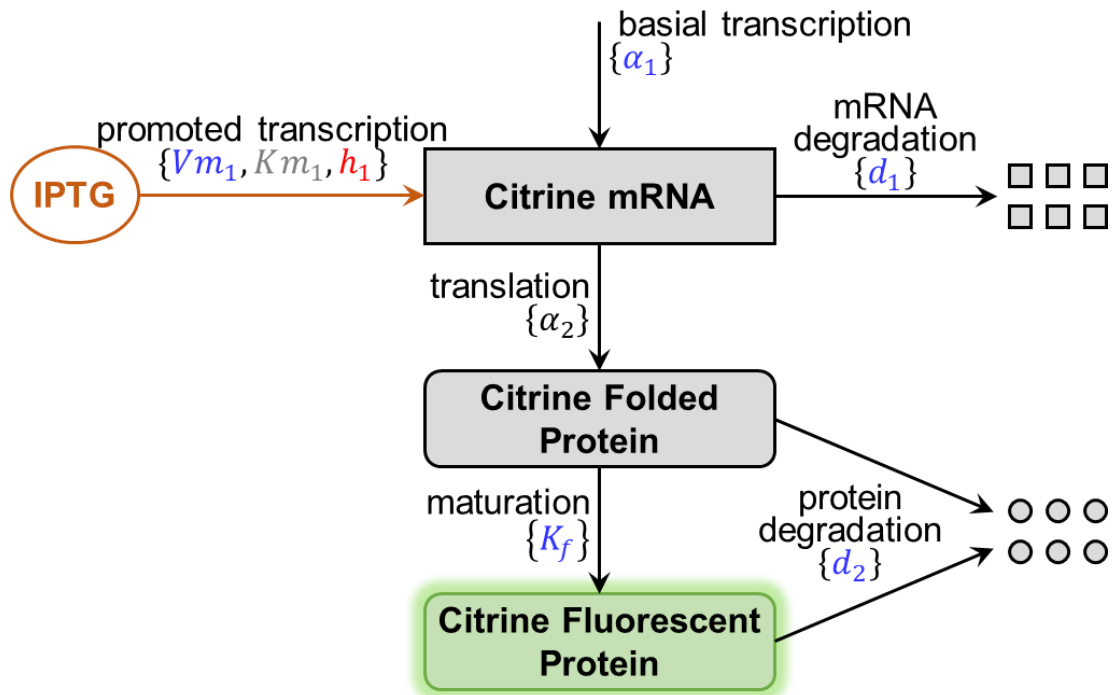
Figure 3-12. Results of FIM-based parameter clustering.
Parameters are coloured by clusters.

3.3.2 Parameter Clustering with Different Parameter Value Sets

Figure 3-13 and Figure 3-14 show the shallow-OED-based clustering results of the 30 randomised initial guesses. The results suggest that the clustering depends on the parameter values, but some common patterns remain the same in most of the trials.
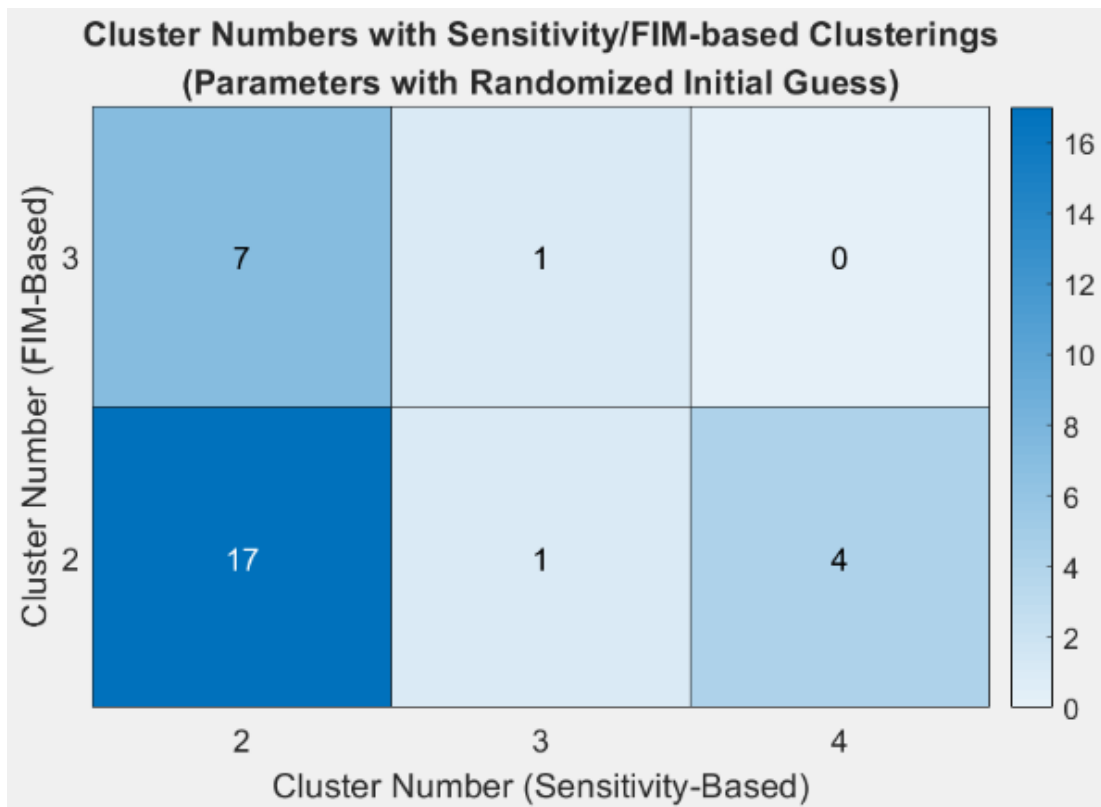
Figure 3-13. Cluster numbers of randomised initial guesses.
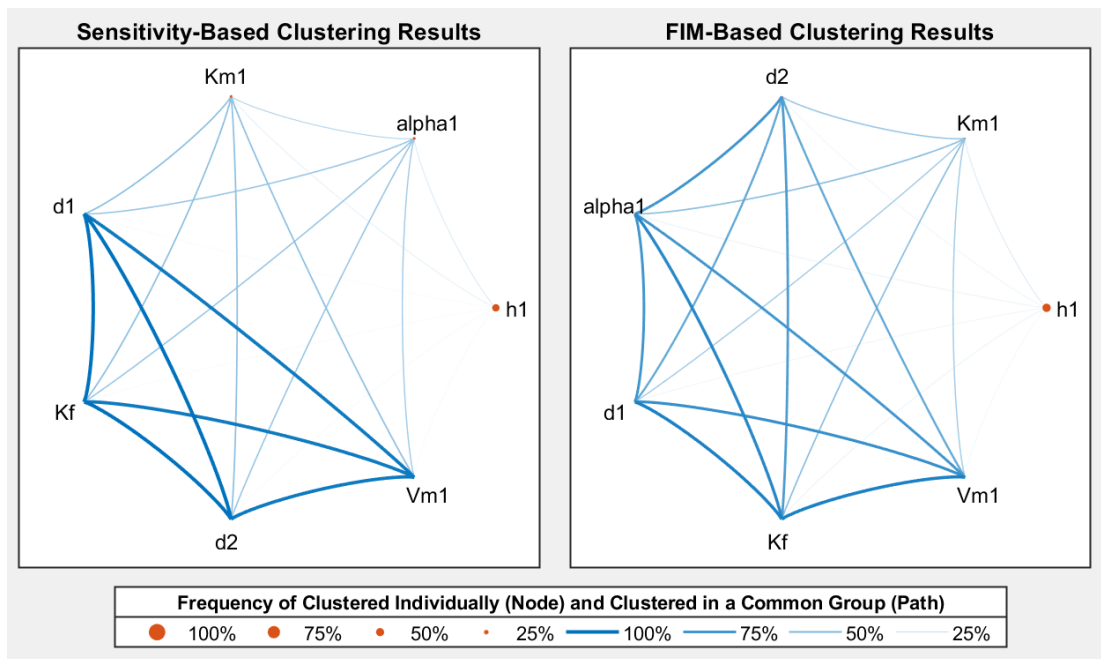
Figure 3-14. Clustering results with randomised initial guesses.

From Figure 3-13 and Figure 3-14, it can be seen that:

1. Both of the sensitivity-based and FIM-based clustering results show their robustness against inaccurate initial guess to an extent. Almost all the connections between parameters remain in 75% of the trials;
2. For sensitivity-based ones, there is a cluster of 4 elements in almost all the results and h_1 tends to be separated into an individual cluster;
3. FIM-based clustering gives a slightly different pattern that 5 parameters are closely related, but these parameters are not always in the same cluster;
4. Summing up the sensitivity-based and FIM-based results: h_1 is in an individual cluster for most of the cases. Km_1 is clustered into a separate cluster in about 25% of the trials. α_1 is clustered as a unique cluster in more than 25% of the sensitivity-based results, but FIM-based analysis suggests that α_1 is more connected with the other parameters.

3.3.3 Clustering with an Unidentifiable Parameter

As mentioned in previous section 2.2.1, there would be an identifiability issue when estimating all the model parameters at the same time. To be precise, α_2 and the overall scale for α_1 and Vm_1 are structurally unidentifiable. However, because of the relatively complex entanglement of these three parameters, the classic pairwise search cannot find this problem [27] (because three parameters must change together with a particular pattern to achieve identical model predictions). And sensitivity-based clustering does not find this problem either (Figure 3-15 and Figure 3-16) because this approach works with cosine distances between sensitivity vectors for each parameter, which can be considered as a pairwise method. On the other hand, the FIM-based method considers the overall accuracy for each parameter cluster and this approach shows the potential of solving this type of problem. The FIM-based clustering always put the unidentifiable parameters together no matter what values they have.
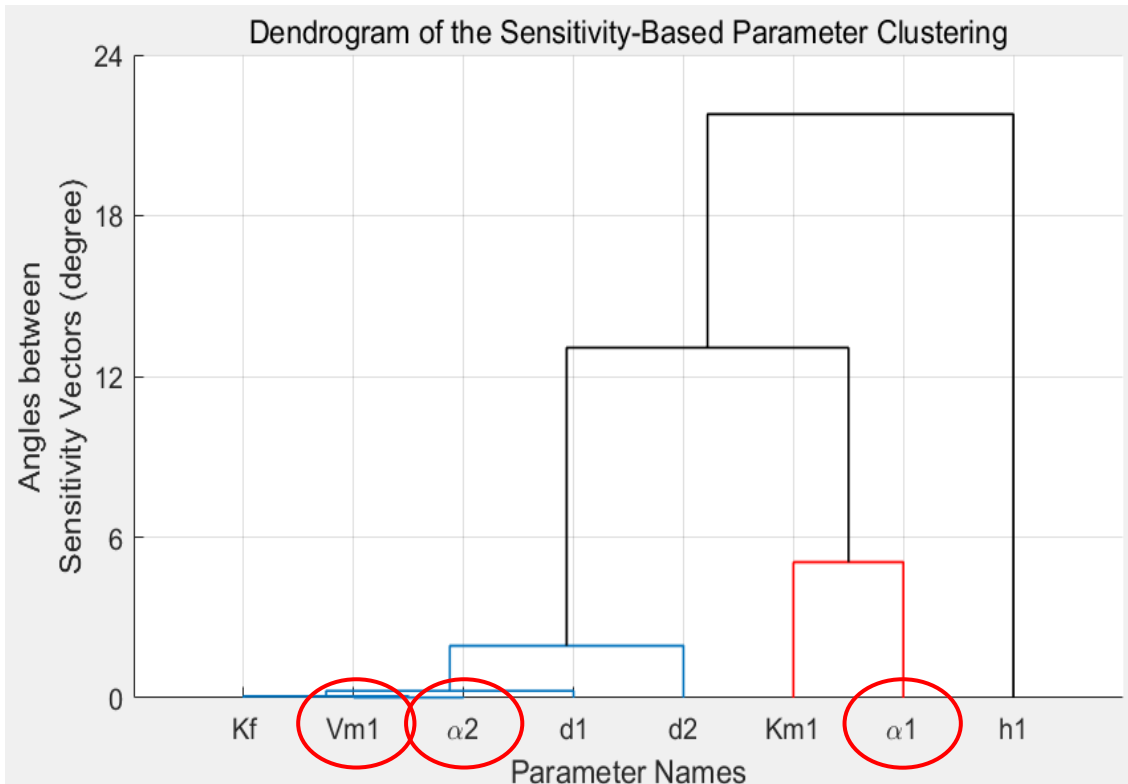
Figure 3-15. Dendrogram of the sensitivity-based clustering results. Red circles mark the parameters of identifiability issues. It can be seen that the sensitivity-based method cannot find the connection between these special parameters.
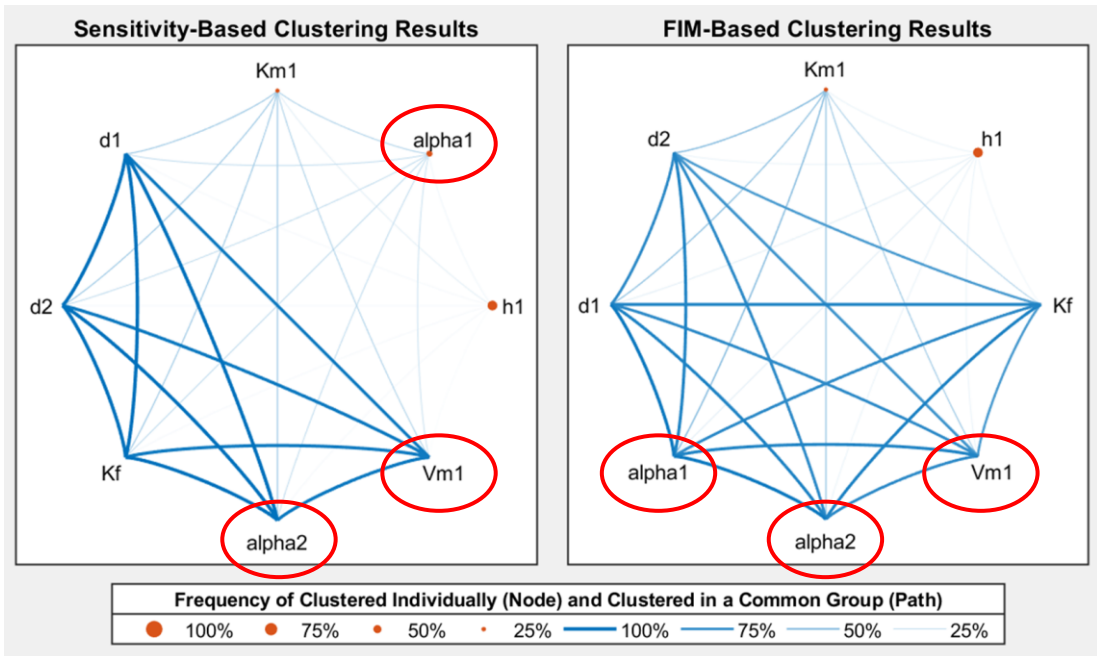
Figure 3-16. Comparison of clustering results with varied parameter values. FIM-based clustering always put the unidentifiable parameters in a common cluster, not matter how their values change.

3.3.4 Experiments with Random Stimuli vs Off-Line OED
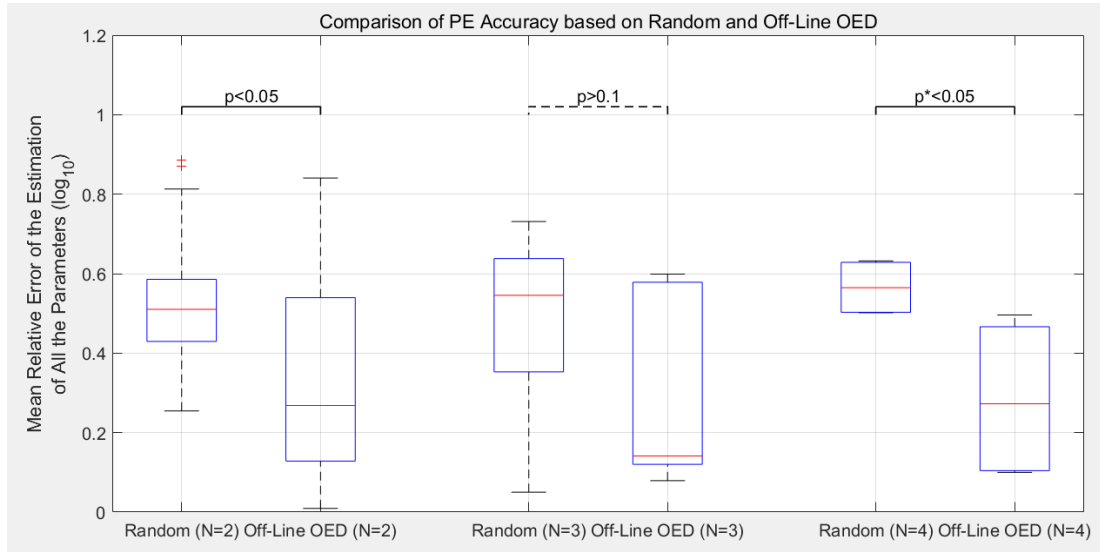
Figure 3-17. Comparison of the PE accuracy based on random and off-line OED experiments.
p values stand for one-way t-test results (normal distributions), p* value is for one-sided Mann-Whitney U-tests. The random and off-line cases for N=3 do not pass Kolmogorov-Smirnov normality test.

Similar to the definition from Bandiera et al.[27], the mean relative error is defined as follows:

$$\varepsilon^j = \frac{1}{N_p} \sum_{i=1}^{N_P} \left| \log_{10} \frac{p_i^j}{p_i^*} \right| \tag{3-7}$$

where j is the trial index, N_p is the total number of calibrated parameters (7 in this case), i is the parameter index, p_i^* is the true parameter value, and p_i^j is the estimated parameter value in the jth trial.

Since the number of sub-experiments depends on the clustering result, there are 29 trials with 2 sub-experiments (N=2), 9 trials with 3 experiments (N=3), and 4 trials with 4 experiments (N=4).

Figure 3-17 shows that for all the three sub-experiment numbers, off-line OED gives a smaller mean relative error of the estimation than random stimuli-based estimations.

3.3.5 Off-Line OED vs On-Line OED

The numbers of experiments are the same as the comparison in section 3.3.3 and. For the cases with all the three sub-experiment numbers, on-line OEDs give smaller mean relative errors of the estimation compared to off-line OEDs.
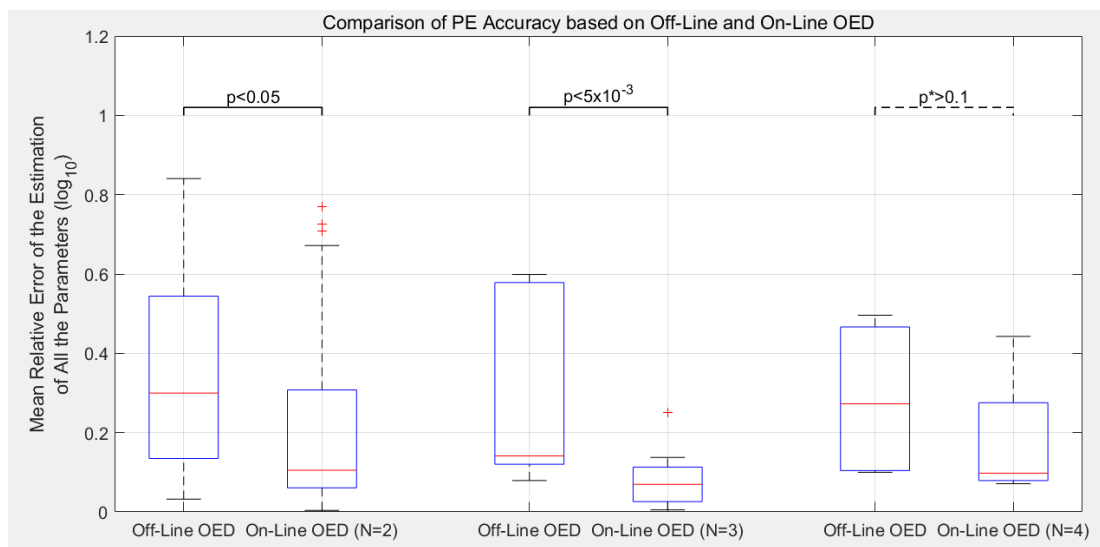
Figure 3-18. Comparison of the PE accuracy based on off-line and on-line OED experiments.
p values stand for one-way t-test results (normal distributions), p* value is for one-sided Mann-Whitney U-tests. The off-line case for N=4 does not pass Kolmogorov-Smirnov normality test.

3.3.6 On-Line OED vs OED with Parameter Clustering

Figure 3-19. Comparison of the PE accuracy with off/on-line OED
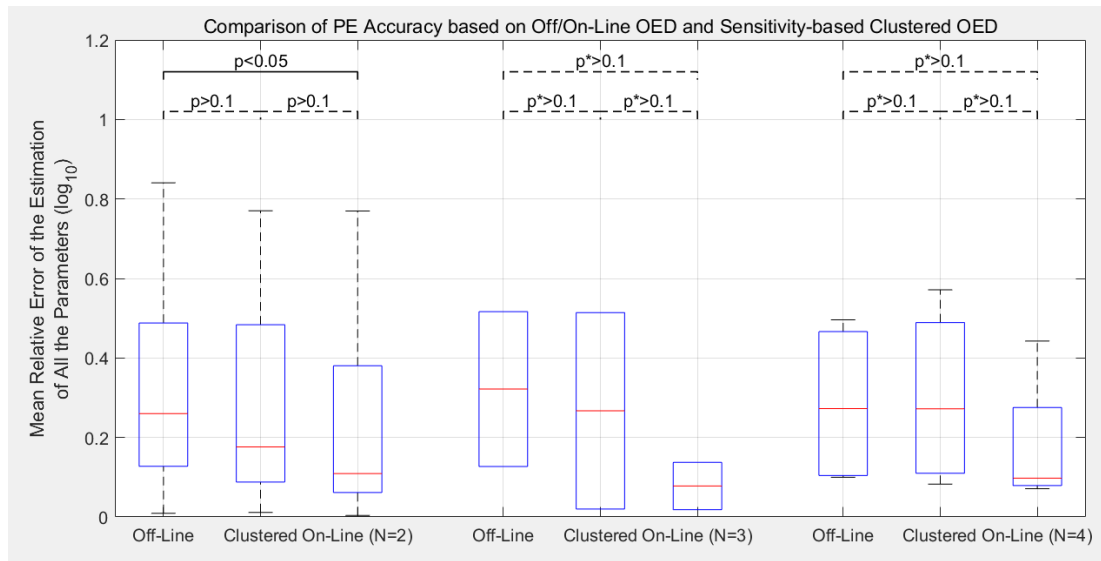
and sensitivity-base-clustered OED.
p values stand for one-way t-test results (normal distributions), p*
value is for one-sided Mann-Whitney U-tests.
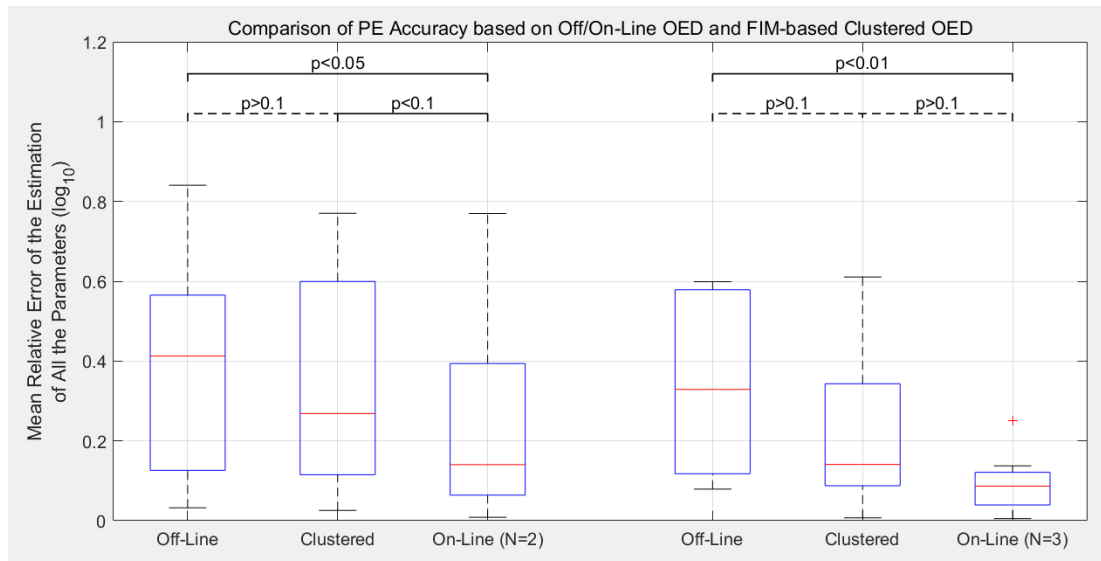
Figure 3-20. Comparison of the PE accuracy with off/on-line OED
and FIM-base-clustered OED.
p values stand for one-way t-test results. All the distributions pass
the Kolmogorov-Smirnov normality test.

As shown in Figure 3-19, for the sensitivity-base-clustered OED, there are 24

trials with 2 sub-experiments (N=2), 2 trials with 3 experiments (N=3), and 4

trials with 4 experiments (N=4). The clustered-OED are not worse than the off-line OED and therefore better than the traditional experimental design (random stimuli). However, because the parameter values do not update between the sub-experiments, the PE accuracy from clustered-OED is not as high as the on-line OED.

As shown in Figure 3-20, for the FIM-base-clustered OED, there are 22 trials with 2 sub-experiments (N=2) and 8 trials with 3 experiments (N=3). The clustered-OED are slightly better than the off-line OED and therefore also better than the traditional experimental design (random stimuli). Similar to the sensitivity-based ones, because the parameter values do not update between the sub-experiments, the PE accuracy from clustered-OED is not as high as the on-line OED.

3.4 Discussions

3.4.1 Parameter Clustering with Different Experimental Designs

Figure 3-8 shows that even with shall-search OED (the calculation takes less than 2 min), the optimised experiments would significantly improve some parameters' sensitivity.

This plot also shows that for some model parameters which have decisive effects on the model behaviours (such as the Hill-coefficient of the dose-response curve), random stimuli can already give relatively high sensitivity to the parameters. It is worthy to recall that previous theories and experiments have shown that random stimuli are the best experimental design to calibrate a linear model [266-268].

Figure 3-10 shows that the clustering number's optimality mainly depends on the criterion (Gap/Silhouette scores) but not the clustering algorithm

(Hierarchical/K-means algorithms). Since K-means takes more time to calculate and is less robust, Hierarchical clustering seems to be a better option for this type of tasks.

Another advantage of the Hierarchical algorithm compared to K-means is that Hierarchical clustering can give plot like Figure 3-11, which helps researchers visualise the sensitivity connections between parameters.

Another message from section 3.3.1 is that the FIM-based clustering and sensitivity-based clustering may give different results. However, the results are not entirely irrelevant, and differences could be caused by the other way of evaluating the similarity between in-cluster parameters.

3.4.2 Parameter Clustering with Different Parameter Value Sets

The optimal cluster number could vary on the initial estimation of the parameters (Figure 3-13) and the clustering results (Figure 3-14). However, most of the connections remain the same in most of the initial estimations.

For every *a-prior* probability distribution or value set of the model parameters, by definition, there would be one corresponded clustering result. As shown in this case study, the clustering result may vary with the parameter values. By applying the Monte Carlo method and varying the parameter values (in the feasible range), researchers could investigate how robust the connections are between the parameters. A modified version of the arc diagram (Figure 3-14) is particularly designed for visualising this information: Thicker arcs stand for more robust connections, and bigger joints highlight the parameters which are less "sociable".

Moreover, for the sensitivity-based clustering, benefiting from the hierarchical clustering, the dendrogram (Figure 3-11) can also show how strong the

connections are. This diagram can be generated base on one parameter value set. In this case study, the dendrogram and arc diagram indeed confirm each other.

3.4.3 Experiments with Random Stimuli and OEDs

Figure 3-17 to Figure 3-20 show the PE accuracy based on random stimuli and different OED approaches. Overall, the OED experiments give significantly better estimations compared to the traditional random stimuli. On-line OED outcompetes the off-line OED and clustered OED. It could because it is the only approach that updates the parameters during the experiments.

FIM-base-clustered OED shows better performance than the off-line OED, and sensitivity-base-clustered OED is also better than the off-line OED but not as significant as the FIM-based one. It could be explained as the FIM-based approach consider the correlation between parameters in a more structural way.

3.4.4 Extending the Parameter Clustering to Larger Models

The work in this chapter shows the parameter clustering on a non-linear model with seven parameters. The parameters are generally clustered into "linear" and "non-linear" parts. How the proposed methods perform for larger models with more parameters is an interesting and valuable question to investigate in future works, it is also a tool for researchers to explore the mechanisms of biological networks. Yet to the best of the author's knowledge, this problem was not focused in most of the previous studies which involved parameter clustering (such as the example mentioned in section 3.1.4).

For larger models, it is reasonable to expect there would be more parameter clusters other than just the "linear" and "non-linear" groups. For example, larger biological models usually involve reactions with faster and slower reaction rates, and the most efficient stimuli corresponding to these reactions have different frequencies. So the parameters for faster and slower reactions may be clustered into different groups. Moreover, both the sensitivity-based and FIM-based approaches could work with given cluster numbers. Researchers may also analyse the models by running the clustering with a pre-decided cluster number.

Chapter 4: Automatic Experiment Abnormality Screening based on Experimental Images

During the lab-work for the microfluidic experiments in chapter 2, some abnormal experimental conditions appeared in the middle of experiments and caused numbers of invalid experimental data. This chapter develops and validates an automatic experiment abnormality screening (AEAS) system supported by convolutional neural networks (CNNs). The system learns the features of six abnormal experiment conditions from images taken in the actual microfluidic experiments and achieves seconds-level identification with the desired accuracy. Moreover, the proposed levelled-training method is helpful to increase the chance of convergence for deep networks.

4.1 Introduction

4.1.1 Requirement of Automatic Abnormality Diagnosis

During the lab-work for the microfluidic experiments in chapter 2, some abnormal experimental conditions appeared in the middle of experiments several times and caused numbers of invalid experimental data. The conditions can be classified into six groups: high bleed-through, device movement, collapsed chamber, bubbles in the oil, blurry images, and dark images. The details of these conditions are given in the method section (section 4.2.1).

Like previous microfluidic experiments [219, 269], the identification of abnormal conditions is completed by visual check of hundreds of microscope images by the end of the experiment. Although this method is the simplest (for setting up) and, to the best of our knowledge, the only existing approach that

can identify all different abnormal conditions in one test [270, 271], it cannot identify and stop the experiment when the condition appears to prevent the waste of time. The accuracy of condition identification with bare-eye-checking is also sub-ideal for some inconspicuous features (particularly in some cases of device movement and chamber collapse). Moreover, it involves laborious efforts from skilled experts and would be impractical for mass parallel microfluidic experiments. Therefore, to achieve a more efficient and reliable parallel microfluidic experiment system, it is of great value to develop an automatic experiment abnormality screening (AEAS) system.

4.1.2 Related Work in Biological Experiments

As mentioned above, there are not yet automated screening system for abnormal experimental conditions for microfluidics [271]. The previous study on biological experiments focused on two related topics: fault screening for microfluidic biochips, and standardisation and quality control (QC) systems of biological experiments.

The automated screening of faults in biochips has been a popular topic in the last decade in the field of microfluidic manufacturing [270, 272-278]. However, these studies focus on the flow-based chips (Figure 4-1) and digital microfluidic chips (Figure 4-2) rather than the continuous-flow chips (Figure 4-3), which is the category of chip used in this experiment. Because of the different mechanism, the testing techniques of the chips are completely different. For the flow-based microfluidic chips, air pressure is applied in control channels to adjust the media flow in the flow channels. Hence the testing is to screen the channel blockages and leakage by applying pressure in one port and measure the pressure readings in related channels [270, 274, 275]. For the digital microfluidic chips, electrical flows are applied to the chip to move the media droplets by electrowetting on dielectric (EWOD), so the testing is to check the faults on the chip by traversing the test droplet between array cells of the chip

and tracking its location [276-278]. Moreover, these automated testing systems are mainly for off-line tests that only screen the faults in the device, not the abnormal conditions during experiments. Therefore, although these studies are related topics, they cannot provide an efficient reference for the work in this chapter.
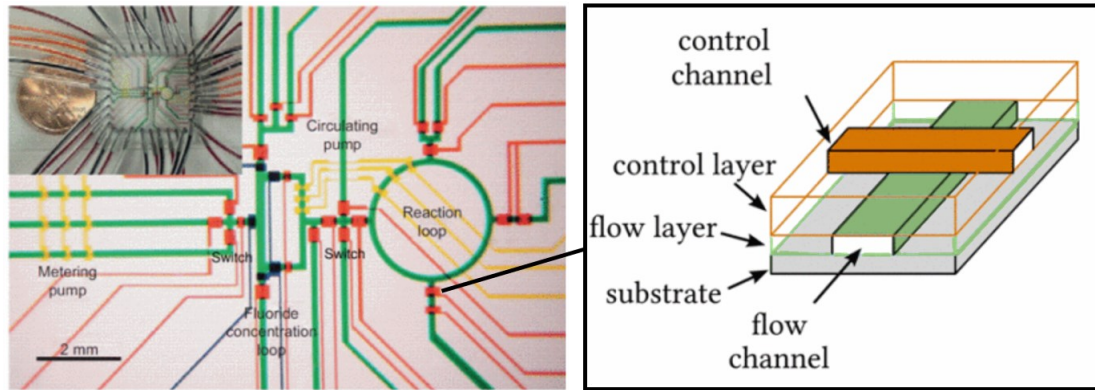
(a)   (b)

Figure 4-1. An example of flow-based biochips (a) and the structure design of control valve (b) [275, 279].
Flow channels are shown in green and control channels are in yellow and red.
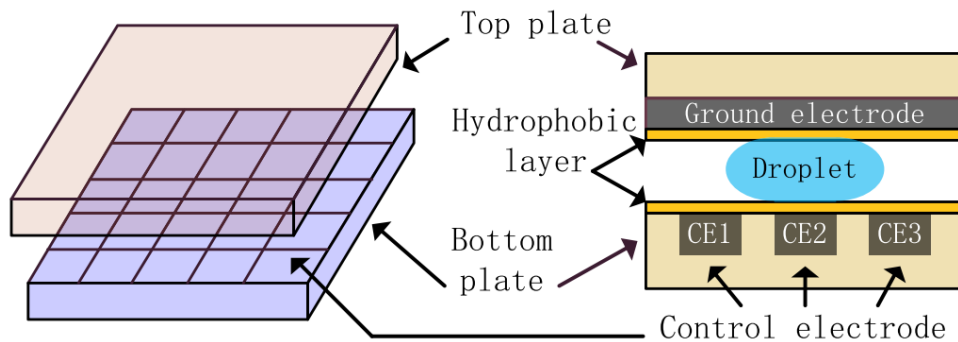
Figure 4-2. Oblique drawing of a digital microfluidic device and its cross section [278].
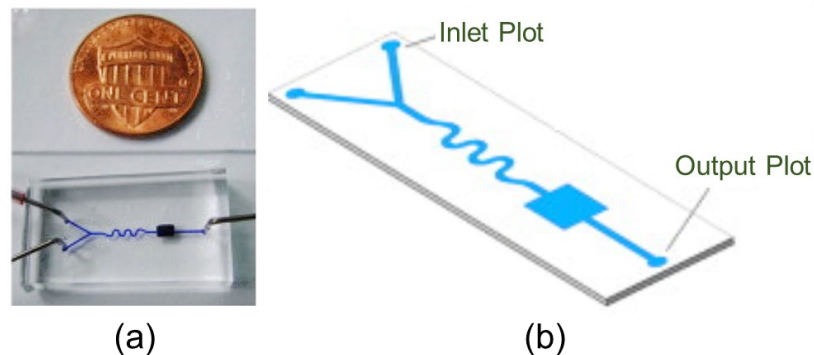
(a)   (b)

Figure 4-3. An example of continuous-flow microfluidic chips (a) and its channel design (b) [280].

In biology, the diagnosis of abnormal experimental condition has been considered an essential aspect of the standardisation and quality control (QC) systems of experiments [281, 282]. In the last decade, many groups focused on the QC of liquid chromatography (LC) tandem mass spectrometry (MS) experiments [283-286]. It is because carrying out an LC-MS experiment is generally an intricate process [285], and this experimental technique is adopted in the Human Proteome Project (HPP) and studies of cancer to identify, quantify, and characterize the human proteome [286-290]. These works on fault diagnosis are mainly based on either the spike-in method (spike some prepared benchmarks in the middle of samples to test) or experts' checking or statistic-tests of manually-extracted features [285, 291, 292]. The considerations of these methods are shown in Table 4-1.

Table 4-1. Considerations of current diagnosis methods in biological experiments as a reference.

Diagnosis method	Consideration
Spike-in Method Add benchmark samples in the middle of testing samples, validate the experiment with the readings of benchmarks	Cannot adopt because the chip works with one sample in each experiment
Experts' Manual Checking Check the raw data or manually designed features	Cannot adopt because labour-intensity is exactly the issue to solve in this chapter
Statistical Analysis of Manually Designed Features Manually design the features to extract and apply statistical analysis	Not ideal because extracting and validating the image features for all the diagnosis cases could be an intensive work

Although for different reasons, these three methods for diagnosis abnormal experimental conditions cannot be directly applied to the work in this thesis, a concept from the third method, i.e., applying statistical theory to analyse the features from the experimental output, shows a promising route. For feature extraction and inferring of 2D images, some recent studies in machine learning have shown some successful applications and great potential. These studies would be introduced in the section below.

4.1.3 Image-Based Inferring Networks in Machine Learning

In the last decade, the performance record of image recognition has been continuously redefined because of the rapid development in machine learning (ML) [293]. Figure 4-4 shows how ML gets adopted in this kind of task and takes on more functions after achieving significant benefits (from left to right).
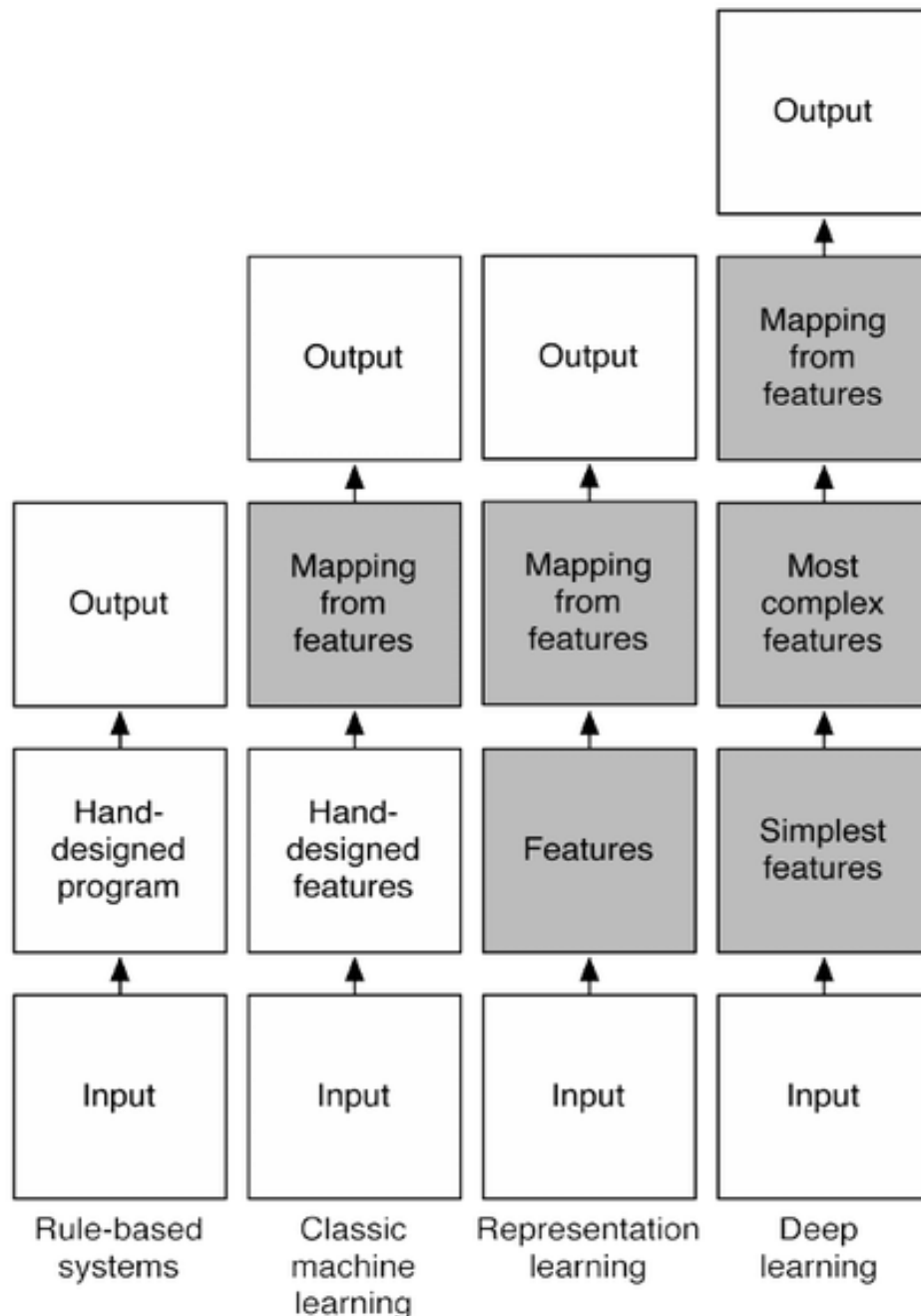
Figure 4-4. Stages of the application of machine learning in image recognition tasks [294].

In Figure 4-4, grey blocks are the functions supported by machine learning. In the early stages (classic machine learning), some shallow machine learning algorithms were used to infer image recognition based on manually extracted features. Two representative methods are support vector machines (SVM) [295, 296] and artificial neural networks (ANN) [297, 298]. Although these shallow machine learning approaches prove their value for image recognition [297, 299], many studies pointed out that the representation of the data, i.e., the features extraction, limits further performance improvements [293, 300]. Some later works attempted to introduce multisensory data fusion algorithms based on machine learning (representation learning) to image recognition [301-303]. However, for diagnosis tasks that involve abstract and transferable features with complex properties such as shift-invariancy, deeper neural networks (deep learning) are required to achieve acceptable performance [304-306].

Deeper networks are constructed by multiple layers that carry out relatively more straightforward processes. The layers closer to the input layer (shallow layers) will handle simple features, while the layers closer to the output layer (deep layers) can capture more complex and abstract features. There are several variants of deep learning network structures that are specialised for different types of input format and purposes. For example, recurrent neural networks (RNN) pass some of the signals from deep layers back to shallow layers. This mechanism simulates the "memory" effect and helps context-depend tasks like natural language processing and fault diagnosis based on continuous-time data [307]. Convolutional neural networks (CNN) is specialised for tasks with 2D and 3D input data because these networks have convolutional layers that can effectively extract the features from these images [298, 308].

The value of deep learning in the field of abnormality screening tasks has been proved in many industrial applications [306, 309-312]. As shown in Figure 4-4, compared to classic machine learning approaches such as classic SVM, one

of its biggest advantages is that the screening systems can work in on-line in the end-to-end form without any pre-processing [312]. The CNNs in particular, is shown to be one of the best options for image-based anomaly diagnosis for their efficiency and state-of-the-art accuracy [313-315].

Based on the background search above, the finally decided approach is to develop the automatic experiment abnormality screening (AEAS) system supported by convolutional neural networks (CNN) to analyse the images taken during the experiment.

4.1.4 Novelty of the Chapter

Previous works in microfluidics show the great potential of mass-parallel experiments with this practicality [219]. The real-time fault diagnosis system would be vital to make these experiments efficient because early detections can save valuable time and cost. On the other hand, recent development in the convolutional neural network proved its power in smart image processing. So far, there are not yet works investigating how well CNNs can be used for fault diagnosis in microfluidic experiments or other image-based microscopy experiments.

It would be the first time applying CNNs for the real-time fault diagnosis tasks of an experimental platform. In this chapter, three CNNs are trained for fault diagnosis base on experimental images. Their accuracies are validated with image data from real experiments. Moreover, this chapter also presents and practically tests the effectiveness of several improvements in a few aspects, including image pre-processing, network training, and network visualisation.

This work provides an example with valuable experiences of the application of CNN-based abnormality screening in bioengineering experiments.

4.2 Methods

4.2.1 Details of Appeared Abnormal Experimental Conditions

As introduced in section 4.1.1, t six abnormal conditions appeared in the microfluidic experiments in chapter 2 multiple times. The details are given below (notice "rerunning the experiment" in the following paragraphs stands for restarting from the very beginning step, i.e., redo the media/cells preparation for the experiment and use a new microfluidic chip):

1. **High Bleed-Through**
   - Reason: The signal bleed-through from input fluorescence (Sulforhodamine) to the output fluorescence channel (Citrine) was so strong that it significantly affects the accurate reading of the output signal. (Figure 4-5)
   - Feature: Signal bleed-through is very strong, and the Sulforhodamine signal can be observed in the Citrine channel. The Citrine signal is non-zero even in the background pixels of the cell chamber.
   - Solution: Signal compensation can partially solve this problem. The preferred solution is reducing the input-reporter concentration and rerunning the experiment.
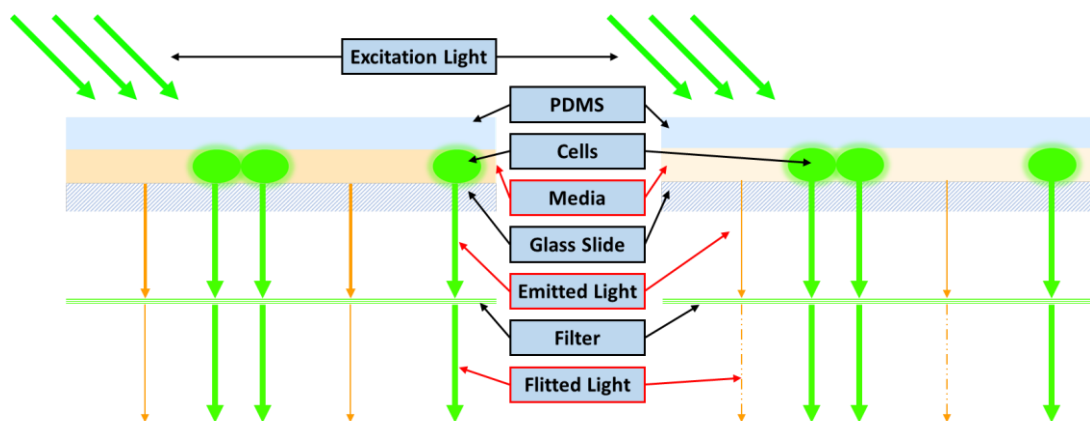
Figure 4-5. Illustration of the high bleed-through problem. For normal experiments (right), the background light (orange lights) signal should become negligible after passing the filter; when there is too much fluorescent dye in the media (left), the passed background signal would be strong enough to affect the reading of the signal (calculated as cell signal minus background signal).

2. Device Movement

- Reason: The microfluidic chip is fixed to the microscope stage frame before the start of experiments. The stress in the fixing tape will be gradually released in the first few hours of the experiment. If the stress is very imbalanced, the device may move with a small distance. In extreme cases, the chamber may completely move out of the microscope view. (Figure 4-13)

- Feature: The location of the device will move during the experiment.

- Solution: A classic approach is the alignment of images, and it does not work with extreme cases that most of the cell chamber area moves out of the view (severe data loss and contamination). The solution is rerunning the experiment with a device appropriately fixed.
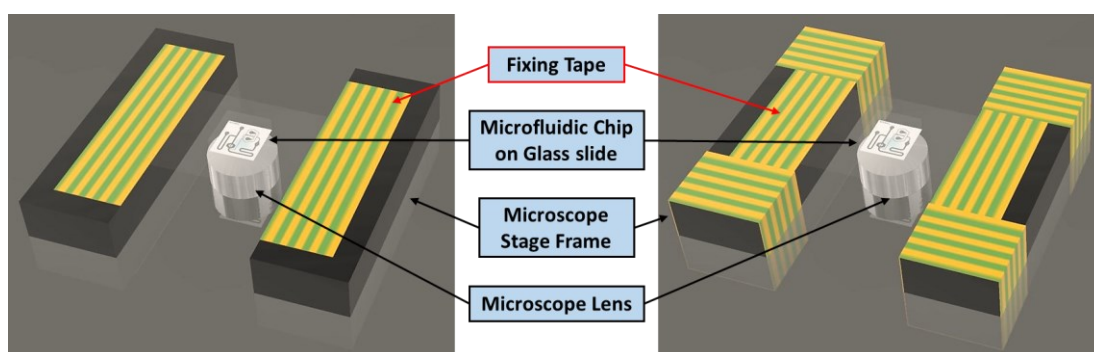
Figure 4-6. Illustration of the device movement problem.
Stress in the fixing tape will be gradually released during the experiment; Fixing the device like the right-hand side is helpful to reduce the chance of device movement.

3. Collapsed Chamber

- Reason: Usually, cells cannot get into a collapsed chamber. However, if the device gets applied with unexpected pressure from above, they may get trapped in the collapsed chamber. (Figure 4-7)

- Feature: The feature is not easy (but possible) to check with bare eyes. There would be a very faint border between the collapsed region and normal regions in the chamber. As the experiment running, cells cannot grow into the collapsed regions, and the cells that get trapped will die as running out of nutrients.

● Solution: If the collapse can be noticed before starting the experiment, it may get solved by applying a gentle positive pulse from the waste port. If it is noticed during the experiment, there are no better solutions other than rerunning the experiment.
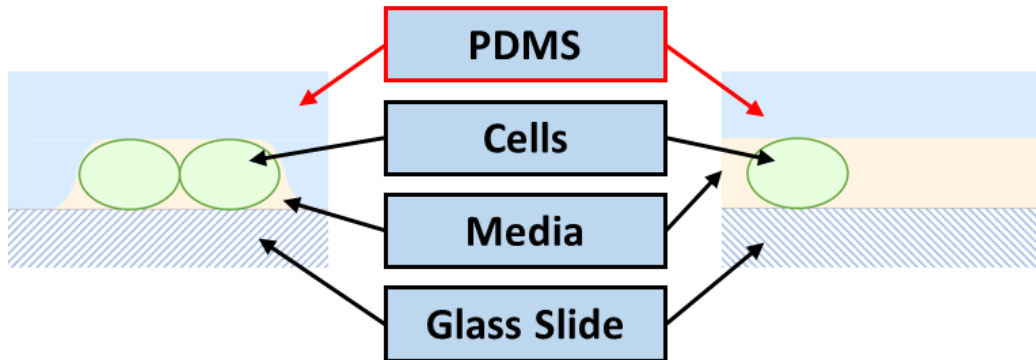
Figure 4-7. Illustration of the collapsed chamber problem.
When cells get trapped in collapsed part of the chamber, they would have very limited resources and no space for budding.

4. **Bubbles in Oil**
   ● Reason: Occasionally, there might be air bubbles in the oil between the oil lens and the device. It usually happens when the oil is not adequate, and the device moved around during the preparation (to take images of multiple positions). (Figure 4-8)
   ● Feature: The region affected by bubbles will be significantly darker than the other regions and out of focus.
   ● Solution: If this problem is identified before starting the experiment, clearing the lens and adding the oil should solve it. If it happens during the experiment, there are no better solutions except rerunning the experiment.
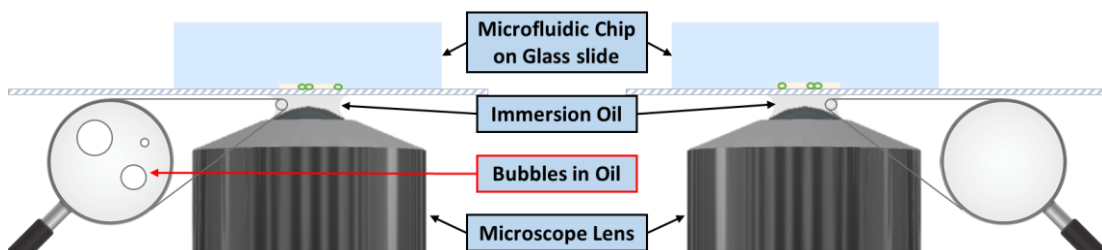
Figure 4-8. Illustration of the bubbles in oil problem.
When there are bubbles in immersion oil (left) and moved right under the chamber, there would be focusing problems.

5. Blurry Images

- Reason: There is a small chance that the Nikon Perfect Focus System (PFS) [316] does not work as expected, and thus the images taken are out of focus. (Figure 4-9)

- Feature: The images would be blurry because they are taken out of the focus.

- Solution: Restarting the perfect-focusing-system may solve the problem, but sometimes it does not help. Probably rerunning the experiment is the only solution if the restarting could not solve the problem.
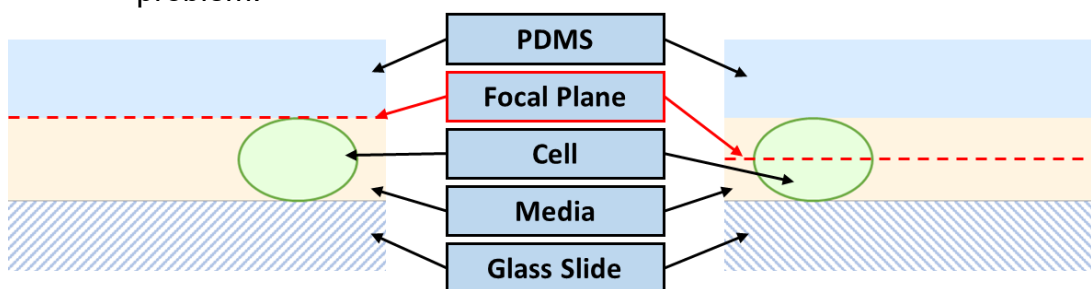
Figure 4-9. Illustration of the blurry Images problem.
The focal plane should be at the middle of cells (right); when it is
above or below this hight, the images taken would be out of focus.

6. Dark Images

- Reason: There are mainly two causes of this problem: Because of the physical limits of the computer, imaging action may be triggered slightly later than the scheduled time, to catch-up with the schedule, the time for exposure will get reduced and results in a darker image; Another case is when taking images of multiple channels, there is a tiny chance that the exposure starts before the filters get in position, it will also shorten the actual exposure time and leads to darker images. (Figure 4-10)

- Feature: The images would be significantly darker than the average level.

- Solution: Labelling these images and removing them in later processes is a solution. However, if there are too many dark images, the experiment would lose so much information that it will be no

longer comparable to other experiments of the same duration. Rerunning the experiment would be the only solution. Moreover, adding longer pauses between imaging actions is helpful to prevent this problem.
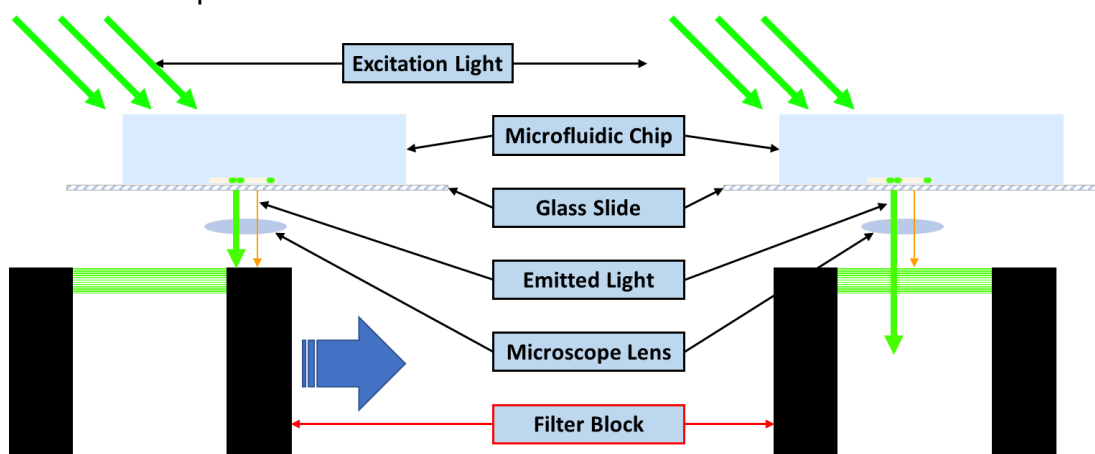
Figure 4-10. Illustration of the dark images problem.
If the exposure starts when the filter block has not yet arrived in position (left), the images would be significantly darker than what it should be.

4.2.2 Experimental Images Used for Network Training

The amount of available data of each category is listed in Table 4-2, and this would be the data source for the training and validation of CNNs. Notice that the data size for normal experiments does not match with all the experiments in chapter 2. It is because previous works [317] have shown the negative effect of imbalanced data sizes for categories. In other words, significantly more data of only one category is not necessarily beneficial. Moreover, this study also used a pre-processing method for CNN training called "image augmentation" to balance the sample sizes of different categories (details of this method and other pre-processing steps are explained in section 4.2.3).

It is essential to determine which images should be given to the network (network input) as evidence for abnormality screening. The most direct option

is to use the most updated images of all the channels (DIC, Sulforhodamine, and Citrine as introduced in chapter 2) at the sampling time.

Table 4-2. Data on different fault categories for CNN training.

Category	No. Experiments	No. Sampling Time Points
Normal	5	1,218
High Bleed-Through	4	131
Device Movement	5	445
Collapsed Chamber	3	156
Bubbles in Oil	3	136
Blurry Images	3	30
Dark Images	3	25

However, only the latest images of the three channels are not enough to identify all the abnormal conditions. One of the most prominent examples is the case of device movement. The device movement is usually slow, and the DIC images are not blurry at all the sampling times; however, when checking these DIC images as a time sequence, it would be obvious to see that the device gradually moved. It means that the information of the previous images should also be given to the network. Therefore, the mean and standard deviations (std.) for each channel are also provided as input images. They are also chosen because both the mean and standard deviation are non-negative and have the maximum limit in values. It is a beneficial property for faster and more robust CNN training [318, 319]. Thus, the decision is to use images of 3*3=9 channels as the input of the diagnosis network, i.e., latest/mean/std images of the DIC/Sulforhodamine/Citrine channel (as shown in Figure 4-11) shows an example of how these images help identify the abnormal condition. Notice that the shape of the device can be seen in the std image of the DIC channel (mid-left) because the location of the device moved during the experiment. In experiments without problems, the corresponded image should not show the device's shapes but only show some small "flowers" because of cell growth.
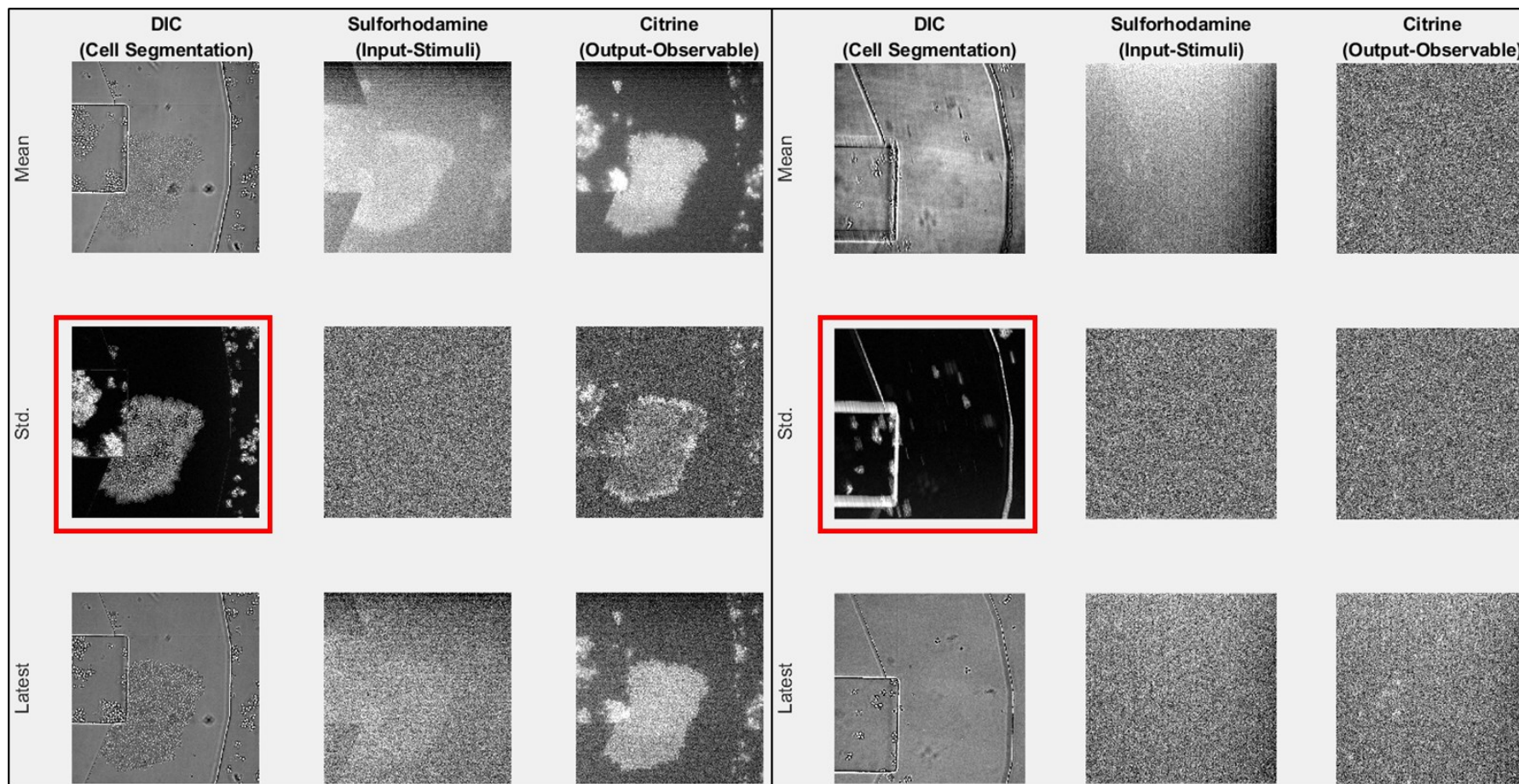
Figure 4-11. Example of input images for the diagnosis network (normal condition and device movement).
Notice that when the device moves, the shape of the device can be clearly seen in the std. image of the DIC channel
(red rectangles), which is not the case in a normal experimental condition.

4.2.3 Pre-processing of the Experimental Images

Image pre-processing is the stage that modifies the image data before the start of network training. The image pre-processing in this project involves two aspects: augmentation and normalisation. These two procedures are of different purpose and are different in methodology, and they shall be introduced one by one in the following paragraphs.

Image augmentation is adding new samples by artificially modifying the original image samples [320]. Many previous works have proved that augmentation is an efficient way to improve network accuracy and generalisation, and also reduce overfitting [321, 322]. Furthermore, augmentation is also a convenient, practical and commonly-adopted procedure to enhance the data sets with limited size and difficulty accumulating new samples. This limit is prevalent in many similar tasks, such as training the networks of fault diagnosis or data analysis of clinical trials [323].

Classic augmentations include multiple transformations, e.g., shifting, resizing, cropping, rotating, flipping, and distorting [320, 324, 325]. Some of these transformations are not suitable for this task. For example, resizing was mainly for making the network recognise objects of different sizes (like cars and apples, which naturally have varying sizes because of the visual angle or inner property), but in this case, the used lens and chip design are fixed so the images of the chamber are impossible to appear with a different size. The detailed reasoning and decisions are shown in Table 4-3. The final decision is to adopt the 0/90/180/270-degree-rotations and flipping the images. In this way, each 9-channel-image of a sampling point can generate seven more augmented images, and the data size increases to 8 times larger. As shown in Figure 4-12, shifting and rotating is reasonable to add, but these methods work better with more image data for this purpose, which could be considered in future works.

Normalisation is another pre-processing procedure. In machine learning, it is initially introduced to ensure all the input values are in a comparable numerical scale, so that each sample for training contribute almost equally to the loss function (the function that evaluates how well the trained network performs). It is helpful to achieve better training performance [318]. However, this project adopted this method mainly for another purpose, i.e. reducing the size of the input data (or dimensionality reduction [326, 327]). As mentioned in section 4.1.1, microfluidic experiments have great potential for massive parallel experiments. These experiments would lead to the transmission and processes of massive high-resolution images, which is a practical challenge in the industry [326].

Table 4-3. Reasoning of the augmentation methods selection. Green ones are the adopted methods.

Transform	Details	Reasoning	Decision
Shifting	moving around the entire image	no data to fill the space after moving	not adopted
Resizing	zoom in/out of the image	the microfluidic chip size does not change	not adopted
Cropping	cut a subset of the image as the input	cropping may miss the evidence on images (e.g., collapsed chamber)	not adopted
Rotating	rotate the image with a random angle	the microfluidic chip only bound to the glass with specific rotations (0/90/180/270 degrees) + should let the network "know" the direction of the chip does not matter	only with 0/90/180/270 degrees
Flipping	flip the image up-down or left-right	shape of the chamber region is symmetric	adopted.
Distorting	uneven elastic distortions of the image	the shape of microfluidic chips is fixed	not adopted.

![Shape of the Chip icon]	Shape of the Chip
![Cell Chamber icon]	Cell Chamber
☐	Current Imaging Region
![New Images icon]	New Images Required for Argumentation
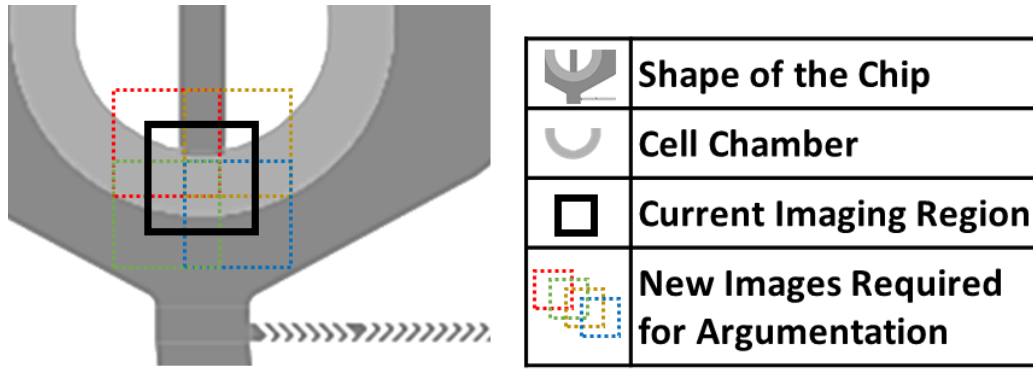
Figure 4-12. Illustration of four additional images for image augmentation. Because of the complex shape of the chip and the unpredictiable layout of cells, it is impossible to automatically predict and generate the extended images for some certain augmentation methods (shifting and rotating).

The core of dimensionality reduction is cut down the data size without omitting important information of the abnormal conditions [327]. In this project, the normalisation combines the latest/mean/std images into one, and thus reduce the total channel number from 9 to 3. The detailed expression of the normalisation for each pixel is given in Eq. 4-3. It is a variation of the 'z-score' normalisation [328].

$$
Y_{t,norm} = \begin{cases} Y_{max} & (Y_t \geq Y_{mean} + 3 \cdot \sigma_Y) \\ \left(\dfrac{Y_t - Y_{mean}}{6 \cdot \sigma_Y} + \dfrac{1}{2}\right) Y_{max} & (otherwise) \\ 0 & (Y_t \leq Y_{mean} - 3 \cdot \sigma_Y) \end{cases} \qquad (4-1)
$$

where $Y_{t,norm}$ is the normalised value, Y_t is the pixel intensity of the latest image, Y_{mean} is the averaged intensity of the same pixel since the first experimental image, σ_Y is the standard deviation of the intensity since the first image, Y_{max} is the numerical value of the maximum intensity (since the data is stored as uint16 integers, $Y_{max} = 2^{16} - 1 = 65535$).

The normalised images have three channels instead of nine so they can be shown as a coloured picture by mapping DIC to Red, Sulforhodamine to Green, and Citrine to Blue (Figure 4-13).
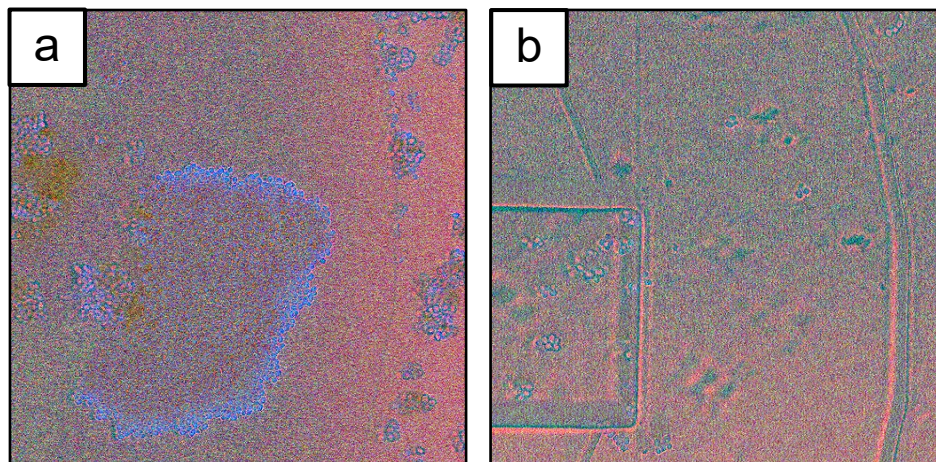
Figure 4-13. Example of normalised input images.
These images correspond to the two samples in Figure 4-11 (a is for normal condition and b is for device movement). Notice the shape of the device can be seen in b but not in a. The purple shapes suggest that the cells are growing as expected in the device.

4.2.4 General Structure and Layers of CNNs

Convolutional neural networks are a category of artificial neural networks that is specialised for image processing. Similar to other artificial neural networks, CNNs are constructed in layers. The data passes from the input layer, then goes through a stack of layers that processes the information, and finally reaches the output layer that provides the final recognition result.

Table 4-4 shows the commonly adopted layers in CNN structures categorised by the different purposes of the layer. It is essential to point out that recent studies in machine learning (such as works on attention models [329], dropout layers [330], and capsule networks [331]) keep extending this list.

Table 4-4. Table of commonly adopted CNN layers [332].

Layer Category	Purpose	Layer used in this Project (MATLAB Deep Learning Toolbox)
Input Layer	read the input data	imageInputLayer
Convolution Layer	carry out convolutional calculations	convolution2dLayer
Activation Layer	activate/deactivate the effect of certain pixels to the next layer	reluLayer
Normalisation Layer	normalise the data to reduce the sensitivity to network initialisation	batchNormalisationLayer
Pooling Layer	reduce the data size (down-sampling)	maxPooling2dLayer
Fully-Connected Layer	summarise all the data from the previous layer	fullyConnectedLayer
Output Layer	summarise the data and give the final output	softmaxLayer

In 2020 Wang et al. developed the famous CNN Explainer that can visualise how CNN transforms the input images into classification predictions [333]. Figure 4-14 is an example with a picture of a ladybug. The input image on the left is 64*64 pixels in size. It contains three channels (red, green and blue). The intensity information of these channels is processed through different layers from left to right and finally reach the output layer with a classification that gives the final result.
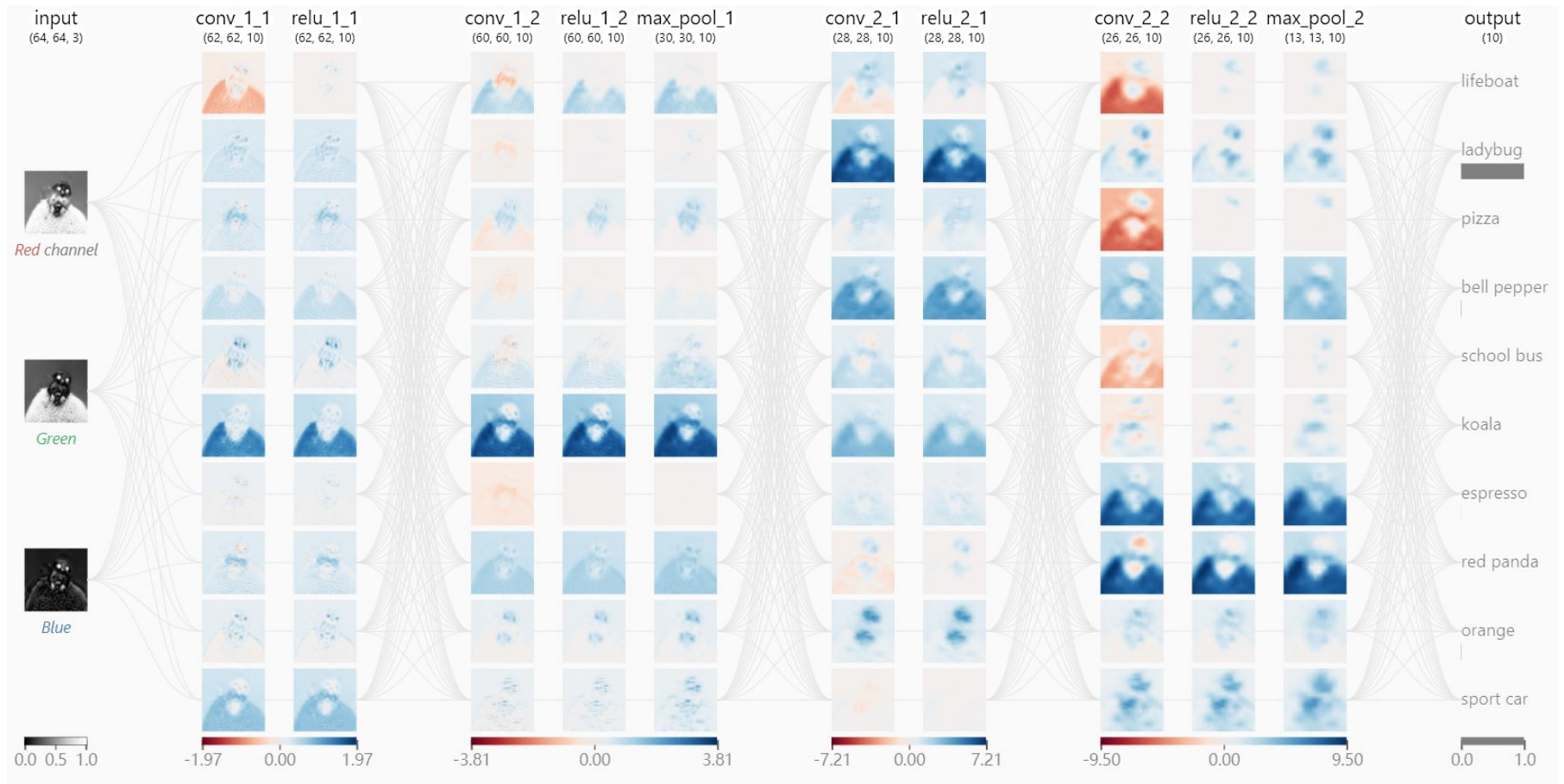
Figure 4-14. Example of a CNN identifying a picture of a ladybug [333].
The names at the top shows the types of these layers.

Wang et al.'s example CNN covers most layer types in Table 4-4, and they also provide impressive visual explanations for these layers (Figure 4-15).
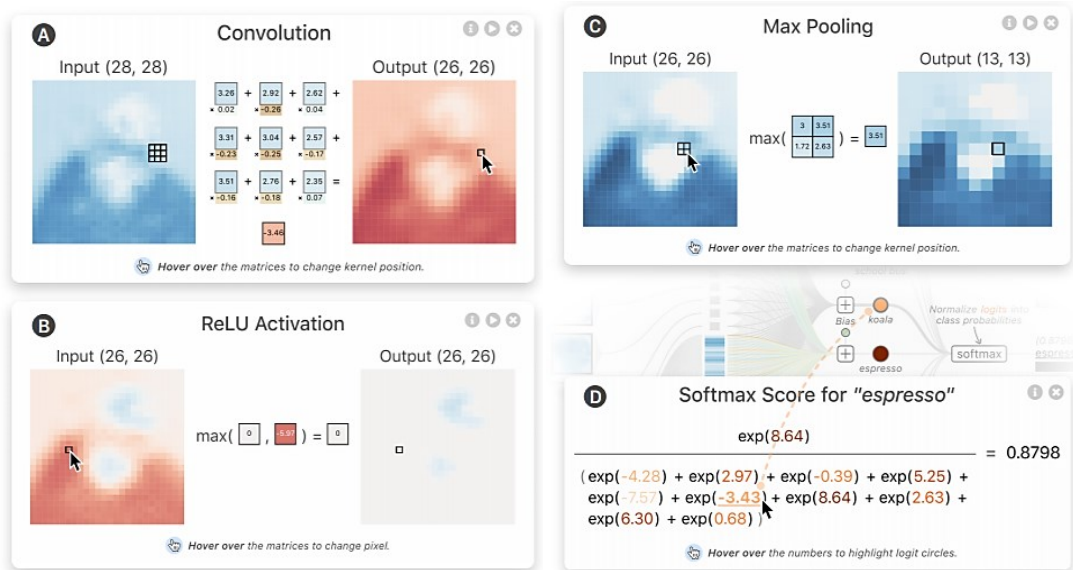
Figure 4-15. Visual explanations of the layers in example CNN [333].

The detailed functions of these layers, including the ones which are not included in Wang et al.'s paper [333], are given in Table 4-5.

Table 4-5. Table of detailed functions of adopted CNN layers.

Layer Name	Input/Output Format	Detailed Functions
Convolution 2D (3*3)	local image stacks to scalar	$y = \sum_{elements} \sum_{i=1}^{N_x} \left(\boxed{}_{x_i} \odot \boxed{}_{w_i} \right) + \beta$
ReLu	scalar to scalar	$y = max(0, x)$
Max Pooling (2*2)	local image to scalar	$y = max \left(\boxed{}_{x} \right)$
Soft Max	scalar array to scalar array	$y_i = \dfrac{e^{x_i}}{\sum_{j=1}^{N_x} e^{x_j}}$
Batch Normalisation	scalar to scalar	$y = \gamma \dfrac{x - \bar{x}}{\sigma_x} + \beta$
Fully-Connected Layer	scalar array to scalar	$y = \sum_{i=1}^{N_x} x_i w_i + \beta$

In Table 4-5, x are the input signals from neurons of the previous layer, w is the weight matrix or array, β is the offset, i and j are the neuron indexes, γ is the scaling factor, $\bar{x}$ and σ_x are the mean and standard deviation of the input during the training, $\odot$ represents the component-wise multiplication.

4.2.5 Selected CNN Architectures

Some of the previous practical attempts of CNN-based abnormality screening have shown that even shallow CNNs can achieve a satisfying diagnostic accuracy [311, 326, 334]. Therefore, this project focused on three networks as follows:

1. **Simple-DLN**
    Simple-DLN is short for the simple deep learning network. This network is introduced in MathWorks' online training course [335, 336] and the user guide for MATLAB Neural network toolbox [337] as a typical example of CNN for image classification. In the original example, the network was designed for greyscale images of hand-written characters of 0-9 (i.e. the MNIST database [338]). It is also chosen in some highly-cited literature studies for solving relatively simple image processing tasks and illustrate the mechanism of CNN [339, 340]. This network is chosen because it is one of the smallest but still very representative convolutional learning networks which is designed for classification tasks of comparable difficulty as the problem to solve in this project.

2. **Tiny-VGG**
    Tiny-VGG is short for the *tiny Visual Geometry Group network*. This network is a simplified variation of the original VGG network family [341]. It is designed for identifying ten different everyday items based on coloured images [333]. Moreover, this network structure is adopted in a study to identify 10 different abnormal working conditions of a rotor-

bearing system based on thermal images and achieved state-of-the-art accuracy [311], which is worthy of reference considering the similarity of the task.

3. **VGG-11**

VGG-11 stands for *Visual Geometry Group network* with 11 layers. This network comes from the original paper for the VGG family [341] and is designed to identify 1,000 different items based on coloured images. Although it is the smallest VGG network in the original paper, the number of parameters is about 1,000 times of the two previous networks. Moreover, this network almost meets the calculation power limit of a computer not particularly designed for CNN training (i.e., no AI computing sticks, multiple high-performance GPUs, or computer clusters). The purpose of choosing this network is to use it as an example and see whether adopting deeper networks is worthy for practical condition diagnosis, as for many researchers managing the computing devices for training deeper networks would be costly and time-consuming.

Figure 4-16 shows the selected CNNs (left) and the detailed configuration designs (right). The grey rectangles in the VGG-11 layout stands for repeating the corresponding layer unit. Similar to the notations in Simonyan et al.'s work [341], the convolutional layers are denoted as "conv<filter size>-< neuron number>", the maxpooling layers are denoted as "maxpool<pool size>", the fully connected layers are denoted as "FC-< neuron number>", and batch-norm and ReLu layers are not shown for brevity.

Figure 4-17 compares the size of selected CNNs with popular pre-trained CNNs available [342]. Notice that these networks are designed for smaller input images (331*331*3 channels at most, the images in this task is 512*512*3 or 9 channels), so if the network is adjusted for this task, their sizes would be larger. It can be seen that Simple-DLN and Tiny-VGG are significantly smaller than the popular networks, and VGG-11 is the smallest in the VGG family.
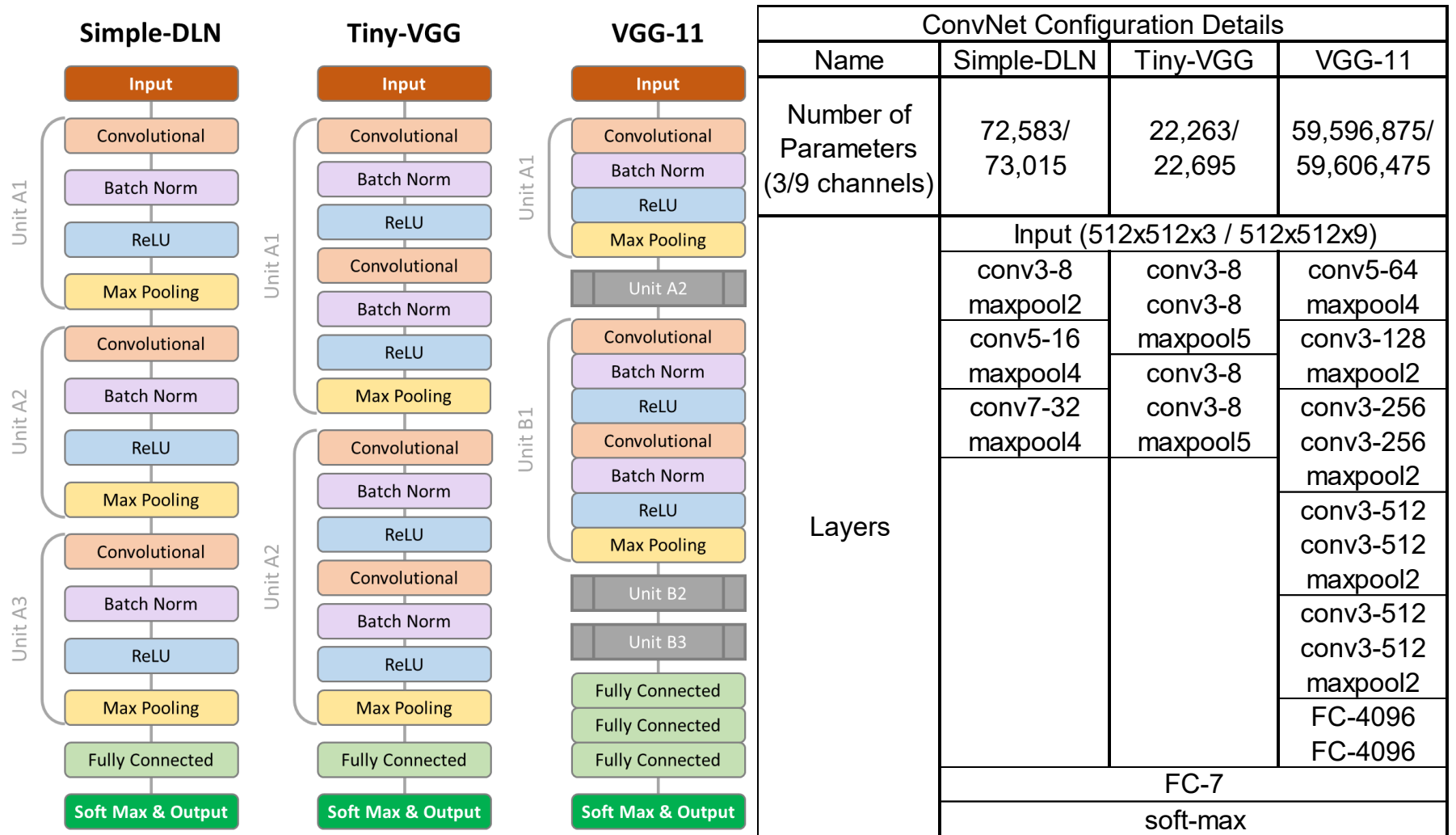
Figure 4-16. Layout (left) and configurations (right) of the selected CNNs.

ConvNet Configuration Details			
Name	Simple-DLN	Tiny-VGG	VGG-11
Number of Parameters (3/9 channels)	72,583/ 73,015	22,263/ 22,695	59,596,875/ 59,606,475
Layers	Input (512x512x3 / 512x512x9)		
	conv3-8 maxpool2	conv3-8 conv3-8	conv5-64 maxpool4
	conv5-16 maxpool4	maxpool5	conv3-128 maxpool2
	conv7-32 maxpool4	conv3-8 conv3-8 maxpool5	conv3-256 conv3-256 maxpool2
			conv3-512 conv3-512 maxpool2
			conv3-512 conv3-512 maxpool2
			FC-4096 FC-4096
	FC-7		
	soft-max		

Figure 4-17. Comparing the size of the chosen networks with popular pre-trained CNNs [342].

4.2.6 Algorithm for Network Training

The algorithm used for training the network, i.e. tuning the network parameters, is known as the Stochastic Gradient Descent with Momentum (SGDM), an improved variation of the classic backpropagation algorithm [336]. The SGDM algorithm introduced a term for momentum, and this can reduce the oscillation along the searching path of steepest descent towards the optimum. Moreover, since not all the networks have a pre-trained version (and the pre-trained networks are for different input image sizes), the training for all the trials starts with randomised parameter values rather than pre-trained values, so that the final accuracies are comparable.

Like most of the CNN training tasks, the images are divided into two groups: the training and validation groups. The training group is only used to tune the network parameters and is not for the validation (the procedure that uses the network to identify some images and calculate the ratio of correct results), and the validation group is used oppositely. During the training, the training images would be given to the network in batches and calculate the loss and gradient for adjusting the parameter values. Every image for training will be given to the network once and only once in an epoch, then the order of the images will be shuffled before the next epoch to speed up the training and minimize the effect of the presentation order to the network. Every five epochs, the validation images will be given to the network and classified. The validation during the training is used to check the overfitting. When the network is overfitting to the training group, the performance will only improve with time on the training images but not the validation images

Figure 4-18 shows a progress plot of a typical training (3-channel Simple-DLN). The final validation accuracy is different from the last validation point during the training because the batch normalisation layers operate differently during the training and final validation [343]. The wite-grey stripes correspond to the 30 epochs.
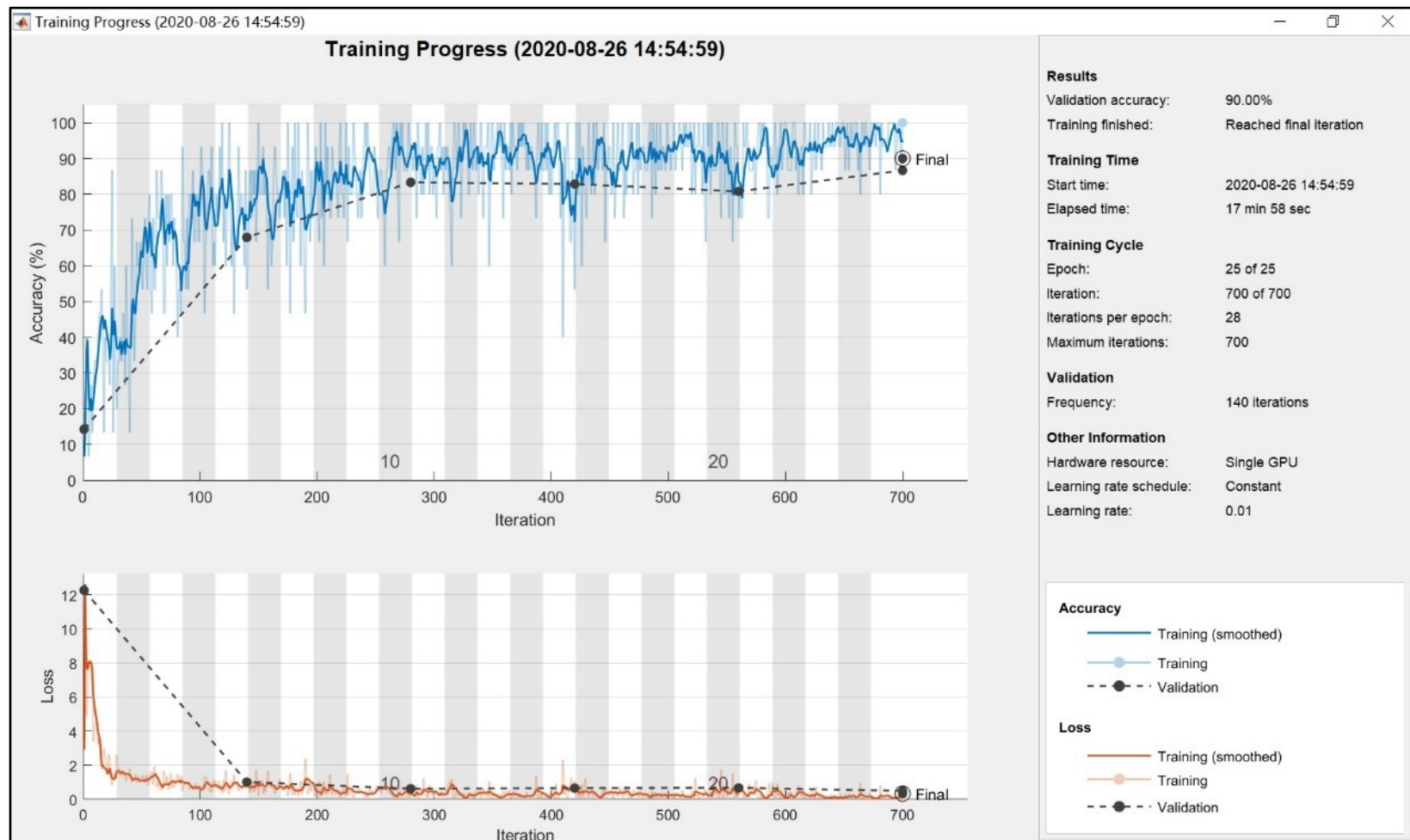
Figure 4-18. Progress plot of a network training with 3-channel Simple-DLN.
The wite-grey stripes correspond to the 30 epochs. The detailed trainng settings (which is the same for all the training) are shown on the right. A sign of overtraining is the accuracy for the training set remains higher than the validation set.

As mentioned in section 4.2.1, the numbers of images for different experimental conditions are not the same. According to the previous works done by Hensman et al. [317], the imbalanced training data will significantly reduce the performance of the final network. Therefore, the number of images for each case is set to be 200 because the smallest data set case (dark images) has 25 original images which correspond to 25*8=200 images after augmentation. About the division of the training and validation set, the initial plan without levelled training (see section 4.2.7 for details) was to set the training-validation ratio as 150:50 (4-fold training). Practices show that 30 epochs (or 30*150*7=31,500 images) seem to be a proper length of the training that balances the final performance and the overfitting.

4.2.7 Levelled Training for VGG-11

The practice shows that almost all the training trials of Simple-DLN and Tiny-VGG achieve 80%+ accuracy. However, when training the VGG-11, the network shows a meagre chance of convergence, and all the trials give 1/7 accuracy (completely random or constant output). It is understandable as VGG-11 has about 1,000 times more parameters than the other two networks (although VGG-11 is still one of the smallest deep learning networks in computer science), which is a very typical situation of over-parameterisation [344, 345]. Generally speaking, deeper networks have a higher degree of freedom and can learn more complex features. However, it is also a more difficult task to capture the same critical features in a bigger feature space.

In recent machine learning studies, there are mainly four categories of techniques for solving the difficulties caused by over-parameterisation: parameter pruning and quantisation, knowledge distillation, low-rank factorisation, and transferred or compact convolutional filters [344]. However, except for a subset of the last category known as transfer learning, the other methods involve a significant change in the network structure, which is beyond the scope of this investigation [346]. Nevertheless, there are no pre-trained

parameter sets for the networks used in this study, as they are still relatively shallow and previous studies on this aspect focused on deeper networks such as VGG-16/19 and Res-Net to maximise the reward of the pre-training process [347, 348]. More importantly, as mentioned in section 4.2.5, the accuracy comparison between networks is one valuable work in this chapter and using a pre-trained network would break the fairness of comparison. Therefore, transfer learning is not an option for this study.

For this task, a new approach is carried out, which is named "levelled-training" (Figure 4-19). It is straightforward to carry out and does not involve any modifications of the network. In the early epochs of training, only a subset of training data is given to the network. As expected, the network will learn both the "true" features and some features that only work in the smaller training data set (It is also the fundamental reason for over-fitting [349-351]). Then in later epochs, the entire training data set shall be given to the network, so that the network will learn which of the learned features are "true" features and which can be only used for "cheating" on the sub data set. This method reduces the difficulty of the training task because it increases the number of features that "works". Before using this method, the networks with an initial setting far from the optimal values are likely to pace randomly because all the neighbouring value sets perform almost equally bad (a case of gradient vanishing problem [352-354]). After adopting the "levelled-training", this situation is less likely to happen. Also, as shown in Figure 4-19, this method introduces extra "cheating" features for the network to learn, which only works on the tiny training data set. Therefore, become a double-edged sword: when the feature space is large (bigger networks), "levelled-training" can speed up the convergence, when the feature space is small (small networks), "levelled-training" may distract the feature learning.

In this task, the three training settings are shown in Table 4-6. Training setting 1 is the one without "levelled training". All the settings have the same total

number of images for training, which corresponds to the same number of evaluations.
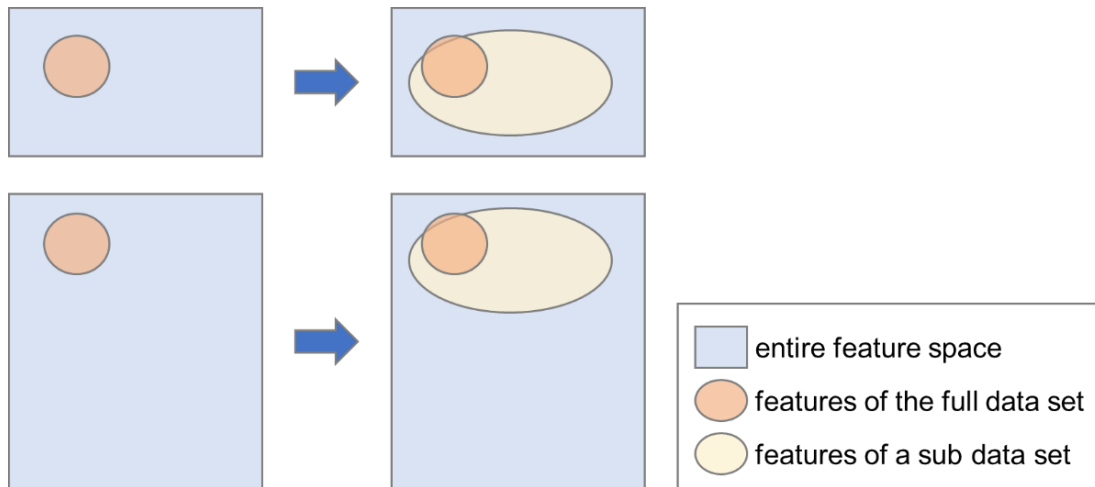
Figure 4-19. Illustration of the "Levelled-training".
Depends on the network depth, this approach is a double-edged sword: when the feature space is small (shallow networks shown at the top), "levelled-training" may distract the feature learning; when the feature space is large (deeper networks, shown at the bottom), "levelled-training" can speed up the convergence by broaden the valid feature space.

Table 4-6. Three different settings of the levelled-training.

Training Setting	Setting 1	Setting 2	Setting 3
Low-Level Training (60 images/ category)	0 epoch 0 images	25 epochs 10,500 images	50 epochs 21,000 images
High-Level Training (150 images/category)	30 epochs 31,500 images	20 epochs 21,000 images	10 epochs 10,500 images

Apart from the purpose of improving convergence, levelled training also corresponds to a practical situation. In the beginning, an engineer may notice the abnormal working condition, but there is not much data for training as the samples of abnormal conditions are very limited. Later as time goes, there might be more data available. This study will also answer how would this affects network training and the final accuracy.

4.2.8 Visualisation of the Extracted Features in CNNs

For a long time, the visualisation (or visual explanation) of the learned features of CNNs is a challenging but important task [347, 355]. It is of great value because it is constructive to understand which of the features are captured by the network so that users can check whether the network focuses on reasonable aspects of the graphical information. It is also helpful to find whether unexpected features (such as background colours) mislead the CNN training. The theories and methods for visualisation developed rapidly in recent years. Till the time of writing this thesis, there are mainly four popular method classes in this field. The detailed reasoning is given as follows:

1. **Activation Map**
   Activation maps display the activation levels of a specific CNN layer. It can tell how different layer neurons focus on different features (Figure 4-20) [356, 357]. This method is adopted in this project, and the results show that it is helpful to illustrate the interested features captured by the trained networks.
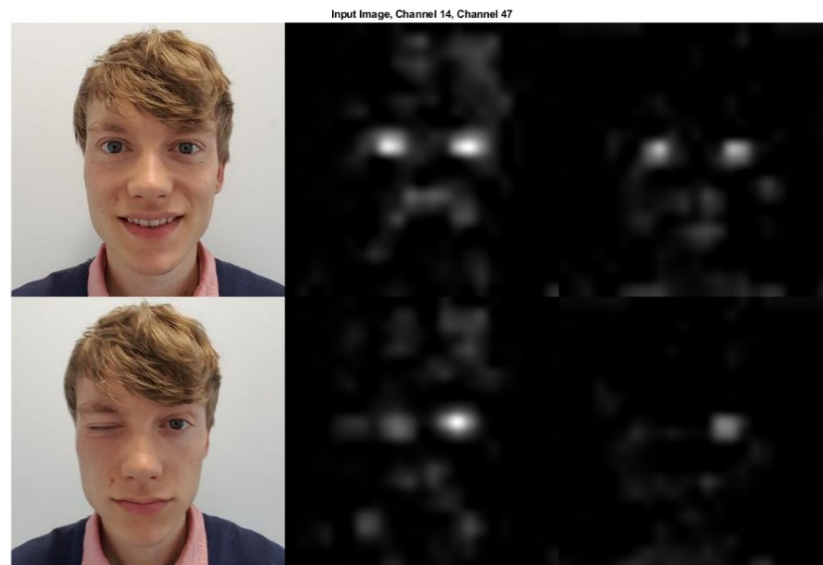
Figure 4-20. Activation map example provided by MathWorks [358]. This map shows that two neurons of CNN focus on the presents of opened human eyes.

2. Class Activation Mapping (CAM) and Gradient-Weighted CAM

The CAM graphs show how different image areas contribute to the activation of the final answer [359]. In other words, which part of the input image make the CNN decide the output. In this project, the Gradient-Weighted CAM (as a generalisation of CAM [355]) is adopted. The method is helpful for feature explanation and is even more helpful when the decision is incorrect (Figure 4-21).

buckle, 0.14911
sock. 0.087194
mailbag, 0.056052
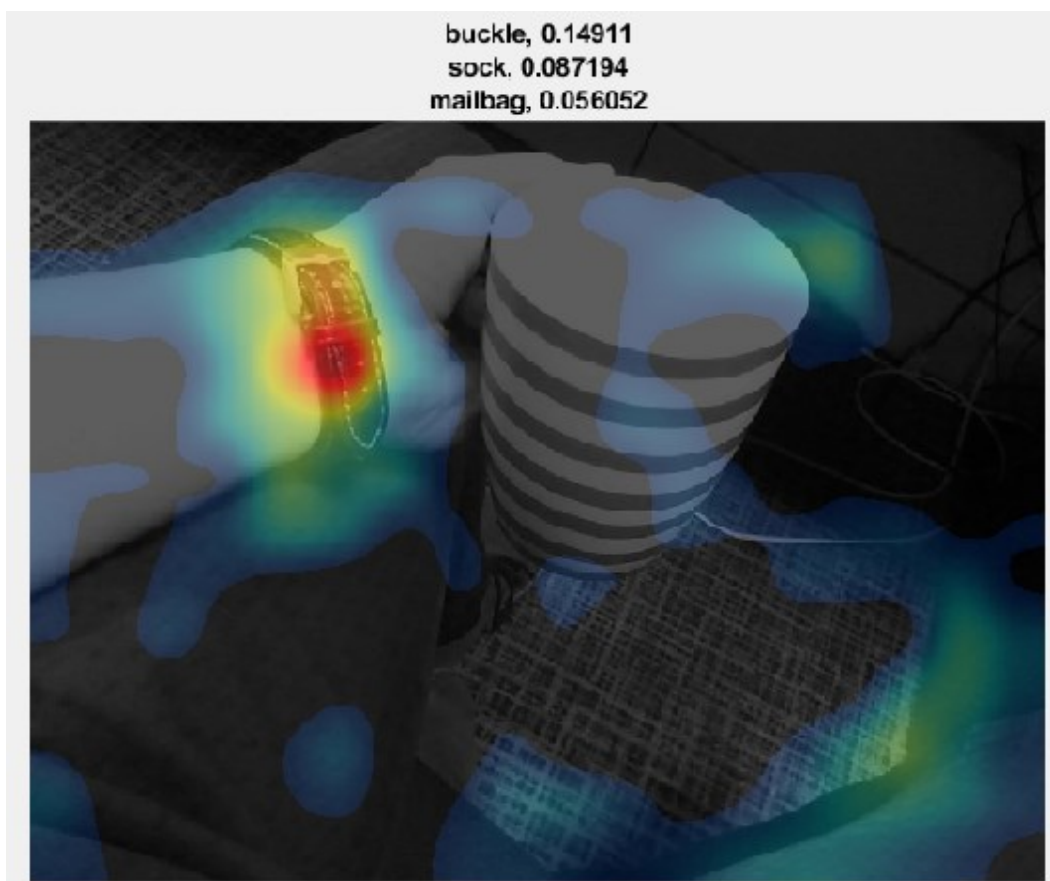
Figure 4-21. CAM of a CNN processing an image of a cup [360].
This image of a cup is recognised as a buckle because the network
pays more attention to the watch on the arm.

3. Feature Map / Deep Dream Image

Feature maps are generated images that could lead to similar CNN activation responses at a specific layer compared to a specific input image [361, 362]. Figure 4-22 shows an example of a VGG-19 network. They are also called Deep dream images because for deeper (closer to the output layer) layers, the feature maps are usually crowded with rescaled and relocated features, which could be kind of "bizarre".

This method is not adopted in this project because the channel of these maps corresponds to the input channel number, so it only works best with the 3-channel-input networks. Nevertheless, different from Figure 4-22, in this work, the three channels are not RGB but for DIC/Sulforhodamine/Citrine, which is not very intuitive. Moreover, the network in this project is trained to work with images with any four directions of the microfluidic chip. As a result, the features are symmetric, relocated, and overlapped. The feature maps in this project only show a few features like the shape of the chamber (relocated and not clear at all) but nothing else.
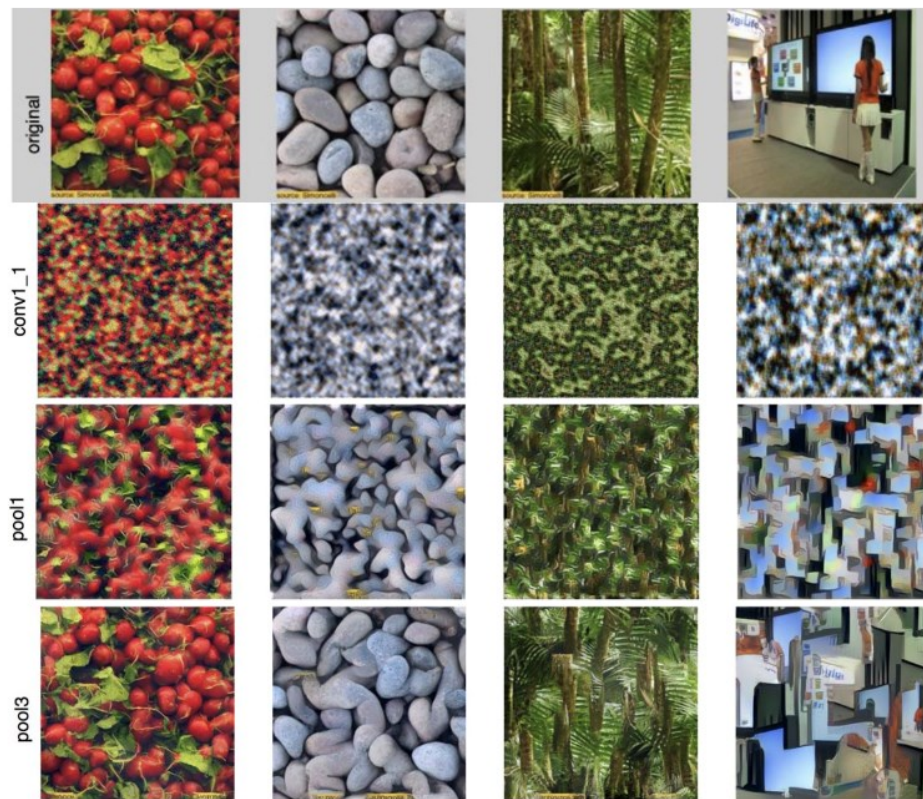
Figure 4-22. Example of feature maps of a VGG-19 network [362]. Lower rows correspond to deeper layers.

4. Occlusion Sensitivity Map

When generating these maps, the input image is partially covered with occluding mask. According to how the output changes with these masks, the map can show which parts of the picture are the more critical or informative for the final classification [363].

This method is not adopted in this project because practices in this study find that the occluding masks affect not only the fault diagnosis but also the understanding of the device orientation. The central area is always considered "very important" in most sample cases because it contributes least to the orientation judgment and is only informative for abnormality diagnosis. Oppositely, the area which actually contributes most to the diagnosis is not highlighted.

4.3 Results

4.3.1 Results of the Network Training

The tables below show the accuracy of trained networks. There are ten trials for each training setting. Two channel numbers correspond to two different image-pre-processing methods given in section 4.2.3, i.e., with/without normalisation. Three training methods stand for three ratios of the low/high-level training as introduced in section 4.2.7 (method 1 do not have low-level trainings and method 3 give the highest time ratio to the low-level training) Chance of convergence is the percentage of trials with validation accuracy larger than 14.29% (1 out of 7, i.e., constant or completely random output).

Table 4-7. Results of the CNN training.

Network	Features	3 Channels (With Normalisation)			9 Channels (Without Normalisation)		
		Method 1	Method 2	Method 3	Method 1	Method 2	Method 3
Simple-DLN	Accuracy (mean±std)	96.80±3.76	96.74±1.69	92.86±4.25	99.34±0.57	98.17±1.95	98.37±1.20
	Best Trial	99.14	98.57	98.29	100	100	100
	Chance of Convergence	100%	100%	100%	100%	100%	100%
Tiny-VGG	Accuracy (mean±std)	87.37±5.92	87.29±4.06	84.49±3.40	97.57±1.46	96.13±3.77	97.94±2.16
	Best Trial	96.00	93.71	89.14	99.71	99.14	100
	Chance of Convergence	90%	60%	100%	100%	90%	100%
VGG-11	Accuracy (mean±std)	88.14±3.86	79.22±14.31	70.16±4.78	-	55.57±6.14	-
	Best Trial	92.00	93.71	75.14	-	61.71	-
	Chance of Convergence	30%	70%	90%	0%	20%	0%

Table 4-8. Wilcoxon-rank-sum-tests of the median validation accuracies.

Comparison		Simple-DLN	Tiny-VGG
3-channels vs 9-channels	Method 1	9-channels better (p*<0.01)	9-channels better (p*<1e-4)
	Method 2	9-channels better (p*<0.05)	9-channels better (p*<0.01)
	Method 3	9-channels better (p*<0.01)	9-channels better (p*<0.001)
Between Methods (3-channels)	Method 1 vs 2	not significantly different (p*>0.1)	not significantly different (p*>0.1)
	Method 1 vs 3	method 3 worse (p*<0.01)	not significantly different (p*>0.1)
	Method 2 vs 3	method 3 worse (p*<0.05)	not significantly different (p*>0.1)
Between Methods (9-channels)	Method 1 vs 2	not significantly different (p*>0.1)	not significantly different (p*>0.1)
	Method 1 vs 3	not significantly different (p*>0.1)	not significantly different (p*>0.1)
	Method 2 vs 3	not significantly different (p*>0.1)	not significantly different (p*>0.1)

There are six different network configurations in Table 4-7 (training methods do not change the network configuration), and only Simple-DLN (9 channel ver.) and Tiny-VGG (9 channel ver.) achieved 100% accuracy for the validation set. Table 4-8 shows that although the normalisation reduced the bandwidth for abnormality screening to 1/3, it also leads to information loss and reduces the screening accuracy.

To investigate the performances of the other four networks that did not achieve 100% accuracy, and understand what mistakes these networks are making, the validation results of the best networks of these four configurations are shown in Sanky charts from Figure 4-23 to Figure 4-26.

Figure 4-23 shows that the best simple-DLN (3 channel ver.) is too strict for the standard of normal experiments, and confuses some blurry images with moving images. The confusion makes some sense, as for 3-channel cases, both the moving device and blurry images will lead to blurriness in the image, the difference is in the direction of blurriness. Figure 4-24 and Figure 4-25 suggest these two networks have an unacceptable level of both type I and type

II errors for normal experiments. The results in Figure 4-26 is the worst, and the network makes many severe mistakes in the diagnosis. Although VGG-11 for 9 channels has the largest number of learnable parameters, over-parameterisation seems to cause severe problems in this case.
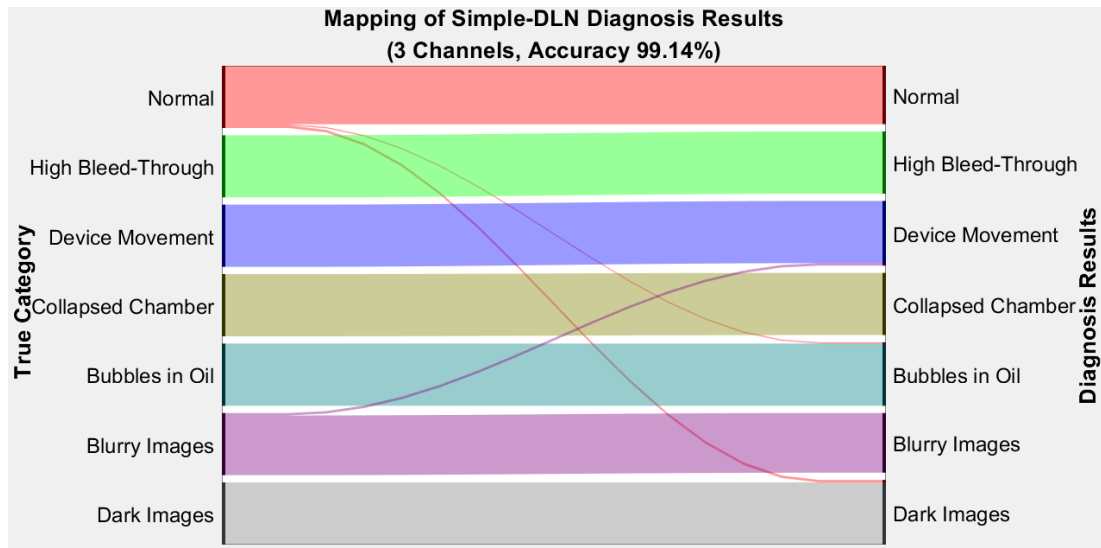
Figure 4-23. Sanky charts of validation results for the best Simple-DLN (3 channel ver.).
This plot shows how the validation images of true categories (left) are identified by the network (right).
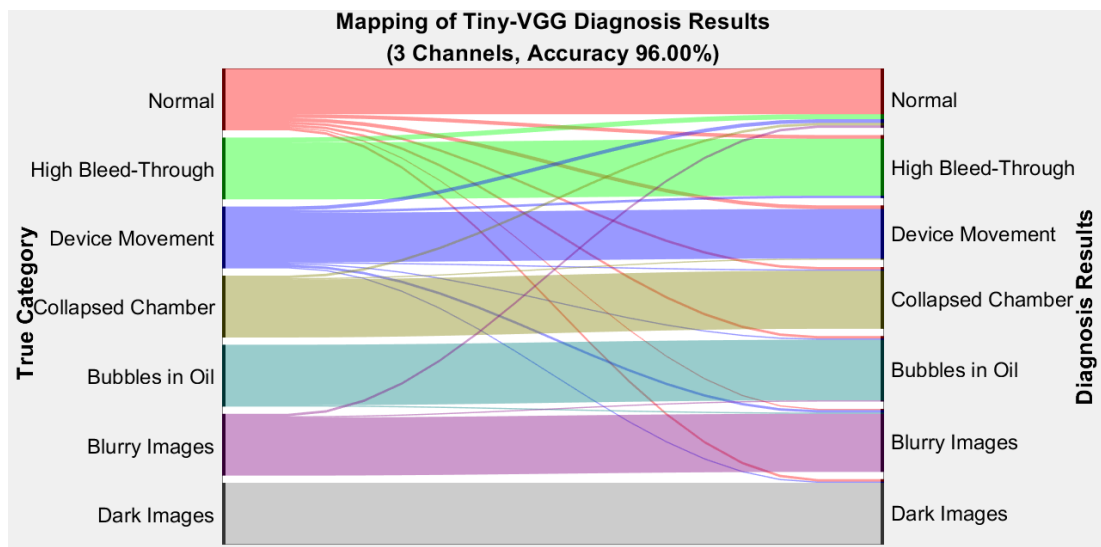
Figure 4-24. Sanky charts of validation results for the best Tiny-VGG (3 channel ver.).
This plot shows how the validation images of true categories (left) are identified by the network (right).
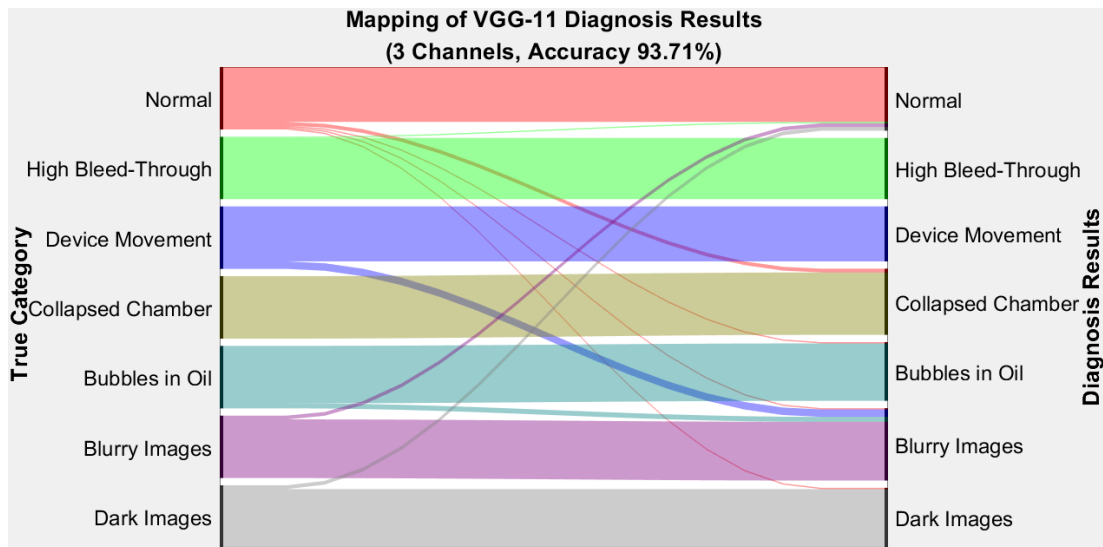
Figure 4-25. Sanky charts of validation results for the best VGG11 (3 channel ver.).
This plot shows how the validation images of true categories (left) are identified by the network (right).
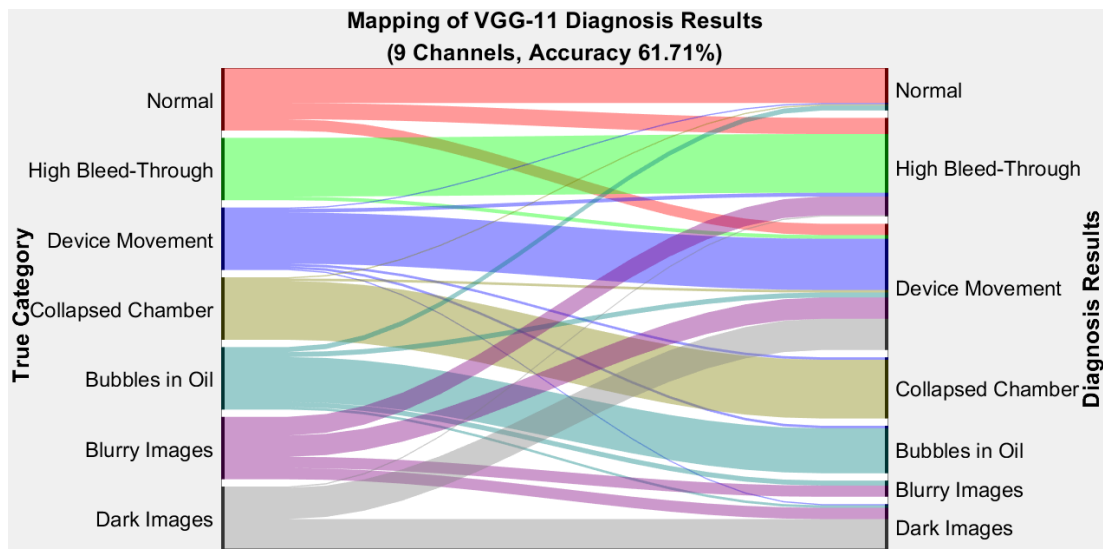
Figure 4-26. Sanky charts of validation results for the best VGG11 (9 channel ver.).
This plot shows how the validation images of true categories (left) are identified by the network (right).

4.3.2 Comparison of Different Training Methods
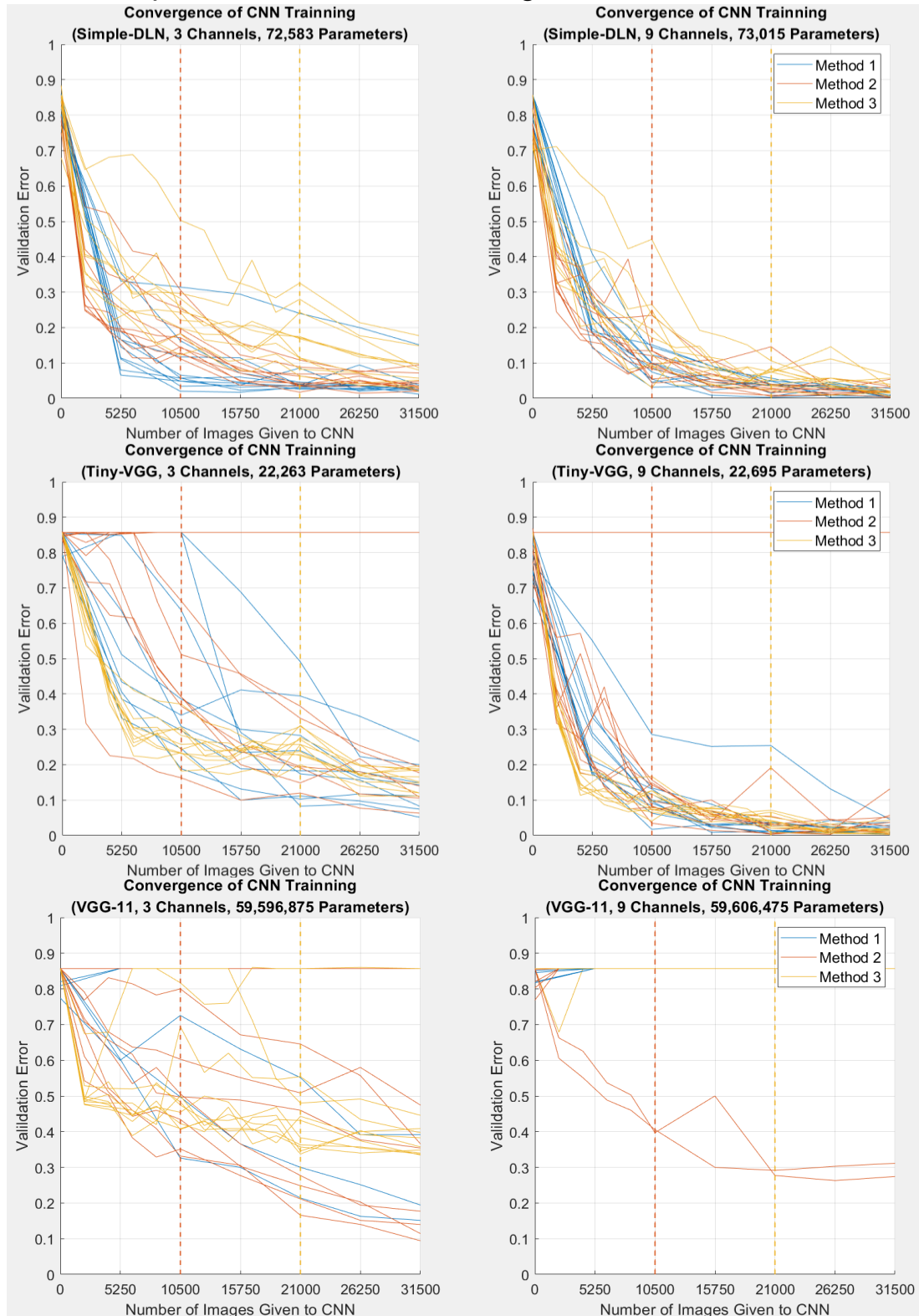
Figure 4-27. Convergence curves of the CNN training with three training methods.
method 1 do not have low-level trainings and method 3 give the highest time ratio to the low-level training.

As listed in Table 4-6, three training methods differ in the ratio of low/high-level training. High-level training means training the network with all the images from the training set (150 images per category) in every epoch; low-level training means using a fixed subset of the training images (60 images per category) for the CNN training. Method 1 trains the network with high-level all the time (classic method), method 2 and 3 starts with low-level training and switches to high-level at 1/3 or 2/3 of the entire training session as marked with dashed lines in Figure 4-27.

In most cases, low-level training leads to faster convergence at the beginning (red and yellow lines are lower than blue lines), but very soon, convergence slows down until switching to high-level training. A clear drop at the training level switch can be seen in some network configurations, particularly simple-DLN (3 channel ver.) and tiny-VGG (3 channel ver.).

For VGG-11, a network with 1,000 times more parameters than the other two, levelled training seems to increase convergence chance. The comparison between methods 2 and 3 suggests that the switching time is also an interesting aspect to investigate. The meagre chance of convergence with 9-channel inputs shows that these deeper networks may need different training strategies than shallower networks, e.g., loading pre-trained networks or training with much more training data.

4.3.3 Visualisation of the CNNs

Gradient-weighted CAM and activation maps of the best networks of the six configurations are given in Figure 4-28 to Figure 4-30. One sample image is randomly selected from each category to generate these plots.

As introduced in section 4.2.8, GW-CAM represents the network's overall attention, so there can only be one GW-CAM for one input image. Activation

maps show how different channels (or neurons) of a layer get activated by the input (remember that the core mechanism that backups the CNNs is the inference base on the neurons activated for different image features). The selected neurons are the eight neurons of the largest variances in activation with different image inputs. Here the activation maps are generated for the last ReLu layer to capture the most complex features with minimum resolution lost.
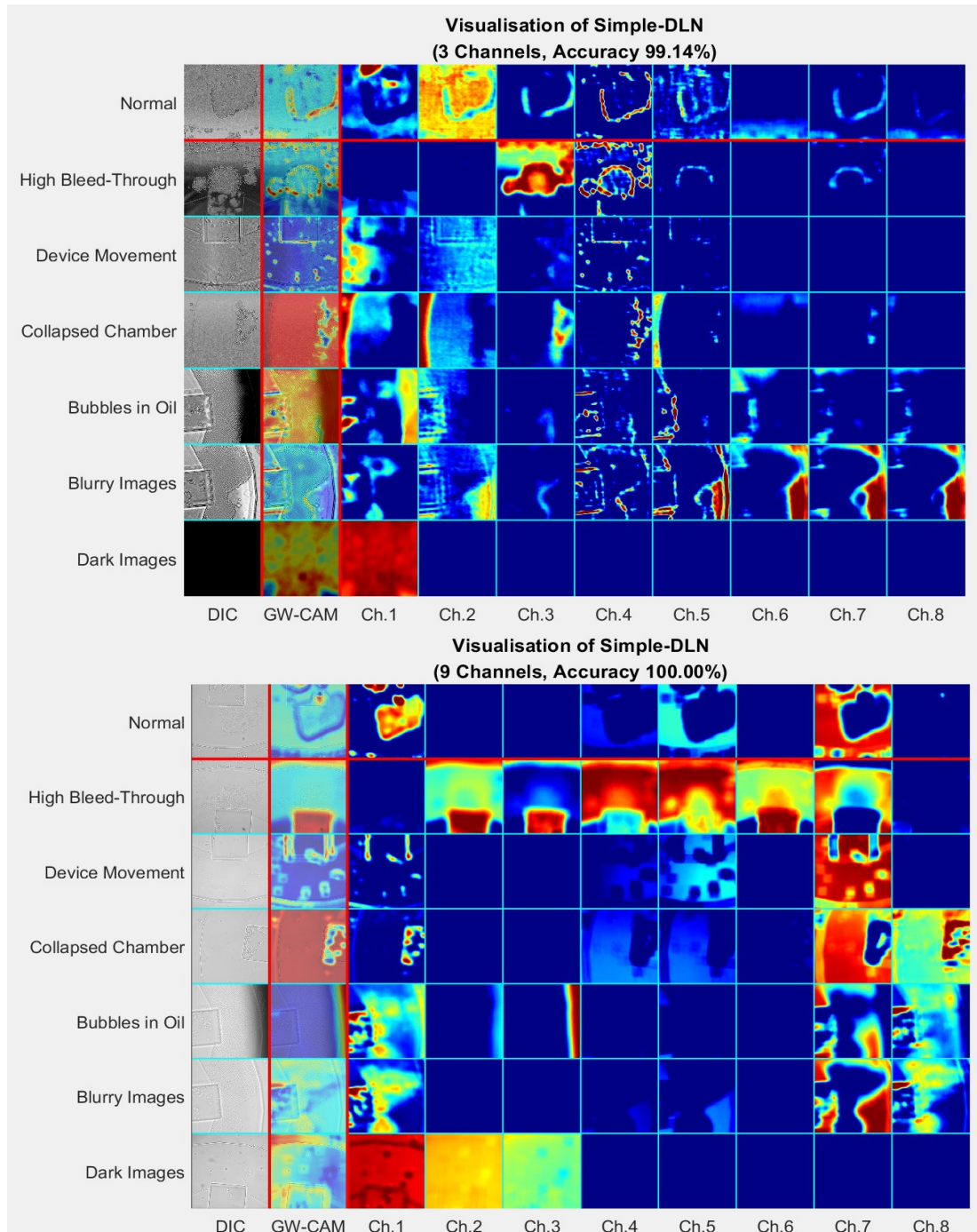
Figure 4-28. Visualisation results of the best simple-DLN networks.
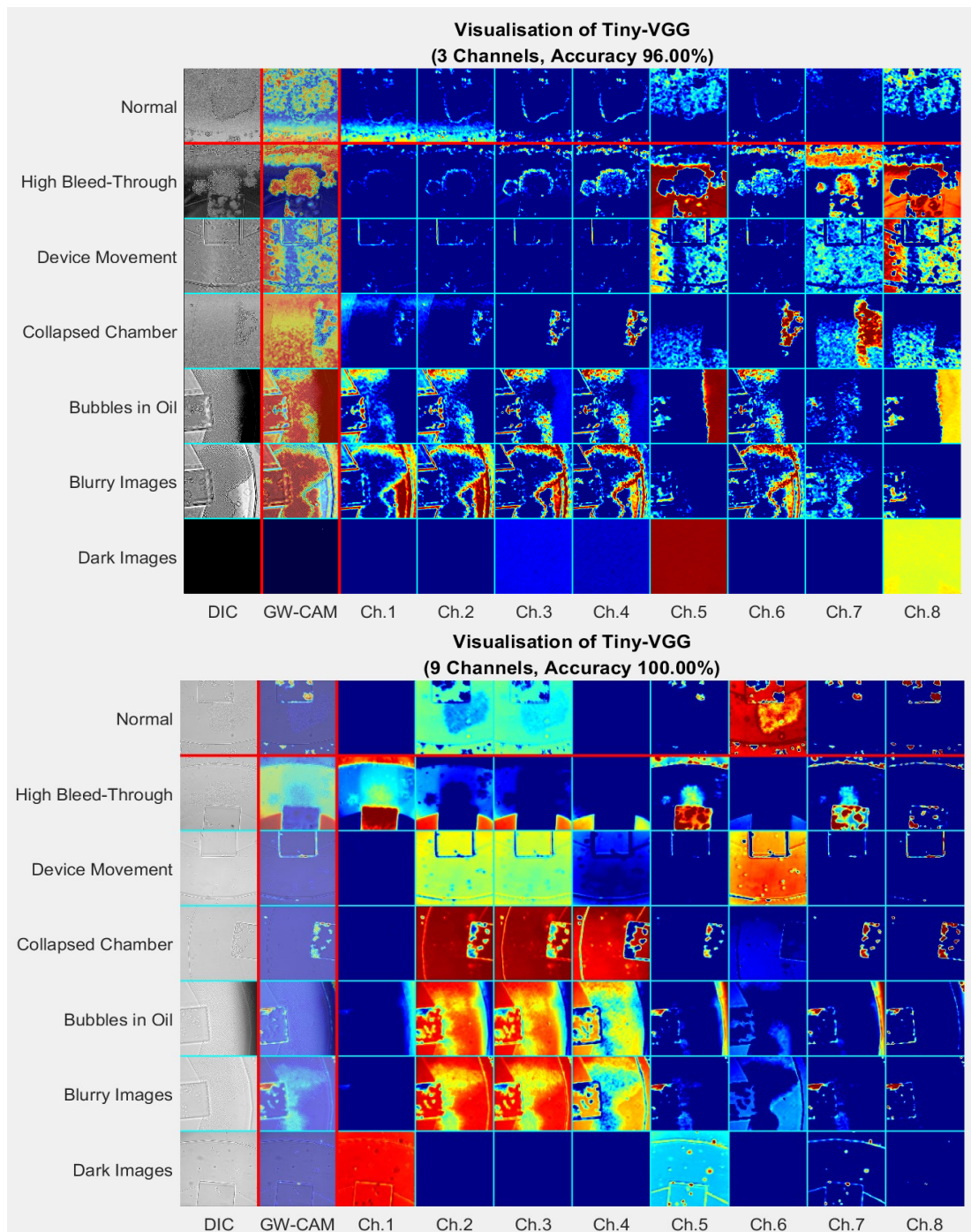
Figure 4-29. Visualisation results of the best tiny-VGG networks.

From the figures, it can be seen that there are some channels/neurons gives very similar activation maps. In other words, they focus on identical features. This suggests that network pruning (a procedure removing these replicated neurons) could be applied to reduce the network size while remaining the accuracy.

For VGG-11, the layer is the second last ReLu because the resolution of the last ReLu is too low (8 by 8 pixels).
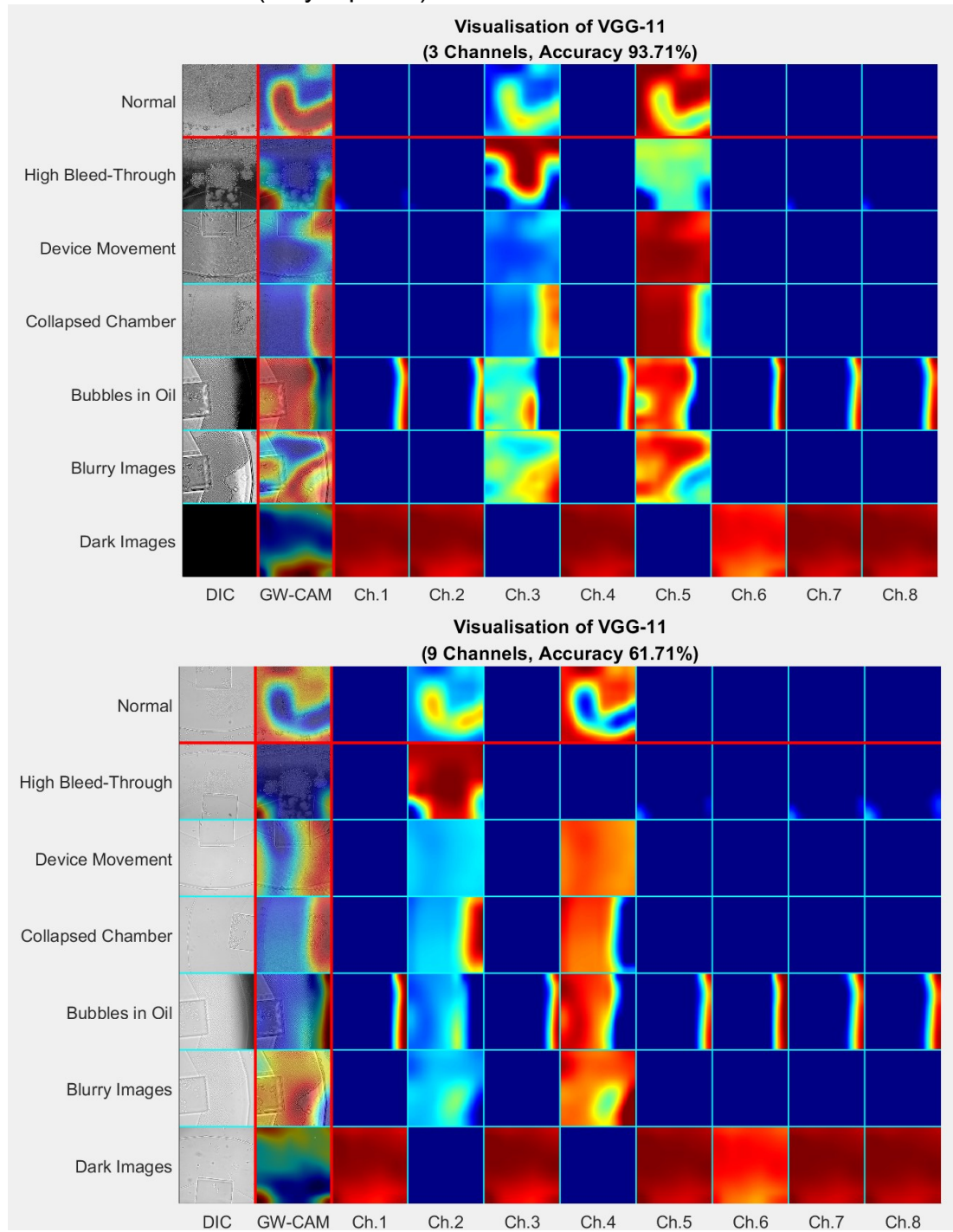
Figure 4-30. Visualisation results of the best VGG-11 networks.

The information density of these graphs is very high. Combining with the figures in section 4.3.1, experts can obtain lots of information to guide the network's configuration, training, and application.

In all of these figures, some channels capture similar features. It suggests that pruning can be applied to reduce the network size without losing accuracy [364, 365], and an optimised network may only need 3-5 channels (neurons) in the last layers. GWCAMs show that the networks can easily identify the yeast cells and pay attention to the colony's borders.

In Figure 4-28, to identify blurry images, 3-channel simple-DLN tends to look for a particular pattern of the blurred empty chamber (channel 6 to 8). However, when the chamber is almost full, this feature would disappear. It might be why some blurry images are identified as devise movement problems in Figure 4-23.

Figure 4-29, shows that the activation maps of the normal experiment for 3-channel tiny-VGG have some common styles to some failures, such as device movement. In other words, the 3-channel network does not capture unique features for normal experiments, so the error level is high in Figure 4-24. Also, for device movement, the 3-channel network seems to rely on sub-ideal features compared to the 9-channel network. Nevertheless, it might because the training was not long enough, supported by Figure 4-27.

In Figure 4-30, for both 3-channel and 9-channel networks, it seems that only two to three channels provide the majority of evidence for the inference. Furthermore, the features in the 9-channel network are blurrier than the 3-channel ones. It may explain why the 9-channel network performs worse than the 3-channel network.

4.4 Discussion

4.4.1 Comparison of Channels and Networks

From Table 4-7 in section 4.3.1, although simple-DLN has the smallest parameter number, it shows the best performance, with 99.14% accuracy for 3-channel inputs and 100% accuracy for 9-channel inputs.

From Table 4-8, the performance of two smaller networks (simple-DLN and tiny-VGG) is significantly better with 9-channel inputs compared to 3-channel ones. It is understandable because image pre-processing (the procedure that reduces the channel number, section 4.2.3) with the cost of losing some information.

The deeper network (VGG-11) shows different behaviour and performs better with 3-channel images. VGG-11 also performs the worst in both accuracy and chance of convergence. A possible reason is the over-parameterisation [366-368]. In other words, the entire feature space for VGG is too large that the difficulty of finding specific valid features is higher than shallower networks.

4.4.2 Comparison of training Methods

According to Figure 4-27 in section 4.3.2, low-level training (training with a smaller data set) may speed up the convergence and increase the chance of convergence. Nevertheless, high-level training is also necessary to furtherly increase validation accuracy.

As mentioned in section 4.2.7, the introduction of low-level training is not only for improving the convergence but also for investigating how the limited training data at an early stage affects the final accuracy. According to Table 4-8, the levelled training does not significantly reduce the validation accuracy, except in method 3 cases with Simple-DLN. In other words, if training the

network needs time, and there are expected to be more data available for training, starting the low-level training while waiting for the new data may be a wise choice, as it will save time while maintaining the accuracy. As long as the high-level training (training with all the data) takes a high enough share of the entire training time, the final accuracy should be not significantly worse than using high-level training all the time. Although, the exact necessary share for high-level training still needs further investigation.

4.4.3 Visualisation of CNN

The visualisation practices in section 4.4.3, the combination of activation map, gradient-weighted class activation mapping (GW-CAM), and the Sanky charts of validation results show extraordinary power to explain the network and guide the configuration, training, and application of the network. This combination is particularly helpful for the trainer to know why the network makes mistakes and how to adjust the training.

The activation map shows the interested features for each neuron of a layer to understand if there are any missing or misleading features in the network. Moreover, suppose the network is confused with two image categories. In that case, the network may capture sub-optimal features for one of the categories, and the activation map of these two categories would be very similar. The activation map could give hints of the features to adjust the images for training to let the network pick up with better features.

GW-CAM is helpful when there are too many neurons in a layer, as GW-CAM shows one overall attention map of the CNN, which plays a similar role as the activation map. In practice, having both the activation map and GW-CAM together is better because activation maps give detailed features, and GW-CAM helps check if there are some cross-neuron features.

4.4.4 Future Works with More Advanced Networks

The work in this thesis verifies the application of CNN for increasing the automation level of anomaly screening for microfluidic experiments, and it is yet at the early stage of exploration. As introduced, machine learning is an attractive and active field and there are new theories and network mechanisms proposed every year. It is very exciting to investigate how these results can improve the accuracy and efficiency in this task: Residual neural networks may lead to better generalization ability and accuracy while remaining a small memory usage [369, 370]. According to the visualised mapping of current networks, network pruning is an approach that can remove the repetitive neurons and reduce the network size without losing classification accuracy [364, 365]. Moreover, the hybrid CNN–SVM approach has shown state-of-the-art accuracy in image-based anomaly detection tasks [371-374].

4.5 Chapter Summary

4.5.1 Fault Diagnosis

This chapter's work shows that CNN can be used in microfluidic experiments and diagnose six faults that once appeared in practices. Even shallow CNNs with less than 23,000 parameters could achieve very high validation accuracy (achieve 100% for multiple times in the trials). Analysis with visualisation suggests the network can be furtherly pruned to smaller sizes without accuracy lost. The CNNs of this size can be trained and run on a computer without high-end GPU or computing clusters. In comparison, most of the pre-trained CNNs in the field of computer science have 1,200,000-150,000,000 parameters [341, 375]. With this work, mass-application of parallel microfluidic experiments can be carried out with a higher automation level and better control of abnormal experimental conditions.

Among the three CNNs introduced in section 4.2.5, after the same amount of training with the random initialisation, simple-DLN shows the best performance; tiny-VGG gives slightly worse accuracy, probably because of the over-parameterisation VGG-11 performances the worse.

Image pre-processing reduces the input channel number from 9 to 3. By reducing the data size or bandwidth, but this procedure reduces the validation accuracy.

4.5.2 Training the CNN

Levelled training is introduced in this chapter. This training method divides the CNN training into two sub-parts: low-level training, which only uses a subset of training images; and high-level training, which uses all the images. Results show that this method increases the convergence speed and the chance of convergence (particularly for deeper networks like VGG-11). Moreover, for the long-time CNN training with limited initial accessibility of all the data (a practical situation in the industry), with this method, the training can start earlier without waiting for all the data set, and the final accuracy is not significantly worse than training with full data set all the time. Moreover, the minimum share for the high-level training to maintain the final accuracy is a question to answer in future works.

Considering the connection between CNNs and the human brain's neural networks, levelled training results are also consistent with human learning. It is easier to start with simpler tasks in education and a small number of questions and then move to more complicated tasks. Learning with all kinds of tasks since the beginning leads to a heavier cognitive load and usually results in a slower learning speed [376]. On the other hand, keeping practising with a small number of simple tasks is also not efficient to further develop the skills [377].

This study also shows that the combination of activation map, gradient-weighted class activation mapping (GW-CAM), and the Sanky charts of validation results helps explain the network and guide the configuration, training, and application of the network. This combination is particularly helpful to understand the mechanism of the incorrect diagnosis and infer the adjustments for the training.

Chapter 5:   Conclusions and Perspectives

This thesis explores, practices, and examines several methods to improve the efficiency of model calibration experiments for synthetic biological products in the three aspects: experimental technique, optimal experimental design (OED), and automatic experiment abnormality screening (AEAS). In this thesis, a model of an orthogonal inducible promoter in *S. cerevisiae* is used as a benchmark. A series of microfluidic experiments are carried out on this cell strain to support the research.

5.1 Conclusions

- **Adopting microfluidics experimental technology is helpful to improve the efficiency of model calibration.** The comparisons on the benchmarking model suggest that with the same experiment duration, microfluidic experiments lead to models that provide more accurate behaviour predictions of never-encountered stimuli patterns than those derived from flow-cytometry experiments.

- **The study confirmed the benefit of current optimal experimental design (OED) approaches for model calibration.** Compared to randomised input designs, current OED approaches lead to more accurate estimations of model parameters.

- **Parameter clustering reduces the computational cost of the OED and improves the current off-line OED approach.** OED with clustering leads to equally or even more informative experiments than the classic off-line OED (optimise all the experiments at the very beginning). However, it is not as informative as the on-line OED (an approach that optimises every

sub-experiment before carrying them out). Clustered-OED may be a better solution when the parameter estimation and OED are not physically allowed to carry out during the experiment.

● **The developed automatic experiment abnormality screening (AEAS) system with convolutional neural networks (CNNs) can diagnosis abnormal experimental conditions with the desired accuracy.** This thesis' work shows that CNN can be used in microfluidic experiments and diagnosis the six abnormal conditions in seconds with the desired accuracy. The trained networks achieve 100% accuracy multiple times in the validation.

● **AEAS system with CNNs of small size can already achieve very high diagnostic accuracy.** Practices show that even shallow CNNs with less than 23,000 parameters could achieve very high validation accuracy (achieve 100% for multiple times in the trials).

5.2 Judgement of the Results

● **Comparison between flow-cytometry and microfluidic microscopy.** The work in this thesis compared the current microfluidic microscopy experiments and flow-cytometry experiments for calibrating a benchmarking model. Although in this study microfluidics shows higher informativeness for model calibration, it is very important to point out that there are improvements published every year for both of the techniques. So this work should not be considered as a certain and definite comparison, but rather an example of comparing different experimental techniques and to provide evidence for deciding experimental designs and understanding the difference between results with the other technique.

- **Optimal experimental design with parameter clustering.** This study explored a new approach to guide the optimal experimental design with the clustering of model parameters. Different from previous studies, this approach does not decide which parameters to fix/fit, but leave this point with researchers (which is a practical need) and decide which parameter set to focus on in each sub-experiment. Although the work in this thesis shows the advantage of this approach, the work is based on in-silico experiments and a relatively small model. To achieve a more solid understanding of this approach, it would be needed to validate this work with in-vivo experiments and with larger models. How the parameter clustering would be like for complex models is a very interesting and valuable point to investigate.

- **Automatic experiment abnormality screening based on experimental images.** This work explores the possibility of increasing the automation level with convolutional neural networks. Although the networks achieve state-of-the-art accuracy, because of the practical limits, the sample size is very small compared to the database for more general purposes. Moreover, recent developments in the related field may also achieve accurate diagnosis with smaller networks. How to generate a better training data set with limited resources, and would network achieve better performance with more recent designs are the questions to answer in the future.

5.3 Future Perspectives

- **Wet-lab validations of the on-line OED approach would be very helpful to confirm its benefit in model calibration.** The study based on in-silico experiments suggests that on-line OED would lead to the most accurate parameter estimations. Wet-lab experimental data would be strong evidence to validate this conclusion.

- **Furtherly improve the structure of the CNNs for abnormality screening applications.** Machine learning is a very active research field. Many newly proposed techniques have the potential of furtherly improving the accuracy and performance of the network, such as residual neural networks and network Pruning.

Appendices

Appendix A: Protocol of Microfluidic Experiments

Outline of the protocol:

This protocol describes the procedures of an in-vivo experiment for model inferencing, on genetically modified yeast strain (YST_526) in a microfluidic device (MFD0005a) with a fluorescence microscope (Nikon Ti2-E). The experiment is supported by a MATLAB/Python experimental platform developed by our group.

Media preparation

6*50ml Test Sample Vial Tube

Tube No.	Use	Total Volume	SC Media	Glucose 20%	IPTG 100μM	Sulforhodamine 1mg/ml
1	Syringe 1	10ml	8.89ml	1ml (2%)	100μl (1μM)	10μl (1μg/ml)
2	Syringe 2-5	40ml	36ml	4ml (2%)		
3	Cells Overnight, OD check, and Re-diluted Cells	30ml	27ml	3ml (2%)		

All these media need to be filtered with 0.22μm filter.

Cell preparation

- Move the media from tube 3 to 4*50ml vial tubes, 5 ml each, and 1 ml to an Eppendorf.

- Use a P200 pipette pick a colony of average size (not too big) from the petri dish and re-suspend it in the Eppendorf. Then transport 60ul to one of the tubes. Prepare another one with 1 whole colony into another tube.

- Place the two 50ml tubes in the incubator at temperature 30.0 degree, 250RPM, grow overnight.

- Measure the OD of the culture. Then dilute them to OD=0.075 in the other 2 tubes.

- Put the two tubes with diluted culture into the incubator.

- Calculate the time to wait based on the following data (OD for the mid-exponential phase is 0.3):

Time (min)	OD
0	0.075
110	0.100
190	0.150
290	0.300

- Finally measure and take note of the OD of the culture in 50ml and load the one closer to 0.3 to syringe 6.

Syringes' preparation

- Prepare 6 25G needles and 6 lengths of tubing[1] (4x (1 arm) and 2x (2 arms)).

- For each length of tubing, connect one extremity to the 25G needle.

- Connect 6 x 50ml syringes to the prepared tubing. Upon connection, fill and empty the syringe with air 3 times, to ensure all debris will be removed from the tubing. Label the syringes with sequential numbers from 1 to 6 (1 and 2 with the longer tubing) and anchor the connection pin on a side (at the height of 10 ml, to prevent backflow), using autoclave tape.

- For each syringe (1-5), remove the piston and, using a p1000 (or P200), load 1 ml of fluid (according to the table below) at the bottom of it (make contact with the inside of the leur stub adapter). This reduces the formation of bubbles that would prevent the flow of fluid in the tubing. Using a 10 ml stripette, load the residual 9 ml of the same fluid (let it run on the syringes wall before reaching the bottom).

- Cover each syringe with parafilm and, using the tip of scissors, create a hall on it.

Syringe No.	Content	
1	SC media 2% Glucose 1mM IPTG 1ug/ml Sulforhodamine B	8.89ml SC media 1ml 20% Glucose 100µl IPTG 10µl Sulforhodamine B
2	SC media 2% Glucose 0 IPTG	9ml SC media 1ml 20% Glucose
3	SC media 2% Glucose	9ml SC media 1ml 20% Glucose
4	SC media 2% Glucose	9ml SC media 1ml 20% Glucose
5	SC media 2% Glucose	4.5ml SC media 0.5ml 20% Glucose
6	cells with PLac + IPTG	5ml cells with PLac + IPTG

Gently move the syringes to the microscopy room and attach them: 1 and 2 to the actuators (2 should be higher than 1 and both higher than all other syringes); 4 and 5 (23 cm from the height of the stage) and 3 (18 cm from the height of the stage).2

- For each syringe (1-5), detach the connection pin and watch the fluid flow through the microbore tubing line to the exit point at the leur stub. Double check that no bubbles are present.

Wetting the chip

- Prepare a syringe for wetting. Connect a 5 ml syringe to a short length of tubing (using a 25G needle as for the others). Fill and empty the syringe with air for 3 times to remove potential debris from the tubing. Fill the syringe with media. Let media flow in the line, to remove all air bubbles.
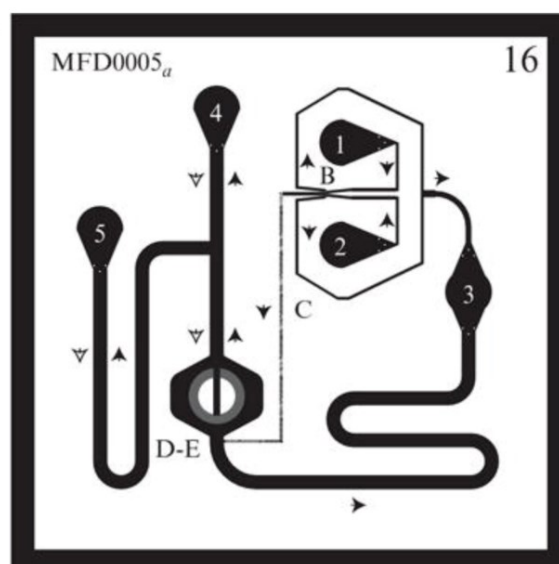
- Place the device in the lid of a petri dish (acting as a chip holder), eventually securing it with tapes on two sides.

- Connect the syringe to port 5 and apply a gentle pressure (too much pressure will lift the chip off the coverslip). When droplets form, move the connector to port 4. Repeat the procedure for 3, 1, 2, until there are no more air bubbles in the chip. Use a Kim wipe to remove excess of media on the surface of the chip.

- Wash the hands with ethanol before wetting the device, and avoid touching any kind of paper to avoid bringing fibres into the device.

- Wetting the device is better in the order as follows:

- Brief wetting: 5,4,1,2,3. (4 and 3 with more pressure and longer time);

- Remove air bubbles in 1&2. (for small bubbles, try gently suck the bubbles out and push the media in very slowly, the bubbles may not return back to the chip);

- Remove air bubbles near port 3. Then check if there are any bubbles from port 3 to port 4 (which includes the chamber), push to remove any bubbles;

- Remove air bubbles near port 4, and check again if there are any bubbles in any place except in port 5.

- Remove air bubbles near port 5.

- Double-check all the places.

Connecting syringes

- If possible, operate at the height of the stage.

- Check for the absence of air bubbles on the line, hence connect syringe 5 to its port.

- Connect syringe 4, 1, 2 and 3.

- At 10x magnification, examine the chip for absence of air bubbles and debris in the chamber and in the channels and in the ports. Check that the chamber and the channels are not collapsed (they would look darker). Check the junction is properly made. In this way, you are verifying that the chip is properly wet.

Calibration

- Bond the device to the frame. and put the chip under 40x oil lens
- Adjust the storiette mixing ratio on the "flow control" panel, the syringe 1, 2 should move to the corresponded heights based on the default setting. Adjust the syringes to the heights that the entire input channel is filled by the media from syringe 1 by snapping fluorescence images of the DAW junction.
- Click the "calibration" button on the "flow control" panel. The software will start carrying out the calibration procedure (about 20 min).
- The first 30 sec of the calibration is mainly for analysis the images and mark the region of interest (then input channel). There would be a figure displayed, which shows the result of alignment. If the region is not correct (very unlikely), user should stop the calibration (Press Ctrl+C in the main window of MATLAB), slightly adjust the position and/or focus, and try again.
- During the calibration, the syringes would generate triangle input and oscillate for 3 times. A series of fluorescence images would be taken.
- After the calibration, the result of calibration would be displayed as a plot of fitted function and the data points (mixing ratio vs. height of the syringe 1 above the stage), and another figure showing the absolute error of the fitted function. The result of calibration will also be saved to the current folder, so that user may check or load these results in the future.
- The calibrated result would be directly applied to the current platform.

Loading cells

- Adjust the mixing ratio to achieve the initial concentration for the experiment
- Prepare the cell syringe and attach it higher than 4 and 5.
- Disconnect the temporary cell syringe (5) and plug in the actual cell syringe (6).

(To avoid damages to the chip, you may want to build a small stage to rest the device when it is off the microscope. To do that, take a petri dish lid, cover its top with a few layers of tissue, make it flat and spray ethanol onto it before putting the device on.)

- Move syringe number 4 to an upper position (still below cells).
- Observe the cell loading, flickering the tube to increase cells flow.
- When you are satisfied with the number of cells in the trap, adjust the syringes to the running position. You should slowly bring syringe 4 to the level of the others (be gentle or you will lose cells from the trap). Bring cells to the same height, disconnect cells from port 5 and connect again syringe 5.
- Verify the absence of air bubbles in the ports (especially port 5).

Carrying out the experiment

- Move the stage to the position of the DAW junction and the chamber; mark these positions by clicking "Mark point" button on the "point visiting" panel.
- Adjust the exposure time for the images of different channels, if the exposure time is set to 0, the software would not take the corresponded images for that position.
- Click the "load input design" button on the "Flow control" panel and select the file of the experimental design we want to carry out. The input patter should be shown in the figure above the button.
- Setup the frequency and duration of sampling, it will also decide how long the experiment going to last.
- Click the "start acquisition" and check the contents on the checklist displayed to start the experiment. The images would be saved to the folder shown at the top of the software.

Dispose the equipment

- Double check the experiment is finished and the software is running properly.
- Wet the tissue on the device holder made with a petri dish lid.
- Switch off the auto focusing system and lower the lens to the minimum height.
- Release the upper part of the microscope, gently unscrew the inner stage and move the stage to the holder.
- Unplug the tubes on the device, and fix the free ends back to the corresponding syringes.
- Gently remove the device from the stage, try not to break the glass part. If the glass breaks, collect the glass pieces by a piece of tape with extra caution. Chuck the device and glass into the sharp bin.
- Clean the lens with dry tissue and then again with tissue wetted by ethanol. Wipe the stages of the microscope with ethanol as well.
- Put the syringes on the tube rack and move to the sink.
- For the syringes without cells, turn on the tap and pour the media into the sink.
- Add Virkon solution to the syringe with cells; update the label with date. Dispose the media with cells after 2+hs.
- To dispose the syringes with needles, firstly double check there are no media in the syringe then cut the connected tube with scissors and DO NOT take off the needles. Chuck the syringes with needles together into the sharp bin, and chuck the tubing into the bio waste bag.

Appendix B: Mock-Up Design of the Platform GUI

Mock-up design is used to guide the development of the graphical user interface (GUI). In this project, this work is done with Balsamiq Wireframes toolkit. The advantage of this toolkit is that it is completely online, and so the developer would not need to install any software and users can access and edit it on any computer with internet. This design refers to the design of a control platform made by Dr. Ivan Clark which corresponds to Tag 1.

Screenshot of the online editing page:
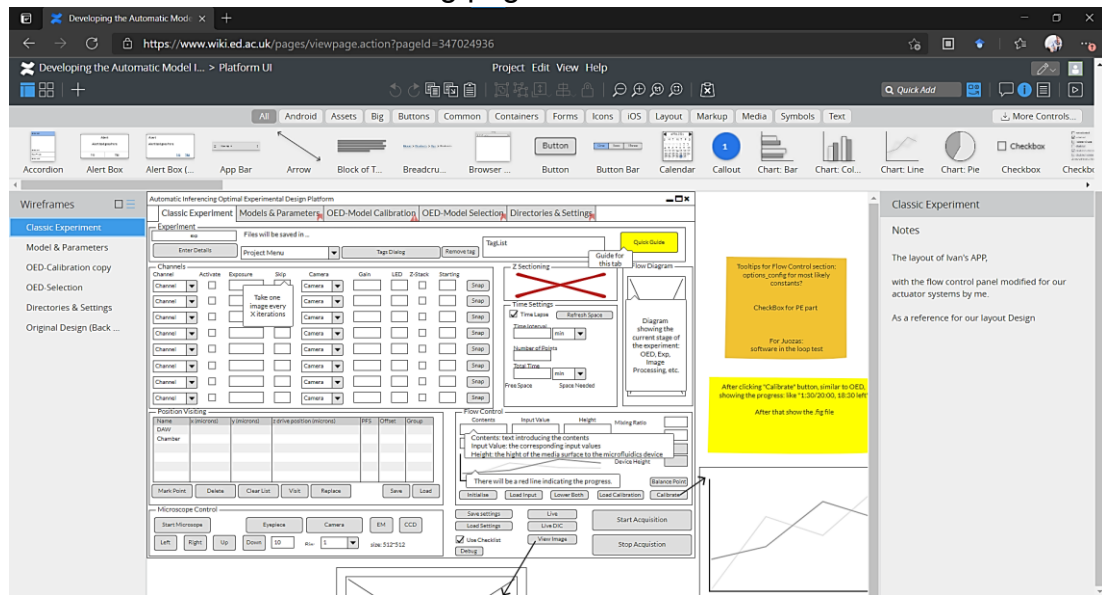
Tag 1: Tab for carrying out a designed experiment.
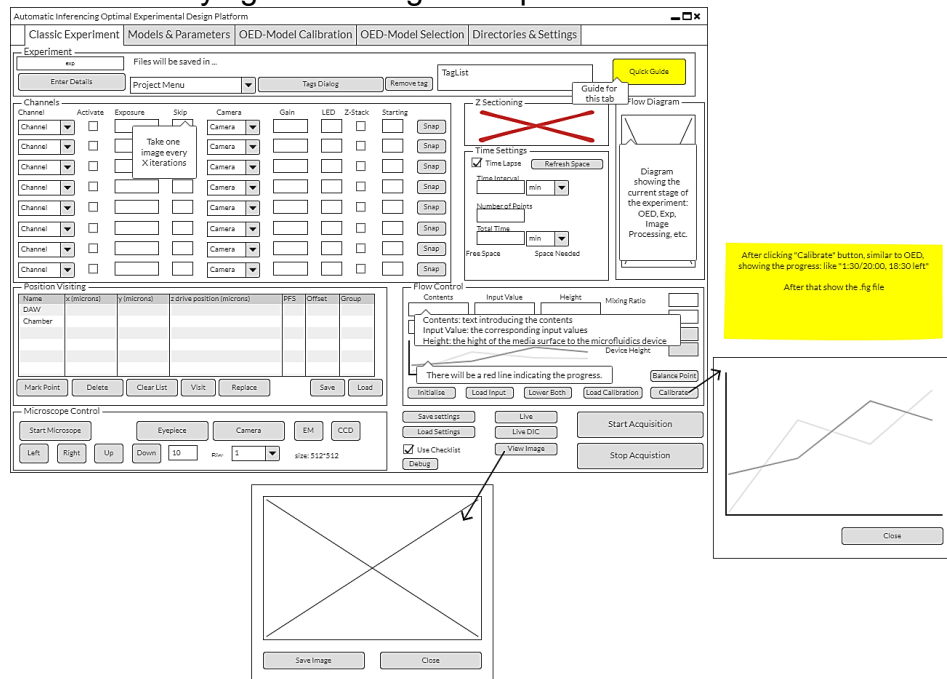
Tag 2: Tab for the list of models and calibration with old experimental data.

Tag 3: Tab for model calibration and corresponding OED.

Tag 4: Tab for model selection and corresponding OED.

Tag 5: Tab for the field directories and settings.
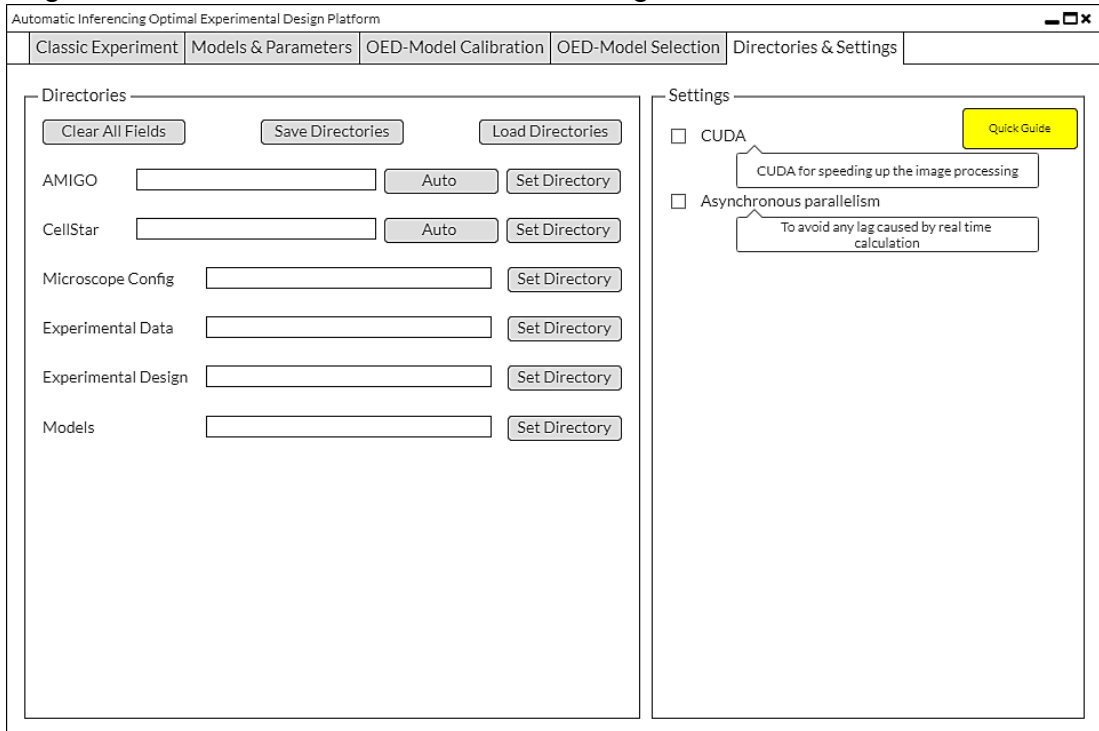
Screenshot of an early-stage GUI demo in MATLAB (corresponds to Tag 5):
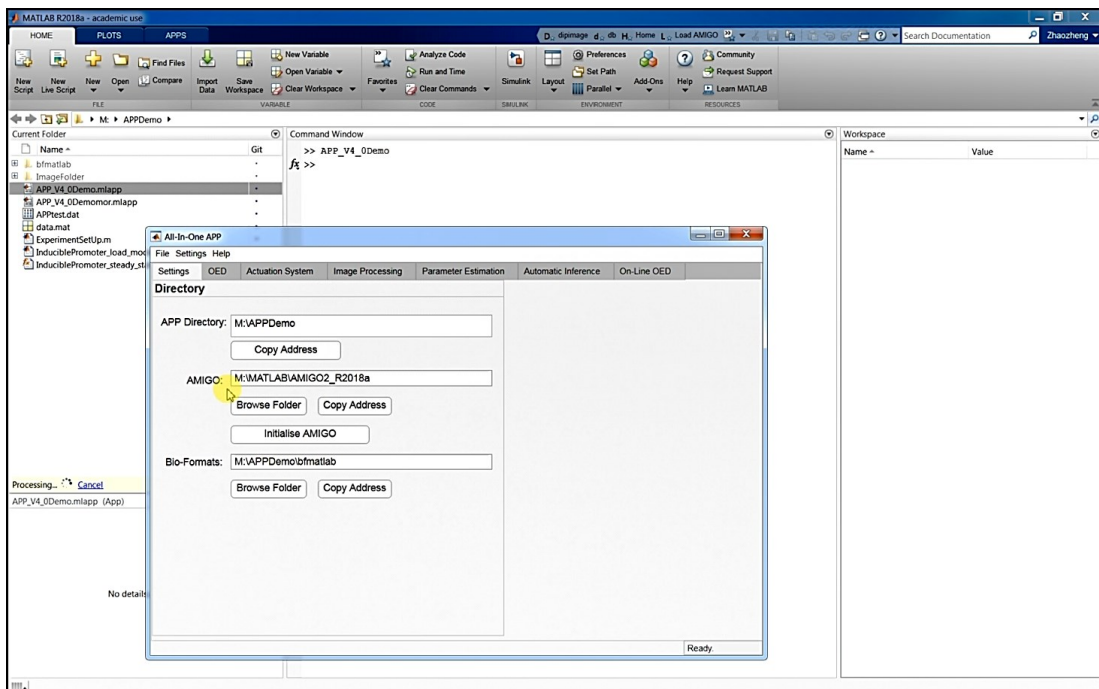
Screenshot of the Python-version GUI (corresponds to Tag 1):

The Python version is developed together with Juozas Pazera.
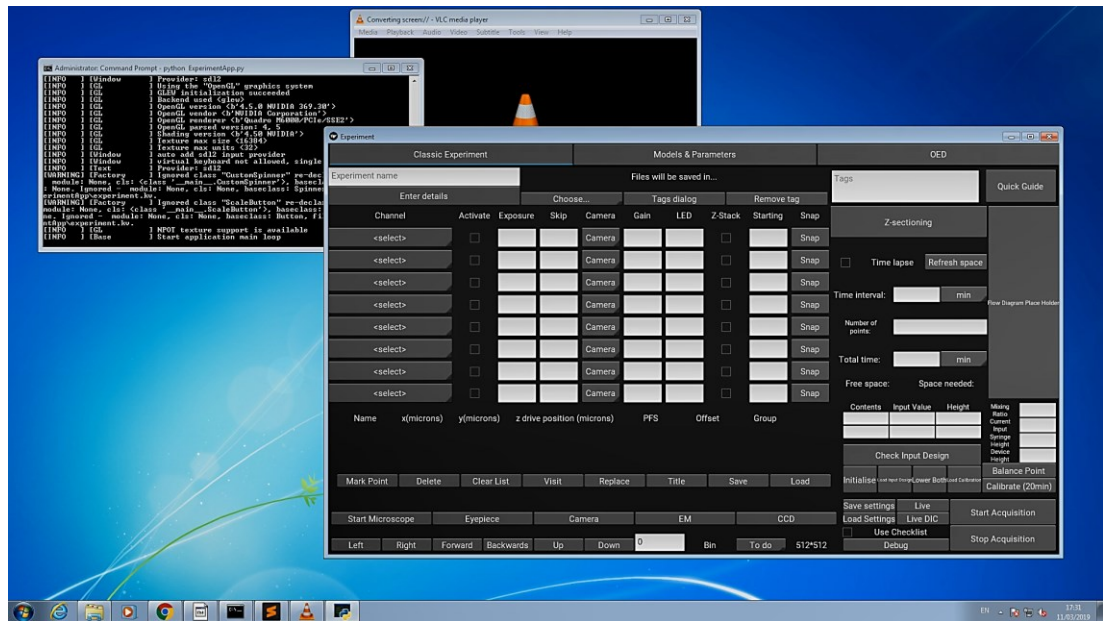
Appendix C: Protocol of Making the Microfluidic Chips

Mixing the PDMS

- In preparation for this stage, obtain a pair of nitrile disposable gloves, a disposable plastic knife/tip and a disposable paper (or plastic) cup.
- PDMS should be mixed in a 1:9 ratio of curing agent and PDMS base. The PDMS base is much more viscous than the curing agent. The way to calculate the mass we need is shown below, we measure the diameter d of the ROI for PDMS pouring within the wafer:

$$M = \rho V = \rho \cdot \frac{\pi}{4} d^2 h \approx \frac{19}{44} d^2,$$

where the density of the mixture is $\rho = 1.1 g/cm^3$, the thickness of the device is $h = 0.5 cm$.

- So, the mass for PDMS base is about $\frac{171}{440} d^2$, the mass for the agent is about $\frac{19}{440} d^2$ (mass in g and diameter in cm).
- On a scale, slowly pour the PDMS base in the cup. Then add the curing agent.
- Take the cup off the scale and begin to stir the PDMS mixture using the disposable plastic knife. Stir the mixture until it turns white and foamy for approximately 3 minutes (make sure they are well mixed, but try not to touch the cup too hard).

Pouring PDMS

- Slowly pour the mixture on the master, selecting a point without features. Given the viscosity of the PDMS, you might have to use a spatula to get all of it onto the wafer.
- Degas it again if needed (remember to keep it flat!).

Degassing the PDMS

Note: A degassing step can be performed before pouring the mixture on the master

- Degas either before or after pouring PDMS on the wafer.

- Put the covered wafer in a degassing desiccator. Check that the valve is open and apply vacuum for 10-15 min. This should allow the bubbles to move to the surface and pop. Hence close the valve, turn off the vacuum and slowly open the valve.

- You may repeat the procedure multiple times, but avoid applying vacuum for more than 15 min.

- If a few bubbles will still be present, you may try to pop them with a sharp needle, or remove it with an airgun. Be cautious as the airgun might remove PDMS from the devices.

Curing

- While keeping the wafer as flat as possible, place it in the oven at 65°C for 3 hours.

Removing the PDMS layer

- Take the wafer out of the oven and let it cool down to room temperature.

- Using a sterile blade, cut the PDMS following the perimeter of old PDMS the master (be careful not to damage the master!). Given the consistency of PDMS, you do not need to go all the way through the thickness of PDMS, the depth is proper when you can see air goes into the cut.

- Gently remove the PDMS from the master and place it on a clean petri dish with the features upwards.

- Cover the top with tapes, and cut the device with a razor blade.

- Punch the holes on the chip, and cover the side with features with tapes again.

Bond the Device

- Turn on the plasma cleaner in advance.

- Perform two cycles of vacuum, plasma and pressure release before processing any device.

- Prepare a piece of clean cover glass (clean it with lens wipes). Remove the cover tape of one microfluidics device and remove dust using tape.
- Place the glass and the device (channels upwards!) in the chamber.
- Turn on the vacuum pump. After 45s, turn the plasma cleaner to the HIGH level for 30s, adjust the valve to get the maximum brightness of the plasma in the chamber.
- Turn the plasma cleaner to OFF level, and turn off the vacuum pump.
- Let the air into the chamber very slowly.
- Open the chamber and bond the device; remember to leave the device falling onto the coverslip from a 30-45° degree angle.
- If not sure if the device is properly bounded, try gently pushing the corner of the device. If there is a sign of bad bonding. If the device is not properly bonded, try repeating the procedure.
- Put the bonded device into the oven at 60-65°C for around 15 min.

References

[1] Nandagopal N, Elowitz MB. Synthetic biology: integrated gene circuits. science. 2011;333(6047):1244-8.

[2] Elowitz M, Lim WA. Build life to understand it. Nature. 2010;468(7326):889-90.

[3] Clarke LJ. Synthetic biology–pathways to commercialisation. Engineering Biology. 2019;3(1):2-5.

[4] Clarke L, Adams J, Sutton P, Bainbridge J, Birney E, Calvert J, et al. A synthetic biology roadmap for the Uk. 2012.

[5] Kitney R, Freemont P. Synthetic biology–the state of play. FEBS letters. 2012;586(15):2029-36.

[6] Hanczyc MM. Engineering Life: A Review of Synthetic Biology. Artificial life. 2020;26(2):260-73.

[7] Clarke L, Kitney R. Developing synthetic biology for industrial biotechnology applications. Biochemical Society Transactions. 2020;48(1):113-22.

[8] Song L, Zhou Y. The COVID‑19 Pandemic and Its Impact on the Global Economy: What Does It Take to Turn Crisis into Opportunity? China & World Economy. 2020;28(4):1-25.

[9] Wisner S. Synthetic Biology Investment Reached a New Record of Nearly $8 Billion in 2020 — What Does This Mean For 2021. 2021.

[10] Sterling J. Synthetic Biology Struts Down the Fashion Runway. Mary Ann Liebert, Inc. 140 Huguenot Street, New Rochelle, NY 10801-5215 (914 …; 2021.

[11] Chen YY, Galloway KE, Smolke CD. Synthetic biology: advancing biological frontiers by building synthetic systems. Genome biology. 2012;13(2):1-10.

[12] Way JC, Collins JJ, Keasling JD, Silver PA. Integrating biological redesign: where synthetic biology came from and where it needs to go. Cell. 2014;157(1):151-61.

[13] Church GM, Elowitz MB, Smolke CD, Voigt CA, Weiss R. Realizing the potential of synthetic biology. Nature Reviews Molecular Cell Biology. 2014;15(4):289-94.

[14] Clarke L, Kitney R. Synthetic biology in the UK–an outline of plans and progress. Synthetic and systems biotechnology. 2016;1(4):243-57.

[15] Endy D. Foundations for engineering biology. Nature. 2005;438(7067):449-53.

[16] Knight T. Idempotent vector design for standard assembly of biobricks. 2003.

[17] Canton B, Labno A, Endy D. Refinement and standardization of synthetic biological parts and devices. Nature biotechnology. 2008;26(7):787-93.

[18] Jinek M, Chylinski K, Fonfara I, Hauer M, Doudna JA, Charpentier E. A programmable dual-RNA–guided DNA endonuclease in adaptive bacterial immunity. science. 2012;337(6096):816-21.

[19] Hillson N, Caddick M, Cai Y, Carrasco JA, Chang MW, Curach NC, et al. Building a global alliance of biofoundries. Nature communications. 2019;10(1):1-4.

[20] Clarke LJ. Synthetic biology UK: progress, paradigms and prospects. Engineering Biology. 2017;1(2):66-70.

[21] Opgenorth P, Costello Z, Okada T, Goyal G, Chen Y, Gin J, et al. Lessons from two design–build–test–learn cycles of dodecanol production in Escherichia coli aided by machine learning. ACS synthetic biology. 2019;8(6):1337-51.

[22] Cravens A, Payne J, Smolke CD. Synthetic biology strategies for microbial biosynthesis of plant natural products. Nature communications. 2019;10(1):1-12.

[23] Dominguez AA, Lim WA, Qi LS. Beyond editing: repurposing CRISPR–Cas9 for precision genome regulation and interrogation. Nature reviews Molecular cell biology. 2016;17(1):5.

[24] Komor AC, Badran AH, Liu DR. CRISPR-based technologies for the manipulation of eukaryotic genomes. Cell. 2017;168(1-2):20-36.

[25] Barteneva NS, Fasler-Kan E, Vorobjev IA. Imaging flow cytometry: coping with heterogeneity in biological systems. Journal of Histochemistry & Cytochemistry. 2012;60(10):723-33.

[26] Gach PC, Iwai K, Kim PW, Hillson NJ, Singh AK. Droplet microfluidics for synthetic biology. Lab on a Chip. 2017;17(20):3388-400.

[27] Bandiera L, Hou Z, Kothamachu VB, Balsa-Canto E, Swain PS, Menolascina F. On-line optimal input design increases the efficiency and accuracy of the modelling of an inducible synthetic promoter. Processes. 2018;6(9):148.

[28] Ruess J, Parise F, Milias-Argeitis A, Khammash M, Lygeros J. Iterative experiment design guides the characterization of a light-inducible gene expression circuit. Proceedings of the National Academy of Sciences. 2015;112(26):8148-53.

[29] Nielsen J, Keasling JD. Engineering cellular metabolism. Cell. 2016;164(6):1185-97.

[30] Qin J, Zhou YJ, Krivoruchko A, Huang M, Liu L, Khoomrung S, et al. Modular pathway rewiring of Saccharomyces cerevisiae enables high-level production of L-ornithine. Nature communications. 2015;6(1):1-11.

[31] Li Y, Li S, Thodey K, Trenchard I, Cravens A, Smolke CD. Complete biosynthesis of noscapine and halogenated alkaloids in yeast. Proceedings of the National Academy of Sciences. 2018;115(17):E3922-E31.

[32] Yang Y, Geng B, Song H, Hu M, He Q, Chen S, et al. Progress and perspective on development of non-model industrial bacteria as chassis cells for biochemical production in the synthetic biology era. Sheng wu Gong Cheng xue bao= Chinese Journal of Biotechnology. 2021;37(3):874-910.

[33] Sarnaik A, Liu A, Nielsen D, Varman AM. High-throughput screening for efficient microbial biotechnology. Current opinion in biotechnology. 2020;64:141-50.

[34] Zhang J, Chen Y, Fu L, Guo E, Wang B, Dai L, et al. Accelerating strain engineering in biofuel research via build and test automation of synthetic biology. Current Opinion in Biotechnology. 2021;67:88-98.

[35] Funke M, Buchenauer A, Schnakenberg U, Mokwa W, Diederichs S, Mertens A, et al. Microfluidic biolector—microfluidic bioprocess control in microtiter plates. Biotechnology and bioengineering. 2010;107(3):497-505.

[36] McKinnon KM. Flow cytometry: an overview. Current protocols in immunology. 2018;120(1):5.1. -5.1. 11.

[37] Ma C, Tan ZL, Lin Y, Han S, Xing X, Zhang C. Gel microdroplet–based high-throughput screening for directed evolution of xylanase-producing Pichia pastoris. Journal of bioscience and bioengineering. 2019;128(6):662-8.

[38] Qiao Y, Zhao X, Zhu J, Tu R, Dong L, Wang L, et al. Fluorescence-activated droplet sorting of lipolytic microorganisms using a compact optical system. Lab on a Chip. 2018;18(1):190-6.

[39] Jian X, Guo X, Wang J, Tan ZL, Xing Xh, Wang L, et al. Microbial microdroplet culture system (MMC): An integrated platform for automated, high‐throughput microbial cultivation and adaptive evolution. Biotechnology and bioengineering. 2020;117(6):1724-37.

[40] Mair F, Prlic M. OMIP‐044: 28‐color immunophenotyping of the human dendritic cell compartment. Cytometry Part A. 2018;93(4):402-5.

[41] Brummelman J, Haftmann C, Nunez NG, Alvisi G, Mazza EM, Becher B, et al. Development, application and computational analysis of high-dimensional fluorescent antibody panels for single-cell flow cytometry. Nature protocols. 2019;14(7):1946-69.

[42] Wang J, Shi D, Zheng M, Ma B, Cui J, Liu C, et al. Screening, separation, and evaluation of xanthine oxidase inhibitors from Paeonia lactiflora using chromatography combined with a multi‐mode microplate reader. Journal of separation science. 2017;40(21):4160-7.

[43] Szymula KP, Magaraci MS, Patterson M, Clark A, Mannickarottu SG, Chow BY. An open-source plate reader. Biochemistry. 2018;58(6):468-73.

[44] Lei Z, Xie D, Mbogba MK, Chen Z, Tian C, Xu L, et al. A microfluidic platform with cell-scale precise temperature control for simultaneous investigation of the osmotic responses of multiple oocytes. Lab on a Chip. 2019;19(11):1929-40.

[45] Scheler O, Postek W, Garstecki P. Recent developments of microfluidics as a tool for biotechnology and microbiology. Current opinion in biotechnology. 2019;55:60-7.

[46] Lohasz C, Rousset N, Renggli K, Hierlemann A, Frey O. Scalable microfluidic platform for flexible configuration of and experiments with

microtissue multiorgan models. SLAS TECHNOLOGY: Translating Life Sciences Innovation. 2019;24(1):79-95.

[47] Xue C-D, Li Y-J, Na J-T, Wang Y-X, Yu H-J, Liu B, et al. A microfluidic platform enabling real-time control of dynamic biochemical stimuli to biological cells. Journal of Micromechanics and Microengineering. 2020;30(9):095011.

[48] Täuber S, von Lieres E, Grünberger A. Dynamic environmental control in microfluidic single‐cell cultivations: From concepts to applications. Small. 2020;16(16):1906670.

[49] Chen P, Li S, Guo Y, Zeng X, Liu B-F. A review on microfluidics manipulation of the extracellular chemical microenvironment and its emerging application to cell analysis. Analytica Chimica Acta. 2020;1125:94-113.

[50] Cavazzuti M. Design of experiments. Optimization methods: Springer; 2013. p. 13-42.

[51] Rodriguez-Fernandez M, Mendes P, Banga JR. A hybrid approach for efficient and robust parameter estimation in biochemical pathways. Biosystems. 2006;83(2-3):248-65.

[52] Moles CG, Mendes P, Banga JR. Parameter estimation in biochemical pathways: a comparison of global optimization methods. Genome research. 2003;13(11):2467-74.

[53] Burger B, Maffettone PM, Gusev VV, Aitchison CM, Bai Y, Wang X, et al. A mobile robotic chemist. Nature. 2020;583(7815):237-41.

[54] Oguntimein GB, Rodriguez M, Dumitrache A, Shollenberger T, Decker SR, Davison BH, et al. Anaerobic microplate assay for direct microbial conversion of switchgrass and Avicel using Clostridium thermocellum. Biotechnology letters. 2018;40(2):303-8.

[55] Mühlmann M, Kunze M, Ribeiro J, Geinitz B, Lehmann C, Schwaneberg U, et al. Cellulolytic RoboLector–towards an automated high-throughput screening platform for recombinant cellulase expression. Journal of biological engineering. 2017;11(1):1-19.

[56] Fialkov AB, Lehotay SJ, Amirav A. Less than one minute low-pressure gas chromatography-mass spectrometry. Journal of Chromatography A. 2020;1612:460691.

[57] Morgan AJ, Hidalgo San Jose L, Jamieson WD, Wymant JM, Song B, Stephens P, et al. Simple and versatile 3D printed microfluidics using fused filament fabrication. PloS one. 2016;11(4):e0152023.

[58] Cardoso RM, Silva PR, Lima AP, Rocha DP, Oliveira TC, do Prado TM, et al. 3D-Printed graphene/polylactic acid electrode for bioanalysis: Biosensing of glucose and simultaneous determination of uric acid and nitrite in biological fluids. Sensors and Actuators B: Chemical. 2020;307:127621.

[59] Hesselman MC, Odoni DI, Ryback BM, de Groot S, van Heck RG, Keijsers J, et al. A multi-platform flow device for microbial (co-) cultivation and microscopic analysis. PLoS One. 2012;7(5):e36982.

[60] CUI J, ZHANG B, Ma Y, Fu X, Wang M, Liu C. Engineering platforms for synthetic biology research. Bulletin of Chinese Acade‐my of Sciences. 2018;33(11):1249-57.

[61] Nuñez I, Matute T, Herrera R, Keymer J, Marzullo T, Rudge T, et al. Low cost and open source multi-fluorescence imaging system for teaching and research in biology and bioengineering. PloS one. 2017;12(11):e0187163.

[62] Kostov Y, Harms P, Randers‐Eichhorn L, Rao G. Low‐cost microbioreactor for high‐throughput bioprocessing. Biotechnology and Bioengineering. 2001;72(3):346-52.

[63] Mousavi PS, Smith SJ, Chen JB, Karlikow M, Tinafar A, Robinson C, et al. A multiplexed, electrochemical interface for gene-circuit-based sensors. Nature chemistry. 2020;12(1):48-55.

[64] Qin Z, Peng R, Baravik IK, Liu X. Fighting COVID-19: Integrated micro- and nanosystems for viral infection diagnostics. Matter. 2020.

[65] Kay E, Wren BW. Recent advances in systems microbiology. Current opinion in microbiology. 2009;12(5):577-81.

[66] Prot JM, Aninat C, Griscom L, Razan F, Brochot C, Guillouzo CG, et al. Improvement of HepG2/C3a cell functions in a microfluidic biochip. Biotechnology and bioengineering. 2011;108(7):1704-15.

[67] Kumar M, Kuzhiumparambil U, Pernice M, Jiang Z, Ralph PJ. Metabolomics: an emerging frontier of systems biology in marine macrophytes. Algal research. 2016;16:76-92.

[68] Leclerc E, Hamon J, Claude I, Jellali R, Naudot M, Bois F. Investigation of acetaminophen toxicity in HepG2/C3a microscale cultures using a system biology model of glutathione depletion. Cell biology and toxicology. 2015;31(3):173-85.

[69] Santo VE, Rebelo SP, Estrada MF, Alves PM, Boghaert E, Brito C. Drug screening in 3D in vitro tumor models: overcoming current pitfalls of efficacy read‐outs. Biotechnology journal. 2017;12(1):1600505.

[70] Degasperi A, Fey D, Kholodenko BN. Performance of objective functions and optimisation procedures for parameter estimation in system biology models. NPJ systems biology and applications. 2017;3(1):1-9.

[71] Yoo C, Cooper GF. A computer-based microarray experiment design-system for gene-regulation pathway discovery. Conference A computer-based microarray experiment design-system for gene-regulation pathway discovery, vol. 2003. American Medical Informatics Association, p. 733.

[72] Lam SH, Karuturi RKM, Gong Z. Zebrafish spotted-microarray for genome-wide expression profiling experiments: data acquisition and analysis. Zebrafish: Springer; 2009. p. 197-226.

[73] Rojas I, Ortuño F. Bioinformatics and Biomedical Engineering: 5th International Work-Conference, IWBBIO 2017, Granada, Spain, April 26–28, 2017, Proceedings, Part I: Springer, 2017.

[74] Behzadi P, Najafi A, Behzadi E, Ranjbar R. Microarray long oligo probe designing for Escherichia coli: an in-silico DNA marker extraction. Central European journal of urology. 2016;69(1):105.

[75] Ranjbar R, Behzadi P, Najafi A, Roudi R. DNA microarray for rapid detection and identification of food and water borne Bacteria: from dry to wet lab. The open microbiology journal. 2017;11:330.

[76] Islam S, Wells CA. Ten simple rules for navigating the computational aspect of an interdisciplinary PhD. Public Library of Science San Francisco, CA USA; 2021.

[77] Lam SH, Mathavan S, Gong Z. Zebrafish spotted-microarray for genome-wide expression profiling experiments. Part I: array printing and hybridization. Zebrafish: Springer; 2009. p. 175-95.

[78] ThermoFisher Ltd. How a Flow Cytometer Works. ThermoFisher; 2021.

[79] Chattopadhyay PK, Hogerkorp CM, Roederer M. A chromatic explosion: the development and future of multiparameter flow cytometry. Immunology. 2008;125(4):441-9.

[80] Kim H, Min K-I, Inoue K, Kim D-P, Yoshida J-i. Submillisecond organic synthesis: Outpacing Fries rearrangement through microfluidic rapid mixing. Science. 2016;352(6286):691-4.

[81] Kim H, Nagaki A, Yoshida J-i. A flow-microreactor approach to protecting-group-free synthesis using organolithium compounds. Nature communications. 2011;2(1):1-6.

[82] Yoshida Ji, Kim H, Nagaki A. Green and sustainable chemical synthesis using flow microreactors. ChemSusChem. 2011;4(3):331-40.

[83] Garcia-Betancur JC, Yepes A, Schneider J, Lopez D. Single-cell analysis of Bacillus subtilis biofilms using fluorescence microscopy and flow cytometry. JoVE (Journal of Visualized Experiments). 2012(60):e3796.

[84] Marjanovič I, Kandušer M, Miklavčič D, Keber MM, Pavlin M. Comparison of flow cytometry, fluorescence microscopy and spectrofluorometry for analysis of gene electrotransfer efficiency. The Journal of membrane biology. 2014;247(12):1259-67.

[85] Weston SA, Parish CR. New fluorescent dyes for lymphocyte migration studies: analysis by flow cytometry and fluorescence microscopy. Journal of immunological methods. 1990;133(1):87-97.

[86] van Gestel J, Vlamakis H, Kolter R. New tools for comparing microscopy images: quantitative analysis of cell types in Bacillus subtilis. Journal of bacteriology. 2015;197(4):699-709.

[87] Lindner PFO, Hitzmann B. Experimental design for optimal parameter estimation of an enzyme kinetic process based on the analysis of the Fisher information matrix. Journal of theoretical biology. 2006;238(1):111-23.

[88] Bardow A. Optimal experimental design of ill-posed problems: The METER approach. Computers & Chemical Engineering. 2008;32(1-2):115-24.

[89] Remli MA, Mohamad MS, Deris S, Samah AA, Omatu S, Corchado JM. Cooperative enhanced scatter search with opposition-based learning schemes for parameter estimation in high dimensional kinetic models of biological systems. Expert Systems with Applications. 2019;116:131-46.

[90] Egea JA, Balsa-Canto E, Garcia M-SG, Banga JR. Dynamic optimization of nonlinear processes with an enhanced scatter search method. Industrial & Engineering Chemistry Research. 2009;48(9):4388-401.

[91] Van Laarhoven PJ, Aarts EH. Simulated annealing. Simulated annealing: Theory and applications: Springer; 1987. p. 7-15.

[92] Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. science. 1983;220(4598):671-80.

[93] Kennedy J, Eberhart R. Particle swarm optimization. Conference Particle swarm optimization, vol. 4. IEEE, p. 1942-8.

[94] Mezura-Montes E, Coello CAC. Constraint-handling in nature-inspired numerical optimization: past, present and future. Swarm and Evolutionary Computation. 2011;1(4):173-94.

[95] Fogel DB. Evolutionary computation: toward a new philosophy of machine intelligence: John Wiley & Sons, 2006.

[96] Storn R, Price K. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of global optimization. 1997;11(4):341-59.

[97] Glover F, Laguna M, Martí R. Fundamentals of scatter search and path relinking. Control and cybernetics. 2000;29(3):653-84.

[98] Bazaraa MS, Sherali HD, Shetty CM. Nonlinear programming: theory and algorithms: John Wiley & Sons, 2013.

[99] Yeniay O. A comparative study on optimization methods for the constrained nonlinear programming problems. Mathematical Problems in Engineering. 2005;2005(2):165-73.

[100] Colville A. A comparative study of nonlinear programming codes. Conference A comparative study of nonlinear programming codes. Princeton University Press, p. 487-502.

[101] Boender C, Romeijn H. Stochastic methods. Handbook of global optimization. Springer, Berlin; 1995.

[102] Bernaerts K, Versyck KJ, Van Impe JF. On the design of optimal dynamic experiments for parameter estimation of a Ratkowsky-type growth kinetics at

suboptimal temperatures. International journal of food microbiology. 2000;54(1-2):27-38.

[103] Banga JR, Versyck KJ, Van Impe JF. Computation of optimal identification experiments for nonlinear dynamic process models: a stochastic global optimization approach. Industrial & Engineering Chemistry Research. 2002;41(10):2425-30.

[104] Czitrom V. One-factor-at-a-time versus designed experiments. The American Statistician. 1999;53(2):126-31.

[105] Asghar A, Abdul Raman AA, Daud WMAW. A comparison of central composite design and Taguchi method for optimizing Fenton process. The Scientific World Journal. 2014;2014.

[106] Ferreira SC, Bruns R, Ferreira H, Matos G, David J, Brandão G, et al. Box-Behnken design: an alternative for the optimization of analytical methods. Analytica chimica acta. 2007;597(2):179-86.

[107] Jaynes J, Ding X, Xu H, Wong WK, Ho CM. Application of fractional factorial designs to study drug combinations. Statistics in medicine. 2013;32(2):307-18.

[108] Huan X, Marzouk YM. Simulation-based optimal Bayesian experimental design for nonlinear systems. Journal of Computational Physics. 2013;232(1):288-317.

[109] Gnügge R, Dharmarajan L, Lang M, Stelling Jr. An orthogonal permease–inducer–repressor feedback loop shows bistability. ACS synthetic biology. 2016;5(10):1098-107.

[110] Adan A, Alizada G, Kiraz Y, Baran Y, Nalbant A. Flow cytometry: basic principles and applications. Critical reviews in biotechnology. 2017;37(2):163-76.

[111] Mazza EMC, Brummelman J, Alvisi G, Roberto A, De Paoli F, Zanon V, et al. Background fluorescence and spreading error are major contributors of variability in high‐dimensional flow cytometry data visualization by t‐distributed stochastic neighboring embedding. Cytometry Part A. 2018;93(8):785-92.

[112] Nettey L, Giles AJ, Chattopadhyay PK. OMIP‐050: A 28‐color/30‐parameter fluorescence flow cytometry panel to enumerate and characterize

cells expressing a wide array of immune checkpoint molecules. Cytometry Part A. 2018;93(11):1094-6.

[113] Starkuviene V, Pepperkok R. The potential of high‑content high‑throughput microscopy in drug discovery. British journal of pharmacology. 2007;152(1):62-71.

[114] Rimon N, Schuldiner M. Getting the whole picture: combining throughput with content in microscopy. Journal of cell science. 2011;124(22):3743-51.

[115] Sokolov M, Ritscher J, MacKinnon N, Bielser JM, Brühlmann D, Rothenhäusler D, et al. Robust factor selection in early cell culture process development for the production of a biosimilar monoclonal antibody. Biotechnology progress. 2017;33(1):181-91.

[116] Totaro D, Rothbauer M, Steiger MG, Mayr T, Wang HY, Lin YS, et al. Downscaling screening cultures in a multifunctional bioreactor array‑on‑a‑chip for speeding up optimization of yeast‑based lactic acid bioproduction. Biotechnology and bioengineering. 2020;117(7):2046-57.

[117] Whitesides GM. The origins and the future of microfluidics. nature. 2006;442(7101):368-73.

[118] Weibel DB, Whitesides GM. Applications of microfluidics in chemical biology. Current opinion in chemical biology. 2006;10(6):584-91.

[119] Breslauer DN, Lee PJ, Lee LP. Microfluidics-based systems biology. Molecular Biosystems. 2006;2(2):97-112.

[120] Takayama S, Ostuni E, LeDuc P, Naruse K, Ingber DE, Whitesides GM. Subcellular positioning of small molecules. Nature. 2001;411(6841):1016-.

[121] Bennett MR, Pang WL, Ostroff NA, Baumgartner BL, Nayak S, Tsimring LS, et al. Metabolic gene regulation in a dynamically changing environment. Nature. 2008;454(7208):1119-22.

[122] Danino T, Mondragón-Palomino O, Tsimring L, Hasty J. A synchronized quorum of genetic clocks. Nature. 2010;463(7279):326-30.

[123] Hou X, Zhang YS, Trujillo-de Santiago G, Alvarez MM, Ribas J, Jonas SJ, et al. Interplay between materials and microfluidics. Nature Reviews Materials. 2017;2(5):1-15.

[124] Ferry MS, Razinkov IA, Hasty J. Microfluidics for synthetic biology: from design to execution. Methods in enzymology. 2011;497:295-372.

[125] Thorsen T, Maerkl SJ, Quake SR. Microfluidic large-scale integration. Science. 2002;298(5593):580-4.

[126] Kaminski TS, Scheler O, Garstecki P. Droplet microfluidics for microbiology: techniques, applications and challenges. Lab on a Chip. 2016;16(12):2168-87.

[127] Chiu DT, Demello AJ, Di Carlo D, Doyle PS, Hansen C, Maceiczyk RM, et al. Small but perfectly formed? Successes, challenges, and opportunities for microfluidics in the chemical and biological sciences. Chem. 2017;2(2):201-23.

[128] Balagaddé FK, You L, Hansen CL, Arnold FH, Quake SR. Long-term monitoring of bacteria undergoing programmed population control in a microchemostat. Science. 2005;309(5731):137-40.

[129] Falk T, Mai D, Bensch R, Çiçek Ö, Abdulkadir A, Marrakchi Y, et al. U-Net: deep learning for cell counting, detection, and morphometry. Nature methods. 2019;16(1):67-70.

[130] Arganda-Carreras I, Kaynig V, Rueden C, Eliceiri KW, Schindelin J, Cardona A, et al. Trainable Weka Segmentation: a machine learning tool for microscopy pixel classification. Bioinformatics. 2017;33(15):2424-6.

[131] Haberl MG, Churas C, Tindall L, Boassa D, Phan S, Bushong EA, et al. CDeep3M—Plug-and-Play cloud-based deep learning for image segmentation. Nature methods. 2018;15(9):677-80.

[132] Rusk N. Deep learning. Nature Methods. 2016;13(1):35-.

[133] Webb S. Deep learning for biology. Nature. 2018;554(7690):555-8.

[134] Sadanandan SK, Ranefall P, Le Guyader S, Wählby C. Automated training of deep convolutional neural networks for cell segmentation. Scientific reports. 2017;7(1):1-7.

[135] Bertaux F, Carrillo SS, Fraisse A, Aditya C, Furstenheim M, Batt G. Enhancing multi-bioreactor platforms for automated measurements and reactive experiment control. bioRxiv. 2020.

[136] Harrigan P, Madhani HD, El-Samad H. Real-time genetic compensation defines the dynamic demands of feedback control. Cell. 2018;175(3):877-86. e10.

[137] Milias-Argeitis A, Summers S, Stewart-Ornstein J, Zuleta I, Pincus D, El-Samad H, et al. In silico feedback for in vivo regulation of a gene expression circuit. Nature biotechnology. 2011;29(12):1114-6.

[138] Paik S-M, Sim S-J, Jeon NL. Microfluidic perfusion bioreactor for optimization of microalgal lipid productivity. Bioresource technology. 2017;233:433-7.

[139] Frey O, Rudolf F, Schmidt GW, Hierlemann A. Versatile, simple-to-use microfluidic cell-culturing chip for long-term, high-resolution, time-lapse imaging. Analytical chemistry. 2015;87(8):4144-51.

[140] Steel H, Habgood R, Kelly C, Papachristodoulou A. Chi. Bio: An open-source automated experimental platform for biological science research. bioRxiv. 2019:796516.

[141] Steel H, Habgood R, Kelly CL, Papachristodoulou A. In situ characterisation and manipulation of biological systems with Chi. Bio. PLoS biology. 2020;18(7):e3000794.

[142] Pouzet S, Banderas A, Le Bec M, Lautier T, Truan G, Hersen P. The Promise of Optogenetics for Bioproduction: Dynamic Control Strategies and Scale-Up Instruments. Bioengineering. 2020;7(4):151.

[143] Takahashi N, Duprez L, Grootjans S, Cauwels A, Nerinckx W, DuHadaway J, et al. Necrostatin-1 analogues: critical issues on the specificity, activity and in vivo use in experimental disease models. Cell death & disease. 2012;3(11):e437-e.

[144] Liu R, Lin S, Rallo R, Zhao Y, Damoiseaux R, Xia T, et al. Automated phenotype recognition for zebrafish embryo based in vivo high throughput toxicity screening of engineered nano-materials. PloS one. 2012;7(4):e35014.

[145] Huang M, Bai Y, Sjostrom SL, Hallström BM, Liu Z, Petranovic D, et al. Microfluidic screening and whole-genome sequencing identifies mutations associated with improved protein secretion by yeast. Proceedings of the National Academy of Sciences. 2015;112(34):E4689-E96.

[146] Cornaglia M, Krishnamani G, Mouchiroud L, Sorrentino V, Lehnert T, Auwerx J, et al. Automated longitudinal monitoring of in vivo protein

aggregation in neurodegenerative disease C. elegans models. Molecular neurodegeneration. 2016;11(1):1-13.

[147] Espah Borujeni A, Mishler DM, Wang J, Huso W, Salis HM. Automated physics-based design of synthetic riboswitches from diverse RNA aptamers. Nucleic acids research. 2016;44(1):1-13.

[148] Zhang YS, Aleman J, Shin SR, Kilic T, Kim D, Shaegh SAM, et al. Multisensor-integrated organs-on-chips platform for automated and continual in situ monitoring of organoid behaviors. Proceedings of the National Academy of Sciences. 2017;114(12):E2293-E302.

[149] Nandania J, Peddinti G, Pessia A, Kokkonen M, Velagapudi V. Validation and automation of a high-throughput multitargeted method for semiquantification of endogenous metabolites from different biological matrices using tandem mass spectrometry. Metabolites. 2018;8(3):44.

[150] Beal J, Overney C, Adler A, Yaman F, Tiberio L, Samineni M. Tasbe flow analytics: a package for calibrated flow cytometry analysis. ACS synthetic biology. 2019;8(7):1524-9.

[151] Fedorec AJ, Karkaria BD, Sulu M, Barnes CP. Single strain control of microbial consortia. bioRxiv. 2020:2019.12. 23.887331.

[152] Soffer G, Perry JM, Shih SC. Real-Time Optogenetics System for Controlling Gene Expression Using a Model-Based Design. Analytical Chemistry. 2021.

[153] Müller-Hill B. The lac operon: de Gruyter, 2011.

[154] Lewis M, Chang G, Horton NC, Kercher MA, Pace HC, Schumacher MA, et al. Crystal structure of the lactose operon repressor and its complexes with DNA and inducer. Science. 1996;271(5253):1247-54.

[155] Da Silva NA, Srikrishnan S. Introduction and expression of genes for metabolic engineering applications in Saccharomyces cerevisiae. FEMS yeast research. 2012;12(2):197-214.

[156] Gligorovski V, Sadeghi A, Rahi SJ. Multidimensional single-cell benchmarking of inducible promoters for precise dynamic control in budding yeast. bioRxiv. 2020.

[157] Li W-Z, Sherman F. Two types of TATA elements for the CYC1 gene of the yeast Saccharomyces cerevisiae. Molecular and cellular biology. 1991;11(2):666-76.

[158] Gilchrist MA, Wagner A. A model of protein translation including codon bias, nonsense errors, and ribosome recycling. Journal of theoretical biology. 2006;239(4):417-34.

[159] Pelechano V, Chávez S, Pérez-Ortín JE. A complete set of nascent transcription rates for yeast genes. PloS one. 2010;5(11):e15442.

[160] Wang Y, Liu CL, Storey JD, Tibshirani RJ, Herschlag D, Brown PO. Precision and functional specificity in mRNA decay. Proceedings of the National Academy of Sciences. 2002;99(9):5860-5.

[161] Belle A, Tanay A, Bitincka L, Shamir R, O'Shea EK. Quantification of protein half-lives in the budding yeast proteome. Proceedings of the National Academy of Sciences. 2006;103(35):13004-9.

[162] Gordon A, Colman-Lerner A, Chin TE, Benjamin KR, Richard CY, Brent R. Single-cell quantification of molecules and rates using open-source microscope-based cytometry. Nature methods. 2007;4(2):175-81.

[163] Broude NE. Analysis of RNA localization and metabolism in single live bacterial cells: achievements and challenges. Molecular microbiology. 2011;80(5):1137-47.

[164] Zenklusen D, Singer RH. Analyzing mRNA expression using single mRNA resolution fluorescent in situ hybridization. Methods in enzymology. 2010;470:641-59.

[165] Kreutz C. An easy and efficient approach for testing identifiability. Bioinformatics. 2018;34(11):1913-21.

[166] Simpson MJ, Baker RE, Vittadello ST, Maclaren OJ. Practical parameter identifiability for spatio-temporal models of cell invasion. Journal of the Royal Society Interface. 2020;17(164):20200055.

[167] Gábor A, Villaverde AF, Banga JR. Parameter identifiability analysis and visualization in large-scale kinetic models of biosystems. BMC systems biology. 2017;11(1):1-16.

[168] Villaverde AF, Tsiantis N, Banga JR. Full observability and estimation of unknown inputs, states and parameters of nonlinear biological models. Journal of the Royal Society Interface. 2019;16(156):20190043.

[169] Mitra ED, Hlavacek WS. Parameter estimation and uncertainty quantification for systems biology models. Current opinion in systems biology. 2019;18:9-18.

[170] Raue A, Steiert B, Schelker M, Kreutz C, Maiwald T, Hass H, et al. Data2Dynamics: a modeling environment tailored to parameter estimation in dynamical systems. Bioinformatics. 2015;31(21):3558-60.

[171] Balsa-Canto E, Alonso AA, Banga JR. An iterative identification procedure for dynamic modeling of biochemical networks. BMC systems biology. 2010;4(1):1-18.

[172] Hong H, Ovchinnikov A, Pogudin G, Yap C. Global identifiability of differential models. Communications on Pure and Applied Mathematics. 2020;73(9):1831-79.

[173] Lira-Parada PA, Pettersen E, Biegler LT, Bar N. Implications of dimensional analysis in bioreactor models: Parameter estimation and identifiability. Chemical Engineering Journal. 2021:129220.

[174] Balsa-Canto E, Henriques D, Gábor A, Banga JR. AMIGO2, a toolbox for dynamic modeling, optimization and control in systems biology. Bioinformatics. 2016;32(21):3357-9.

[175] Gomes GF, de Almeida FA, Alexandrino PdSL, da Cunha SS, de Sousa BS, Ancelotti AC. A multiobjective sensor placement optimization for SHM systems considering Fisher information matrix and mode shape interpolation. Engineering with Computers. 2019;35(2):519-35.

[176] Manesso E, Sridharan S, Gunawan R. Multi-objective optimization of experiments using curvature and fisher information matrix. Processes. 2017;5(4):63.

[177] Kaschek D, Mader W, Fehling-Kaschek M, Rosenblatt M, Timmer J. Dynamic modeling, parameter estimation, and uncertainty analysis in R. Journal of Statistical Software. 2019;88(1):1-32.

[178] Hultström J, Manneberg O, Dopf K, Hertz HM, Brismar H, Wiklund M. Proliferation and viability of adherent cells manipulated by standing-wave ultrasound in a microfluidic chip. Ultrasound in medicine & biology. 2007;33(1):145-51.

[179] Huberts DH, Lee SS, Gonzalez J, Janssens GE, Vizcarra IA, Heinemann M. Construction and use of a microfluidic dissection platform for long-term imaging of cellular processes in budding yeast. Nature protocols. 2013;8(6):1019-27.

[180] Kumar P, Vriens K, Cornaglia M, Gijs M, Kokalj T, Thevissen K, et al. Digital microfluidics for time-resolved cytotoxicity studies on single non-adherent yeast cells. Lab on a Chip. 2015;15(8):1852-60.

[181] Ruan Z, Giordano M. The use of NH4+ rather than NO3− affects cell stoichiometry, C allocation, photosynthesis and growth in the cyanobacterium Synechococcus sp. UTEX LB 2380, only when energy is limiting. Plant, cell & environment. 2017;40(2):227-36.

[182] Giordano M, Wang Q. Microalgae for industrial purposes. Biomass and Green Chemistry. 2018:133-67.

[183] Toyoda A, Sugiyama M, Furihata S, Nishizumi K, Omori T, Itagaki H. Development of a modified 3T3 Neutral Red Uptake Phototoxicity Test protocol for evaluation of poorly water-soluble substances. The Journal of toxicological sciences. 2017;42(5):569-77.

[184] Janousek S, Vlkova A, Jirova G, Kejlova K, Krsek D, Jirova D, et al. Qualitative and quantitative analysis of certain aspects of the cytotoxic and genotoxic hazard of hospital wastewaters by using a range of in vitro assays. Alternatives to Laboratory Animals. 2021;49(1-2):33-48.

[185] Edelstein AD, Tsuchida MA, Amodaj N, Pinkard H, Vale RD, Stuurman N. Advanced methods of microscope control using µManager software. Journal of biological methods. 2014;1(2).

[186] Vaisberg EA, Lenzi D, Hansen RL, Keon BH, Finer JT. An infrastructure for high‐throughput microscopy: instrumentation, informatics, and integration. Methods in enzymology. 2006;414:484-512.

[187] Versari C, Stoma S, Batmanov K, Llamosi A, Mroz F, Kaczmarek A, et al. Long-term tracking of budding yeast cells in brightfield microscopy: CellStar and the Evaluation Platform. Journal of The Royal Society Interface. 2017;14(127):20160705.

[188] Bagwell CB, Adams EG. Fluorescence spectral overlap compensation for any number of flow cytometry parameters. Annals of the New York Academy of Sciences. 1993;677(1):167-84.

[189] Herzenberg LA, Tung J, Moore WA, Herzenberg LA, Parks DR. Interpreting flow cytometry data: a guide for the perplexed. Nature immunology. 2006;7(7):681-5.

[190] Lehmann EL, Romano JP. Generalizations of the familywise error rate. Selected Works of EL Lehmann: Springer; 2012. p. 719-35.

[191] Proschan MA, Brittain EH. A primer on strong vs weak control of familywise error rate. Statistics in medicine. 2020;39(9):1407-13.

[192] Egea JA, Martí R, Banga JR. An evolutionary method for complex-process optimization. Computers & Operations Research. 2010;37(2):315-24.

[193] Grossmann IE. Global optimization in engineering design: Springer Science & Business Media, 2013.

[194] Papamichail I, Adjiman CS. A rigorous global optimization algorithm for problems with ordinary differential equations. Journal of Global Optimization. 2002;24(1):1-33.

[195] Floudas CA. Deterministic global optimization: theory, methods and applications: Springer Science & Business Media, 2013.

[196] Chachuat B, Singer AB, Barton PI. Global methods for dynamic optimization and mixed-integer dynamic optimization. Industrial & Engineering Chemistry Research. 2006;45(25):8373-92.

[197] Jalali F, Seader J, Khaleghi S. Global solution approaches in equilibrium and stability analysis using homotopy continuation in the complex domain. Computers & Chemical Engineering. 2008;32(10):2333-45.

[198] Bhargava V, Fateen S, Bonilla-Petriciolet A. Cuckoo search: a new nature-inspired optimization method for phase equilibrium calculations. Fluid Phase Equilibria. 2013;337:191-200.

[199] Le Thi HA, Nguyen DM, Dinh TP. Globally solving a nonlinear UAV task assignment problem by stochastic and deterministic optimization approaches. Optimization Letters. 2012;6(2):315-29.

[200] Floudas CA, Pardalos PM, Adjiman CS, Esposito WR, Gümüş ZH, Harding ST, et al. Nonconvex optimization and its applications. Deterministic global optimization: theory, methods and applications. 2000;37.

[201] Adjiman CS, Androulakis IP, Floudas CA. Global optimization of mixed-integer nonlinear problems. AIChE Journal. 2000;46(9):1769-97.

[202] Smith EM, Pantelides CC. A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of nonconvex MINLPs. Computers & Chemical Engineering. 1999;23(4-5):457-78.

[203] Sahinidis NV. BARON: A general purpose global optimization software package. Journal of global optimization. 1996;8(2):201-5.

[204] Misener R, Floudas CA. GloMIQO: Global mixed-integer quadratic optimizer. Journal of Global Optimization. 2013;57(1):3-50.

[205] Gau C-Y, Schrage LE. Implementation and testing of a branch-and-bound based method for deterministic global optimization: operations research applications. Frontiers in global optimization: Springer; 2004. p. 145-64.

[206] McCormick GP. Nonlinear programming; theory, algorithms, and applications. 1983.

[207] Rangaiah GP. Evaluation of genetic algorithms and simulated annealing for phase equilibrium and stability problems. Fluid Phase Equilibria. 2001;187:83-109.

[208] Bonilla-Petriciolet A, Rangaiah GP, Segovia-Hernández JG. Constrained and unconstrained Gibbs free energy minimization in reactive systems using genetic algorithm and differential evolution with tabu list. Fluid phase equilibria. 2011;300(1-2):120-34.

[209] Srinivas M, Rangaiah GP. A study of differential evolution and tabu search for benchmark, phase equilibrium and phase stability problems. Computers & chemical engineering. 2007;31(7):760-72.

[210] Srinivas M, Rangaiah G. Differential evolution with tabu list for global optimization and its application to phase equilibrium and parameter estimation problems. Industrial & engineering chemistry research. 2007;46(10):3410-21.

[211] Bonilla-Petriciolet A, Segovia-Hernández JG. A comparative study of particle swarm optimization and its variants for phase stability and equilibrium calculations in multicomponent reactive and non-reactive systems. Fluid Phase Equilibria. 2010;289(2):110-21.

[212] Zhang H, Fernández-Vargas JA, Rangaiah GP, Bonilla-Petriciolet A, Segovia-Hernández JG. Evaluation of integrated differential evolution and

unified bare-bones particle swarm optimization for phase equilibrium and stability problems. Fluid Phase Equilibria. 2011;310(1-2):129-41.

[213] Glover F, Laguna M, Martí R. Scatter search. Advances in evolutionary computing: Springer; 2003. p. 519-37.

[214] Banga JR, Balsa-Canto E, Moles CG, Alonso AA. Dynamic optimization of bioprocesses: Efficient and robust numerical strategies. Journal of Biotechnology. 2005;117(4):407-19.

[215] Balsa-Canto E, Vassiliadis VS, Banga JR. Dynamic optimization of single-and multi-stage systems using a hybrid stochastic− deterministic method. Industrial & engineering chemistry research. 2005;44(5):1514-23.

[216] Zhou T, Zhou Y, Sundmacher K. A hybrid stochastic–deterministic optimization approach for integrated solvent and process design. Chemical Engineering Science. 2017;159:207-16.

[217] Lagarias JC, Reeds JA, Wright MH, Wright PE. Convergence properties of the Nelder--Mead simplex method in low dimensions. SIAM Journal on optimization. 1998;9(1):112-47.

[218] Crane MM, Clark IB, Bakker E, Smith S, Swain PS. A microfluidic system for studying ageing and dynamic single-cell responses in budding yeast. PloS one. 2014;9(6):e100042.

[219] Ferry MS, Razinkov IA, Hasty J. Microfluidics for synthetic biology: from design to execution. Methods in enzymology: Elsevier; 2011. p. 295-372.

[220] Beal J, Weiss R, Yaman F, Davidsohn N, Adler A. A method for fast, high-precision characterization of synthetic biology devices. 2012.

[221] Tsigkinopoulou A, Hawari A, Uttley M, Breitling R. Defining informative priors for ensemble modeling in systems biology. Nature protocols. 2018;13(11):2643-63.

[222] Arendt PD, Apley DW, Chen W, Lamb D, Gorsich D. Improving identifiability in model calibration using multiple responses. Journal of Mechanical Design. 2012;134(10).

[223] Etemadi N. An elementary proof of the strong law of large numbers. Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete. 1981;55(1):119-22.

[224] Faller D, Klingmüller U, Timmer J. Simulation methods for optimal experimental design in systems biology. Simulation. 2003;79(12):717-25.

[225] Bandiera L, Cabeza DG, Balsa-Canto E, Menolascina F. Bayesian model selection in synthetic biology: factor levels and observation functions. IFAC-PapersOnLine. 2019;52(26):24-31.

[226] Fedorov VV, Hackl P. Model-oriented design of experiments: Springer Science & Business Media, 2012.

[227] N. Politis S, Colombo P, Colombo G, M. Rekkas D. Design of experiments (DoE) in pharmaceutical development. Drug development and industrial pharmacy. 2017;43(6):889-901.

[228] Durakovic B. Design of experiments application, concepts, examples: State of the art. Periodicals of Engineering and Natural Sciences. 2017;5(3).

[229] Paulo F, Santos L. Design of experiments for microencapsulation applications: A review. Materials Science and Engineering: C. 2017;77:1327-40.

[230] Yu P, Low MY, Zhou W. Design of experiments and regression modelling in food flavour and sensory analysis: A review. Trends in Food Science & Technology. 2018;71:202-15.

[231] Spall JC. Monte Carlo computation of the Fisher information matrix in nonstandard settings. Journal of Computational and Graphical Statistics. 2005;14(4):889-909.

[232] Machado VC, Tapia G, Gabriel D, Lafuente J, Baeza JA. Systematic identifiability study based on the Fisher Information Matrix for reducing the number of parameters calibration of an activated sludge model. Environmental Modelling & Software. 2009;24(11):1274-84.

[233] Hassanein W, Kilany N. DE-and EDPM-compound optimality for the information and probability-based criteria. Hacettepe Journal of Mathematics and Statistics. 2019;48(2):580-91.

[234] Retout S, Mentré F. Further developments of the Fisher information matrix in nonlinear mixed effects models with evaluation in population pharmacokinetics. Journal of Biopharmaceutical Statistics. 2003;13(2):209-27.

[235] Hamilton DC, Watts DG. A quadratic design criterion for precise estimation in nonlinear regression models. Technometrics. 1985;27(3):241-50.

[236] Srinath S, Gunawan R. Model-based Design of Experiment for Kinetic Parameter Identification: Beyond the Fisher Information Matrix. Conference Model-based Design of Experiment for Kinetic Parameter Identification: Beyond the Fisher Information Matrix.

[237] McGree J, Eccleston J. Probability‐based optimal design. Australian & New Zealand Journal of Statistics. 2008;50(1):13-28.

[238] Chu Y, Hahn J. Parameter set selection via clustering of parameters into pairwise indistinguishable groups of parameters. Industrial & Engineering Chemistry Research. 2009;48(13):6000-9.

[239] Kanada Y. Optimizing neural-network learning rate by using a genetic algorithm with per-epoch mutations. Conference Optimizing neural-network learning rate by using a genetic algorithm with per-epoch mutations. IEEE, p. 1472-9.

[240] Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, et al. Caffe: Convolutional architecture for fast feature embedding. Conference Caffe: Convolutional architecture for fast feature embedding. p. 675-8.

[241] Kravaris C, Hahn J, Chu Y. Advances and selected recent developments in state and parameter estimation. Computers & chemical engineering. 2013;51:111-23.

[242] Dai W, Bansal L, Hahn J. Parameter set selection for signal transduction pathway models including uncertainties. IFAC Proceedings Volumes. 2014;47(3):815-20.

[243] Lee D, Jayaraman A, Kwon JS-I. Identification of a time-varying intracellular signalling model through data clustering and parameter selection: application to NF-κ B signalling pathway induced by LPS in the presence of BFA. IET systems biology. 2019;13(4):169-79.

[244] Nienałtowski K, Włodarczyk M, Lipniacki T, Komorowski M. Clustering reveals limits of parameter identifiability in multi-parameter models of biochemical dynamics. BMC systems biology. 2015;9(1):1-9.

[245] Singh A, Jayaraman A, Hahn J. Modeling regulatory mechanisms in IL‐6 signal transduction in hepatocytes. Biotechnology and bioengineering. 2006;95(5):850-62.

[246] Chu Y, Jayaraman A, Hahn J. Parameter sensitivity analysis of IL-6 signalling pathways. IET Systems Biology. 2007;1(6):342-52.

[247] Fischer P, Hilfiker‑Kleiner D. Role of gp130‑mediated signalling pathways in the heart and its impact on potential therapeutic aspects. British journal of pharmacology. 2008;153(S1):S414-S27.

[248] Lang R, Pauleau A-L, Parganas E, Takahashi Y, Mages J, Ihle JN, et al. SOCS3 regulates the plasticity of gp130 signaling. Nature immunology. 2003;4(6):546-50.

[249] Bashor CJ, Patel N, Choubey S, Beyzavi A, Kondev J, Collins JJ, et al. Complex signal processing in synthetic gene circuits using cooperative regulatory assemblies. Science. 2019;364(6440):593-7.

[250] Bashor CJ, Horwitz AA, Peisajovich SG, Lim WA. Rewiring cells: synthetic biology as a tool to interrogate the organizational principles of living systems. Annual review of biophysics. 2010;39:515-37.

[251] Ferrell Jr JE, Xiong W. Bistability in cell signaling: How to make continuous processes discontinuous, and reversible processes irreversible. Chaos: An Interdisciplinary Journal of Nonlinear Science. 2001;11(1):227-36.

[252] Babu B, Angira R. Modified differential evolution (MDE) for optimization of non-linear chemical processes. Computers & chemical engineering. 2006;30(6-7):989-1002.

[253] Onwubolu GC, Babu B. New optimization techniques in engineering: Springer, 2013.

[254] Avriel M. Nonlinear programming: analysis and methods: Courier Corporation, 2003.

[255] Vilas C, Balsa-Canto E, García M-SG, Banga JR, Alonso AA. Dynamic optimization of distributed biological systems using robust and efficient numerical techniques. BMC systems biology. 2012;6(1):1-16.

[256] Salton G, Buckley C. Term-weighting approaches in automatic text retrieval. Information processing & management. 1988;24(5):513-23.

[257] Jang J, Smyth AW. Model updating of a full-scale FE model with nonlinear constraint equations and sensitivity-based cluster analysis for updating parameters. Mechanical Systems and Signal Processing. 2017;83:337-55.

[258] Shahverdi H, Mares C, Wang W, Mottershead J. Clustering of parameter sensitivities: examples from a helicopter airframe model updating exercise. Shock and Vibration. 2009;16(1):75-87.

[259] Hartigan JA, Wong MA. Algorithm AS 136: A k-means clustering algorithm. Journal of the royal statistical society series c (applied statistics). 1979;28(1):100-8.

[260] David A, Vassilvitskii S. K-means++: The Advantages of Careful Seeding. Conference K-means++: The Advantages of Careful Seeding. p. 1027-35.

[261] Ester M, Kriegel H-P, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. Conference A density-based algorithm for discovering clusters in large spatial databases with noise, vol. 96. p. 226-31.

[262] Everitt BS, Landau S, Leese M, Stahl D. Hierarchical clustering. Cluster analysis. 2011;5:71-110.

[263] The MathWorks. Choose Cluster Analysis Method. Help Center2020.

[264] The MathWorks. linkage - agglomerative hierarchical cluster tree. 2020.

[265] Lallement G, Piranda J. Localization methods for parametric updating of finite elements models in elastodynamics. Conference Localization methods for parametric updating of finite elements models in elastodynamics. p. 579-85.

[266] Kirk RE. Experimental design. Handbook of Psychology, Second Edition. 2012;2.

[267] Waite TW, Woods DC. Minimax efficient random experimental design strategies with application to model-robust design for prediction. Journal of the American Statistical Association. 2021:1-14.

[268] Vieira RK, Soares RC, Pinheiro SC, Paiva OA, Eleutério JO, Vasconcelos RP. Completely random experimental design with mixture and process variables for optimization of rubberized concrete. Construction and Building Materials. 2010;24(9):1754-60.

[269] Wu H, Huang B, Zare RN. Construction of microfluidic chips using polydimethylsiloxane for adhesive bonding. Lab on a Chip. 2005;5(12):1393-8.

[270] Hu K, Bhattacharya BB, Chakrabarty K. Fault diagnosis for flow-based microfluidic biochips. Conference Fault diagnosis for flow-based microfluidic biochips. IEEE, p. 1-6.

[271] Hassanin H, Mohammadkhani A, Jiang K. Fabrication of hybrid nanostructured arrays using a PDMS/PDMS replication process. Lab on a Chip. 2012;12(20):4160-7.

[272] Davids D, Datta S, Mukherjee A, Joshi B, Ravindran A. Multiple fault diagnosis in digital microfluidic biochips. ACM Journal on Emerging Technologies in Computing Systems (JETC). 2006;2(4):262-76.

[273] Chakrabarty K. Design automation and test solutions for digital microfluidic biochips. IEEE Transactions on Circuits and Systems I: Regular Papers. 2009;57(1):4-17.

[274] Hu K, Yu F, Ho T-Y, Chakrabarty K. Testing of flow-based microfluidic biochips: Fault modeling, test generation, and experimental demonstration. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2014;33(10):1463-75.

[275] Zhu Y, Li B, Ho T-Y, Wang Q, Yao H, Wille R, et al. Multi-channel and fault-tolerant control multiplexing for flow-based microfluidic biochips. Conference Multi-channel and fault-tolerant control multiplexing for flow-based microfluidic biochips. IEEE, p. 1-8.

[276] Ho T-Y, Chakrabarty K, Pop P. Digital microfluidic biochips: recent research and emerging challenges. Conference Digital microfluidic biochips: recent research and emerging challenges. p. 335-44.

[277] Chakrabarty K, Fair RB, Zeng J. Design tools for digital microfluidic biochips: toward functional diversification and more than moore. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2010;29(7):1001-17.

[278] Huang X, Xu C, Zhang L. An Efficient Algorithm for Optimizing the Test Path of Digital Microfluidic Biochips. Journal of Electronic Testing. 2020:1-14.

[279] Elvira KS, i Solvas XC, Wootton RC, Demello AJ. The past, present and potential for microfluidic reactor technology in chemical synthesis. Nature chemistry. 2013;5(11):905.

[280] Jiang X, Shao N, Jing W, Tao S, Liu S, Sui G. Microfluidic chip integrating high throughput continuous-flow PCR and DNA hybridization for bacteria analysis. Talanta. 2014;122:246-50.

[281] Chiva C, Olivella R, Borràs E, Espadas G, Pastor O, Solé A, et al. QCloud: A cloud-based quality control system for mass spectrometry-based proteomics laboratories. PLoS One. 2018;13(1):e0189209.

[282] Mateos J, Carneiro I, Corrales F, Elortza F, Paradela A, Del Pino MS, et al. Multicentric study of the effect of pre-analytical variables in the quality of plasma samples stored in biobanks using different complementary proteomic methods. Journal of proteomics. 2017;150:109-20.

[283] Carr SA, Abbatiello SE, Ackermann BL, Borchers C, Domon B, Deutsch EW, et al. Targeted peptide measurements in biology and medicine: best practices for mass spectrometry-based assay development using a fit-for-purpose approach. Molecular & Cellular Proteomics. 2014;13(3):907-17.

[284] Choi M, Eren-Dogu ZF, Colangelo C, Cottrell J, Hoopmann MR, Kapp EA, et al. ABRF Proteome Informatics Research Group (iPRG) 2015 Study: Detection of Differentially Abundant Proteins in Label-Free Quantitative LC–MS/MS Experiments. Journal of proteome research. 2017;16(2):945-57.

[285] Bittremieux W, Tabb DL, Impens F, Staes A, Timmerman E, Martens L, et al. Quality control in mass spectrometry‐based proteomics. Mass Spectrometry Reviews. 2018;37(5):697-711.

[286] Bittremieux W, Walzer M, Tenzer S, Zhu W, Salek RM, Eisenacher M, et al. The HUPO-PSI Quality Control Working Group: Making quality control more accessible for biological mass spectrometry. Anal Chem. 2017;89(8):4474-9.

[287] Lane L, Bairoch A, Beavis RC, Deutsch EW, Gaudet P, Lundberg E, et al. Metrics for the Human Proteome Project 2013–2014 and strategies for finding missing proteins. Journal of proteome research. 2014;13(1):15-20.

[288] Omenn GS, Lane L, Lundberg EK, Beavis RC, Nesvizhskii AI, Deutsch EW. Metrics for the Human Proteome Project 2015: progress on the human proteome and guidelines for high-confidence protein identification. Journal of proteome research. 2015;14(9):3452-60.

[289] Zhang B, Wang J, Wang X, Zhu J, Liu Q, Shi Z, et al. Proteogenomic characterization of human colon and rectal cancer. Nature. 2014;513(7518):382-7.

[290] Slebos RJ, Wang X, Wang X, Zhang B, Tabb DL, Liebler DC. Proteomic analysis of colon and rectal carcinoma using standard and customized databases. Scientific data. 2015;2(1):1-15.

[291] Stanfill BA, Nakayasu ES, Bramer LM, Thompson AM, Ansong CK, Clauss TR, et al. Quality control analysis in real-time (QC-ART): A tool for real-time quality control assessment of mass spectrometry-based proteomics data. Molecular & Cellular Proteomics. 2018;17(9):1824-36.

[292] Locati MD, Terpstra I, de Leeuw WC, Kuzak M, Rauwerda H, Ensink WA, et al. Improving small RNA-seq by using a synthetic spike-in set for size-range quality control together with a set for data normalization. Nucleic acids research. 2015;43(14):e89-e.

[293] Zhao R, Yan R, Chen Z, Mao K, Wang P, Gao RX. Deep learning and its applications to machine health monitoring. Mechanical Systems and Signal Processing. 2019;115:213-37.

[294] Shorten C. Machine Learning vs. Deep Learning. Towards Data Science; 2018.

[295] Tax D, Ypma A, Duin R. Support vector data description applied to machine vibration analysis. Conference Support vector data description applied to machine vibration analysis, vol. 54. Citeseer, p. 15-23.

[296] Widodo A, Yang B-S. Support vector machine in machine condition monitoring and fault diagnosis. Mechanical systems and signal processing. 2007;21(6):2560-74.

[297] Shatnawi Y, Al-Khassaweneh M. Fault diagnosis in internal combustion engines using extension neural network. IEEE Transactions on Industrial Electronics. 2013;61(3):1434-43.

[298] Wen L, Li X, Gao L, Zhang Y. A new convolutional neural network-based data-driven fault diagnosis method. IEEE Transactions on Industrial Electronics. 2017;65(7):5990-8.

[299] You D, Gao X, Katayama S. WPD-PCA-based laser welding process monitoring and defects diagnosis by using FNN and SVM. IEEE Transactions on Industrial Electronics. 2014;62(1):628-36.

[300] Bengio Y, Courville A, Vincent P. Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence. 2013;35(8):1798-828.

[301] Wang J, Xie J, Zhao R, Mao K, Zhang L. A new probabilistic kernel factor analysis for multisensory data fusion: Application to tool condition monitoring. IEEE Transactions on Instrumentation and Measurement. 2016;65(11):2527-37.

[302] Malhi A, Gao RX. PCA-based feature selection scheme for machine defect classification. IEEE Transactions on Instrumentation and Measurement. 2004;53(6):1517-25.

[303] Wang J, Xie J, Zhao R, Zhang L, Duan L. Multisensory fusion based virtual tool wear sensing for ubiquitous manufacturing. Robotics and Computer-Integrated Manufacturing. 2017;45:47-58.

[304] Jia F, Lei Y, Guo L, Lin J, Xing S. A neural network constructed by deep learning technique and its application to intelligent fault diagnosis of machines. Neurocomputing. 2018;272:619-28.

[305] Abdelgayed TS, Morsi WG, Sidhu TS. Fault detection and classification based on co-training of semisupervised machine learning. IEEE Transactions on Industrial Electronics. 2017;65(2):1595-605.

[306] Shao H, Jiang H, Zhang X, Niu M. Rolling bearing fault diagnosis using an optimization deep belief network. Measurement Science and Technology. 2015;26(11):115002.

[307] Funahashi K-i, Nakamura Y. Approximation of dynamical systems by continuous time recurrent neural networks. Neural networks. 1993;6(6):801-6.

[308] Lu C, Wang Y, Ragulskis M, Cheng Y. Fault diagnosis for rotating machinery: A method based on image processing. PloS one. 2016;11(10):e0164111.

[309] Smith WA, Randall RB. Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study. Mechanical Systems and Signal Processing. 2015;64:100-31.

[310] Amar M, Gondal I, Wilson C. Vibration spectrum imaging: A novel bearing fault classification approach. IEEE transactions on Industrial Electronics. 2014;62(1):494-502.

[311] Xia M, Han G, Zhang Y, Wan J. Intelligent fault diagnosis of rotor-bearing system under varying working conditions with modified transfer CNN and thermal images. IEEE Transactions on Industrial Informatics. 2020.

[312] Canizo M, Triguero I, Conde A, Onieva E. Multi-head CNN–RNN for multi-time series anomaly detection: An industrial case study. Neurocomputing. 2019;363:246-60.

[313] Napoletano P, Piccoli F, Schettini R. Anomaly detection in nanofibrous materials by CNN-based self-similarity. Sensors. 2018;18(1):209.

[314] Van Wyk F, Wang Y, Khojandi A, Masoud N. Real-time sensor anomaly detection and identification in automated vehicles. IEEE Transactions on Intelligent Transportation Systems. 2019;21(3):1264-76.

[315] Zhang L, Cheng B. Transferred CNN based on tensor for hyperspectral anomaly detection. IEEE Geoscience and Remote Sensing Letters. 2020;17(12):2115-9.

[316] Keogh C. Bringing Live Cells into Focus. Imaging & Microscopy. 2008;10(1):36-7.

[317] Hensman P, Masko D. The impact of imbalanced training data for convolutional neural networks. Degree Project in Computer Science, KTH Royal Institute of Technology. 2015.

[318] Drozdzal M, Chartrand G, Vorontsov E, Shakeri M, Di Jorio L, Tang A, et al. Learning normalized inputs for iterative estimation in medical image segmentation. Medical image analysis. 2018;44:1-13.

[319] Ulyanov D, Vedaldi A, Lempitsky V. Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:160708022. 2016.

[320] Bloice MD, Stocker C, Holzinger A. Augmentor: an image augmentation library for machine learning. arXiv preprint arXiv:170804680. 2017.

[321] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Communications of the ACM. 2017;60(6):84-90.

[322] Ciregan D, Meier U, Schmidhuber J. Multi-column deep neural networks for image classification. Conference Multi-column deep neural networks for image classification. IEEE, p. 3642-9.

[323] Hauberg S, Freifeld O, Larsen ABL, Fisher J, Hansen L. Dreaming more data: Class-dependent distributions over diffeomorphisms for learned data augmentation. Conference Dreaming more data: Class-dependent distributions over diffeomorphisms for learned data augmentation. p. 342-50.

[324] Perez L, Wang J. The effectiveness of data augmentation in image classification using deep learning. arXiv preprint arXiv:171204621. 2017.

[325] He Z, Ye X, He Z, Li Y. Embedding Rotate-and-Scale Net for Learning Invariant Features of Simple Images. Conference Embedding Rotate-and-Scale Net for Learning Invariant Features of Simple Images. Springer, p. 131-42.

[326] San Martin G, López Droguett E, Meruane V, das Chagas Moura M. Deep variational auto-encoders: A promising tool for dimensionality reduction and ball bearing elements fault diagnosis. Structural Health Monitoring. 2019;18(4):1092-128.

[327] Guo S, Yang T, Gao W, Zhang C. A novel fault diagnosis method for rotating machinery based on a convolutional neural network. Sensors. 2018;18(5):1429.

[328] The MathWorks. imageInputLayer. The MathWorks, Inc.; 2020.

[329] Chen L, Zhang H, Xiao J, Nie L, Shao J, Liu W, et al. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. Conference Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. p. 5659-67.

[330] Wu H, Gu X. Towards dropout training for convolutional neural networks. Neural Networks. 2015;71:1-10.

[331] Paoletti ME, Haut JM, Fernandez-Beltran R, Plaza J, Plaza A, Li J, et al. Capsule networks for hyperspectral image classification. IEEE Transactions on Geoscience and Remote Sensing. 2018;57(4):2145-60.

[332] The MathWorks. List of Deep Learning Layers. The MathWorks, Inc.; 2020.

[333] Wang ZJ, Turko R, Shaikh O, Park H, Das N, Hohman F, et al. CNN Explainer: Learning Convolutional Neural Networks with Interactive Visualization. arXiv preprint arXiv:200415004. 2020.

[334] Yang B, Lei Y, Jia F, Xing S. An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings. Mechanical Systems and Signal Processing. 2019;122:692-706.

[335] The MathWorks. Create Simple Deep Learning Network for Classification. The MathWorks, Inc.; 2020.

[336] Murphy KP. Machine learning: a probabilistic perspective: MIT press, 2012.

[337] Beale MH, Hagan MT, Demuth HB. Neural network toolbox. User's Guide, MathWorks. 2010;2:77-81.

[338] Deng L. The mnist database of handwritten digit images for machine learning research [best of the web]. IEEE Signal Processing Magazine. 2012;29(6):141-2.

[339] Bai S, Kolter JZ, Koltun V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:180301271. 2018.

[340] Paluszek M, Thomas S. Finding Circles with Deep Learning. Practical MATLAB Deep Learning: Springer; 2020. p. 43-63.

[341] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556. 2014.

[342] The MathWorks. Pretrained Deep Neural Networks. The MathWorks, Inc.; 2021.

[343] The MathWorks. trainNetwork - Train neural network for deep learning. The MathWorks, Inc.; 2020.

[344] Cheng Y, Wang D, Zhou P, Zhang T. A survey of model compression and acceleration for deep neural networks. arXiv preprint arXiv:171009282. 2017.

[345] Cheng J, Wang P-s, Li G, Hu Q-h, Lu H-q. Recent advances in efficient computation of deep convolutional neural networks. Frontiers of Information Technology & Electronic Engineering. 2018;19(1):64-77.

[346] He T, Zhang Z, Zhang H, Zhang Z, Xie J, Li M. Bag of tricks for image classification with convolutional neural networks. Conference Bag of tricks for image classification with convolutional neural networks. p. 558-67.

[347] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. Imagenet large scale visual recognition challenge. International journal of computer vision. 2015;115(3):211-52.

[348] Zhou B, Khosla A, Lapedriza A, Torralba A, Oliva A. Places: An image database for deep scene understanding. arXiv preprint arXiv:161002055. 2016.

[349] Liu W, Wen Y, Yu Z, Yang M. Large-margin softmax loss for convolutional neural networks. Conference Large-margin softmax loss for convolutional neural networks, vol. 2. p. 7.

[350] Xu Q, Zhang M, Gu Z, Pan G. Overfitting remedy by sparsifying regularization on fully-connected layers of CNNs. Neurocomputing. 2019;328:69-74.

[351] Thanapol P, Lavangnananda K, Bouvry P, Pinel F, Leprévost F. Reducing Overfitting and Improving Generalization in Training Convolutional Neural Network (CNN) under Limited Sample Sizes in Image Recognition. Conference Reducing Overfitting and Improving Generalization in Training Convolutional Neural Network (CNN) under Limited Sample Sizes in Image Recognition. IEEE, p. 300-5.

[352] Zhang X, Fang Z, Wen Y, Li Z, Qiao Y. Range loss for deep face recognition with long-tailed training data. Conference Range loss for deep face recognition with long-tailed training data. p. 5409-18.

[353] Hochreiter S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems. 1998;6(02):107-16.

[354] Shin H-C, Roth HR, Gao M, Lu L, Xu Z, Nogues I, et al. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE transactions on medical imaging. 2016;35(5):1285-98.

[355] Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-cam: Visual explanations from deep networks via gradient-based localization. Conference Grad-cam: Visual explanations from deep networks via gradient-based localization. p. 618-26.

[356] Kudo M, Toyama J, Shimbo M. Multidimensional curve classification using passing-through regions. Pattern Recognition Letters. 1999;20(11-13):1103-11.

[357] Kwaśniewska A, Rumiński J, Rad P. Deep features class activation map for thermal face detection and tracking. Conference Deep features class activation map for thermal face detection and tracking. IEEE, p. 41-7.

[358] The MathWorks. Visualize Activations of a Convolutional Neural Network. The MathWorks, Inc.; 2021.

[359] Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A. Learning deep features for discriminative localization. Conference Learning deep features for discriminative localization. p. 2921-9.

[360] The MathWorks. Investigate Network Predictions Using Class Activation Mapping. The MathWorks，Inc.; 2021.

[361] Krizhevsky A. One weird trick for parallelizing convolutional neural networks. arXiv preprint arXiv:14045997. 2014.

[362] Gatys LA, Ecker AS, Bethge M. Texture synthesis using convolutional neural networks. arXiv preprint arXiv:150507376. 2015.

[363] Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. Conference Visualizing and understanding convolutional networks. Springer, p. 818-33.

[364] He Y, Zhang X, Sun J. Channel pruning for accelerating very deep neural networks. Conference Channel pruning for accelerating very deep neural networks. p. 1389-97.

[365] Lin S, Ji R, Yan C, Zhang B, Cao L, Ye Q, et al. Towards optimal structured cnn pruning via generative adversarial learning. Conference Towards optimal structured cnn pruning via generative adversarial learning. p. 2790-9.

[366] Handa A, Bloesch M, Pătrăucean V, Stent S, McCormac J, Davison A. gvnn: Neural network library for geometric computer vision. Conference gvnn: Neural network library for geometric computer vision. Springer, p. 67-82.

[367] Li H, Xu Z, Taylor G, Studer C, Goldstein T. Visualizing the loss landscape of neural nets. arXiv preprint arXiv:171209913. 2017.

[368] Hoffer E, Banner R, Golan I, Soudry D. Norm matters: efficient and accurate normalization schemes in deep networks. arXiv preprint arXiv:180301814. 2018.

[369] He K, Zhang X, Ren S, Sun J. Identity mappings in deep residual networks. Conference Identity mappings in deep residual networks. Springer, p. 630-45.

[370] Han D, Kim J, Kim J. Deep pyramidal residual networks. Conference Deep pyramidal residual networks. p. 5927-35.

[371] Niu X-X, Suen CY. A novel hybrid CNN–SVM classifier for recognizing handwritten digits. Pattern Recognition. 2012;45(4):1318-25.

[372] Sun X, Park J, Kang K, Hur J. Novel hybrid CNN-SVM model for recognition of functional magnetic resonance images. Conference Novel hybrid CNN-SVM model for recognition of functional magnetic resonance images. IEEE, p. 1001-6.

[373] Shima Y, Nakashima Y, Yasuda M. Handwritten digits recognition by using CNN alex-net pre-trained for large-scale object image dataset. Conference Handwritten digits recognition by using CNN alex-net pre-trained for large-scale object image dataset. p. 36-40.

[374] Amraee S, Vafaei A, Jamshidi K, Adibi P. Abnormal event detection in crowded scenes using one-class SVM. Signal, Image and Video Processing. 2018;12(6):1115-23.

[375] Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. arXiv preprint arXiv:160207360. 2016.

[376] Van Merriënboer JJ, Sweller J. Cognitive load theory in health professional education: design principles and strategies. Medical education. 2010;44(1):85-93.

[377] Van Merriënboer JJ, Kester L, Paas F. Teaching complex rather than simple tasks: Balancing intrinsic and germane load to enhance transfer of learning. Applied Cognitive Psychology: The Official Journal of the Society for Applied Research in Memory and Cognition. 2006;20(3):343-52.