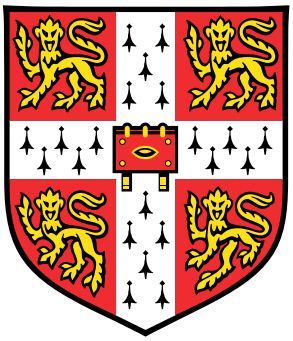
Cross-Modality Profiling of High-Content Microscopy Images with Deep Learning




Jan Oscar Cross-Zamirski

Department of Applied Mathematics and Theoretical Physics
Department of Psychology
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

Downing College April 2023

Declaration

This thesis is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the Preface and specified in the text. I state that, except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification at the University of Cambridge, or any other University or similar institution except as declared in the Preface and specified in the text. It does not exceed the prescribed word limit for the relevant Degree Committee.

Jan Oscar Cross-Zamirski
April 2023

Abstract

Title: Cross-Modality Profiling of High-Content Microscopy Images with Deep Learning

In this thesis we investigate the use of deep learning for cross-modality and multi-modal image-based profiling applications. In particular, we explore the utility of the brightfield image modality with deep generative models, and also propose new methods to integrate metadata labels freely obtained in high-content screening into deep learning architectures.

The use of automated microscopy in high-content phenotypic screening of cells treated with compounds or genetic perturbations produces a large amount of high-dimensional data. One example which we focus on in this thesis is Cell Painting, a standardized pipeline used to capture rich cell morphology. Typically, fluorescent stained images are the focus of image-based profiling, where the aim is to extract meaningful features from the images which can be used to represent the biology of the cells, and compare the effects of the treatments. Image-based profiling is central to screening in drug discovery, and is used to guide the selection of drug candidates to take to clinical trials.

With an abundance of large databases of high-dimensional images, deep learning for image-based profiling has risen as its own sub-field, and there are now entire drug discovery pipelines selecting drugs to take to clinical trials built upon deep learning foundations. This is possible due to advances in deep learning in computer vision. In this thesis we explore and adapt recent and powerful deep learning approaches including the generative adversarial network, self-supervised vision transformers, and the diffusion denoising probabilistic model.

A major challenge in this space is to use state-of-the-art frameworks from computer vision in a way which is sensitive to the challenges of drug discovery. Image-based profiling, and particularly deep learning in image-based profiling, is a new and maturing field. In this thesis we tackle the unaddressed challenge of incorporating the cheaper, easier to obtain but classically less informative brightfield modality, as well as using underutilised but freely available metadata alongside the images to guide the training

of deep neural networks. As we are working in interdisciplinary science, we do this while making models which are visually interpretable to the many people working in drug discovery who are not familiar with, or potentially sceptical of deep learning.

This thesis presents the first study to predict all five fluorescent Cell Painting channels from brightfield images. We explore the potential, benefits and limitations of this new approach. Next, we introduce a weakly-self supervised learning framework to learn feature representations which are guided by informative metadata. Finally, we present the first study to use a diffusion model with high-content microscopy images. We generate entire plates of synthetic Cell Painting images of exceptional image quality to make predictions about the information these models are capable of capturing, and investigate if this can also be guided by labels.

Acknowledgements

This work was funded by the UKRI-BBSRC DTP (UK Research and Innovation and the Biotechnology and Biological Science Research Council Doctoral Training Partnership) studentship grant, and AstraZeneca. I would also like to acknowledge the resources provided by the AstraZeneca Scientific Computing Platform (SCP) who made this work possible.

Firstly, I would like to thank my supervisors Professor Carola-Bibiane Schönlieb and Dr. Yinhai Wang for their guidance, expertise, pragmatism and unbroken support.

I am grateful to everyone who has been a part of my academic journey including Professor Zoe Kourtzi, the excellent staff and students who have always been willing to give their time and resources in both the Adaptive Brain Lab and the Cambridge Image Analysis Group, as well as the academic communities at Corpus Christi College and Downing College. Thank you to the Phenomics group at AstraZeneca, especially Adam Corrigan and Chen Qian who have helped me many times.

I am exceedingly grateful for the hard work, valuable input and skilled analysis from Elizabeth Mouchet and Praveen Anand who have contributed greatly to the direction of this project.

In particular, I would like to thank Riku Turkki, Guy Williams and Joseph Giorgio who have all been outstandingly generous to me with their time, trust, knowledge and kindness. Without either one of you, I am sure this thesis would not exist. Thank you.

Finally, my family, friends, team-mates and loved ones are all great. Thanks always.

For Mark, who stuck his neck out.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

Drug discovery is a lengthy and costly endeavour. The time and cost scales associated with developing a single drug are decades and billions of dollars [1]. One of the major bottlenecks in drug discovery is determining the mechanisms of action of candidate compounds prior to advancing to clinical trials, a process which takes many years. After years spent selecting and refining hits and leads, a candidate drug reaching clinical trials will still have a 90% chance of failing [2].

Such long timescales and high rates of failure in the drug discovery process may be preventing the development of vital, life-changing or life-saving treatments. While drug discovery has become more data-driven in recent years, there is still immense cost and time associated with the process. However, there is great hope that artificial intelligence (AI) will bring a new era of efficient, data-driven drug discovery, and significantly accelerate the process.

Machine learning, a branch of AI, has introduced numerous powerful methods for analysis and prediction at all stages of the drug discovery pipeline. Machine learning methods can be used for target identification and validation, compound screening and lead discovery, as well as both preclinical and clinical development [3]. Significant advances have been made, efficiencies increased, and machine learning drug discovery pipelines established which have led to clinical trials [4]. AI in drug discovery is already a multi-billion dollar industry and is expected to continue growing rapdily across the next decade [5].

Despite this, AI has not yet revolutionised drug discovery [6]. It has been widely adopted in certain areas, however there are a number of reasons as to why current advances have fallen short of the impact some predicted. One reason is that, at least by drug discovery timescales, we are still in the infancy of AI in drug discovery. Large AI investment has occurred predominately in the last 5 years, and AI-driven pipelines have

not had time to run to completion. Chris Gibson, the co-founder and CEO of Recursion summarises this: "In 2023, we'll see more clinical trial readouts and pre-clinical studies of AI-discovered drugs that will set the bar for what it takes to be taken seriously as leaders in this space." [7]. It is unlikely these prototype AI pipelines will be successful at first attempt.

Conversely, AI - and particularly deep learning - is maturing across many industries. Marked by an abundance of data, advanced techniques that continue to improve, and ample computing power, the last year in particular has seen mainstream commercial success for AI, for example with large language and generative models. Yet it is still not used extensively in pharmaceutical pipelines. When it comes to capturing chemical and biological properties, trustworthiness, interpretability, and robustness are especially crucial. Deep learning is yet to provide this. Additionally, machine learning can be a victim of its own success. Rapid growth brings its own problems, and deep learning is advancing incredibly fast. By the time an algorithm is developed and implemented, new methods have surpassed its performance - which can mean nothing is adopted into a clinical pipeline, which takes years. Perhaps the dust is yet to settle for these new, groundbreaking AI advances.

Alternatively, some remark that there has not been enough focus on addressing the main issues in drug discovery. The aim of AI needs to be to assist predicting drug properties, in particular with respect to deciding which compounds to take to preclinical and clinical stages. Bender and Cortés-Ciriano [6] stress the importance of using AI to increase the quality of predictions, as opposed to speed. This is due to the extremely high failure rate at the clinical stage. In other words, if you're going to fail its better to fail early as it will save a huge amount of time and cost compared to failing at the same rate slightly quicker. This places an immense burden on making high-quality models and predictions at the earliest stages of the pipeline.

One of the most promising methodologies which has emerged to make accurate and biologically meaningful predictions at an early stage is phenotypic screening [8]. Phenotypic screening offers a high-content and high-throughput way to move away from the slower and more limited alternative of target-based screening. Given the huge amount of data, and restrictions of classical methods in phenotypic screening, there is scope for deep learning to make a large impact in this space [9]. Advances in profiling and specifically machine learning has inspired biotech companies such as Recursion and insitro [8] to focus on profile-based phenotypic screening assays, which have historically been ignored by pharmaceutical companies. The aim is to more efficiently evaluate the

potential of compound intervention by testing hundreds of model systems, allowing for the exploration of the less frequently studied disease spectrum of compounds.

Central to phenotypic screening is high-content microscopy - a way to capture millions of cells treated with hundreds of compounds at high-resolution in an automated fashion. Cell Painting [10] is one of the newer and now widely adopted standardized pipelines for staining, imaging and profiling cells. Advances in computer vision have renewed image-based profiling [9, 11] - the process of extracting phenotypic features from microscopy images to make predictions about mechanisms of the compounds used to treat the cells. The potential of image-based profiling is immense, and machine learning may hold the key to providing automated and reliable ways to efficiently and accurately select new drug candidates at an early stage. Today, deep learning is one of the most heavily researched and fast moving fields. Computer vision has enabled automatic image analysis of massive datasets to identify biological and phenotypic changes associated with disease, protein binding, mechanism of action and target prediction.

Computer vision breakthroughs guide studies in applied fields such as drug discovery, however this brings its own challenges. Traditionally, models in biology and medicine are designed to be used as a tool to solve problems or make predictions driven by the data. They are naturally based on the prevalent techniques of the time, such as the convolutional neural network. Yet there are differences between what is useful for general computer vision applications, and what is required for drug discovery tasks. This is particularly challenging as generalist AI models are becoming so capable that they can outperform many specialised, purpose-built models on a multitude of complex tasks. How we as researchers adapt to interact with these immensely powerful models (for which an end goal could be considered artificial general intelligence) will become important in the future. A deep understanding of both the machine learning techniques as well as the data and challenges in drug discovery will likely yield the most success.

In this thesis, we tackle challenges in the fields of phenotypic screening and image-based profiling with novel deep learning techniques. We introduce methods to address some of the drawbacks of the fluorescent staining process, and to incorporate under-utilised and meaningful modalities and metadata from high-content screens. We do this with the broader drug discovery challenges in mind, and aim to develop models which are understandable by and interpretable to biologists.

A hurdle for AI is to increase biological insight of, and trust in *black box* deep learning methods. With this in mind, every model presented in this thesis has visual interpretability. We provide holistic quantifications of performance with image-level

metrics as well as with extensive downstream benchmarking, placing our models in typical screening and image-based profiling pipelines. Our goal is not just to create algorithms that temporarily achieve the best performance by a specific measure, but to examine how well deep learning methods work with real data, and to understand how these models behave. This knowledge is essential for deep learning to be adopted into successful drug discovery pipelines.

1.1 Thesis overview and contribution

In Chapter 2 we present the background, concepts and non-mathematical methods required for the thesis. We provide high-level overviews of the fields of microscopy imaging, phenotypic profiling, drug discovery and machine learning, and the intersections between them.

In Chapter 3 we detail the mathematical and machine learning methods required for the experimental chapters. We discuss computer vision network architectures, techniques used to learn representations from high-content microscopy images, deep generative models and the metrics used to evaluate their performance.

Chapters 4, 5 and 6 are the three experimental chapters of this thesis. Summaries of these chapters and their contributions are presented as follows:

1.1.1 Label-free prediction of Cell Painting from brightfield images

We investigate label-free Cell Painting by predicting all five fluorescent Cell Painting channels from brightfield images with two deep learning models. Our contributions are:

- This study introduces the first model to predict all five Cell Painting channels from brightfield images.
- We show incorporating adversarial loss improves the quality of the predicted Cell Painting images when compared to an absolute loss function.
- We present an extensive evaluation of the predicted Cell Painting images using brightfield images from unseen batches, including a detailed breakdown comparing the predicted morphological features to the ground truth.
- We explore the potential of downstream clustering and toxicity analysis with label-free Cell Painting.

This work was done in collaboration with Elizabeth Mouchet, Guy Williams, Carola-Bibiane Schönlieb, Riku Turkki and Yinhai Wang. I implemented the models and the experiments in python, and performed the analysis and evaluation of the images. I implemented the image pre-processing and Cell Profiler pipeline together with Guy Williams. Together with Riku Turkki, I performed the methodology development, data interpretation and statistical analysis. AstraZeneca provided the data used in this study, particularly Elizabeth Mouchet and Guy Williams who conducted the laboratory work and imaging.

The work presented in this chapter was released as a preprint in 2021 and accepted for publication by *Nature Scientific Reports* in 2022.

1.1.2 Self-supervised learning of phenotypic representations with weak labels

With a self-supervised framework we use weak label information to guide learning phenotypic representations from high-content fluorescent microscopy images of cells. Our contributions are:

- We present WS-DINO, a novel method to incorporate free metadata into a self-supervised framework for image-based profiling.
- We outperform all previous models in the literature on the BBBC021 dataset in not-same-compound and not-same-compound-and-batch mechanism of action prediction.
- We present a competitive method which does not require single-cell segmentation as a pre-processing step to learn high-quality phenotypic representations.
- With self-attention maps we can show our model is weighting on biologically meaningful structures in the images, increasing interpretability and confidence in a deep learning approach.

This work was done in collaboration with Elizabeth Mouchet, Guy Williams, Carola-Bibiane Schönlieb, Riku Turkki and Yinhai Wang. I implemented the models and the experiments in python, and performed the analysis and evaluation of the images. I implemented the image pre-processing together with Guy Williams. Together with Riku Turkki, I performed the methodology development and data interpretation.

The work presented in this chapter was released as a preprint in 2022 and accepted to *NeurIPS 2022 Workshop on Learning Meaninful Representations of Life.*

1.1.3 Class-guided image-to-image diffusion: Cell Painting from brightfield images with class labels

We extend the work of Chapter 4 by using a diffusion model to predict Cell Painting from brightfield. We introduce a novel way to incorporate class labels into the image-to-image framework and compare the different methods. We address the merits of using generative models to enhance brightfield images vs using the brightfield outright in profiling tasks. Our contributions are:

- We introduce and implement the first general framework for multi-modal conditional diffusion for paired images with labels.
- We apply our multi-modal conditional diffusion model to cross-modality prediction of 5-channel Cell Painting fluorescent microscopy from 3-channel brightfield images.
- We show that incorporating label information into training can increase prediction on the downstream target matching task with both extracted biological features and a transfer learning approach.
- We present a number of visualisations to compare the profiling properties of the predicted Cell Painting images against both the real Cell Painting images and the brightfield channels.

This work was done in collaboration with Praveen Anand, Elizabeth Mouchet, Guy Williams, Carola-Bibiane Schönlieb and Yinhai Wang. I implemented the models and the experiments in python, and performed the analysis and evaluation of the images. Guy Williams implemented the image pre-processing and Cell Profiler pipeline. Together with Praveen Anand and Guy Williams, I performed the methodology development and data interpretation. AstraZeneca provided the data used in this study, which is now publicly available.

The work presented in this chapter was released as a preprint and submitted for publication in 2023.

Chapter 2

Preliminaries

This chapter details the background, concepts and predominately non-mathematical methods required for this thesis. The aim of this chapter is to give background to the work with an overview of relevant methods. Additionally, we provide context to the thesis with high-level overviews of the fields of microscopy imaging, phenotypic profiling, drug discovery and machine learning, and the intersections between them. We start by introducing the imaging modalities used in high-content screening (HCS) and then move to image-based profiling. We introduce the extra data types which can be used in image-based profiling, and place classical methods and machine learning analysis within the drug discovery pipeline while discussing the associated challenges. Finally, we introduce the basic concepts of machine learning and the subfield of deep learning. Specific machine learning methodologies, architectures and models are beyond the scope of this chapter and are covered in Chapter 3.

2.1 Drug screening

Drug screening technologies are the processes "by which potential drugs are identified and optimized before selection of a candidate drug to progress to clinical trials" [12]. Screening involves the testing of many potential drugs to assess their impact on a disease. The screening process can often be simplified to a three-step process [13] of i) designing a model system to closely mimic a disease condition ii) producing a disease-associated response with selected stimuli and iii) quantifying the response with readouts and features [9]. Screens can vary from very simple to very complex, with several different approaches which include target-based screening [14], genomic screening [15], and phenotypic screening [13, 9, 8].

Target-based screening was the method of choice in drug discovery for the majority of the past three decades [8]. In target-based screening, compounds are identified which interact with specific targets, proteins, enzymes, or receptors in the body. *Target* is a broad term to describe "a range of biological entities which may include for example proteins, genes and RNA" [16]. Target-based screening aims to identify compounds that bind to defective pathways or proteins in order to treat different diseases. Examples of diseases which can be treated with drugs discovered from target-based studies include cancer, cardiovascular diseases, neurodegenerative diseases (Alzheimer's and Parkinson's), diabetes and many others [17].

Despite the prevalence of target-based methods, phenotypic screening has also been very successful in identifying both first-in-class and best-in-class drugs [8, 18, 19]. Phenotypic screening strategies are those where hit or lead compounds are selected without any previous knowledge of the drug target's underlying mechanisms of action or role in disease [20]. Effects of the compounds are quantified through observable phenotypic characteristics, allowing for the identification of compounds which have therapeutic potential without knowledge of the specific targets.

Genomic screening involves the use of genomic technologies, such as DNA microarrays or RNA sequencing [15], to identify genes or pathways that are involved in biological processes or diseases. By identifying changes in genomic features such as gene expression, it is possible to gain insights into the underlying mechanisms of a disease, and design or identify new drugs accordingly. CRISPR gene editing technology can also be used to determine gene and protein function in disease [21]. Genomic screening is not necessarily separate from phenotypic screening, and automated, high-throughput phenotypic screening can be used to analyse huge numbers of genetic variations and compounds to assess their impact on specific genes or pathways [22].

Between 1999 and 2008, it was shown that phenotypic screening contributed to the discovery of 28 first-in-class small-molecule drugs, compared to just 17 drugs from target-based approaches, despite target-based screening being the predominant method in drug discovery at the time [18]. Since this research, phenotypic screening has experienced a resurgence [23], fuelled further by advances in omics, profiling and computational approaches including machine learning (which we will discuss later in this chapter) [19]. There are disadvantages of both target-based and phenotpyic screening approaches, however it seems there may be significantly more potential for progress to be made in phenotypic screening, which is not limited by the availability of suitable targets, and may be able to address incompletely understood complexity of diseases [8]. Phenotypic approaches may be able to provide a more realistic representation of the *in*

vivo effects of a compound, by taking into account the complex interactions between different biological pathways and systems.

Machine learning studies have attempted to overcome traditional issues with phenotypic screens such as difficulty in target deconvolution, identifying hits and being harder to scale up. The ability of machine learning and deep learning to analyse large amount of data and identify previously unknown relationships holds immense promise for drug screening, and many recent studies have explored this [24, 25].

In modern drug discovery, a variety of different screening approaches (including high-content CRISPR screens [21]) are used to gain insights into the underlying mechanisms of a disease and to identify potential targets [16]. Combined screens which are primarily phenotypic, but can incorporate target or mechanism of action (MOA) information can be used to support clinical candidates to progress more drugs more quickly to the clinical trial phase [19].

Machine learning models which can synthesise multi-modal data types could have a large impact in this space. Large pharmaceutical companies have traditionally focused on target-based screens of large compound libraries, which can be more efficiently processed using streamlined, customized assays. However, there is evidence that high-content phenotypic profiling is more effective at capturing biological information [9]. Alongside this, advances in profiling and specifically machine learning has inspired biotech companies such as Recursion and insitro [4] to focus on profile-based phenotypic screening assays. The aim of this approach is to more efficiently evaluate the potential of compound intervention by testing hundreds of model systems, allowing for the exploration of the less frequently studied disease spectrum of compounds [9]. Fundamental to these new methods are high-content phenotypic screening (Section 2.2) and image-based profiling (Section 2.4).

2.2 High-content screening

High-content screening (HCS) (or high-content imaging (HCI)) [26], is the term for a set of technologies used to efficiently and rapidly analyse a large number of compounds to detect and assess their impact on a disease. Specifically where the images are capable of yielding *high-content* multi-parametric data. HCS is a high-throughput technique which specifically refers to the use of automated microscopy and image analysis to measure and quantify cellular or molecular characteristics. HCS technology is at the forefront of drug discovery and is used at all stages of the drug discovery and development pipeline [27].

Many phenotypic screens are high-content in nature and will utilize automated microscopy machines such as the CellVoyager CV8000 (Yokogawa) [28] to acquire large numbers of images, followed by computational image analysis (classical or machine learning) to measure cellular or molecular characteristics. Analysis of whole cells and components of cells results in multiple phenotypic parameters which are used to quantify compound activity or to identify specific biological effects based on the given treatments. The image analysis, feature extraction and utilization of images as data sources is referred to as image-based profiling [9, 11] (Section 2.4).

We summarize the image-based profiling pipeline in Fig. 2.1. The cell samples are generally prepared in 384-well microplates, although smaller and larger plates can also be used. Cells are treated with the chosen compounds or genetic perturbations, and incubated for a set time period before being fixed and stained [9]. The automated microscope captures images from each well, producing multiple image channels for unique stains and and z-planes. Typically tens of gigabytes of images are captured for each 384-well plate [11].

HCS is applied to all aspects of drug discovery including primary and post-primary compound screening, multivariate drug profiling and early evaluation of ADMET (absorption, distribution, metabolism, excretion and toxicity) properties [30, 31]. Additionally, HCS data is used in target/MOA identification and in functional genomics to identify and characterise genetic interactions. The pipeline we have described is not unique to any particular stains, cells, treatments or image-capture techniques.

Although ubiquitous in application, HCS is primarily used in compound library screens to identify hits or leads early in the drug discovery pipeline. HCS is used academia to advance techniques for drug discovery, but also in systems biology with RNA or genetic interference assays [27]. In recent years, there have been several academic collaborations that have released large publicly available HCS datasets including the Broad Bioimage Benchmark Collection (BBBC) from the Broad Institute [32] and Target-2 [33], a Cell Painting database (Section 2.3) from the Joint Undertaking for Morphological Profiling with Cell Painting (JUMP CP) [34].

High-content screening (and indeed all types of drug screening) is necessary in drug discovery as it is practically impossible to test the efficacy and safety of all candidate compounds *in vivo* (in human or animal studies). High-content, high-throughput screening assays provide *in vitro* pipelines to test large numbers of drug candidates in objective, quantifiable and reproducible ways. New computational and deep learning methods which can be referred to as *in silico* [35] are also appearing as alternatives to replace, enhance, or accelerate some parts of the traditional HCS assay. Developing

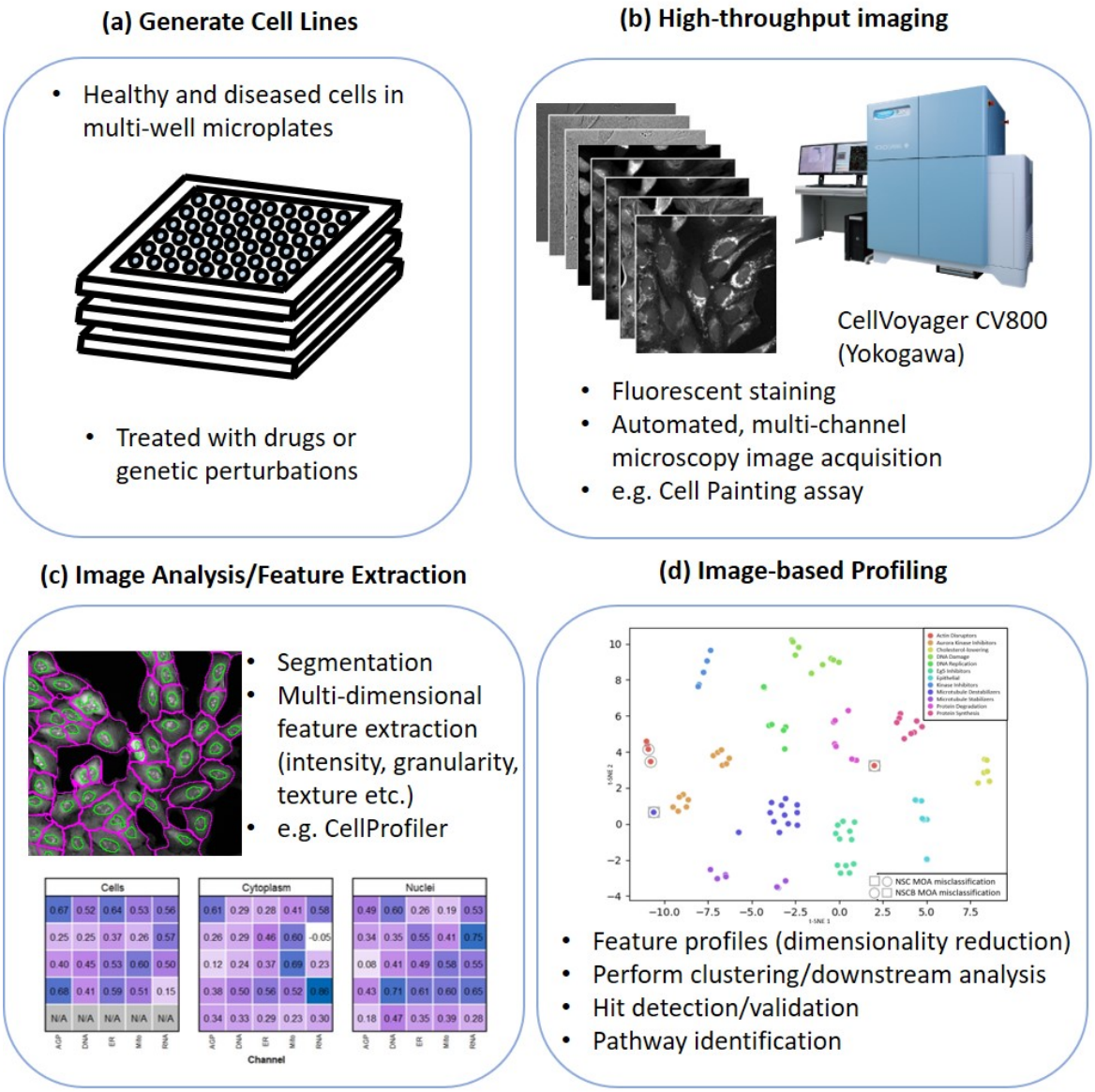


Fig. 2.1 A typical high-content microscopy screening pipeline. **(a)** Cell line generation in multi-well microplates. **(b)** high-throughput image acquisition after fluorescent staining of the cells (Section 2.2.1). **(c)** Computational image analysis and feature extraction **(d)** Image-based profiling of the extracted features (Section 2.4). Adapted from similar figures from Chandrasekaran *et al.* [9] and Chessel and Carazo Salas [29]

and testing these models forms a significant part of this thesis (Chapter 4 and Chapter 6).

2.2.1 Fluorescent staining

Fluorescent staining is one of the most powerful and important techniques in high-content microscopy screening [36] as it allows for multiple cellular structures, molecules and biomarkers to be imaged. Samples of cells in multi-well plates are stained with dyes which fluoresce at specific wavelengths of light. For each wavelength corresponding to the stain, images are captured with automated confocal microscopes [37] as a single grayscale image (channel). One advantage of fluorescent microscopy is that multiple stains can be applied and imaged simultaneously (up to a limit). An example of three simultaneously captured fluorescent channels for a single group of cells is shown in Figure 2.2

Fluorescent dyes are added to a sample to fluoresce under particular wavelengths of light. There are two main ways to achieve fluorescence: either add a small molecule or antibody-based dye, or genetically engineer a protein to express a fluorescent tag. One example is Hoechst, 4',6-diamidino-2-phenylindole (DAPI) which exhibits around a 20-fold enhancement of fluorescence upon binding to AT regions of dsDNA, when excited by a 405nm wavelength (violet) laser [40]. Typically DAPI is used to stain the nucleus, and the images captured from cells stained with this dye can be used to segment and count cells, measure DNA content and apoptosis (cell death). There are a number of ways to stain specific parts of the cells, including immunofluorescent staining where a primary antibody is used to bind to the desired protein or molecule, followed by a secondary, fluorescent antibody to bind to the primary antibody [41].

Cell microscopy imaging with fluorescent staining is very effective at observing drug activity and cell behavior, providing understanding of the MOA [31] as well as a toxicity assessment which is vital to exclude mutagenic or carcinogenic activity of drug candidates [36]. Fluorescent stained images are of high resolution and detail, and are used routinely as the inputs for image-based profiling (Section 2.4), followed by analysis using dedicated software such as CellProfiler [42, 43] designed to quantify thousands of phenotypic features.

However, there are a number of drawbacks to fluorescent microscopy. It can be expensive, time consuming and labour-intensive. Applying dyes to cells is an intrusive process which can permanently damage the cells or alter their behaviour [44]. Dyes can be cytotoxic, and can cause damage to cells or subcellular structures [45].

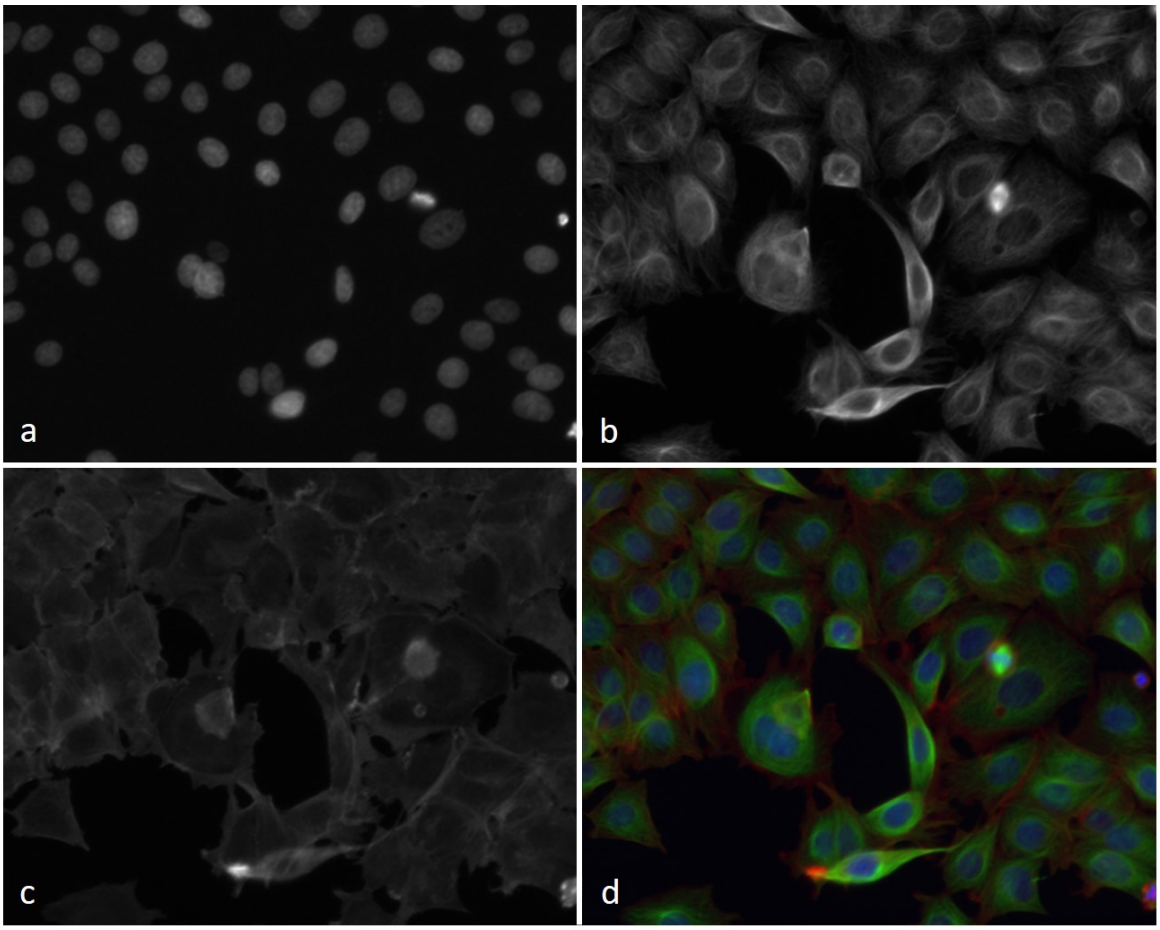


Fig. 2.2 A typical example of fluorescent stained MCF-7 breast cancer cells from the publicly available BBBC021 dataset [38, 39]. **(a)** DNA (DAPI) channel **(b)** β-tubulin channel **(c)** F-actin channel **(d)** Colour composite image of the F-actin channel (red), DNA channel (blue) and the β-tubulin channel (green). Each image is a quarter-crop of a field of view of a single well from one of the plates in the dataset.

Photobleaching is another potential problem where samples become less fluorescent when exposed to light, causing unwanted intensity variations over time [36].

Typically, there are a maximum of six stains which can be applied simultaneously across five imaging channels [46] in order to avoid spectral overlap. Furthermore, certain combinations of dyes are restricted due to the particular wavelength the dye can be imaged at. These technical limitations can hinder the ability of the scientist to capture morphological information from the unstainable subcellular compartments. Therefore, image-based assays which rely on fluorescent staining are restricted by the finite number of imaging channels available.

Imaging of live cells in time-lapse experiments can be problematic with fluorescent microscopy not just due to interference with the cells' mechanics. Application of the dyes and light can permanently damage the cells due to a phototoxic effect [47]. Commonly used dyes which bind to DNA, such as Hoechst 33342, have been proven to cause apoptosis when used in computerized time-lapse microscopy [48]. Although strategies exist to reduce phototoxity, such as using fast-switching LED lamps, the acquisition of this non-standard equipment can place further requirements and complications on an already laborious and expensive experimental procedure [47, 49]. Because of this, there is interest in using cheaper, quicker, less damaging alternatives such as brightfield to perform high-throughput screening and image-based profiling.

2.2.2 Brightfield microscopy

Brightfield microscopy is one of the most straightforward and inexpensive techniques to image samples with a microscope. It is a form of optical microscopy, where transmitted light illuminates the samples and structures, causing blocked light to be absorbed which creates a dark image on a bright background. In contrast to fluorescent imaging, minimal preparation is required to acquire brightfield images as no fluorescent dyes need to be applied. Due to its simplicity, brightfield microscopy is often performed simultaneously to fluorescent microscopy as part of a multispectral approach [50]. Typically brightfield images are acquired across multiple z-planes - examples shown in Fig. 2.3.

Even though brightfield images are noisy (Fig. 2.3), it is still possible to use brightfield to visualise and segment cell structures [51, 52], observe phenotypic information from cells, and perform MOA prediction for different drugs [53]. While brightfield imaging overcomes many drawbacks of fluorescent labelling, it lacks the specificity and clear separation of the structures of interest. The major drawback of brightfield imaging is the low contrast, which can make it difficult to detect internal cell structures.

Due to the success of fluorescent staining, it is likely brightfield has been underutilised in screening and profiling. Studies have explored the potential to analyse cells without disrupting their physiology by using brightfield images to perform tasks such as accurately detecting the inhibition of DNA-to-RNA transcription [54]. So called "label-free" [35] approaches have shown promise in augmenting the information available in brightfield when trained with fluorescent signals as a domain transfer problem [55–57], or simply by using brightfield as the input to perform classification tasks [53, 58]. The limit of brightfield's potential is still unknown, and very few studies exist which attempt to learn phenotypic representations from brightfield input. If

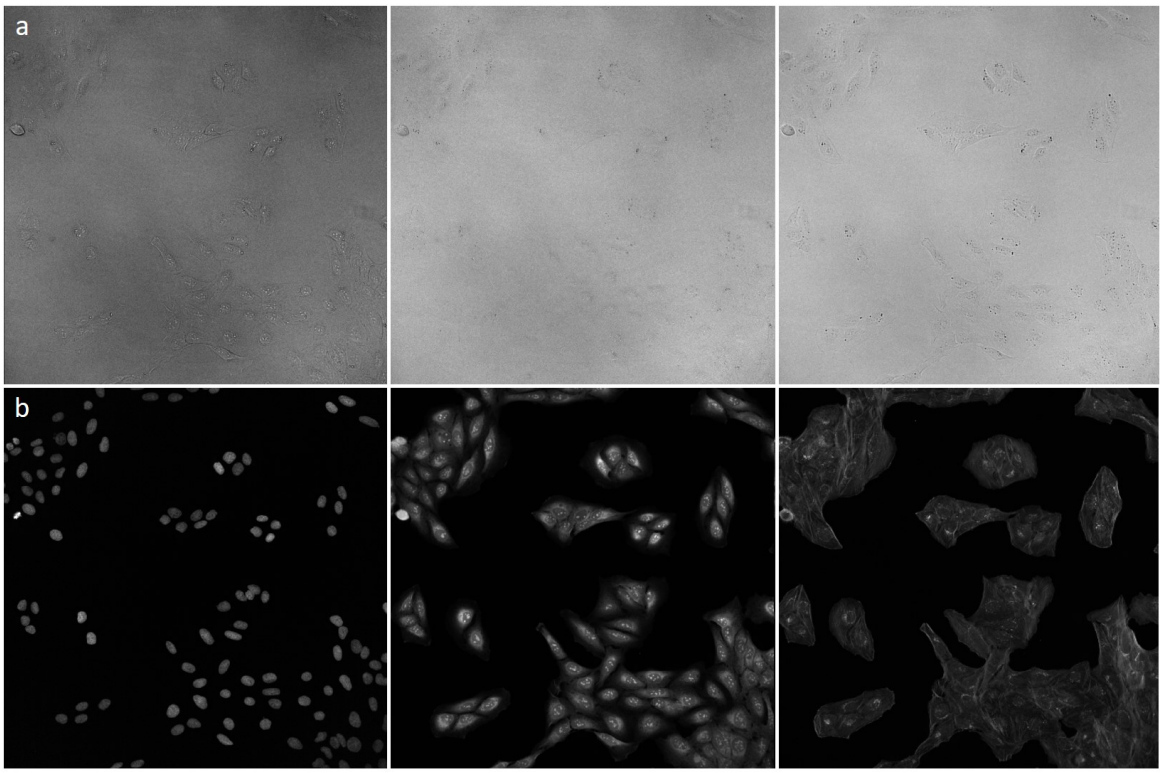


Fig. 2.3 A typical example of paired brightfield and fluorescent images from the publicly available JUMP-CP dataset [33]. Row **(a)** shows three brightfield z-planes. Row **(b)** shows three fluorescent channels of the same cells: (i) nucleus, (ii) endoplasmic reticulum, (iii) actin, Golgi, plasma membrane. Each image is a full a field of view of a single well from one of the plates in the dataset.

robust label-free methods existed which could challenge the quality and interpretability of feature profiles learned from fluorescent images, it would have a great cost- and time-saving impact on drug discovery, as well as opening up potential novel label-free applications. We present studies investigating such methods in Chapters 4 and 6.

2.3 Cell Painting

Cell Painting [10] is a high-content image-based assay designed to reveal rich cellular morphology. It can be applied in drug discovery to predict bioactivity [59], assess toxicity [60] and understand mechanisms of action of chemical and genetic perturbations [61, 62]. It is an assay designed to be used for image-based profiling [9], with standardized and regularly updated pipelines to guide both the generation of images and the

computational techniques to extract thousands of morphological features from the images [63].

In Cell Painting, cell phenotypes are captured with six generic fluorescent dyes imaged across five channels: nucleus (DNA), endoplasmic reticulum (ER), nucleoli, cytoplasmic RNA (RNA), actin, Golgi, plasma membrane (AGP) and mitochondria (Mito) [10]. In total there are eight cell and organelle components imaged across the five channels (Fig. 2.4). High-resolution images are captured with a digital camera, generally 1000×1000 or 2000×2000 pixels in size per field of view. A typical field size of 13.0mm by 13.0mm results in a pixel size of $6.5 - 13.0$µm [28]. Often three brightfield z positions - one equal to the lowest fluorescence position, and one 5µm above and below that - are captured alongside the five fluorescent channels (Fig. 2.3).

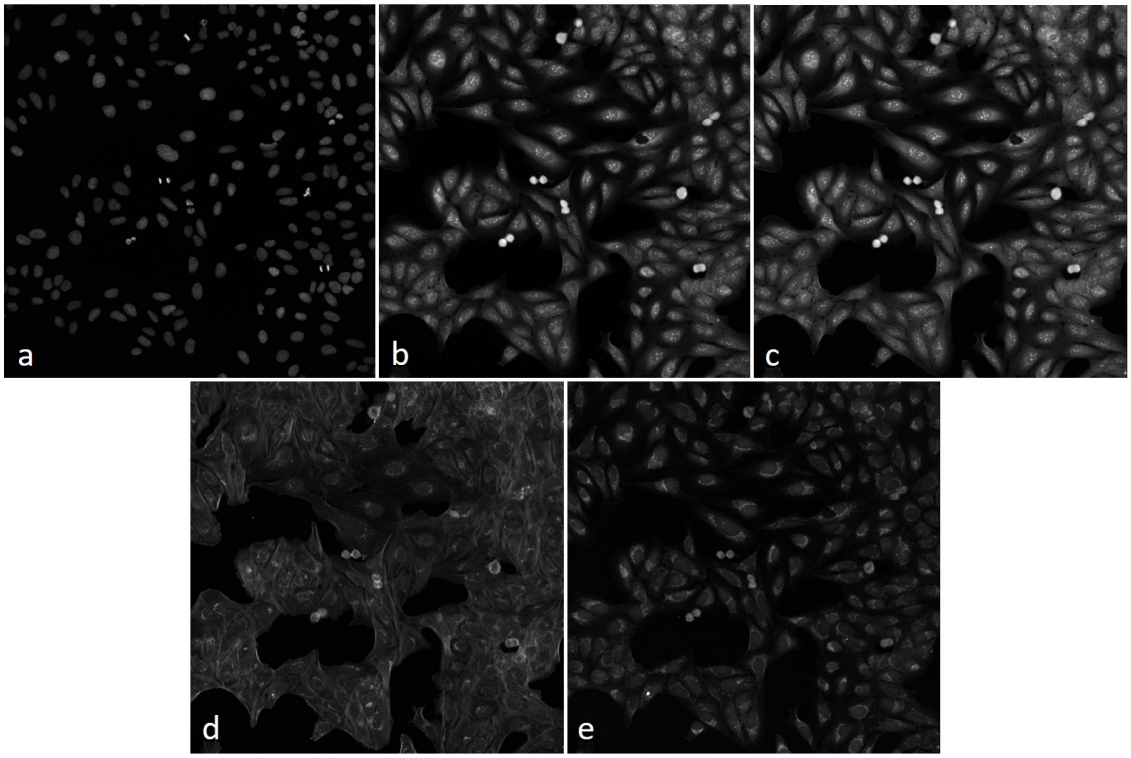


Fig. 2.4 A typical example of the five Cell Painting channels from the publicly available JUMP-Target-2 dataset [33]. **(a)** nucleus (DNA), **(b)** endoplasmic reticulum (ER), **(c)** nucleoli, cytoplasmic RNA (RNA), **(d)** actin, Golgi, plasma membrane (AGP), **(e)** mitochondria (Mito). Each image is a full a field of view of a single well from one of the plates in the dataset.

Despite being a relative newcomer to the field, Cell Painting is now considered the most popular assay in image-based profiling (as of late 2022) [63]. It has been used in a variety of studies, including to identify COVID-19 treatments [64], profile mutations

in lung cancer [65], and to evaluate the toxicity of environmental chemicals [66]. Cell Painting is used extensively by pharmaceutical companies, for example Recursion who have implemented Cell Painting and machine learning to support clinical stage pipelines from as early as 2019 [4]. The Cell Painting consortium JUMP-CP [34] was established in the belief that Cell Painting will "transform drug discovery by relieving a major bottleneck in the pharma pipeline: determining the mechanism of action of potential therapeutics prior to introduction into patient". JUMP-CP has many partners and collaborators including the Broad Institute of MIT and Harvard, AstraZeneca, Bayer, Janssen, Pfizer and Google Research, just to name a few.

With such backing, the potential of Cell Painting is clear, however it has taken time to be adopted. There are many established phenotypic screens in drug discovery, and initially standardized pipelines were repurposed instead of replaced [4]. Additionally, there are some downsides to the approach. Cell Painting, particularly at a large scale, is expensive due to the time, labour, equipment and reagent costs. Additionally there are many challenges regarding experimental site and batch variation which can be a problem in image-based profiling. There are many alternative, cost and time effective high-throughput techniques such as nuclear magnetic resonance [67] and mass spectrometry [68]. The compromise with the highest throughput methods is quality [69], and Cell Painting has so much interest in both academia and industry for its performance in MOA prediction while still being able to efficiently screen a large number of cells.

Whether Cell Painting will succeed is to be seen, but new initiatives to scale-up Cell Painting by releasing large public datasets such as the JUMP-CP dataset [33] will make it an industry-standard methodology in data-driven drug discovery for the foreseeable future.

2.4 Image-based profiling

Image-based profiling [9] is the process of extracting features from high-content images of cells. The aim is to extract "unbiased representations that capture morphological cell states" [70]. These features are used to build profiles which can assess bioactivity and MOA of the cells in response to treatments with chemical compounds and/or genetic perturbations. In drug discovery these phenotypic profiles are used to compare new treatments with known ones, and downstream applications include routine lead compound identification, drug target screening through CRISPR technology and toxicity assessment [60]. Cell Painting is an example of an image-based profiling assay [10].

In this section we focus on the computational component of image-based profiling. Image-based profiling can be used for (i) making predictions or identifying drugs or drug targets with specific features, or (ii) to profile perturbations more globally by reflecting the biology of the system. The goal of the latter is to extract multivariate features to uncover meaningful relationships, which is the domain of computer vision algorithms trained on large datasets [71]. The main challenge is to find techniques that best capture the biological information in images which can be used to perform downstream tasks. Traditionally, this is achieved with classical feature selection software such as CellProfiler [42, 43], however recent advances in machine learning methods, computing power and big-data approaches to high-content imaging has turned the field to deep learning methods [9].

Most image-based profiling methods are based on single cell segmentation [72, 73]. In fact, image-based profiling was virtually defined by single-cell techniques until deep learning enabled multi-scale approaches using the full field of view [74, 75]. While single-cell methods have yielded excellent results and have been the go-to-methods for both classical and machine learning studies [72, 73, 76–78], they are not without drawbacks. When applied to large datasets, single-cell segmentation can lead to high computational demands in pre-processing images with algorithms [43]. Additionally, working with single-cells requires feature aggregation to the population level and may necessitate further computational requirements for corrections due to cellular heterogeneity [79]. These are issues worth bearing in mind as the field drives towards larger datasets [33, 80].

2.4.1 Classical features in image-based profiling

Classical, or hand-crafted, features are the method of choice for many studies in both academia and pre-clinical pipelines [81–83]. CellProfiler is [42, 43] a popular open-source software for image analysis, and the original paper [42] is one of the most cited papers in the field of cell imaging. CellProfiler papers are cited over 1000 times per year [43], and the rate of citation isn't slowing. Even with the rise of machine learning, classical methods remain universal and necessary, not least to benchmark novel machine learning studies.

CellProfiler can be used to perform illumination correction, segment nuclei, cells and cytoplasm, and extract morphological features from each of the channels. Single cell measurements of fluorescence intensity, texture, granularity, density, location and various other features can be quantified as feature vectors. CellProfiler employs pixel-

based correlation and thresholding algorithms, alongside in-built machine learning for specific tasks. Pipelines from multiple studies are shared in public repositories [84].

For now, hand-crafted features have a number of advantages over machine learning features. Interpretability [85] and reproducibility [86] are actively researched challenges in deep learning, and specifics of these will vary significantly from model to model. Deep learning is a rapidly advancing field, and by the time a method is tested on a dataset, an abundance of new methods claiming superior performance have often been released. In this climate, it is hard to challenge well established and easily accessible software packages like CellProfiler (or commercial counterparts) to be accepted as the universally trusted benchmark in image-based profiling.

2.4.2 Machine learning in image-based profiling

However, there are several arguments for using deep learning methods instead of hand-crafted features in phenotypic profiling. Classical features may not capture all the variations in the cell phenotype. It is important for phenotypic representations to be able to detect and express subtle changes in cell morphology and treatment effect, and there is mounting evidence that deep learning methods may be more effective in achieving this than classical methods [9, 72, 73, 77, 80, 87, 88]. Early results in phenotypic profiling have been described as 'disappointing' [9], and deep learning advancements have renewed phenotypic drug discovery. Given the rate of advancement of deep learning (particularly computer vision) does not appear to be slowing, it is expected that the power and expressivity of these models will only improve. In the future it is possible deep learning could replace classical image processing altogether.

Classical features are agnostic to the size of the dataset, which can be an advantage in smaller studies (no overfitting issues). Deep learning, on the other hand, becomes more capable with access to larger training datasets [89]. Even though a software package such as CellProfiler may be more accessible to biologists than complex deep learning models, it still requires manual parameter adjustment for different experimental setups. A reproducible and high-throughput pipeline for large compound library screening should aim to be as automated as possible, and machine learning offers this automation.

Some of the earliest machine learning models were segmentation based, and set out to solve the problem of manual parameter adjustment in hand-crafted approaches [90]. More recently, deep learning has permeated into virtually all aspects of image-based profiling. Deep learning in cell image analysis is a broad, vibrant and rapidly advancing space [91].

Examples of prevalent deep learning methods in image-based profiling, roughly in chronological order, are transfer learning (Section 3.2.3) with Inception [87] and Deep Metric [73] Convolutional Neural Networks (CNNs), supervised Multiscale-CNN [75], weakly-supervised [77] representation learning (Section 3.2), and unsupervised, reconstruction-based methods [92, 78]. The examples in this paragraph are from studies which all use the same dataset, BBBC021 [38, 39], and are presented simply as a small window into the multitude of deep learning approaches in image-based profiling [9].

Image-based profiling is, to a large extent, guided by deep learning research and advances in parallel fields such as medical imaging. Currently popular are unsupervised methods such as self-supervised learning [93] using a contrastive loss [94]. Self-supervised algorithms such as SwAV [95] and DINO [96] give consistently richer embeddings for downstream tasks compared to pretrained supervised baseline models in medical imaging [97]. Self-supervised models have performed very well learning feature representations from single cell microscopy images [72, 88], and recently on the image-level with a multi-scale approach [98]. We explore some of these methods in Chapter 3.

Finally, there are a number of unique applications of deep learning to high-content microscopy data. Several studies have examined the use of deep learning for cross-modality prediction, specifically for reconstructing fluorescent images from transmitted light (such as brightfield) images [35, 55, 56]. Deep Generative methods (Section 3.3) are becoming increasingly powerful, and there is some evidence that features from these models can be used to characterize cell morphology [92, 99]. Generative models have the advantage of outputting images, allowing for visual interpretation of model performance. This property alone could make deep learning models more palatable for biologists, yet the potential use of generative models in downstream, drug discovery applications such as MOA and target prediction is yet to be fully explored.

We revisit this topic, provide more in depth discussion, and present in detail some of the machine learning methods used in image-based profiling in Chapter 3.

2.4.3 HCS metadata in image-based profiling

In addition to high-content images, there are a number of extra pieces of information naturally acquired through the experimental screening process [100] (Fig. 2.1.a-b). This metadata is free and meaningful information which can be used to further guide the learning of feature representations in image-based profiling.

Examples of free information which is always available alongside the images as paired labels include: the compound/concentration pair (treatment), the experimental

batch or plate, and whether or not the image is from the known control perturbation group (such as dimethyl sulfoxide (DMSO)). Furthermore, as the compound that cells in each image were treated with is always known, then this generally means chemical structure and gene expression of that compound is either free or inexpensive to obtain [101].

No unified way to treat this information exists, and until very recently it has been drastically under-utilised in image-based profiling. In the 2021 review *Image-based profiling for drug discovery: due for a machine-learning upgrade?* Chandrasekaran *et al.* described how "the flexible architecture of neural networks enables information to flow in from alternative data sources and formats, as input, or as side information" [9], yet the authors are unable to cite any relevant studies, so propose that as a future direction. Even in medical imaging, there are just a limited number examples of using free *side* or *weak label* information in computer vision network architectures.

Deep learning has been popular for at least a decade and this information has always been readily available, hence it is surprising how under-utilised metadata is in image-based profiling. One reason may be that, as previously mentioned, medical and biological imaging models are strongly guided by advances in computer vision. Academic machine learning communities focus heavily on large public image databases such as ImageNet [102], and these datasets are much more limited in both quantity and type of useful labels. So as medical and biological fields follow machine learning, they do so employing models designed for slightly different tasks. When designing computer vision architectures there is often an emphasis to move away from labels as they can be expensive and not always available.

However this is not always the case, including for problems in HCS, where there is an abundance of freely available metadata. We present a summary of this metadata and studies which have attempted to use it to enhance image-based, machine learning profiling models in Table 2.1. Methods include training a classification network to predict treatment (the *weak* label) as an auxiliary task, then extracting a latent embedding to be used as a feature profile [77, 80], as well as more hand-crafted and multi-network approaches where embeddings are typically concatenated in latent space [101, 103]. Batch correction is also a large issue in image-based profiling, and ways to incorporate batch labels into training neural networks have appeared to attempt to address this [103, 104]. This type of work is important as batch effects are strong in HCS, and CNNs in particular can easily fit to unwanted batch signals in the data rather than morphological information. Generally this is addressed with classical, post-processing feature correction methods such as Typical Variation Normalization

(TVN) [73] or sphering [80]. Another way to overcome the batch effect is by training models across a diverse range of microscopy data - either from multiple datasets [80] or with a large number of plates and replicates, which is the aim of new big data initiatives such as JUMP-Target [33].

There is evidence that chemical structure information is useful for bioactivity prediction [105] and with a combined machine learning approach Moshkov *et al.* [101] showed that chemical structure, high-content images and gene-expression profiles are complimentary for predicting compound activity. Other studies have similarly demonstrated the complimentary nature of Cell Painting and gene expression data in profiling assays [106]. Generative approaches such as CP2Image [99] propose training a network to reconstruct images from their CellProfiler features, to then extract the latent space to be used in profiling tasks. Auotencoder [107] and generative adversarial network (GAN) [92] based approaches have attempted to use latent spaces as feature profiles, as well as a combined approach with compound SMILES (Simplified Molecular Input Line Entry System) [108].

Type	Metadata	Availability	ML Method	Other Use
Experimental variables	Compound/pert	Always	WSL [77, 80]	Feature aggregation
	Concentration	Always	WSL [77, 80]	Feature aggregation
	Control (DMSO)	Always	N/A	Batch correction
	Brightfield	Usually	CNN [53]	Label-free prediction
Experimental conditions	Batch/plate	Always	BEN [104], TEAMs [103]	Batch correction
	Well/field	Always	N/A	Feature aggregation
Additional knowledge	Chemical structure	Always	CS+MO [101]	Benchmarking
	CellProfiler features	Usually	CP2Image [99]	Benchmarking
	Gene Expression	Sometimes	GE+MO [101]	Benchmarking
	MOA/target	Sometimes	Supervised Learning	Evaluation

Table 2.1 Our summary of typically available metadata in HCS with known attempts to use the data in the network of their models for image-based profiling tasks. We exclude referencing studies where MOA/target information is used to train supervised models.

In this section we have attempted to provide an exhaustive overview of metadata guided machine learning models in image-based profiling. Many of the publications we reference are very recent studies from 2022, and it is likely many future studies will incorporate metadata. It is hard to see why one wouldn't use free metadata when there is evidence that it is useful and informative. However, these models often appear overly complicated, demonstrate only a small improvement in very specific cases and may reduce model generalisability. They can be very difficult to implement, and computer

vision is not fully equipped to handle images and extensive metadata. We are clearly a long way off a unified model which incorporates multi modal information in an end-to-end way. Perhaps new frameworks such as generative methods and self-supervised learning will provide natural ways to use extra information in guiding the networks.

There is strong early evidence that these approaches improve performance under specific parameters (e.g boosting a network's performance on a task), however it doesn't mean they are they best choice models for image-based profiling. Generally the models use the labels as extra inputs, required for both training and evaluation - however that assumes some consistency in labels across datasets, which is not always the case - in other words these constructions are dataset specific. Self-supervised methods have achieved robust and state-of-the art performance without using **any** labels [72, 88, 98], which presents somewhat of a crossroads for image-based profiling - why do we need to incorporate extra labels when unsupervised methods are performing so well? In Chapter 5 we explore combining self-supervised frameworks with weak label information for the first time as a new solution to learning feature representations guided by informative metadata.

2.4.4 Challenges

We have mentioned many of the challenges of image-based profiling, and posed some unanswered questions. We summarize some of the main challenges facing the field which motivate the research in this thesis:

- Can we replace fluorescent staining (e.g. with brightfield)?
- What is the best way to learn biologically-relevant and expressive representations?
- Can we make machine learning models that are reproducible, generalisable and interpretable enough to be used in clinical pipelines?
- Can image-level approaches compete with single-cell methods?
- Overcoming the batch effect and adjusting for technical noise.
- Incorporating metadata into computer vision architectures.
- Are labels needed at all? Are self-supervised methods superior?

2.5 The rest of the drug discovery pipeline

The drug discovery pipeline is a multi-step process which is summarised in Fig. 2.5. Typically it takes 10-15 years for a drug to be developed from initial screening to approval for manufacture (e.g. by the Food and Drug Administration (FDA)). It takes around 3-6 years of screening, identifying hits, and validating and optimising leads prior to pre-clinical development, where the candidate drug is tested on animals. After pre-clinical testing there can be up to 5 more years of clinical trials on humans to be approved as safe and efficacious for manufacture and distribution. Based upon data from 2009 to 2018, it is estimated to cost between $300 million and $2.8 billion to develop a new drug [1].

As a very slow and expensive process, there is a great benefit to saving time at any point in the pipeline. 10-15 years is the time taken for a successful drug, but it is important to consider that 90% of drugs fail in clinical trials [2]. This places a greater burden on selecting the right targets in the screening and lead generation phases. One of the most significant bottlenecks in drug discovery is the identification of the correct molecular targets or MOA of a compound. Image-based profiling could be one of the methods used to overcome this bottleneck [9, 11, 71].

There are many ways machine learning is being used to attempt to accelerate drug discovery beyond image-based profiling. Machine learning has been successfully applied in drug discovery for target identification, compound design, biomarker prediction, clinical safety and efficacy prediction as well as to analyse clinical data [3, 110]. Some promising areas of research include using knowledge graphs [111], active learning [112] and generative modelling [113].

Machine learning can be applied at virtually all stages of a long and complex pipeline, including in clinical trials [3]. It is important to note that many of these attempts are yet to be successful [110], and in some areas machine learning is being used speculatively, or exists only in academic studies, rather than in clinical pipelines.

Image-based profiling is still a relative newcomer in drug discovery and only one part of a vast landscape. The impact it will have on drug discovery will be known only in the future. In this thesis we introduce novel methodologies with studies designed to advance image-based profiling for drug discovery.

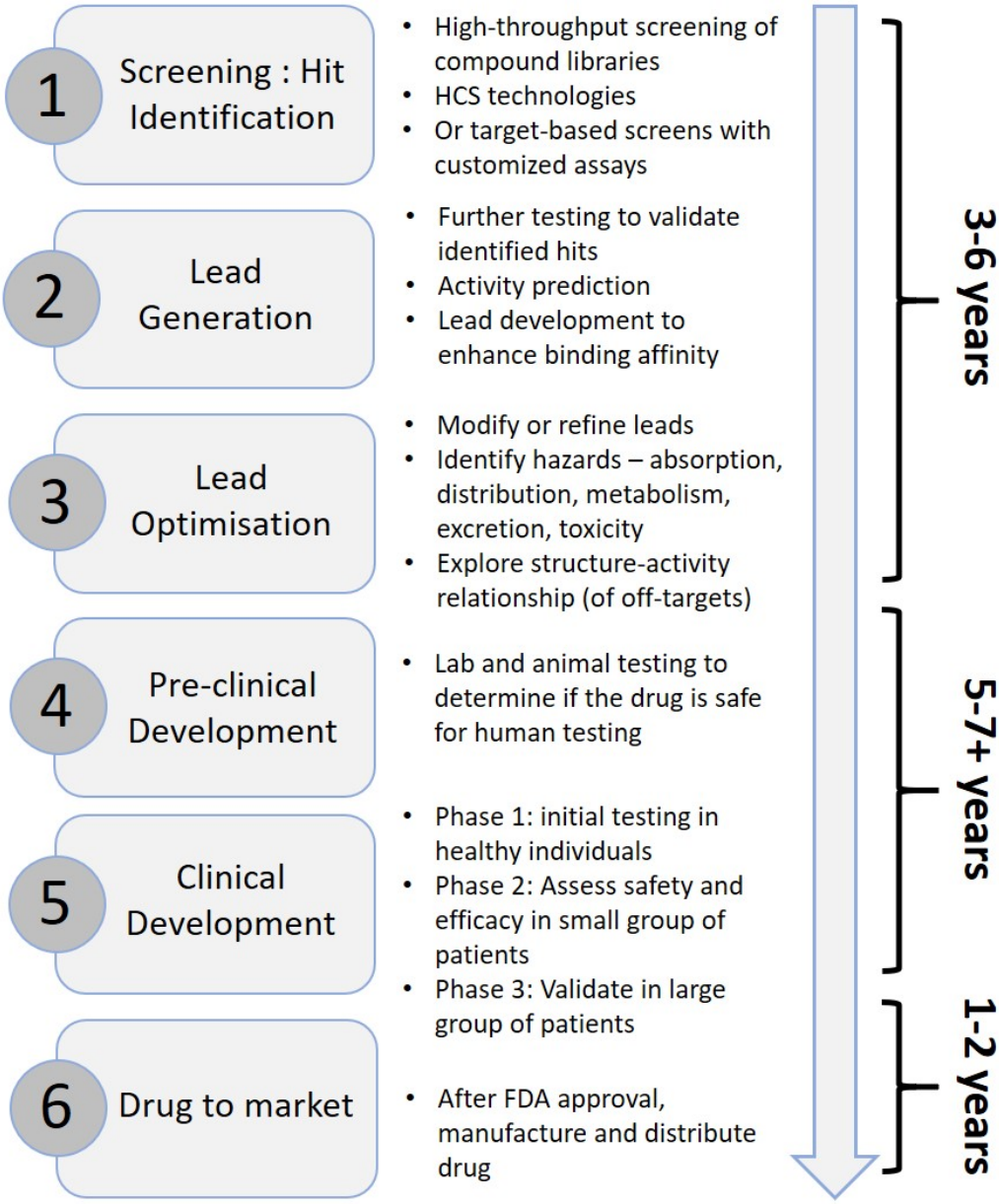


Fig. 2.5 Adapted from Jenkinson *et al.* [109]. Machine learning and image-based profiling can have the greatest impact in stages 1-3 of this pipeline.

2.6 Machine learning

We will now shift our focus from drug discovery to machine learning. The aim of the final two sections of this chapter is to cover the machine learning and deep learning fundamentals, allowing us to begin Chapter 3 without assuming the necessary principles.

Machine learning is a branch of the field of artificial intelligence (AI). Machine learning algorithms are designed to make predictions or decisions about unseen data. These predictions are made based on data the algorithm has previously seen. Machine learning models aim to capture information from training data in a way which allows the model to inform future decisions and/or predictions. Machine learning is at the forefront of many scientific disciplines and there are very few spaces where machine learning has not yet been used in some capacity. If there is data, there is machine learning.

Machine learning is commonly split into three main branches: supervised learning, unsupervised learning, and reinforcement learning. Typical supervised tasks are classification and regression. Unsupervised learning is generally used to cluster and classify unlabelled data. Reinforcement learning is a branch of machine learning to guide informed decision making by rewarding or punishing behaviours. We discuss the machine learning methods used in this thesis in more detail in Chapter 3.

2.6.1 Supervised learning

Supervised machine learning [114] uses labelled training data for training. For the training data set (x_i, y_i), where y_i is the label of the point x_i for $i = 1, \ldots, N$, the aim is to learn a function f such that:

$$f(x) \approx y \tag{2.1}$$

In classification tasks, y_i is a set of discrete labels, for example in a (1/0) or (True/False) binary classification task. If the values taken by y_i are continuous, then the task is a regression task. Linear regression is a statistical technique which could be considered a simple machine learning algorithm. Here the function f is parametrised as:

$$f_\theta(x) = \theta_0 + \theta_1 x \tag{2.2}$$

where θ_0 and θ_1 are unknown constants to be estimated using the dataset. This is achieved thorough minimising a loss function $\mathcal{L}$ such as least squared error:

$$\mathcal{L} = \frac{1}{N} \sum_{i}^{N} \left| f_\theta(x_i) - y_i \right|^2 \tag{2.3}$$

for N data points in the sample data. Minimising $\mathcal{L}$ will result in the best fit function,

$$\min_{\theta_1,\theta_2} \mathcal{L}(\theta_1, \theta_2) = \frac{1}{N} \sum_{i}^{N} \left| \theta_0 + \theta_1 x_i - y_i \right|^2 \tag{2.4}$$

and values of θ can be found by solving the resulting quadratic equations, which would generally be calculated computationally.

Many other supervised algorithms such as K-nearest neighbour (K-NN), decision trees, and random forest models are used for various tasks in machine learning. However, deep learning is one of the most important techniques in computer vision (and hence image-based profiling) for its ability to capture complex relationships in high-dimensional data.

2.7 Deep learning

Almost all machine learning studies we have mentioned in this chapter use deep learning [115]. Deep learning refers to the use of deep (multilayer) neural network architectures such as the convolutional neural network (Section 3.1.1) or the transformer (Section 3.1.4) to learn and output representations of the input data. In this section we introduce the fundamentals of deep learning.

2.7.1 Neural networks

In computer vision, neural network [116] based models outperform all other machine learning approaches on almost every task, given sufficient amounts of training data. Inspired by neural networks in the brain, the artificial neural network (ANN) is used for classification, prediction, clustering, segmentation, pattern recognition and learning feature representations across many disciplines and datasets [117]. ANNs consist of multiple layers of interconnected neurons - nodes which take input signals, perform computational functions then pass their output signal to the next layer(s) of neurons. ANNs have become a mainstay of machine learning as, when compared to other models, they have scaled exceptionally well with the big data revolution [118] and the exponential growth of computational power [119].

The simplest neural network is the perceptron [120] which is an algorithm to map an input vector $\boldsymbol{x}$ to an output value $f(\boldsymbol{x})$:

$$f(\boldsymbol{x}) = \boldsymbol{x} \cdot \boldsymbol{\theta} + b \tag{2.5}$$

where $\boldsymbol{\theta}$ is the weight vector, and b is a constant. Note the similarities to linear regression Eq. 2.2. This could be used in a classification task, for example, where $\boldsymbol{x}$ is the input feature, $\boldsymbol{\theta}$ is to be learned and f predicts a class value of $\boldsymbol{x}$. The perceptron consists of only a single layer of neurons, but in an ANN there are many layers. For a vector $\boldsymbol{x}^i$ which is the input to the i^{th} layer of the network, the $(i+1)^{\text{th}}$ layer is calculated by:

$$x_k^{i+1} = \sigma\left(\sum_j \theta_{kj}^i x_j^i + b_k^i\right) \tag{2.6}$$

where σ is a nonlinear function called the activation function. This is known as a dense layer. Multiple dense layers can form the basis of a *fully connected* deep neural network, where all the neurons between neighbouring layers are connected to each other. One notable change from Eqn. 2.5 is the introduction of the activation function σ. The activation function is used to force non-linearity, allowing the network to capture complex relationships between the input space to the output space. There are many choices for σ, with some well known activations being the sigmoid (or logistic) function:

$$\sigma_{\text{sigmoid}}(x) = \frac{1}{1+e^{-x}}, \tag{2.7}$$

the rectified linear unit (ReLU) [121] function:

$$\sigma_{\text{ReLU}}(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{else} \end{cases} \tag{2.8}$$

and a commly used adaption to ReLu such as LeakyReLU [122]:

$$\sigma_{\text{LReLU}}(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha x & \text{else} \end{cases} \tag{2.9}$$

for a small constant $\alpha < 1$.

In addition to fully connected layers, there are a number of other types of layers including convolutional layers, recurrent layers, pooling layers and attention layers. Convolutional neural networks [123] and transformer networks, which use attention layers [124], are important to many tasks in computer vision and beyond. We explore the specifics of using these layers in computer vision network architectures in Section 3.1.

2.7.2 Training neural networks

In order to train an ANN, two things are required: a loss (or objective) function $\mathcal{L}$ to learn the weights θ, and a dataset of examples to guide the learning through comparisons between the network output and the real data or labels. In the supervised case, this would be the inputs x_i with paired labels y_i.

The difference between the network output and the real data is quantified by a loss function, $\mathcal{L}$ - a cost function used to measure the error of the network's prediction. For a network f_θ parameterised by the trainable weights θ, a supervised training regime will seek to minimise the loss function as follows:

$$\arg\min_{\theta} \sum_{i}^{N} \mathcal{L}(f_\theta(x_i), y_i) \tag{2.10}$$

A simple example is the mean squared error loss, $\mathcal{L}_{\text{L2}}$:

$$\mathcal{L}_{\text{L2}}(f_\theta(x_i), y_i) = \frac{1}{N} \sum_{i}^{N} \left| f_\theta(x_i) - y_i \right|^2 \tag{2.11}$$

Additional terms can be added to loss functions, referred to as *regulariser* terms. These are usually incorporated to increase generalisability and prevent *overfitting*, where the training data is too closely learned by the network in a way which is sub-optimal for perfomance on unseen data.

Fundamental to training a network is the backpropagation algorithm [125], which is used to calculate the gradient of the loss function with respect to the weights. If the input $\rightarrow$ output direction of a network is considered forwards, then the backpropagation computation occurs layer by layer in the opposite direction - from the last layer to the first (backwards).

The weights are usually randomly initialized before training. There are a number of choices for the optimisation algorithm used to update the weights, but a simple choice is (stochastic) gradient descent [126]:

$$\theta_{n+1} = \theta_n - \lambda \nabla_\theta \mathcal{L}(f_\theta(x_{i:i+N}, y_{i:i+N})) \tag{2.12}$$

for the n^{th} update of the weights and a batch size of N samples from the training set. These updates of the weights in the network are repeated until convergence. Stochastic gradient descent refers to the use of single points or smaller batches of data to compute the gradients (as opposed to gradient descent on the entire data set). The parameter λ

can be considered the *learning rate*, which is an example of a *hyperparameter* of the model. There are many hyperparameters which often need to be optimised to best suit the model and the dataset. Another commonly used optimisation algorithm is Adam [127] which incorporates exponential moving averages of the gradient into each weight update, resulting in greater stability and faster convergence.

There are many powerful, open-source tools for implementating deep learning such as PyTorch [128] and Tensorflow [129], which include many necessary mathematical operations wrapped in convenient functions. Deep learning models are often trained on graphics processing units (GPUs) or specialist machine learning tensor processing units (TPUs). Memory constraints of GPUs are often the limiting factor in the depth (and performance) of the network. Some of the most powerful models are trained on giant clusters of GPU/TPUs, which are extremely expensive to build and run, but allow companies such as OpenAI to train models with hundreds of billions of parameters and datasets comparable to the size of the entire internet.

In the next chapter, we describe specific architectures and discuss their complexities in more detail.

Chapter 3

Methods

In this Chapter we detail the mathematical and machine learning methods relevant for this thesis. The aim of this chapter is to explain the concepts required for the experimental chapters (Chapters 4 - 6). We begin with some of the important network architectures used in computer vision, the convolutional neural network, the U-Net, and the vision transformer. Next, we expand the discussion from Section 2.4.2 and explore the machine learning techniques used to learn representations from high-content microscopy images. Finally, we describe deep generative models, specifically the generative adversarial network and the denoising diffusion probabilistic model, and discuss the metrics used to evaluate their performance.

3.1 Network architectures

3.1.1 CNNs: Convolutional neural networks

We have introduced fully-connected ANNs in Section 2.7.1, however the staple of modern computer vision has been the convolutional neural network (CNN) [123]. A convolution operation applies a kernel w of size $(2k+1) \times (2k+1)$ over the image (and successive hidden layers) x:

$$w \star x = \sum_{j,l} w_{j,l} \cdot x_{m+j,n+l}, \quad j,l = -k, -k+1, \ldots, k-1, k \tag{3.1}$$

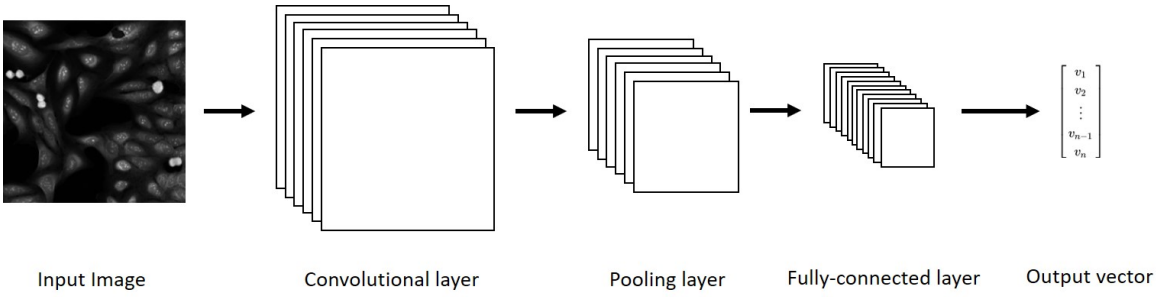


Fig. 3.1 An example of a simple CNN which could be used to perform a classification task. The latent embedding from the fully-connected layer could be extracted and used as a feature representation as in [77]. Adapted from O'Shea and Nash [130].

Hence, for an input image or layer x^i, the output of a convolutional layer (and input to the next layer x^{i+1}) would be given by:

$$x^{i+1} = \sigma(w^i \star x^i + b^i) = \sigma\left(\sum_{j,l\in\{-k,-k+1,\ldots,k-1,k\}} w^i_{j,l} \cdot x^i_{m+j,n+l} + b^i\right) \tag{3.2}$$

for an image of size $m \times n$, where σ is the activation function and b is the bias. The index k sets the size of the kernel, typically just 3×3 pixels ($k = 1$). A common choice of σ is the ReLU activation function which enforces non-linearity in the output (Section 2.7.1). *Padding* is added to the edge of the image (often just pixels of zero value) to ensure the layer outputs are of equal dimension. Without padding the convolution output would be of size $(m - 2k) \times (n - 2k)$.

As shown in Fig. 3.1, there are can be other layers in a CNN. Pooling layers follow convolutional layers to downsample the output. They reduce the dimension by calculating the average or maximum value of the group of pixels with a filter of fixed size. The filter then *strides* to the next group of pixels (stride size > 1 pixel). Fully-connected layers are generally added as the last layer(s) of the network. Fig. 3.1 shows a single greyscale input image, but for multiple channels (such as RBG images), an extra channel dimension is added to the system.

CNNs have led the way in computer vision as they are very good at automatic pattern and feature detection in all kinds of image data. They are efficient and have been extensively studied, with many well known and optimised architectures such as Inception [131] and ResNet [132].

Recently, the vision transformer (ViT) [133] has matched or improved upon CNN performance in a number of computer vision tasks. We explore the ViT later in this section.

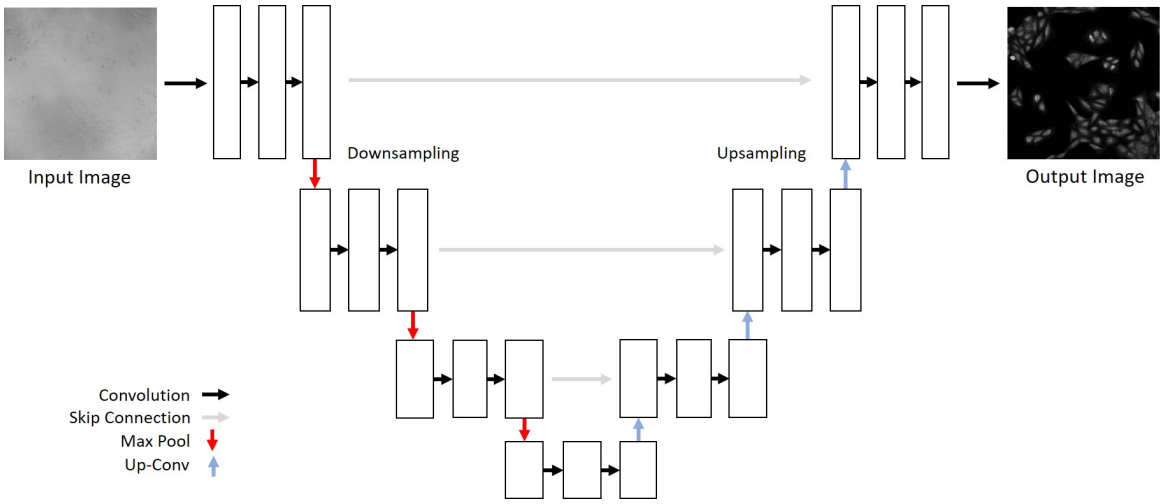


Fig. 3.2 An example of the U-Net architecture displaying the downsampling and upsampling branches of the network, as well as the skip connnections. Adapted from Ronneberger *et al.* [134]

3.1.2 U-Net

CNNs can also be used for image-to-image tasks such as segmentation, inpainting and denoising. One of the most important architectures, originally designed for segmentation, is the U-Net [134]. The U-Net is a fully-convolutional network which has been used to solve reconstruction problems in cellular [55], medical [135] and general imaging problems [136]. Despite being relatively old for a deep learning architecture, U-Net based models still achieve state-of-the-art results in segmentation tasks [137].

The U-Net architecture (shown in Fig. 3.2) features a contracting path that includes a series of convolutional layers and downsampling (max pooling) layers, as well as an expansive path that includes upsampling (up-conv) and concatenation layers. U-Nets allow for retention of spatial information while learning detailed feature relationships. This is achieved in the expansive path through feature concatenation with the corresponding features from the contracting path, via the skip connections. This preserves information which may otherwise have been lost without the skip connections.

3.1.3 Self-attention

Attention [138] is an interaction which can be introduced into a neural network to weight the importance of different regions of content in the input. It was first used for natural language processing (NLP) tasks and gained popularity with the transformer [124] - a network which can process whole sequences of data simultaneously. This is

achieved using attention to weight the relative importance of the sequence's components, and provide global positional context (for example words in sentences).

In a vision context, for an input image (or layer) x, the single-headed attention function is defined as:

$$\text{Attention}(q, k, v)_{ij} = \sum_{a,b \in \mathcal{N}_k(i,j)} \text{softmax}_{ab}\left(q_{ij}^{\intercal} k_{ab}\right) v_{ab} \tag{3.3}$$

where $\mathcal{N}_k(i, j)$ is a region of pixels in positions ab in the box of dimension k centred around the pixel x_{ij} (similar to a convolutional kernel). The functions q, k and v are linear transformations of the input, with weights to be learned:

- Query: $q_{ij} = W_Q x_{ij}$ - feature vector of the element of interest (e.g. word or pixel)
- Key: $k_{ab} = W_K x_{ab}$ - feature vector of other elements
- Value: $v_{ab} = W_V x_{ab}$ - value for each element used in weighted sum

The attention process maps the query vector q and a set of key-value pairs (k, v) to an output vector. The output is a weighted sum of the values, with the weight being determined by the compatibility between the query and the key - normalised by the softmax function:

$$\text{softmax}_{ab}(x_{ab}) = \frac{\exp(x_{ab})}{\sum_{a,b} \exp(x_{ab})} \tag{3.4}$$

More commonly used in transformers is the multi-headed attention function, which allows the model to jointly synthesise information from different feature spaces at different positions. The input is split and an attention function applied multiple times in parallel, each with its own q, k and v. The outputs of each of these heads are concatenated before projection.

Self-attention is attention where q, k and v are all calculated from the same input sequence. For each element in the input sequence, an attention layer computes the similarity of its query with the keys of all the other elements in the sequence. An averaged value vector is returned for each element, which indicates it relative importance in the task. The model learns to focus on the most relevant parts of the input.

In a transformer architecture multi-head self-attention layers are followed by feed-forward layers which take information from both the input data and the attention layers in order to learn capture relationships and perform the chosen tasks. Transformers have been adapted to computer vision tasks - the vision transformer (ViT), which we discuss in the next section. As well as in transformers, self-attention layers can also

be introduced into existing CNN architectures. Incorporating self-attention has been shown to improve performance of diffusion generative models with CNN backbones [139–141].

3.1.4 ViTs: Vision transformers

The vision transformer (ViT) [133] is one of the most exciting developments in computer vision, as it is capable of outperforming CNNs when trained with enough data, while being computationally very efficient.

The ViT first breaks the image into patches (8×8 or 16×16 pixels). Attention layers embed patches instead of every pixel as the computational cost is quadratic in the number of tokens. These visual tokens have a positional embedding as part of the input to the transformer encoder, which is a sequence of multi-attention heads and feed forward layers with skip connections - see Fig. 3.3. The ViT can be summarised in the following steps:

1. Split the input image into patches of size n. Flatten the patches.
2. Linearly project the flattened patches for the initial patch embeddings.
3. Add the [CLS] class token to the patch embeddings.
4. Sum the patch and positional embeddings and input to the transformer encoder.

The transformer encoder consists of normalisation (Norm) layers before each block to help improve training time and performance, multi-head self-attention layers to capture local and global dependencies, and finally multi-layer perceptron (MLP) layers for classification. There are skip connections after each block in the encoder to allow some information to flow linearly through the network (i.e. bypassing layers with non-linear activations). Patches are fed into the encoder then reconnected into a single feature embedding before being passed to a final MLP head for classification. This head can be used as a low-dimensional feature representation of the image e.g. for clustering tasks [96].

Typically a ViT will be pre-trained for a classification task on a large dataset and then fine-tuned on a smaller dataset for the desired application. There is evidence that ViTs may require extensive pre-training on large datasets to be competitive with CNNs [142], but once these model weights are known, it does make transfer learning (Section 3.2.3) more computationally efficient. This pre-training is also possible to do in a self-supervised fashion (Section 3.2.2) [96].

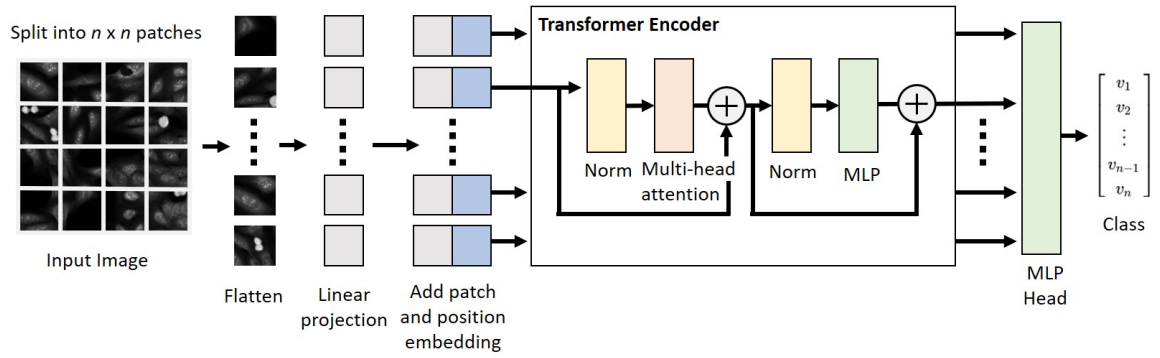


Fig. 3.3 The vision transformer architecture. Adapted from similar figures from Dosovitskiy *et al.* [133] and Vaswani *et al.* [124].

Compared to CNNs, ViTs are more resilient against various forms of image distortions, including adversarial attacks and permutations [143]. However, they may not have the inductive bias of CNNs, such as the ability to recognize translations and local patterns. Capturing global and long-range relationships comes at the cost of requiring more data for training. Additionally, ViT performance can depend heavily on augmentations, hyperparameters, optimisers and network depth.

While the performance of the ViT compared to the CNN is dependent on a number of parameters, many modern CNNs have adopted techniques from transformers to mimic the powerful global attention behaviour [144]. These hybrid architectures have performed particularly well in generative models [139–141].

Given the ViT can better capture global interactions across an image [145] it may be more suited to tasks in medical and biological imaging [146]. Conceptually this makes ViTs good candidates for high-content microscopy studies. We explore the potential of image-based profiling with a ViT in a self-supervised setting for the first time in Chapter 5.

Using attention layers has the further advantage of being able to visualise self-attention maps of the attention heads. This makes attention-based networks significantly more interpretable, which is very useful in a field where replacing classical feature extraction with black-box techniques in clinical pipelines has proven difficult. It may also help to reveal biological insights as it is possible to visually highlight which structures the network weights most heavily on. We present self-attention maps of high-content images in studies in Chapters 5 and 6.

3.2 Representation learning

In this section, we expand upon the discussion in Section 2.4.2 and summarise some of the most important strategies used to learn representations in image-based profiling.

"Representation learning is a set of methods that allows a machine to be fed with raw data and to automatically discover the representations needed for detection or classification." - Yann LeCun [115].

Representation learning is sometimes referred to as feature learning. The aim is to convert input data (such as images and metadata) into a reduced-dimension feature space which accurately and efficiently represent the data structure. It is clear from this definition that representation learning and image-based profiling are almost two ways of saying the same thing. Successfully profling high-content images involves learning an informative and useful feature space which can capture complex interactions and perform well on downstream prediction tasks.

There are a number of different approaches for learning feature representations from high-content images with machine learning, and the chosen method is usually task-specific. Typically, models are based around unsupervised clustering, with features extracted from the latent space of a trained network. One way to evaluate the quality of the representation is with a nearest-neighbour or distance metric to matching targets in the multi-dimensional feature space.

As we have discussed in Chapter 2, deep learning often outperforms classical approaches in downstream profiling tasks. One downside compared to handcrafted features is that machine learning often lacks both interpretability and reproducibility. Additionally, quantitative metrics can be cherry-picked to favour the study, and classification accuracy alone (especially on a single dataset) is not enough to make any model the best model to use in practice.

Representation learning can be particularly challenging with high-content microscopy images. Image-based profiling models need to be able to find features sensitive to changes in biology rather than noisy and irrelevant information (e.g. unwanted batch effects). This burden is partly on the dataset used for training, partly on pre- and post-proccessing, and partly on the network design. In image-based profiling, self-supervised methods have been popular and successful in the last couple of years [72, 98]. While generative models were initially unsuccessful at capturing feature representations [92], it is likely that they will see a resurgence in the near future due to improvements in diffusion and transformer based models.

3.2.1 Weakly supervised learning

Caicedo *et al.* [77] proposed the use of a weakly supervised framework to learn feature embeddings from single-cell cropped microscopy images. Weakly supervised learning is essentially transfer learning from an auxiliary task trained on the same dataset. The network is pre-trained to predict a freely available *weak* label. Their framework considered n single cells $X_i = \{x_k\} \forall k \in \{1, ..., n\}$, each treated with compound $Y_i \in Y$. They defined the mean profile of a compound Y_i:

$$\mu(X_i) = \frac{1}{n} \sum_{k=1}^{n} \boldsymbol{\phi}_\theta(x_k) \in \mathbb{R}^m \tag{3.5}$$

where $\boldsymbol{\phi}_\theta(x_k) \in \mathbb{R}^m$ is a mapping function paramaterised using a CNN. Two compounds, Y_i and Y_j, have an unknown relationship $Z_{i,j}$ that can be approximately measured as the similarity between their mean profiles:

$$Z_{i,j} = \rho\left(\mu(X_i), \mu(X_j)\right) \tag{3.6}$$

The network $\boldsymbol{\phi}_\theta(x_k)$ is trained as a multi-class classification function to predict Y from a single cell input as an *auxiliary task*, but the *main goal* is to uncover treatment relations $Z_{i,j}$. In one of their experiments, a CNN was trained on the BBBC021 dataset [38, 39], to predict the treatment as the weak label (*auxiliary task*). From a layer of the network, an embedding was extracted to represent a morphological profile, which was evaluated by predicting the MOA class as the *main goal*.

This method is important to the concepts in this thesis as it elegantly incorporates meaningful and free metadata into the network. Similar studies have also used metadata in pre-training classifiers to learn representations [147]. Even though Caicedo's study is from 2018, the weakly-supervised model is still being used a competitive profiling method in late 2022 [80]. It also has the potential to scale to multiple, large datasets (as long as there are overlapping classes between datasets).

Other uses of metadata to guide representation learning

We continue the discussion from Section 2.4.3. More complex machine learning models have attempted to incorporate multiple pieces of metadata (weak labels) for learning representations from cell images, notably using treatment alongside technical variation information to align features across different batches with a *mixture of experts* approach [103]. Older attempts to use metadata alongside images involved extracting classical

features from the image and joining those features with e.g. activity data to form an input to a deep neural network classifier [148]. Sometimes these methods can feel too hand-crafted and non-generalisable - there is a fine balance to strike when using metadata.

We propose that there is promising future work in designing methods to synthesise metadata with image data for learning meaningful representations. The goal of machine learning is typically to understand the distribution of data, and using metadata to aid in this understanding, rather than relying solely on potentially flawed images, may be important in high-content data. These images can be very similar, and the completely unguided or unsupervised approach risks fitting to the batch effect or unwanted background noise. Experimental batch variation signals are so strong that it may be easier for the model to fit to them - learning the path of least resistance. While the supervised method is probably not the optimal approach, framing the training objective as a supervised task with carefully chosen labels will help to learn meaningful differences, and may prevent fitting to unwanted technical variations.

Informing training with multiple sources of data is one aspect to feature learning. Given we don't fully understand how exactly, for example, a convolutional neural network learns features (which can be very unstable and tough to reproduce), it is ever more vital to be able to interpret the models. Generative models may be another way to do this, as they yield visual outputs as their objective. We detail deep generative models later in this chapter.

3.2.2 Self-supervised learning

Unsupervised learning is machine learning without annotations - training with unlabelled datasets. Self-supervised learning also uses no labels, however the key difference from unsupervised learning is that there is a specific task or objective to be optimized. Through design, the network is told what information to focus on in training. This information exists in the training data, and in computer vision tasks, training is often achieved with crops and augmentations [96]. Like with supervised learning, when (pre-)trained in a self-supervised manner, a network can then be applied to various tasks including classification, object detection, and segmentation.

Self-supervised methods have achieved impressive performance in medical imaging and image-based profiling [72, 95, 146]. Self-supervised models are capable of capturing feature representations with excellent clustering properties and robustness [149]. Contrastive learning [150] is a commonly used self-supervised approach, where the network is trained to pull representations of augmented versions of the same class - positive

examples - close together in latent space while pushing negative (different) examples further away from positive ones [151]. The idea behind using a contrastive loss is to train a model to learn clusters based on characteristics of the images (textures, shapes, edges).

Contrastive learning is a form of instance classification - where images are considered to be of certain classes in training. More recently, new self-supervised algorithms such as SwAV [95] and DINO [96] don't use instance instance classification at all. SWaV clusters similar images together in feature space with a CNN, then uses the clusters as pseudo-labels to train a linear classifier. It does not directly compare image features. Knowledge distillation involves the transfer of knowledge from a larger model (teacher) to a smaller one (student). The aim is to train the student model to mimic the teacher's prediction, which results in learning a high-quality representation of the data. We explain DINO in more detail in our study in Chapter 5.

3.2.3 Transfer learning

Transfer learning [152] involves utilising the *pre-trained* weights from one model with a new model for a different task and/or with different data. Pre-training is often done with a large dataset such as 10+ million image ImageNet [102].

One form of transfer learning is where the pre-trained weights are fixed, and the network is used outright on a different dataset for a different task. The second approach is to initialise the network with the pre-trained weights (instead of randomly initialising) and *fine-tune* by re-training on the new data. This could involve freezing many layers and fine-tuning just a few layers, however it may be appropriate to unfreeze all layers. In all cases, transfer learning refers to the recycling of learned data to generalise to a new application.

Transfer learning is useful as it can save computation resource by either eliminating or significantly reducing training time. Using larger and more general datasets can overcome overfitting and learn more general and generalisable features. In image-based profiling, transfer learning of convolutional neural networks has been used quite successfully [73, 87, 153]. Some unsupervised methods perform well without any fine-tuning [96] as enormous datasets such as ImageNet allow networks to learn very meaningful features which can transfer well microscopy images of cells. Pre-training can be also done in a self-supervised setting. These models can then be fine-tuned on a specific dataset for the downstream task, or have feature representations extracted and used as inputs to train a linear classifier for the downstream task [132].

Transfer learning using ImageNet weights has produced strong results in image-based profiling. It may be worrying that features from images of cats, dogs and aeroplanes are just as capable as models trained with real, biological data. Pretraining on large datasets can learn robust object and edge detection, segmentation and texture features. However, it can be very difficult to know if these models have learnt anything with biological meaning. Recently, large labelled microscopy image datasets of comparable size to the large natural image databases (millions of images) [154] have further increased the potential of transfer learning in image-based profiling of cells. Yet there is no strong evidence yet that pre-training with these large datasets of cell images outperforms natural image databases in transfer learning. This may suggest that the networks are not capable of capturing complex cellular biology, or are not being sufficiently guided in the training process.

3.3 Deep generative models

The aim of generative modelling is to learn an approximation to (or to learn how to sample from) the probability distribution of a dataset such that the model is able to generate unseen samples which could have been drawn from the training dataset. Generative modelling is a rapidly advancing domain which became extremely popular after Goodfellow *et al.* introduced the generative adversarial network (GAN) in 2014 [155]. Other generative models include autoregressive models [156], variational autoencoders (VAEs) [157], normalising flows [158] and denoising diffusion probabilistic models (DDPMs) [140].

The power of generative models is that they can capture complicated and high-dimensional distributions. Once learned, an unlimited number of samples can be drawn from the distribution. Generative models are often used in tasks such as image synthesis, natural language processing, and anomaly detection, as well as problems with medical and biological data. Even as recently as 2021, GANs have been considered the state of the art in terms of quality of sample generation. However, this has recently been challenged by DDPMs.

3.3.1 GANs: Generative adversarial networks

A generative adversarial network (GAN) [155] consists of two components: the generator $G(z;\theta_g)$ and the discriminator $D(x;\theta_d)$. The generator and the discriminator are simultaneously trained neural networks parameterized by θ_g and θ_d respectively. $G(z;\theta_g)$

takes input noise variables $p_z(z)$ (typically gaussian) and learns to output samples which the discriminator compares to the real data x from the distribution $\mathbb{P}(x)$. The discriminator learns to determine if the image is real or fake. The objective function $\mathcal{L}_{\text{GAN}}$ from a two-player minimax game is defined as:

$$\min_G \max_D \mathcal{L}_{\text{GAN}}(G, D) = \mathbb{E}_{x\sim\mathbb{P}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] \tag{3.7}$$

In the training loop, first the discriminator is updated to maximise the probability of classifying the image as real or fake, followed by the generator being trained to fool the discriminator by minimising $\log(1 - D(G(z)))$. The loop is iterated until the generator outputs realistic images.

Training this GAN is equivalent to attempting to minimise the Jensen-Shannon divergence between the real data distribution $\mathbb{P}$ and the generated distribution $\mathbb{P}_g$ [155, 159]. This is defined as:

$$\text{JSD}(\mathbb{P} \parallel \mathbb{P}_g) = \frac{1}{2}\text{KL}(\mathbb{P} \parallel \mathbb{P}_A) + \frac{1}{2}\text{KL}(\mathbb{P}_g \parallel \mathbb{P}_A) \tag{3.8}$$

where $\mathbb{P}_A$ is the average distribution and KL is the Kullback-Leibler divergence between the two distributions, given by:

$$\text{KL}(\mathbb{P} \parallel \mathbb{P}_g) = \int_{\mathcal{X}} P(x) \log \frac{P(x)}{P_g(x)} \mathrm{d}x \tag{3.9}$$

where it is assumed both distributions are continuous with densities P and P_g. KL divergence is a way to quantify the difference between the two probability distributions. We define KL divergence here as it appears again in this section.

Wasserstein GANs

When successfully trained, GANs are capable of producing visually high-quality samples. However, they are also well known to suffer from unstable training dynamics [159] such as mode collapse [160], where the generator produces low diversity samples which are over-optimised to fool the discriminator.

The Wasserstein GAN (WGAN) [161] was introduced as an improvement to overcome GAN training instabilities. The proposed modification addresses some issues caused by using Jensen-Shannon divergence as the minimisation objective. Arjovsky *et al.* [161] showed that Jensen-Shannon divergence does not always provide usable gradients, and instead proposed the Wasserstein distance to be used in the loss function,

which is defined as:

$$W(\mathbb{P}, \mathbb{P}_g) = \sup_{\|f\|_L \leq 1} \mathbb{E}_{x\sim\mathbb{P}(x)}[f(x)] - \mathbb{E}_{x\sim\mathbb{P}_g(x)}[f(x)] \tag{3.10}$$

where the supremum is over 1-Lipschitz functions f. The authors show that it possible to solve the following:

$$\max_{w\in\mathcal{W}} \mathbb{E}_{x\sim\mathbb{P}}[f_w(x)] - \mathbb{E}_{z\sim p(z)}[f_w(g_\theta(z))] \tag{3.11}$$

for the family of functions $\{f_w\}_{w\in\mathcal{W}}$ that are all K-Lipschitz for some K. It can also be shown that W can be minimised with gradient descent, with its gradient given by:

$$\nabla_\theta W(\mathbb{P}, \mathbb{P}_g) = -\mathbb{E}_{z\sim p(z)}[\nabla_\theta f(g_\theta(z))] \tag{3.12}$$

To achieve minimising W, a critic is introduced to approximate f. Rather than to differentiate between real and fake outputs, the critic D (previously discriminator) is trained to learn a K-Lipschitz function to minimise the Wasserstein loss between the real data and the generator output. This is achieved with the loss function:

$$\mathcal{L}_D(G, D) \ = \mathbb{E}_{x\sim\mathbb{P}(x)}[D(x)] - \mathbb{E}_{z\sim\mathbb{P}_g(z)}[D(G(z))] + \lambda\mathcal{L}_{\text{GP}} \tag{3.13}$$

Where λ is a weighting parameter for the gradient penalty term introduced by Gulrajani *et al.* [162]:

$$\mathcal{L}_{\text{GP}}(D) \ = \ \mathbb{E}_{\hat{x}\sim\mathbb{P}_{\hat{x}}}[(||\nabla_{\hat{x}} D(\hat{x})||_2 \ - \ 1)^2] \tag{3.14}$$

which ensures the critic has smooth, Lipschitz continuous gradients. $\hat{x} \sim \mathbb{P}_{\hat{x}}$ is from uniformly sampling along straight lines between pairs of points in the real data and generated data distributions [163]. This particular formulation is a Wasserstein GAN with gradient penalty (WGAN-GP).

Conditional GANs (image-to-image)

The input to the generator G for an unconditional GAN is random noise z. Conditional GAN [164] generators can take an input x as well as noise z and learn the mapping $G : \{x, z\} \rightarrow y$. Image conditional GANs (*pix2pix* from Isola *et al.* [165]) can be used to solve image-to-image translation and reconstruction problems.

For a conditional GAN (cGAN), the objective function $\mathcal{L}_{\text{cGAN}}$ from a two-player minimax game is defined as:

$$\mathcal{L}_{\text{cGAN}}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_x[\log(1 - D(x, G(x)))] \tag{3.15}$$

where G is trying to minimise this objective, and D is trying to maximize it:

$$\min_G \max_D \mathcal{L}_{\text{cGAN}}(G, D) \tag{3.16}$$

Pix2pix [165] introduced the model objective function G^*:

$$G^* = \arg \min_G \max_D \mathcal{L}_{\text{cGAN}}(G, D) + \lambda \mathcal{L}_{\text{L1}}(G) \tag{3.17}$$

where $\mathcal{L}_{\text{L1}}$ is the mean absolute distance between the real and predicted images.

$$\mathcal{L}_{\text{L1}} = \mathbb{E}_{x,y,z}[||y - G(x, z)||_1] \tag{3.18}$$

Conditional WGAN

Conditional Wasserstein GANs follow by training to maximise:

$$\mathcal{L}_G(G, D) = \lambda_1 \mathcal{L}_{\text{L1}} - \mathbb{E}_{x,G(x)}[D(x, G(x))] \tag{3.19}$$

while minimising the critic:

$$\mathcal{L}_D(G, D) = \mathbb{E}_{x,y}[D(x, y)] - \mathbb{E}_{x,G(x)}[D(x, G(x))] + \lambda_2 \mathcal{L}_{\text{GP}} \tag{3.20}$$

where λ_2 is a weighting parameter for the gradient penalty term:

$$\mathcal{L}_{\text{GP}}(D) = \mathbb{E}_{x,\hat{x}}[(||\nabla_{\hat{x}} D(x, \hat{x})||_2 - 1)^2] \tag{3.21}$$

Applications of GANs

Further improvements to GAN architectures have been made (recently popular are StyleGAN [166, 167] and BigGAN-Deep [168]), and the state-of-the-art performance has advanced with computing power. Brock *et al.* [168] demonstrated that GANs benefit significantly from scaling number of parameters and batch size, and much of the focus of GAN research is on customising network architectures. Performance of state-of-the-art models is generally benchmarked on tasks such as inpainting, colourization, uncropping, restoration using public datasets such as ImageNet [102].

GANs have been applied extensively in medical imaging fields for tasks such as translation between CT, PET and MRI images [169] and related multi-modal synthesis [170]. General frameworks such as MedGAN [171] can be used for a variety of image-to-image tasks.

In drug discovery, GANs have be used to attempt to generate new molecules (*de novo* drug design) and 3D structures, as well as to generate synthetic data to train other ML models [172]. Image-based profiling applications include conditioning a GAN on Cell Painting feature profiles to generate compounds [108], and using GANs to reduce unwanted batch-specific information by transforming HCS images [173]. Another model, CytoGAN, was trained as preliminary task with the goal of extracting embeddings as feature representations for image-based profiling [92].

Problems with GANs

Despite the success of GANs, they are likely not the optimal solution to best learn a data distribution [174]. Their results in image-based profiling applications are yet to have a large impact on the field. With immense training data and computational power GANs can be very good at producing visually high-quality images, however this does not guarantee capturing the exact distributions or even the correct structures. In addition to training instabilities and mode collapse, GANs can produce feature hallucinations or phantoms which are particularly undesirable in medical and biological applications [175]. While GANs have been considered state-of-the-art models for many imaging tasks for a number of years, recently they have been challenged by more stable models which have achieved comparable or superior image quality.

3.3.2 DDPMs: Denoising diffusion probabilistic models

While various deep generative models including VAEs, autoregressive models and flow models have advanced alongside GANs, it has been diffusion models which have emerged as the most successful alternative. They have several advantages over GANs, including a stable training objective and easy scalability while still being capable of producing high-quality images.

Diffusion models

In a diffusion model [176], the *forward process* is defined as the Markov chain $q(x_{1:T} \mid x_0)$ which gradually adds gaussian noise to the data $x_0 \sim q(x_0)$ over T timesteps. This produces the sequence of samples with increasingly more noise $x_1, \ldots, x_T$ with the

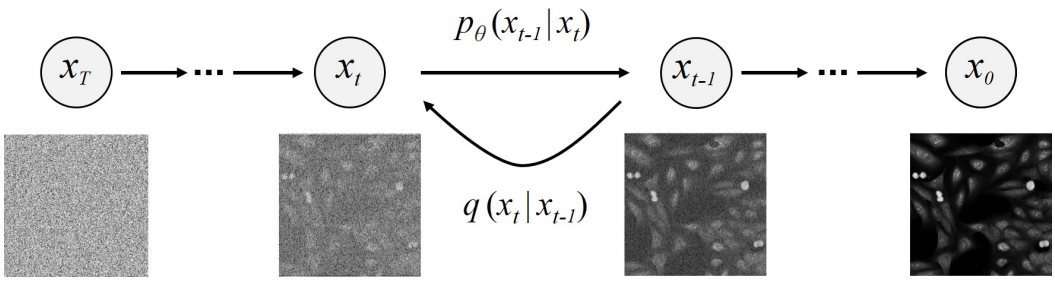


Fig. 3.4 An illustration of the Markov chain forward and reverse sampling processes for diffusion models. Adapted from Ho *et al.* [140]

noise variance schedule $\beta_1, \ldots, \beta_T$.

$$q(x_{1:T} \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1}), \quad q(x_t \mid x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I) \tag{3.22}$$

For a sufficient number of steps (e.g. $T = 1000$ or $T = 2000$), x_T becomes so heavily noised it is assumed to have been sampled from the gaussian distribution. This process of adding noise step-by-step is visualised in Fig. 3.4.

By attempting to sample in the opposite direction (i.e. by finding $q(x_{t-1} \mid x_t)$), it is possible to try and recreate the real image by starting from gaussian noise. This is known as the *reverse process*, which for a model p_θ (which we want to learn) is defined as:

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} \mid x_t), \quad p_\theta(x_{t-1} \mid x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{3.23}$$

The training regime for the forward process involves minimising the variational bound on negative log likelihood:

$$\mathbb{E}[-\log p_\theta(x_0)] \leq \mathbb{E}_q \left[-\log \frac{p_\theta(x_{0:T})}{q(x_{1:t} \mid x_0)} \right] = \mathbb{E}_q \left[-\log p(x_T) - \sum_{t \geq 1} \log \frac{p_\theta(x_{t-1} \mid x_t)}{q(x_t \mid x_{t-1})} \right] = L \tag{3.24}$$

where the variance schedule β_t can either be learned or fixed. Sohl-Dickstein *et al.* [176] show that in the forward process x_t can be sampled in closed form for arbitrary t:

$$q(x_t \mid x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I) \tag{3.25}$$

where $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$. This allows for efficient training of L with random sampling of timesteps t. The authors also show that L can be rewritten in terms of KL (Eqn. 3.9):

$$L = L_0 + L_T + \sum_{t>1} L_{t-1} \tag{3.26}$$

where:

$$\begin{aligned} L_0 &= -\log p_\theta(x_0 \mid x_1) \\ L_T &= \mathrm{KL}\big(q(x_T \mid x_0) \parallel p(x_T)\big) \\ L_{t-1} &= \mathrm{KL}\big(q(x_{t-1} \mid x_t, x_0) \parallel p_\theta(x_{t-1} \mid x_t)\big) \end{aligned} \tag{3.27}$$

This formulation compares $p_\theta(x_{t-1} \mid x_t)$ to the forward process posterior, which is conditioned on x_0. Hence, the KL divergences can be calculated with closed form expressions.

Denoising diffusion probabilistic models

By observing connections between diffusion models and denoising score matching [177], Ho *et al.* [140] showed that a simplified, weighted variational bound objective can be used to train diffusion models:

$$L_{\text{simple}}(\theta) = \mathbb{E}_{t,x_0,\epsilon}\left[\parallel \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t) \parallel^2\right] \tag{3.28}$$

where $\epsilon = \mathcal{N}(0, I)$ and t is uniformly distributed between 1 and T. This is equivalent to training the function $\epsilon_\theta(x_t, t)$ to predict the noise component of the noisy sample x_t. Once this objective is trained with gradient descent, images can be sampled with:

$$x_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}}\epsilon_\theta(x_t, t)\right) + \sqrt{1-\alpha_t}z \tag{3.29}$$

for $t = T, \ldots, 1$ iterations, starting from noise $x_T \sim \mathcal{N}(0, I)$. For $t = 0$, $z = 0$, otherwise z is normally distributed $z \sim \mathcal{N}(0, I)$.

Conditional DDPM

The denoising diffusion probabilistic model (DDPM) was improved further by Dhariwal and Nichol [139] who introduced a number of changes including using a U-Net architecture [134] with self-attention layers [124], as well as the BigGAN [168] residual block for upsampling and downsampling the activations [178].

Ho *et al.* [140] fixed the variance Σ, however Dhariwal and Nichol show that parameterising it as a neural network $\Sigma_\theta(x_t, t)$ results in superior performance [139]. Additionally, they propose two modification to achieve conditional image synthesis with class labels. The first way to incorporate class labels is in training the network. Adaptive group normalisation (AdaGN) is a layer used to incorporate the timestep and class embedding into the residual blocks of the netowrk following a group normalization operation [179]. It is defined as:

$$\text{AdaGN}(h, k) = k_s \text{GroupNorm}(h) + k_b \tag{3.30}$$

where $k = [k_s, k_b]$ is a linear projection of the timestep and class embedding, and h is the activations of the residual block after the first convolution.

The second modification, classifier guidance, enables the use of class information in inference of the trained diffusion model. Sohl-Dickstein *et al.* [176] and Song *et al.* [178] showed this can be achieved using pre-trained classifier gradients to condition the sampling of the diffusion model. First, the classifier $p_\phi(k \mid x_t)$ is pre-trained to predict the class k from noisy images x_t.

The aim is to sample each transition from the distribution:

$$p_{\theta,\phi}(x_t \mid x_{t+1}, k) = Z p_\theta(x_t \mid x_{t+1}) p_\phi(k \mid x_t) \tag{3.31}$$

where $p_\theta(x_t \mid x_{t+1})$ is the unconditional reverse noising process and Z is a normalizing constant. Although it is intractable to sample from the distribution in Eqn. (3.31), it can be approximated as a perturbed guassian distribution [176]:

$$\log(p_\theta(x_t \mid x_{t+1}) p_\phi(k \mid x_t)) \approx \log p(z) + C, \tag{3.32}$$

$$z \sim \mathcal{N}(\mu + \Sigma g, \Sigma), \quad g = \nabla_{x_t} \log p_\phi(k \mid x_t)|_{x_t=\mu} \tag{3.33}$$

where $g = \nabla_{x_t} \log p_\phi(k \mid x_t)$ are the gradients of the classifier and C is a constant which can be ignored. In inference, this shifts the mean of the sampled Gaussian to guide the denoising process towards the given class label k.

In inference, the classifier gradients can be scaled with a constant s. Hence in the sampling loop for $t = T, \ldots, 1$ iterations:

$$x_{t-1} \leftarrow \text{sample from } \mathcal{N}\big(\mu_\theta(x_t) + \underbrace{s\Sigma_\theta(x_t)\nabla_{x_t} \log p_\phi(k \mid x_t)}_{\text{shifted mean}}, \Sigma_\theta(x_t)\big) \tag{3.34}$$

where the loop begins by sampling x_T from $\mathcal{N}(0, I)$ as in Eqn. 3.29.

Other famous diffusion models exist, such as those incorporating natural language text encoders (text-to-image) [180, 181]. The scalability and quality of these models is so impressive that in 2022 there was commercial success for models such as DALL-E 2 which uses a CLIP (contrastive language image pre-training) image embedding to guide image generation with text captions [180].

Image-to-image diffusion

There are new frameworks for image-conditional diffusion such as *Palette* from Saharia *et al.* [141]. Their model outperforms GANs on four tasks: colorization, inpainting, uncropping and JPEG restoration. *Palette* is a denoising diffusion probabilistic model [140] of the form $p(y \mid x)$ which is trained to predict the output image y conditional on the input image x (note the change of notation in this subsection). The noisy image $\widetilde{y}$ is given by:

$$\widetilde{y} = \sqrt{\gamma} y + \sqrt{1-\gamma}\epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \tag{3.35}$$

for Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ and noise level indicator γ. A neural network f_θ is trained to denoise $\widetilde{y}$ for a given x with the loss function:

$$\mathbb{E}_{(x,y)}\mathbb{E}_{\epsilon\sim\mathcal{N}(0,I)}\mathbb{E}_{\gamma}\left\| f_\theta(x, \underbrace{\sqrt{\gamma} y + \sqrt{1-\gamma}\epsilon}_{\widetilde{y}}, \gamma) - \epsilon \right\|_p^p \tag{3.36}$$

where p is the chosen norm (L_1 or L_2). Eqn. 3.36 is the image-conditional version of L_{simple} from Eqn. 3.28.

The reverse diffusion process is computed step-by-step as:

$$y_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}\left(y_t - \frac{1-\alpha_t}{\sqrt{1-\gamma_t}} f_\theta(x, y_t, \gamma_t)\right) + \sqrt{1-\alpha_t}\epsilon_t \tag{3.37}$$

for $t = T, \ldots, 1$ steps. The noise level indicator γ_t is a function of t, and α_t is the noise variance scale parameter (also timestep-dependent).

Image-to-image diffusion models are so recent that there is limited literature of their application on biological or medical data (and nothing in image-based-profiling at the time of writing, to the best of our knowledge). However, we propose that these models could outperform GANs for cross modality prediction across a wide range of data, and we expect to see this in future studies. In Chapter 6 we present the first study to apply these models to high-content microscopy data.

Problems with DDPMs

The main issue with DDPMs are that they require sampling all T steps of the Markov chain to generate an image, which is usually $T = 1000 - 2000$ steps. This makes the model extremely slow compared to sampling once from a GAN, for example. While we don't see this as a major consideration, as most state of the art models require immense computing power for optimal performance, it is a problem for generating big datasets, such as those in HCS. One proposed solution is the denoising diffusion implicit model (DDIM) [182] which can generate high quality samples up to 50 times faster than the DDPM with the same training objective. However, the cost is that the performance is inferior to the DDPM over 100s of steps. Given we expect further improvements in sampling speed without the tradeoff of reducing image quality, we choose not to use DDIM for our studies.

3.4 Evaluating generative models

In medical and biological studies, often the most informative way to evaluate generated images is by testing their utility and performance in real-world tasks, such as in downstream image-based profiling experiments (we do this in Chapter 4 and Chapter 6). However, there are also well known metrics used to quantify the quality of the output at the image-level. When evaluating image-to-image models, a holistic approach is taken where multiple metrics are compared to understand the quality of the prediction.

This can be done by i) comparing the distributions of the data, which can be applied to both conditional and unconditional generative models, and ii) comparing image predictions with ground truth images, which can only be applied to conditional, image-to-image models. We describe a number of these metrics in this section.

3.4.1 Fréchet Inception distance

The Fréchet Inception distance (FID) [183] was introduced specifically to evaluate the performance of GANs, and is now widely adopted as the standard benchmark to evaluate and compare state-of-the art generative models.

FID is an improvement on Inception score (IS) [184] which is a metric also developed to evaluate generative models. For (generated) images $\tilde{x}$, Inception score is defined by:

$$\text{IS} = \exp(\mathbb{E}_{\tilde{x}}\text{KL}(p(y \mid x) \parallel p(y)) \tag{3.38}$$

where KL is the Kullback-Leibler divergence (Eq. 3.9), $p(y \mid x)$ is the conditional label distribution from the trained Inception model [131], and $p(y)$ is the distribution of class probabilities of the dataset used to train the Inception network. IS measures generated image quality by assuming good quality images should be difficult to classify with confidence. While IS correlates well with human judgement, it only evaluates the distribution of the generated images which is less useful for medical and biological datasets as it uses weights from a classifier trained on natural image datasets (e.g. Imagenet [102]). Additionally, IS measures diversity of the dataset which may not be a desirable metric for biological or medical datasets.

Hence we propose that FID is a more useful metric for evaluating high-content microscopy images. The Fréchet distance is evaluated between the features of the real distribution of images and the features of the generated distribution of images. As with IS, FID calculates feature vectors with the activations of the last pooling layer of Inception v3 [131].

For the real images $x \in X$, and the generated predictions $\tilde{x} \in \tilde{X}$ first calculate the features with the trained Inception network: $p(y \mid X)$ and $p(y \mid \tilde{X})$. Next, fit gaussian distributions $\mathcal{N}(\mu_x, \Sigma_x)$ and $\mathcal{N}(\mu_{\tilde{x}}, \Sigma_{\tilde{x}})$ with the mean and variance from $p(X)$ and $p(\tilde{X})$ respectively. The Fréchet Inception distance d_F between two distributions is then defined as:

$$d_F^2(\mathcal{N}(\mu_x, \Sigma_x), \mathcal{N}(\mu_{\tilde{x}}, \Sigma_{\tilde{x}})) = \|\mu_x - \mu_{\tilde{x}}\|_2^2 + \mathrm{Tr}\big(\Sigma_x + \Sigma_{\tilde{x}} - 2(\Sigma_x \Sigma_{\tilde{x}})^{1/2}\big) \tag{3.39}$$

The lower the FID score, the more similar the two distributions are. FID is used to evaluate generative models in medical imaging [185], and very recently in high-content image-based profiling studies [99].

3.4.2 Structural similarity index measure

The structural similarity index measure (SSIM) [186] is used to measure the similarity of two images. It was designed to take into account luminance, contrast, and structure, which are all important factors in how the human visual system perceives images.

The SSIM between two images x and $\tilde{x}$ is defined by:

$$\mathrm{SSIM}(x, \tilde{x}) = \underbrace{\left(\frac{2\mu_x\mu_{\tilde{x}} + C_1}{\mu_x^2 + \mu_{\tilde{x}}^2 + C_1}\right)^{\alpha}}_{\text{luminance}} \cdot \underbrace{\left(\frac{2\sigma_x\sigma_{\tilde{x}} + C_2}{\sigma_x^2 + \sigma_{\tilde{x}}^2 + C_2}\right)^{\beta}}_{\text{contrast}} \cdot \underbrace{\left(\frac{\sigma_{x\tilde{x}} + C_3}{\sigma_x\sigma_{\tilde{x}} + C_3}\right)^{\gamma}}_{\text{structure}} \tag{3.40}$$

where μ and σ are the means and standard deviations of the respective images and C_1, C_2 and C_3 are stabilizing constants set by the range of pixel values in the images. α, β and γ define the relative strength of the three factors (often $\alpha = \beta = \gamma = 1$).

SSIM values are bounded between -1 for completely different images and 1 for identical images ($x = \tilde{x}$). Unlike pixel-based error functions, SSIM is designed to capture structural relationships between images, and is a widely adopted metric in reconstruction tasks to compare real and predicted images.

3.4.3 Pixel-based metrics

Finally, we describe the absolute, pixel-based metrics used to evaluate image similarity. It can be useful to compare pixel-based metrics for structural alignment, however pixel loss metrics are generally less informative for evaluating the quality of image content. For instance, adding salt-and-pepper noise to an image may only result in a small pixel-wise change, but human perception will judge the resulting image as being of lower quality compared to the original. When used as training objectives, pixel-wise loss functions will generate well-aligned but blurry images which can appear low-pass filtered [187].

Mean absolute error

Mean absolute error (MAE), or L1 loss, is the sum of the absolute distances between each pixel value in two images. For two images x and $\tilde{x}$ MAE is defined by:

$$\mathcal{L}_{\text{L1}}(x, \tilde{x}) = \frac{1}{N} \sum_{p=1}^{N} \left| x(p) - \tilde{x}(p) \right| \tag{3.41}$$

where $x(p)$ denotes the pixel value of image x for pixels $p = 1, \ldots, N$.

Mean squared error

Similarly, mean squared error (MSE), or L2 loss, is defined as:

$$\mathcal{L}_{\text{L2}}(x, \tilde{x}) = \frac{1}{N} \sum_{p=1}^{N} \left| x(p) - \tilde{x}(p) \right|^2 \tag{3.42}$$

MSE is also known as the Euclidean loss function.

Peak signal-to-noise ratio

Peak signal-to-noise ratio is the ratio of the maximum power of the signal in data to the power of the noise in the reconstruction [188]. PSNR is often used to quantify compressed images, however it can also be used to quantify image reconstruction quality. PSNR is defined (in dB) as:

$$\mathrm{PSNR}(x, \tilde{x}) = 10 \cdot \log_{10} \left(\frac{\mathrm{MAX}_x^2}{\mathcal{L}_{\mathrm{L2}}(x, \tilde{x})} \right) \tag{3.43}$$

where MAX_x is the maximum pixel value of the real image x, and $\tilde{x}$ is the reconstruction.

For high-content microscopy images, the maximum pixel value can either be chosen using an arbitrary threshold or it could be a very high value (e.g. greater than five standard deviations from the mean). Any kind of scaling or thresholding will significantly alter the PSNR, making the value of the metric limited. For this reason PSNR values cannot be compared between different datasets, however it can be used when comparing different models in the same dataset.

Pearson correlation coefficient

The Pearson correlation coefficient (PCC) can be used to describe the linear correlation between two variables [189]. The PCC of a single image pair $(x, \tilde{x})$ is defined as:

$$\mathrm{PCC}(x, \tilde{x}) = \frac{\mathrm{cov}(x, \tilde{x})}{\sigma_x \sigma_{\tilde{x}}} = \frac{\sum_{p=1}^{N} (x_p - \mu_x)(\tilde{x}_p - \mu_{\tilde{x}})}{\sqrt{\sum_{p=1}^{N} (x_p - \mu_x)^2} \sqrt{\sum_{p=1}^{N} (\tilde{x}_p - \mu_{\tilde{x}})^2}} \tag{3.44}$$

where $\mathrm{cov}(x, \tilde{x})$ is the covariance of the two images x and $\tilde{x}$.

Chapter 4

Label-Free Prediction of Cell Painting from Brightfield Images

In this chapter we investigate label-free Cell Painting by predicting the five fluorescent Cell Painting channels from paired brightfield z-stacks with two deep learning models. We train and validate the models with a dataset representing 1000s of pan-assay interference compounds sampled from 17 unique batches. The model predictions are evaluated using a test set from two additional batches, treated with compounds comprised from a publicly available phenotypic set. In addition to pixel-level evaluation, we process the label-free Cell Painting images with a segmentation-based feature-extraction pipeline in CellProfiler to understand whether the generated images are useful in downstream analysis. The mean Pearson correlation coefficient (PCC) of the images across all five channels is 0.84. Without actually incorporating these features into the model training we achieved a mean correlation of 0.45 from the features extracted from the images. Additionally we identified 30 features which correlated greater than 0.8 to the ground truth. Toxicity analysis on the label-free Cell Painting resulted a sensitivity of 62.5% and specificity of 99.3% on images from unseen batches. We provide a breakdown of the feature profiles by channel and feature type to understand the potential and limitation of the approach in morphological profiling. Our findings demonstrate that label-free Cell Painting has potential above the improved visualization of cellular components, and it can be used for downstream analysis. The findings also suggest that label-free Cell Painting could allow for repurposing the imaging channels for other non-generic fluorescent stains of more targeted biological interest, thus increasing the information content of the assay.

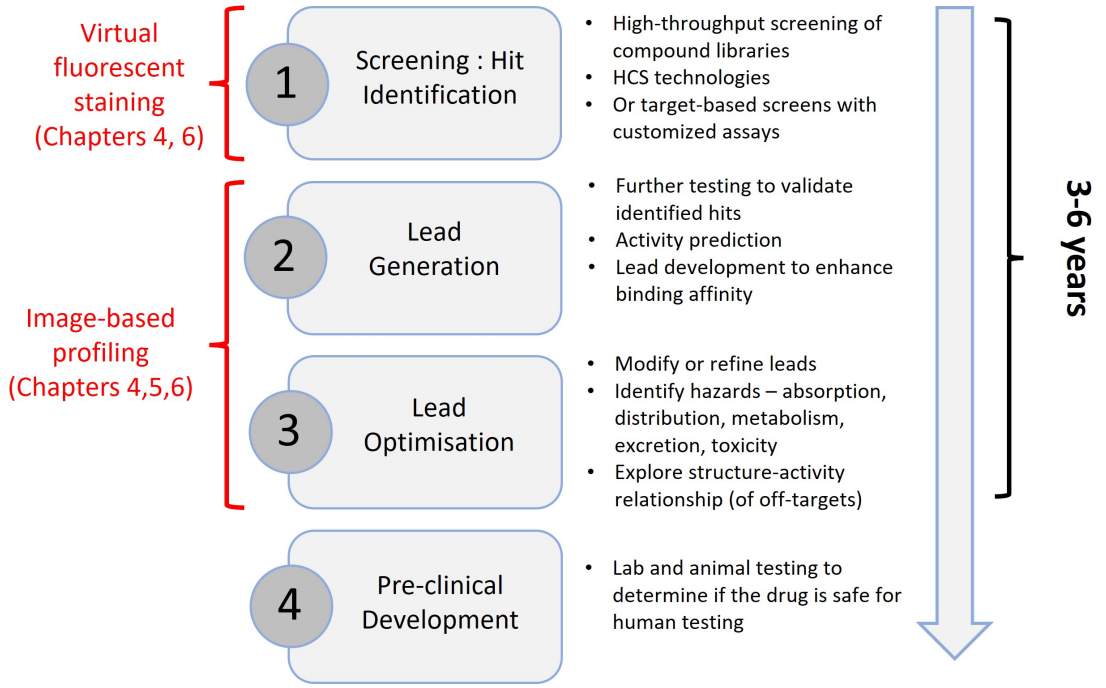


Fig. 4.1 Placing the experimental chapters of this thesis in the drug discovery pipeline. Virtual cell staining methods replace traditional staining in the screening phase. Image-based profiling is employed after the images have been acquired to identify hits and leads which may be taken to clinical trials (Phases 4-6 in Fig. 2.5.)

4.1 Introduction

Advances in deep learning and computer vision have rapidly advanced image-based profiling, and is helping to accelerate drug discovery [9]. It is possible to use transmitted light images such as brightfield as an input to a convolutional neural network to generate the corresponding fluorescent images – so called *in silico* labelling [35]. As this is a method which is designed to replace screening, it occurs very early in the drug discovery pipeline (Fig. 4.1. Image-based profiling is the next step, which also pre-clinical but after the experimental screening phase.

One way to achieve virtual fluorescent staining this is with conditional [164] generative adversarial networks (GANs) [155], introduced by Isola in 2016 (also known as pix2pix) [165]. From the input samples of an underlying unknown joint distribution of multi-modal data, the generator network can be trained to generate complementing paired data. The advantage of the conditional GAN is to overcome the limitations of a pixel-wise loss function such as L1 [190], used in many standard U-Net models, traditionally for simpler segmentation tasks.

In this study we investigate the utility of label free Cell Painting. Prior to this study, image-to-image approaches in Cell Painting are rare, with only known instance of GANs specifically for Cell Painting using Nucleoli, cytoplasmic RNA, Golgi plasma membrane and F-actin as an input to predict the other two fluorescent channels [191]. Fluorescent channels contain significantly more information than a brightfield z-stack, therefore predicting full fluorescent channels from brightfield images is a significantly more challenging task. However, it is a far more useful task to investigate as the staining process has many drawbacks (Section 2.2.1) which could be overcome by using solely brightfield microscopy in a HCS assay.

To the best of our knowledge this work is the first to demonstrate a label-free, five-channel Cell Painting replication from a transmitted light input modality such as brightfield. In this chapter we investigate the quality of the Cell Painting features extracted from the model-predicted image channels by correlating with the ground truth features. Additionally we attempt to replicate clustering by treatment group on a test sample of 273 fields of view from two different batches. This was achieved by training two models on a large, multi-batch dataset: a U-Net trained with L1 loss, and the same U-Net trained as the generator of a conditional Wasserstein [161] GAN. In order to fully interrogate the utility of the label-free Cell Painting image channels, a comparison of both image-level metrics and morphological feature predictions between the two models is presented, contextualizing the non-normalized image metrics [192] and the feature-level evaluation results. In this study, we aimed to investigate the potential of using deep generative models to assist or replace fluorescent staining in future research.

Note: In this chapter we use the term *label* as the fluorescent staining label. We refer to discrete value metadata as *classes* or *class labels*.

4.2 Materials and Methods

4.2.1 Dataset

Cell Culture and Seeding

In this study we used a dataset from AstraZeneca. A summary of the dataset is provided in Table 4.1. A U-2 OS cell line was sourced from AstraZeneca's Global Cell Bank (ATCC Cat# HTB-96). Cells were maintained in continuous culture in McCoy's 5A media (#26600023 Fisher Scientific, Loughborough, UK) containing 10% (v/v) fetal bovine serum (Fisher Scientific, #10270106) at 37°C, 5% (v/v) CO_2, 95%

humidity. At 80% confluency, cells were washed in PBS (Fisher Scientific, #10010056), detached from the flask using TrypLE Express (Fisher Scientific, #12604013) and resuspended in media. Cells were counted and a suspension prepared to achieve a seeding density of 1500 cells per well using a 40 µL dispense volume. Cell suspension was dispensed directly into assay-ready (compound-containing) CellCarrier-384 Ultra microplates (#6057300 Perkin Elmer, Waltham, MA, USA) using a Multidrop Combi (Fisher Scientific). Microplates were left at room temperature for 1h before transferring to a microplate incubator at 37°C, 5% (v/v) CO_2, 95% humidity for a total incubation time of 48 hours.

Compound treatment

All chemical compounds were sourced internally through the AstraZeneca Compound Management Group and prepared in stock solutions ranging 10-50mM in 100% DMSO. Compounds are transferred into CellCarrier-384 Ultra microplates using a Labcyte Echo 555T from Echo-qualified source plates (#LP-0200 Labcyte, High Wycombe, UK). Compounds were tested at multiple concentrations; either 8 concentration points at 3-fold (half log) intervals or 4 concentration points at 10-fold intervals. Control wells situated on each plate consisted of neutral (0.1% DMSO v/v) and positive controls (63nM mitoxantrone, a known and clinically-used topoisomerase inhibitor and DNA intercalator). Compound addition to microplates was performed immediately prior to cell seeding to produce assay-ready plates.

Cell staining

The Cell Painting protocol was applied according to the original method [10] with minor adjustments to stain concentrations. Hank's balanced salt solution (HBSS) 10x was sourced from AstraZeneca's media preparation department and diluted in dH2O and filtered with a 0.22µm filter. MitoTracker working stain was prepared in McCoy's 5A medium. The remaining stains were prepared in 1% (w/v) bovine serum albumin (BSA) in 1x HBSS.

Post incubation with compound, media was evacuated from assay plates using a Blue®Washer centrifugal plate washer (BlueCatBio, Neudrossenfeld, Germany). 30µL of MitoTracker working solution was added and the plate incubated for a further 30 min at 37°C, 5% CO_2, 95% humidity. Cells were fixed by adding 11µL of 12% (w/v) formaldehyde in PBS (to achieve final concentration of 3.2% v/v). Plates were incubated at room temperature for 20 min then washed using a Blue®Washer. 30µL of 0.1% (v/v) Triton X-100 in HBSS (#T8787 Sigma Aldrich, St. Louis, MO, USA)

solution was dispensed and incubated for a further 20 min at room temperature followed by an additional wash. 15µL of mixed stain solution was dispensed, incubated for 30 min at room temperature then removed by washing. Plates were sealed and stored at 4°C prior to imaging.

Imaging

Microplates were imaged on a CellVoyager CV8000 (Yokogawa, Tokyo, Japan) using a 20x water-immersion objective lens (NA 1.0). Excitation and emission wavelengths are as follows for fluorescent channels: DNA (ex: 405nm, em: 445/45nm), ER (ex: 488nm, em: 525/50nm), RNA (ex: 488nm, em: 600/37nm), AGP (ex: 561nm, em: 600/37nm) and Mito (ex: 640nm, em: 676/29nm). The three brightfield slices are from different focal z-planes; within, 4µm above and 4µm below the focal plane. Images were saved as 16-bit .tif files without binning (1996×1996 pixels).

Pre-processing

The selected images were bilinearly downscaled from 1996×1996 to 998×998 pixels to reduce computational overheads. Global intensity normalization was implemented to eliminate intensity differences between batches. For each channel, each 998×998 image was constrained to have a mean pixel value of zero and a standard deviation of one. Corrupted files or wells with missing fields were removed from the usable dataset and replaced with files from the corresponding batch.

4.2.2 Training and test set generation

We sampled from 17 of the 19 batches to select wells for training (Table 4.1). The remaining two batches were used to select the test set, and were excluded in the training process. Compounds from the test set were comprised from a publicly available list of known pharmacologically active molecules, with a known observable phenotypic activity. We randomly sampled 3000 wells for model training and hyperparameter tuning, with the constraint to force an overall equal number of wells per batch. We randomly selected one field of view from each of the four fields in the well, which was the image used in the training set. For model tuning we used 90/10 splits sampled randomly from the training set, before using the full training set to train the final models. The test set contained 273 images and was chosen by sampling randomly within each treatment group across the two remaining batches (treatment breakdown in Table 4.1).

	No. of Batches	No. of 384-well Plates	No. of Images	Controls and Treatment Distribution	Collection Description (no. of compounds)
Total Available	19	140	150,000 (4 fields per well)	2.1% positive controls; 6.4% negative controls; 91.5% treatments	Mixed collection containing a high chemical diversity set with a range of physicochemical properties (∼7,000); a phenotypic set (∼1,000) and a set of pan-assay interference compounds (∼1,000)
Training / Validation Set	17	124	3,000 (1 field of view used per well)	Randomly sampled from above	Randomly sampled from above
Test Set	2	16	273 (1 field of view used per well)	26 positive controls; 77 negative controls; 170 treatments	Publicly available phenotypic set (exclusively in test set)

Table 4.1 The composition of the Training, Validation and Test sets. Choosing data from many batches and experiments should make our models learn more robust, common features, which gives us the best chance of successful prediction on unseen data.

4.2.3 U-Net with L1 loss

Our first model is based on the original U-Net (Section 3.1.2) [134], a convolutional neural network which has proven very effective in many imaging tasks. U-Net architectures have been used to solve inverse reconstruction problems in cellular [55], medical [135] and general imaging problems [136], and are a sensible choice for image-to-image tasks. For segmentation problems, out of the box U-Net based architectures such as nnU-Net [137] have been proven to perform very compared to other state-of-the-art models.

U-Nets involve a number of convolutions in a contracting path with down sampling or pooling layers, and an expansive path with up sampling and concatenations, allowing for retention of spatial information while learning detailed feature relationships. The network captures multi-scale features of images through different resolutions by going up and down the branches of the network.

We adapted a 3 channel RGB channel U-Net model to have 3 input channels and 5 output channels to accommodate our data. An overview of the model network and training is presented in Figure 4.2. There were 6 convolutional blocks in the downsampling path, the first with 32 filters and the final with 1024 filters. Each block performed a 2d convolution with a kernel size of 3 and a stride of 1, followed by a ReLU then batch normalization operation. Between blocks a max pooling operation with a kernel size of 2 and a stride of 2 was applied for downsampling. The upsampling path was symmetric to the downsampling, with convolutions with a kernel of 2 and a stride of 2 applied for upsampling. The corresponding blocks in the contracting and expansive paths were concatenated as in the typical U-Net model. The final layer was a convolution with no activation or batch normalization. In total our network had 31×10^6 trainable parameters.

For pairs of corresponding images (x_i, y_i), where x_i is a 3 channel image from the brightfield input space and y_i is a 5 channel image from the real fluorescent output space, the loss function $\mathcal{L}_{\text{L1}}$ for the U-Net model is:

$$\mathcal{L}_{\text{L1}} = \mathbb{E}_{x,y}[||y - \hat{y}(x)||_1] \tag{4.1}$$

where $\hat{y}_i$ is the predicted output image from the network.

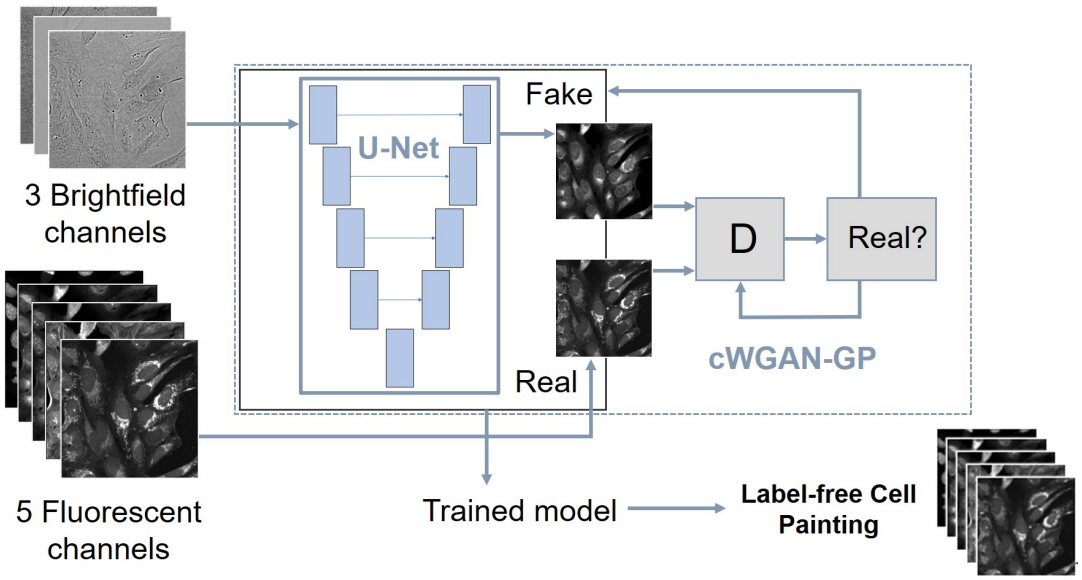


Fig. 4.2 Label-free Cell Painting model summary of the cWGAN-GP

4.2.4 cWGAN-GP: Conditional Wasserstein GAN with gradient penalty

The second model we trained is a conditional [164] GAN [155] (Section 3.3.1), where the generator network G is the same U-Net used in the first model. To overcome difficulties with training *Pix2pix* [165] we opted for a conditional Wasserstein GAN with gradient penalty [162] (cWGAN-GP) approach. This improved Wasserstein GAN was designed to stabilize training, useful for more complex architectures with many layers in the generator.

The Discriminator network D - alternatively the critic in the WGAN formulation - is a patch discriminator [165] with the concatenated brightfield and predicted Cell Painting channels as the eight-channel input. There were 64 filters in the final layer and there were three layers. For the convolutional operations the kernel size is 4 and the stride is 2. The output is the sum of the cross-entropy losses from each of the localized patches.

The L1 loss term enforces low-frequency structure, so using a discriminator which classifies smaller patches (and averages the responses) is helpful for capturing high frequency structures in the adversarial loss. For Wasserstein GANs the discriminator is called the critic as it is not trained to classify between real and fake, instead to learn a K-Lipschtiz function to minimize the Wasserstein loss between the real data and the generator output.

Hence, for our conditional WGAN-GP-based construction we trained the generator to minimize the following objective:

$$\mathcal{L}_G(G, D) \ = \lambda_1 \mathcal{L}_{\text{L1}} \ - \ \lambda_e \mathbb{E}_{x, G(x)}[D(x, G(x))] \tag{4.2}$$

where λ_1 is a weighting parameter for the L1 objective. The notation follows on from that introduced in Section 3.3.1 We introduce λ_e as an adaptive weighting parameter to prevent the unbounded adversarial loss overwhelming L1:

$$\mathcal{L}_{\text{L1}} = \mathbb{E}_{x,y}[||y - G(x)||_1] \tag{4.3}$$

The critic objective, which the network is trained to maximize, is:

$$\mathcal{L}_D(G, D) \ = \mathbb{E}_{x,y}[D(x, y)] - \mathbb{E}_{x, G(x)}[D(x, G(x))] + \lambda_2 \mathcal{L}_{\text{GP}} \tag{4.4}$$

where λ_2 is a weighting parameter for the gradient penalty term:

$$\mathcal{L}_{\text{GP}}(D) \ = \ \mathbb{E}_{x,\hat{x}}[(||\nabla_{\hat{x}} D(x, \hat{x})||_2 \ - \ 1)^2] \tag{4.5}$$

which is used to enforce the K-Lipschitz constraint. $\hat{x}$ is from uniformly sampling along straight lines between pairs of points in the real data and generated data distributions [163].

Failed models

Our dataset was small, the brightfield images are noisy, and many of the training images were over 50% background in the fluorescent channels, which makes this a difficult problem. GANs are hard to train and we experienced many failures when introducing adversarial loss. Firstly, *pix2pix* would not converge at all and all preliminary experiments produced considerably worse results than using L1 loss.

cWGAN-GP was chosen to stablise training, however we found that the adversarial component of the loss would become too large, resulting in phantoms in the background of the images (Fig. 4.3). To overcome this, we introduced the adaptive weighting parameter λ_e (Eq. (4.2)).

Hence, the two models we present in this study are the two models which we could train successfully on this dataset. Quantitative results for any other models are not reported, however we give some examples of the images of the failed models in Fig. 4.3.

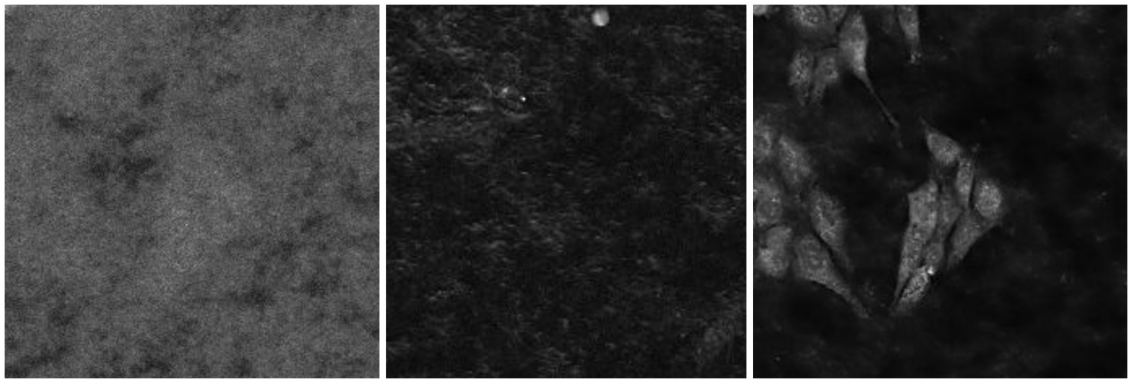


Fig. 4.3 Examples of output images from the failed preliminary models. Left: *pix2pix*. Middle and Right: background phantoms induced with cWGAN-GP without the adaptive weighting parameter λ_e.

Alternative approaches

U-Net is the most common choice for the generator network of the image-to-image conditional GAN due to the efficiency of the skip-connections in preserving structure. However, there are other architectures such as the ResNet [132] which can be used as the generator, as shown in other related studies [193]. The ResNet is built of several residual blocks embedded into in the CNN. The residual blocks consist of convolutional layers, ReLU activation and batch-norm layers (Section 2.7.1).

Our experience with ResNet with trial runs on subsets of our dataset showed the tendency to overfit [194] which is problematic when we have large amounts of background pixels and batch effects to consider. The residual U-Net has been developed to use residual blocks in the U-Net architecture [195], and improved performance was reported on standard datasets using residual blocks to adapt *pix2pix* in *pix2pixHD* [196]. There are many other choices for the generator architecture such as the U-Net++ [197] which is designed to be more powerful than a standard U-Net (Section 3.1.2) by replacing the skip connections with dense convolutional blocks and convolution layers to form a skip path.

In this study we chose a standard U-Net architecture as a benchmark to investigate the effect of adversarial loss in training compared to pixel-wise loss. We revisit label-free Cell Painting prediction from brightfield in Chapter 6 where we use improved U-Net architectures which use residual blocks and self-attention layers (Section 3.1.3) to improve model performance.

4.2.5 Model training and computational details

From the training set, random 256×256 patches were cropped to serve as inputs to the network. No data augmentation was used. The models were trained on the AstraZeneca Scientific Computing Platform with a maximum allocation of 500G memory and four CPU cores. The U-Net model was trained with a batch size of 10 for 15,000 iterations (50 epochs). The optimizer was Adam, with a learning rate of 2×10^{-4} and weight decay of 2×10^{-4}. The total training time for the U-Net model was around 30 hours.

The cWGAN-GP model was trained with a batch size of 4 for an additional 21,000 iterations (28 epochs). The generator optimizer was Adam, with a learning rate of 2×10^{-4} and β_1 of 0 and β_2 of 0.9. The critic optimizer was Adam with a learning rate of 2×10^{-4} and β_1 of 0 and β_2 of 0.9 and a weight decay of 1×10^{-3}. The generator was updated once for every 5 critic updates. The L1 weight is $\lambda_1 = 100$ and the gradient penalty weighting parameter is $\lambda_2 = 10$. $\lambda_e = 1/\text{epoch}$. The total training time for the cWGAN-GP model was an additional 35 hours. For each model, the best performing epoch was selected by plotting the image evaluation metrics of the validation split for each epoch during training. Once the metrics stopped improving (or got worse), training was stopped as the model was considered to be overfit.

Model inference

Input and output images were processed in 256×256 patches. The full-sized output images were stitched together with a stride of 128 pixels. This was chosen as half of the patch size so the edges met along the same line, which reduced the number of boundaries between tiles in the full-sized reconstructed image. Each pixel in the reconstructed prediction image for each fluorescent channel is the median value of the pixels in the four overlapping images. The full-sized, restitched images used for the metric and CellProfiler evaluations were 998×998 pixels for each channel, examples of which are displayed in the Appendix.

4.2.6 Evaluation

Image-level evaluation metrics

The predicted and target images were evaluated with five metrics: mean absolute error (MAE), mean squared error (MSE), structural similarity index measure (SSIM) [186, 188] peak signal-to-noise ratio (PSNR) [188] and the Pearson correlation coefficient (PCC) [35, 198]. MAE and MSE capture pixel-wise differences between the images,

and low values are favorable for image similarity. SSIM is a similarity measure between two images where for corresponding sub-windows of pixels, luminance, contrast, means, variances and covariances are evaluated. The mean of these calculations is taken to give the SSIM for the whole image. PSNR is a way of contextualizing and standardizing the MSE in terms of the pixel values of the image, with a higher PSNR corresponding to more similar images. PCC is used to measure the linear correlation of pixels in the images. For full details of these metrics see Chapter 3.

PSNR is normalized to the maximum potential pixel value, taken as 255 when the images are converted to 8-bit. For this dataset the PSNR appeared as high as the maximum 8-bit pixel values for each image was generally lower than the theoretical maximum value. Only SSIM can be interpreted as a fully-normalized metric, with values between 0 and 1 (1 being a perfect match).

4.2.7 Feature extraction with CellProfiler

A CellProfiler [42, 43] pipeline was utilized to extract image- and cell-level morphological features. The implementation followed the methodology of Way *et al.* 2021 [81], and is chosen as a representative application of Cell Painting. The pipeline we used is included in our GitHub repository (Appendix). CellProfiler was used to segment nuclei, cells and cytoplasm, then extract morphological features from each of the channels. Single cell measurements of fluorescence intensity, texture, granularity, density, location and various other features were calculated as feature vectors.

Features were aggregated using the median value per image. For feature selection, we adopted the following approach:

- Drop missing features - features with $> 5\%$ NaN values, or zero values, across all images
- Drop blocklisted features [199] which have been recognized as noisy features or generally unreliable
- Drop features with greater than 90% Pearson correlation with other features
- Drop highly variable features (>15 SD in DMSO controls)

This process reduced the number of useable features to 611, comparable to other studies. We used the ground truth data only in the feature reduction pipeline to avoid introducing a model bias to the selected features.

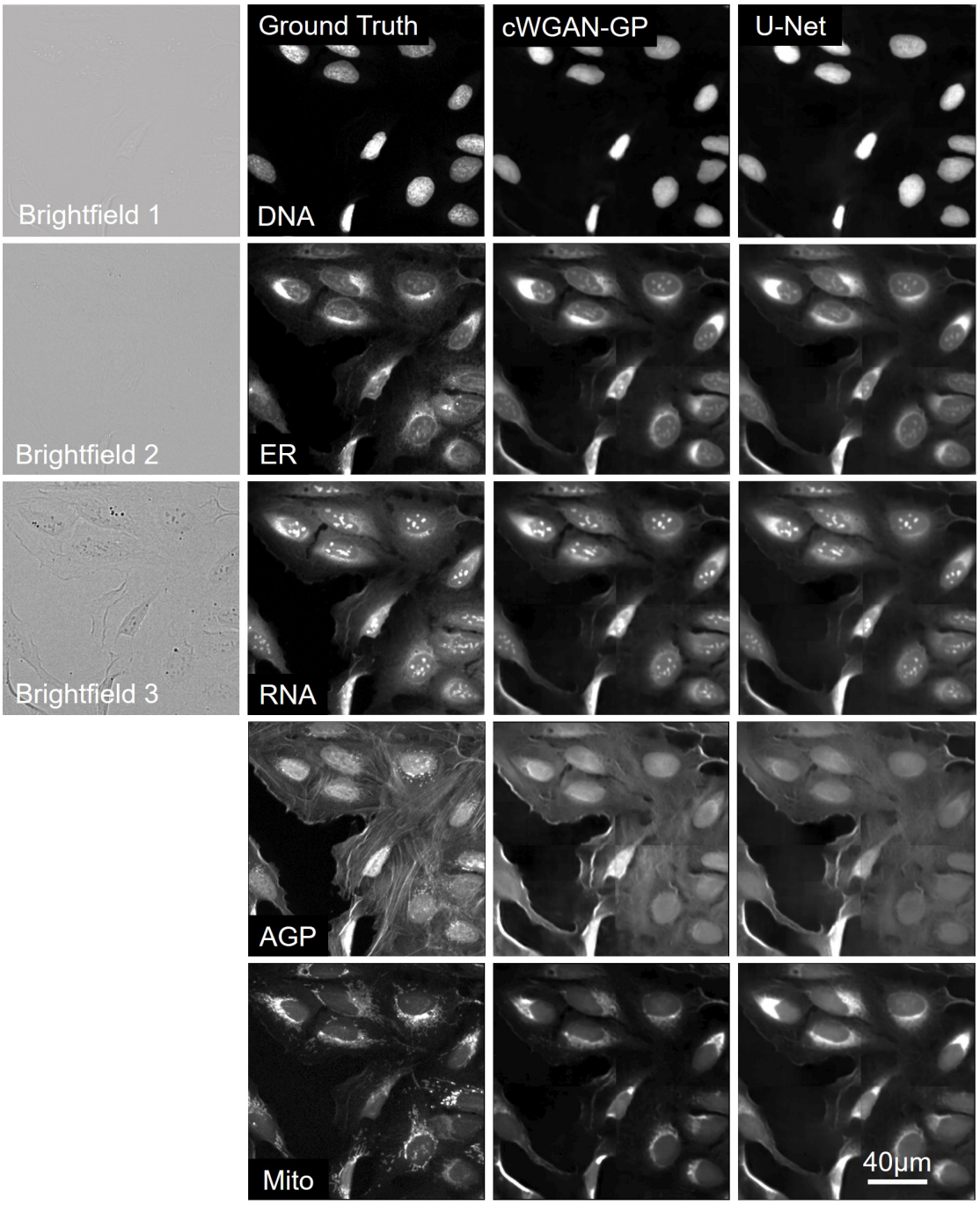


Fig. 4.4 Cropped images of typical brightfield images, ground truth fluorescent, and predicted channels from the test dataset for the U-Net and cWGAN-GP models. Presented at scale of the input and output size used in the model networks (256 x 256 pixels), the first column contains the three brightfield z-planes which form the input, and the subsequent columns are the five fluorescent output channels. The stitching method has been applied to the model predicted images, and small artefacts of this process are visible.

Morphological feature-level evaluation

Pairwise Spearman correlations [200] between the features in test set data were calculated for each model, with the mean values for each feature group grouped into correlation matrices, and visualized as heatmaps [81]. These features were split into several categories – area/shape, colocalization, granularity, intensity, neighbors, radial distribution and texture. We also visualized feature clustering using uniform manifold approximations (UMAPs) [201], implemented in python using the UMAP package [202].

Toxicity prediction

We normalized the morphological profiles to have a mean of zero and standard deviation of one, and classified the compounds with K-NN-classifier ($k = 5$, Euclidean distance) into two groups using the controls as the training class label (positive controls were used as an example of toxic phenotype). To account for the imbalance between the number of positive and negative controls, we sampled with a replacement equal number of profiles from both categories for training. We ran the classifier 100 times and used majority voting for the final classification.

4.3 Results

Channel	Model	SSIM	PSNR	MSE	MAE	PCC
DNA	U-Net	0.38 ± 0.13	51.8 ± 1.7	0.45 ± 0.12	0.41 ± 0.05	0.90 ± 0.06
	cWGAN	**0.39 ± 0.14**	**52.4 ± 1.5**	**0.39 ± 0.10**	**0.39 ± 0.04**	**0.90 ± 0.06**
ER	**U-Net**	**0.49 ± 0.09**	**48.0 ± 1.3**	**1.07 ± 0.30**	**0.58 ± 0.09**	0.86 ± 0.06
	cWGAN	0.48 ± 0.09	47.8 ± 1.3	1.12 ± 0.32	0.61 ± 0.09	**0.87 ± 0.06**
RNA	U-Net	0.59 ± 0.07	56.0 ± 1.6	0.18 ± 0.09	0.27 ± 0.06	0.92 ± 0.06
	cWGAN	**0.60 ± 0.07**	**56.3 ± 1.7**	**0.17 ± 0.10**	**0.26 ± 0.06**	**0.92 ± 0.06**
AGP	U-Net	0.35 ± 0.08	52.4 ± 1.8	0.27 ± 0.10	0.39 ± 0.06	0.80 ± 0.07
	cWGAN	**0.37 ± 0.09**	**54.7 ± 1.3**	**0.23 ± 0.09**	**0.34 ± 0.06**	**0.82 ± 0.06**
Mito	U-Net	0.31 ± 0.06	50.5 ± 1.1	**0.34 ± 0.08**	0.42 ± 0.05	**0.74 ± 0.08**
	cWGAN	**0.35 ± 0.06**	**53.1 ± 1.4**	0.34 ± 0.12	**0.36 ± 0.04**	0.70 ± 0.08

Table 4.2 Image metrics for each channel for the two models. The best performing model for each channel metric is highlighted in bold.

To systematically investigate the utility of the label-free prediction of Cell Painting from brightfield images, we conducted the evaluation on three separate levels: image-

level, morphological feature-level, as well as a downstream analysis on the profile-level to identify toxic compounds. We evaluated the two models on the test set of 148 unique compounds, 26 positive control wells (mitoxantrone), and 77 negative control wells (DMSO) that represented two experimental batches and 12 plates (Table 4.1). Examples and descriptions of cells in these different wells are presented in supplementary Figure A.

4.3.1 Image-level evaluation

The mean values of the image-level metrics for the U-Net/**cWGAN-GP** models respectively were SSIM: 0.42/**0.44**, PSNR: 51.7/**52.9**, MSE: 0.46/**0.45**, MAE: 0.41/**0.39**, and PCC: **0.84**/0.84, with the superior model in bold. cWGAN-GP achieved on average superior image-level performance; however, the difference was always within the standard deviation (Table 4.2). Visual inspection of the model generated images with the ground truth revealed strong resemblance (Figures 4.4, 4.5, 4.8). Most notably, both models struggled in predicting the fine structures in AGP and Mito channels, this was also reflected as lower image-level performance for the channels. Interestingly, only the plasma membrane of AGP channel was successfully predicted, whereas the models were not able to reproduce the actin filaments and the Golgi-apparatus structures. Similarly with the Mito channel, the models performed well overall in predicting the mitochondrial structures, but we observed lack in small granularity and detail that was present in the ground truth. In addition, we observed that the blocking effect from inference time sliding window was visible in the generated images. Nevertheless, DNA, ER, and RNA channels were better predicted, of which RNA achieved the best image-level performance.

4.3.2 Morphological feature-level evaluation

Next, we extracted morphological features at the single-cell level and aggregated them to well-level profiles with the aim to further elucidate the utility of label-free Cell Painting. For consistency and to avoid any potential bias favoring the trained models, we chose to use the ground truth images for feature selection and reduced the profiles extracted from the predicted images accordingly. This resulted in 273 well-aggregated profiles consisting of 611 morphological features.

Using correlation analysis, we deconvoluted the profile similarities according to feature type, cell compartment, and imaging channels for both models (Figure 4.6). Overall, many morphological features extracted from the generated images showed

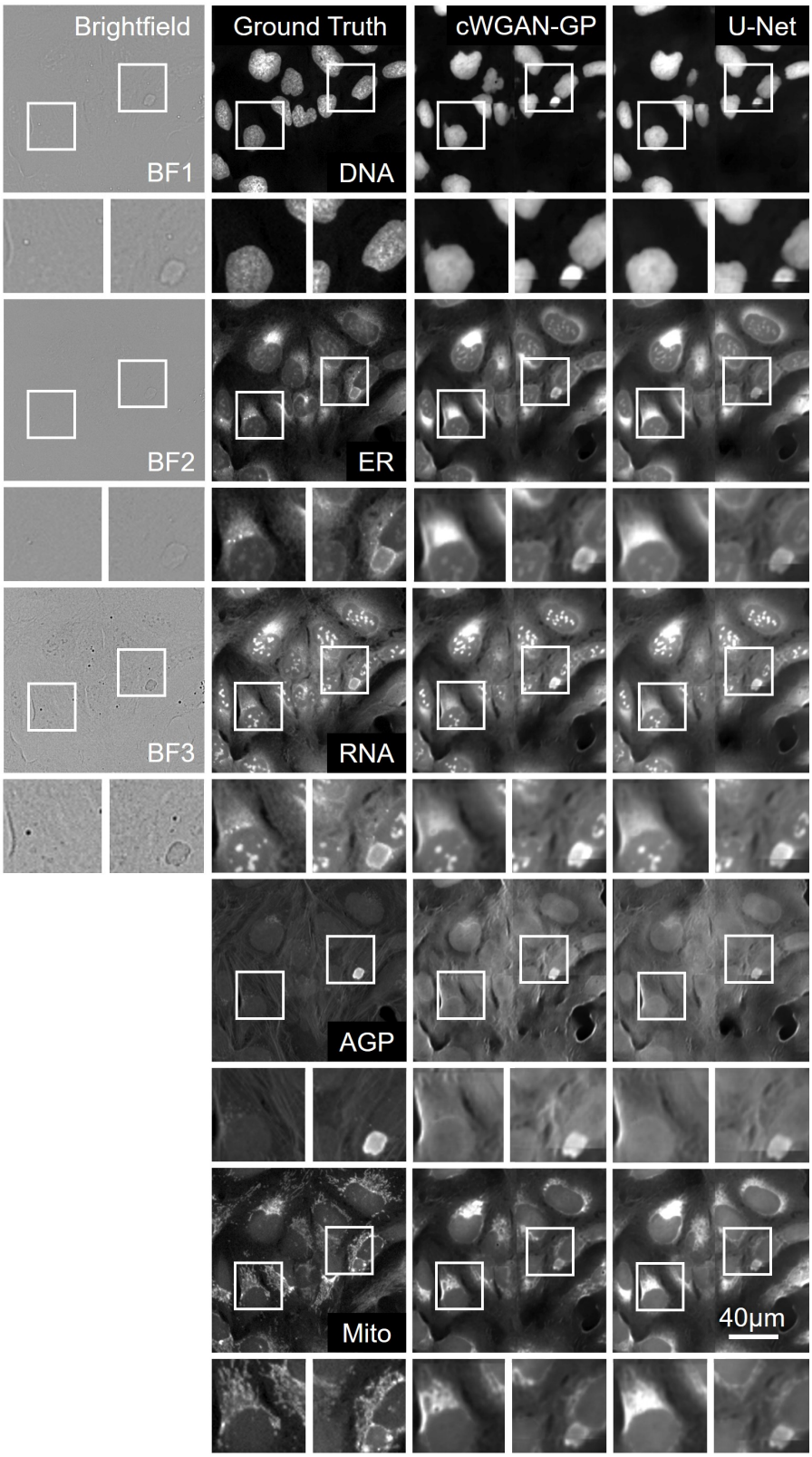


Fig. 4.5 Two small outsets for each image show a magnified view of the selected region of the cell, and can be used to compare details between the ground truth and the two model outputs in each channel. Images are independently contrasted for visualization.

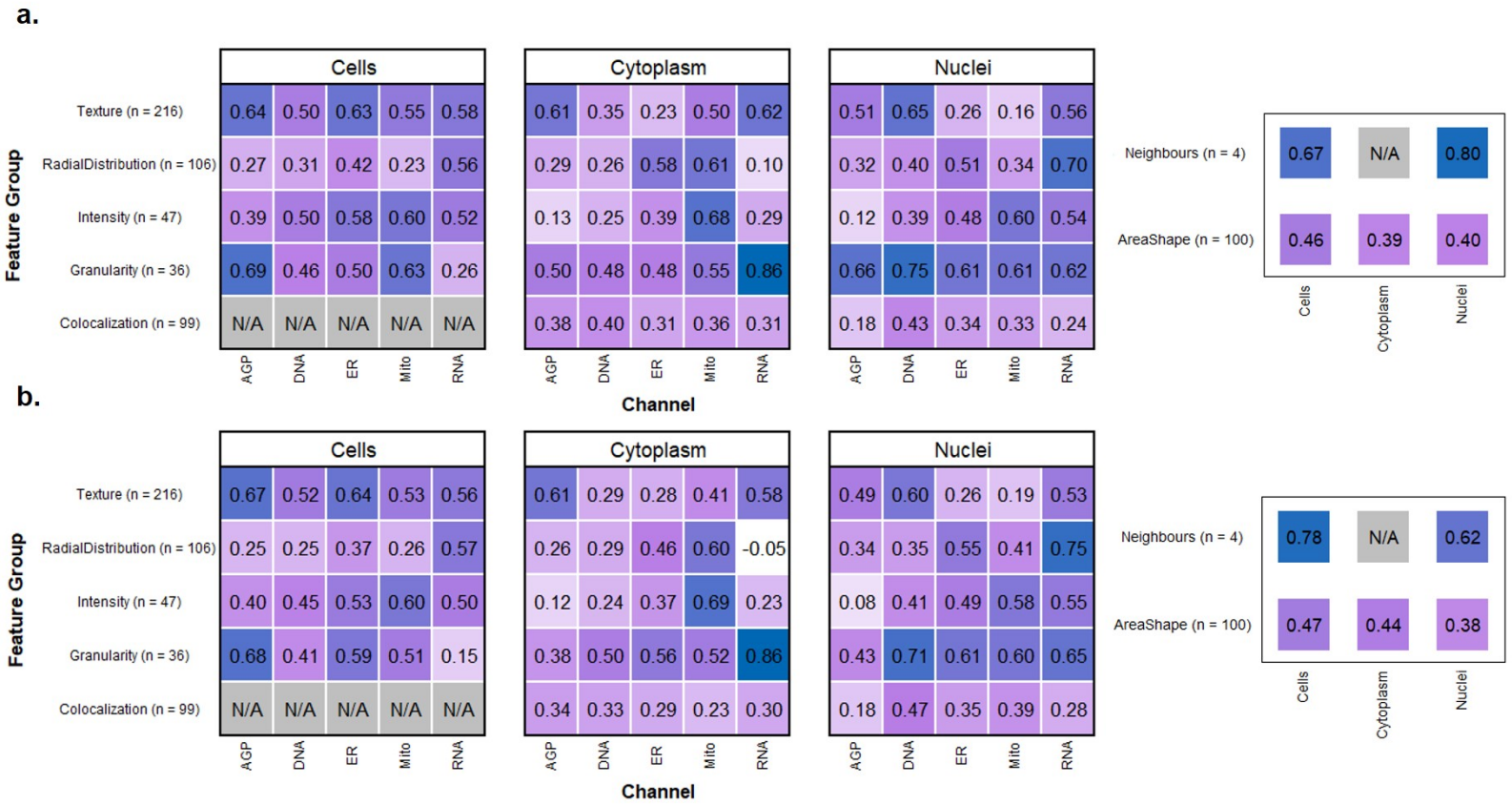


Fig. 4.6 a systematic breakdown of correlation between morphological profiles. The features selected from the CellProfiler pipeline are presented as correlation heatmaps in 5.a (U-Net) and 5.b (cWGAN-GP). The correlations for each model are with the features extracted from the ground truth Cell Painting images, and are aggregated by channel and feature group for cell, cytoplasm, and nuclei objects. The number of features for each feature group is also presented. U-Net mean = 0.43, cWGAN-GP mean = 0.45.

substantial correlation with those extracted from ground truth images. Examples of accurately reproduced (>0.6 correlation) feature groups across both models were texture measurements of the AGP channel in both cells and cytoplasm; intensity measurements of the Mito channel in the cytoplasm, and granularity and texture measurements of the DNA channel within the nuclei. The highest performing feature group were granularity measurements of the RNA channel within the cytoplasm (0.86 correlation in both models). Almost all features correlated positively with the ground truth, and only a small number of features showed close to zero or negative correlation, such as the radial distribution of RNA in cytoplasm and the intensity of the AGP channel in nuclei. The cell colocalization features were not calculated by CellProfiler however the cytoplasm and nuclei colocalization features represent the cell as a whole.

The mean correlation of all the feature groups was 0.43 for the U-Net and 0.45 for cWGAN-GP, supporting the earlier finding of cWGAN-GP's superiority in the image-level evaluation. The feature breakdown by cell group was: 169 cell features (mean correlation: 0.48/**0.50**), 217 nuclei features (**0.43**/0.41), and 225 cytoplasm features (0.38/**0.42**). The feature breakdown by feature type was, in descending order of best to worst mean correlation: 4 neighbors (0.70/**0.74**), 36 granularity (0.53/**0.55**), 216 texture (0.47/**0.49**), 47 intensity (0.46/**0.48**), 103 area/shape (0.42/**0.43**), 106 radial distribution (0.38/**0.39**) and 99 colocalization (0.31/**0.33**). By channel, the mean feature correlations were: RNA: 0.47/**0.49**, AGP: 0.43/**0.44**, DNA: 0.41/**0.44**, Mito: 0.40/**0.43**, ER: **0.4**2/0.40.

The mean correlations for the top 10% of the selected features were 0.80/**0.81**. The mean correlations for the top 50% of the selected features were 0.64/**0.65**. The number of features with a correlation greater than 0.8 were 26/**30** (U-Net/**cWGAN-GP**), and for both models all feature groups and cell compartments had at least one feature with such a strong correlation, except for the colocalization feature group. For the cWGAN-CP model the breakdown of the 30 features with greater than 0.8 correlation was as follows: 12 cell, 11 nuclei, 7 cytoplasm, and this included the feature types: 11 texture, 7 radial distribution, 5 area/shape, 4 granularity, 2 texture, 1 neighbors, 0 colocalization.

Using uniform manifold approximation (UMAP), we visualized the high-dimensional morphological profiles for identification of underlying data structures (Figure 4.7). We observed that all three image sources (ground truth, U-Net, and cWGAN-GP) were separated in the common feature space due to the difference in the ground truth and predicted images. Despite this separation, we also observed that all the image sources maintained the overall data structure. The clear batch effect visible in the ground truth

is also evident in the predicted images, and similarly the clustering of positive and negative control wells respectively is retained, indicating successful model performance. Features extracted from the cWGAN-GP model lie closer to the ground truth features than the U-Net extracted features, yet features from both models are closer to each other than either are to the ground truth.

4.3.3 Profile-level evaluation

Conclusions made from the label-free prediction of Cell Painting images should show agreement with ground truth datasets to be of experimental value. We therefore performed a series of analyses to identify compounds that elicit a comparable toxic phenotype to an established positive control compound, mitoxantrone. Our UMAP (Figure 4.7) showed that, as would be expected, most of the compounds showed greater resemblance to negative (DMSO) controls. Promisingly, some compounds clustered with our positive control compound. To identify these compounds, we trained a K-NN classifier using the control profiles as our training set. The classification resulted in identification of eight compounds in the ground truth, five in the cWGAN-GP model, and eight in the U-Net model profiles. The U-Net model achieved a sensitivity of 62.5% and specificity of 98.0% in toxicity classification whilst the respective values for cWGAN-GP were 50.0% and 99.3%.

4.4 Discussion

As the first full prediction of five-channel Cell Painting from brightfield input, we have presented evidence that label-free Cell Painting is a promising and practicable approach. We show our model can perform well in a typical downstream application, and that many label-free features from the predicted images correlate well with features from the stained images. In addition, we see success in clustering by treatment type from the features. We present indications of which channels and biomarkers can be satisfactorily predicted by testing our model predictions with a comprehensive segmentation-based feature extraction profiling methodology.

Training with an adversarial scheme using a conditional GAN approach has been shown to enhance performance in virtual staining tasks [203]. In preliminary experiments, we tested different GAN models and adversarial weightings, but we chose cWGAN-GP due to its stability of training on our dataset. Increasing the adversarial component of the training objective function resulting in undesirable artifacts in the

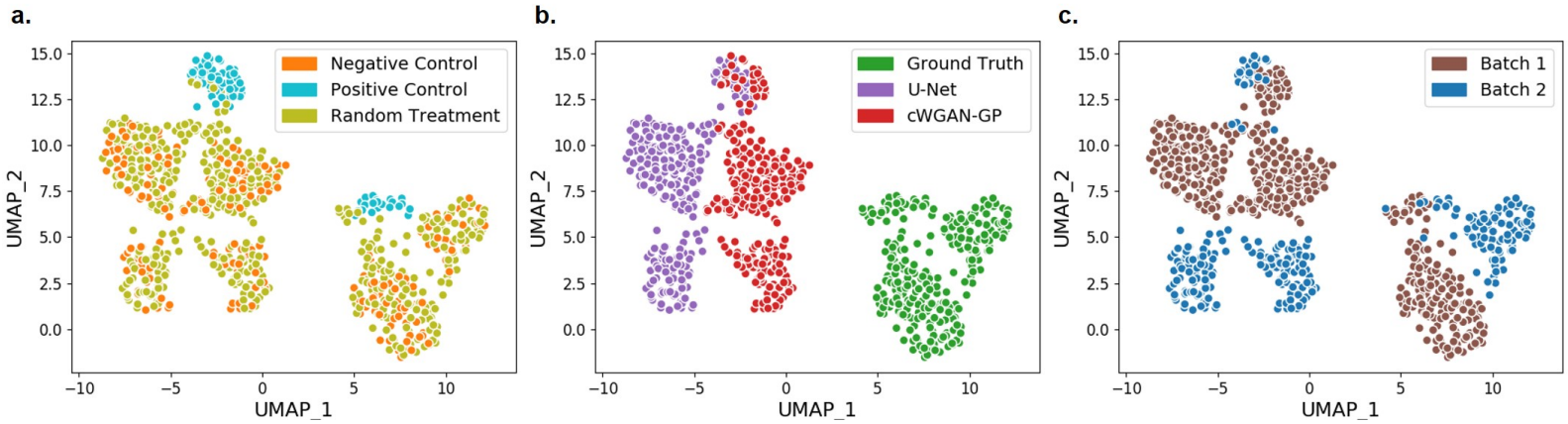


Fig. 4.7 The UMAPs demonstrate that both models could reproduce the separation between treatments and batches seen in the ground truth features and that the clustering patterns are very similar. **(a)** shows each test image labelled by treatment type, **(b)** shows ground truth vs the two models and **(c)** highlights the difference between the two batches. cWGAN-GP was closer to the ground truth than U-Net, although the two models sit much closer to each other in feature space than either to the ground truth.

images which were not true representations of the cells. Our results demonstrate that incorporating adversarial loss results in a small increase in performance over L1 loss based on the pixel-wise evaluation metrics in all channels except ER. Even though the difference in metric values are mostly within one standard deviation, the values are calculated for images generated from the same set of brightfield images (for each model), so it is meaningful that cWGAN-GP achieves superior performance for 18 out of 25 metrics across the channels (Table 4.2). In image-to-image problems, the finer details in some of the predicted channels can be obstructed and blurred with a pixel-wise loss function such as L1 loss [204]. Just a small performance difference is expected as the network and loss function for training both models were very similar, and in the cWGAN-GP model the adversarial component of loss is weighted relatively low compared to L1.

In addition to the metric evaluation, features from images from the cWGAN-GP model show an increase in performance over the U-Net model, with slightly higher mean correlations to the ground truth (Figure 4.6). The strong correlation of specific feature groups increases confidence that extracted morphological feature data from predicted images can be used to contribute overall morphological profiles of perturbations.

Biologically, it is expected that correlations of feature groups within the DNA channel are higher within the nuclei compartment than either cytoplasm or cells since the nuclei compartment is morphologically very distinct from the cytoplasmic region. The high correlation of radial distribution features in the RNA channel suggests that successful visualization of the nucleoli within the nuclear compartment has a large effect on this particular feature group. AGP and Mito channels both contain small and subtle cellular substructures which are typically less than two pixels wide. We postulate that this fine scale information is not present in the brightfield image in our data, making accurate image reproduction of the AGP and Mito channels very challenging regardless of model choice. There are known limitations of brightfield imaging which will always be restrictive. Brightfield images can display heterogeneous intensity levels and poor contrast. Segmentation algorithms have been proven to perform poorly when compared with fluorescence channels, even after illumination correction methods have been applied [163].

In UMAP feature space (Figure 4.7), as well as in the PCA analysis (Supp Figure C), the features extracted from model-predicted images do not overlap with the ground truth features. This highlights limitations of the model but also the challenge of batch effect – it is not expected for the model to exactly predict an unseen batch. The relevant structure of the data is maintained although not absolute values. This is also

notable at the image level, for example the high MSE is likely due to the batch effect causing a systematic difference in pixel values between the training and test batches. Similarity metrics such as SSIM may be more informative in this instance. It is notable that ground truth features from different batches also sit in non-overlapping feature space in our UMAPs. Within batches, the negative controls and treatments would form sub-clusters depending on the treatments in a larger dataset, however we acknowledge our test set was relatively small, resulting in minimal sub-clustering.

Other studies [134] which have used U-Net based models to predict fluorescence from transmitted light or brightfield have evaluated their performance on pixel-level metrics such as PCC [161], SSIM [186, 188] and PSNR [205, 136]. The mean PCC of all channels in our test set is 0.85 (using the best model for each channel), a value which compares favorably with prevailing work in fluorescent staining prediction [35]. In our data two channels (DNA, RNA) exceed a PCC of 0.90 for both models. However, absolute values of image metrics are heavily data dependent so we present these metrics primarily for model comparison.

Such metrics are standard in image analysis but have some limitations for cellular data. Treating each pixel in the image equally is a significant limitation as pixels representing the cellular structures are clearly more important than background (void) pixels [205]. Some channels such as the DNA channel are more sparse than other channels such as AGP, and as such the number of pixels of interest vs background pixels is different. Extracting features from scientific pipelines provides a more biologically relevant and objective evaluation to give deep learning methods more credibility and a greater chance of being practically employed in this field [206].

Despite the small sample size, the models performed well at predicting compounds with phenotypes similar to that of the positive control compound mitoxantrone, with specificities of 98.0% (U-Net) and 99.3% (cWGAN-GP). The sensitivities were 62.5% and 50% respectively. Lower sensitivity values suggest the models are not identifying some potentially toxic or active profiles. There are multiple morphological features present in ground truth images which are linked to toxicity and it is likely that the models cannot capture all of these accurately. The manifestation of cytotoxicity can be most prevalent in the Mito/AGP channels due to mitochondrial dysfunction and gross changes to cytoskeletal processes and structure. We report that the Mito and AGP channels were the least well predicted channels on similarity and correlation metrics (Table 4.2), thus could serve as a reason for loss of sensitivity. Future studies could focus on predicting these two channels in particular. It is also relevant to consider that the concentration of mitoxantrone used was deliberately chosen to elicit a milder

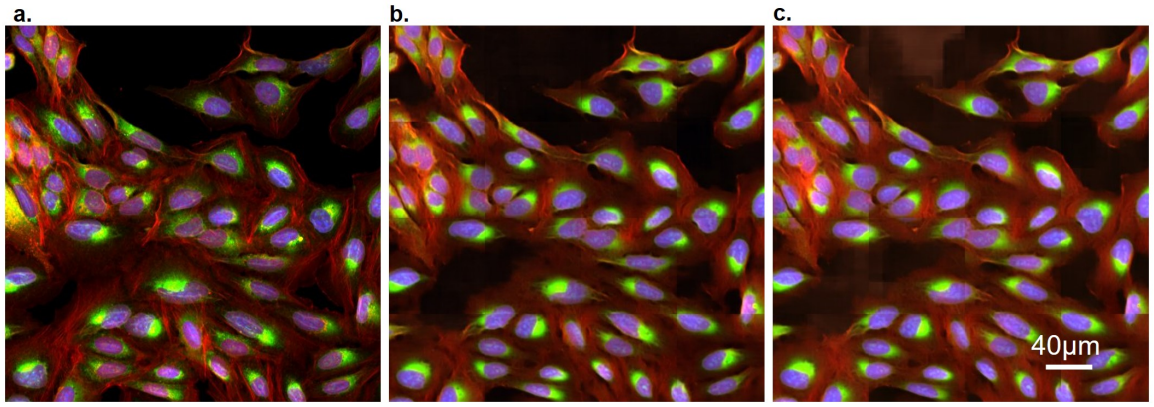


Fig. 4.8 Colored composite image of three channels from the test dataset (Table 1): AGP (red), ER (green) and DNA (blue). a) Ground truth. b) cWGAN-GP model output. c) U-Net model output. Each image is 512x512 pixels.

cytotoxic phenotype with an enlarged cellular area (suggesting a cytostatic effect is occurring) rather than complete cell death and an absence of cells entirely. Very high specificities indicate that label-free Cell Painting is not introducing prediction errors which would lead to false positive identification of a mitoxantrone-like phenotype. This is expected for models with significantly more weight on L1 loss, rather than with a high adversarial weighting which could introduce artifacts or phantom structures into the predicted images.

Despite a lack of sensitivity, we have demonstrated that morphological features extracted from both models are capable of recapitulating a significant portion of the morphological feature space to result in a positive clustering to the chosen control compound, mitoxantrone. We disclose two compounds which are already in the public domain; glipizide, a clinically-used sulfonylurea compound for the treatment of type 2 diabetes, and GW-842470, a phosphodiesterase 4 inhibitor previously evaluated in clinical studies for the treatment of atopic dermatitis (discontinued).

Although comparative or superior results for phenotype classification could likely be achieved with an image classifier trained on the brightfield (see later study done after this work [53]), as a label-free and image-to-image approach there is a lot of promise as a generalist model to perform multiple tasks. The simple visualization of Cell Painting channels is one approach to improve the interpretation of brightfield images. Our results highlight the rich information captured in the brightfield modality, which is currently under-utilized in morphological profiling.

Our approach by itself cannot replace full fluorescent staining, but we have provided evidence that it may be possible to replicate the information of some Cell Painting

channels and feature groups, and that the brightfield modality by itself may be sufficient for certain experimental applications. Importantly, employing such methods may reduce time, experimental cost and enable the utilization of specific imaging channels for experiment-specific, non-generic fluorescent stains. We acknowledge that particular feature groups which predict poorly in our models (such as colocalization features) may result in an inability to identify cellular phenotypes which are characterized fully by these features. In such situations, the replacement of generic stains for phenotype- or target-relevant biomarkers may offer an effective solution to the standard Cell Painting protocol.

One limitation of this study is the dataset used. Future studies will require larger datasets with greater diversity in terms of compound treatments and collection sources (see Chapter 6). Matching the numbers of fields in our training and test sets to the size of a typical dataset used in drug discovery would allow for greater insight into the capabilities of label-free Cell Painting. International collaborations such as the Joint Undertaking of Morphological Profiling using Cell Painting (JUMP-CP) aim to further develop Cell Painting and provide a highly valuable public dataset for this use [34]. It is notable that we have evaluated predictions with downstream features the networks have not seen – simply extracted classically from the images. In future studies, incorporating metadata information in the training of the network itself may improve performance. We investigate this in the next two chapters

We also acknowledge that the brightfield modality may restrict the quantity and quality of information being input into the models. The rationale behind imaging at multiple focal planes in brightfield configuration is to visualize as much cellular substructure as possible. Not all cellular features are visible in a single focal plane, therefore taking information from a z-stack image set will increase the input content to the models. Alternative brightfield imaging modalities such as phase contrast microscopy or differential interference contrast (DIC) microscopy have previously been used for fluorescence channel prediction tasks [55, 207] and can capture a wealth of cellular morphology. It was beyond the scope of this study to investigate the impact of using different brightfield imaging modalities on predictive model performance, yet it should be recognized that all brightfield approaches lack the ability to fully visualize small-scale cellular substructures to an extent [191]. We propose that brightfield imaging in a z-stack configuration serves as a typical process that is widely adopted across both academic and industry laboratories, therefore the methods we present are applicable to a range of instruments with varied imaging capabilities.

In summary, we propose a deep learning approach which can predict Cell Painting channels without the application of fluorescent stains, using only the brightfield imaging modality as an input. Building upon previous work [35, 55] we have predicted the five fluorescent channels and used these images to predict the associated groups of morphological features from a standard image analysis pipeline. We then used the features from the predicted images to assess how information-rich such images are. Finally we have provided a critical evaluation of the predictions using morphological features extracted from images using CellProfiler analysis and the resulting compound profiles. In Chapter 6 we further develop the ideas presented in this chapter, and significantly advance label-free Cell Painting.

The work presented in this chapter was released as a preprint in 2021 and accepted for publication by *Nature Scientific Reports* in 2022. Thank you to the co-authors of the publication.

Cross-Zamirski, J.O., Mouchet, E., Williams, G. Schönlieb, C.B, Turkki, R. and Wang, Y. Label-free prediction of cell painting from brightfield images. *Sci Rep* **12**, 10001 (2022). https://doi.org/10.1038/s41598-022-12914-x

Chapter 5

Self-Supervised Learning of Phenotypic Representations with Weak Labels

In this chapter we introduce WS-DINO, the first self-supervised framework to use weak label information in learning phenotypic representations from high-content fluorescent microscopy images of cells. Our novel method is based on a knowledge distillation approach with a vision transformer backbone (DINO - self-**di**stillation with **no** labels), and we use this as a benchmark model for our study. Using WS-DINO, we fine-tuned with weak label metadata freely available in high-content microscopy screens (treatment and compound), and achieve state-of-the-art performance in not-same-compound mechanism of action prediction on the BBBC021 dataset (98%), and not-same-compound-and-batch performance (96%) using the compound as the weak label. Our method bypasses single cell cropping as a pre-processing step, and using self-attention maps we show that the model learns biologically and structurally meaningful phenotypic profiles.

5.1 Introduction

We have used classical methods to extract feature profiles from (real and predicted) Cell Painting images in Chapter 4. In this chapter we focus developing novel machine learning models to profile high content images. Deep learning methods for phenotyping cells from microscopy images have advanced rapidly alongside cellular imaging technology [208]. Profiles captured from images can be used to quantify activity and perform downstream tasks such as mechanism of action (MOA) prediction of cells which have been treated with compounds of interest. These methods hold promise for accelerating

drug discovery pipelines by overcoming one of the major bottlenecks in drug discovery: identifying a compounds mechanism of action (MOA) and off-target activities [9, 209].

Typically, convolutional neural networks (CNNs - Section 3.1.1) have been the method of choice, however these models can struggle to generalise to new treatments, and can be limited by label information [88]. Rarely are they interpretable, and this may contribute to how machine learning methods have not reached their potential in clinical use [210]. Across medical imaging fields, vision transformers (ViTs) [133] and self-supervised [93] methods have risen as a viable alternative to CNNs with labels [95, 146].

In this study we make the following contributions:

- We introduce Weakly Supervised DINO (WS-DINO), a framework to incorporate weak label information into a self-supervised knowledge distillation construction based on the DINO algorithm [96].
- We implement WS-DINO, and with the BBBC021 dataset [38, 39] we learn representations with two weak labels - treatment and compound. We achieve state-of-the-art results for not-same-compound (98%), and not-same-compound-and-batch (96%) mechanism of action prediction using the compound as the weak label. We show the learnt representations are biologically meaningful and based on cellular structure with self-attention maps.

Additionally, we show that WS-DINO is easily adaptable to the case where strong label information is available for training (i.e. the mechanism of the compound is known). Using MOA labels we achieve exceptional performance on BBBC021 and suggest our approach has promise on larger, real world high-content imaging datasets in drug discovery where it is highly likely at least some known MOA information is available in training

5.2 Background

5.2.1 DINO - self-supervised learning with knowledge distillation

DINO (self-**di**stillation with **no** labels) from Caron *et al.* [96] is a self-supervised learning approach which incorporates aspects of knowledge distillation - the process of transferring knowledge from a larger model (teacher) to a smaller one (student) [211].

In knowledge distillation, the teacher model will typically output probability distributions (soft targets) using a softmax function (Section 3.1.3) which are more diffuse and hence allow the student model to capture the teacher network's uncertainty and knowledge about the relationships between different class labels (hard targets). In training, the student network attempts to match the soft targets while also learning from the class labels (ground truth). Once the student model is trained using knowledge distillation, it can be used for inference with a lower temperature parameter - the value in the exponent of the softmax function which controls the sharpness of the output distribution - to produce more precise and confident predictions. The student network can achieve comparable or superior performance to the teacher.

The DINO framework extended knowledge distillation to the case where no class labels are available for training. The authors cast knowledge distillation as a self-supervised (Section 3.2.2) objective. Uniquely combining self-supervised learning with a Vision Transformer (ViT) [133] (Section 3.1.4) backbone, DINO was trained to learn features shown to perform well at k-NN clustering tasks. Self-supervised training with DINO achieved 78.3% top-1 on ImageNet [102] with a small ViT (8×8 pixel patches).

We chose DINO as the basis for this study for its ability to learn meaningful representations with strong clustering properties, in addition to the output features containing semantic segmentation information. The multiscale approach which reveals these segmentation properties allowed us to bypasses single-cell segmentation as a pre-processing step. The ViT backbone allows for the visualisation of self-attention to reveal segmentation properties and structural weightings.

5.2.2 The DINO model

DINO, summarised in Fig. 5.1, consists of student network g_{θ_s}, parameterized by θ_s, trained with set of image crops to match the teacher network g_{θ_t}, parameterized by θ_t, which sees a different set of crops.

For each image x a set of views V is generated which contains the two global crops $x^{g,1}$ and $x^{g,2}$ as well as eight local crops. The student is passed the set of global and local crops V, while the teacher sees only the global crops. The student network is trained to maximise the agreement between the outputs of g_{θ_s} and g_{θ_t}. To achieve this, the parameters of θ_s are found by minimizing the cross-entropy loss:

$$\min_{\theta_s} \sum_{x \in \{x^{g,1}, x^{g,2}\}} \sum_{\substack{x' \in V \\ x' \neq x}} H(P_t(x), P_s(x')) \tag{5.1}$$

where $H(a, b) = -a \log b$ and P_s and P_t are the probability distributions of the student and teacher respectively, defined by:

$$P_s(x)^{(i)} = \frac{\exp\left(g_s(x)^{(i)}/\tau_s\right)}{\sum_{k=1}^{K} \exp\left(g_s(x)^{(k)}/\tau_s\right)} \tag{5.2}$$

for K dimensions and temperature parameters $\tau_s > 0$ and $\tau_t > 0$ [96]. Teacher weights θ_t are frozen during each epoch of student training and updated iteratively with an exponential moving average based on the previous weights of the student network with the formula: $\theta_t \leftarrow \lambda\theta_t + (1 - \lambda)\theta_s$, where λ is the momentum parameter.

Both the student and the teacher network have a ViT [133] backbone (Section 3.1.4). The inputs are small patches of fixed size ($N \times N$) in a non-overlapping grid, which are embedded (alongside a positional embedding) with a linear layer. An additional learnable [CLS] (class) token (in this case not related to any class label) with a projection head output h is added to the embeddings and sent to the transformer encoder. The attention mechanism (Section 3.1.3) allows the ViT to synthesise information across the whole image as self-attention layers in the transformer globally update the attention token embeddings.

DINO directly predicts the output of the teacher network with a standard cross-entropy loss. This is achieved with a momentum encoder [212]. Through an ablation analysis in the original paper [96] it is shown that this momentum encoder is vital for strong performance. Additionally, centering and sharpening in the teacher network are implemented to avoid collapse. Centering stops one single dimension from overwhelming the output by encouraging collapse to the uniform distribution. Sharpening is applied simultaneously to balance centering which prevents collapse in presence of the momentum teacher.

This approach sacrifices stability to reduce reliance on batch statistics, as the centering operation relies solely on first-order batch statistics. This is equivalent to introducing a bias term c to the teacher network, denoted as $g_t(x) \leftarrow g_t(x) + c$. The value of the center c is updated using an exponential moving average (ema), which leads to effective performance for different batch sizes. The update equation for c is given by:

$$c \leftarrow mc + (1 - m)\frac{1}{B}\sum_{i=1}^{B} g_{\theta_t}(x_i) \tag{5.3}$$

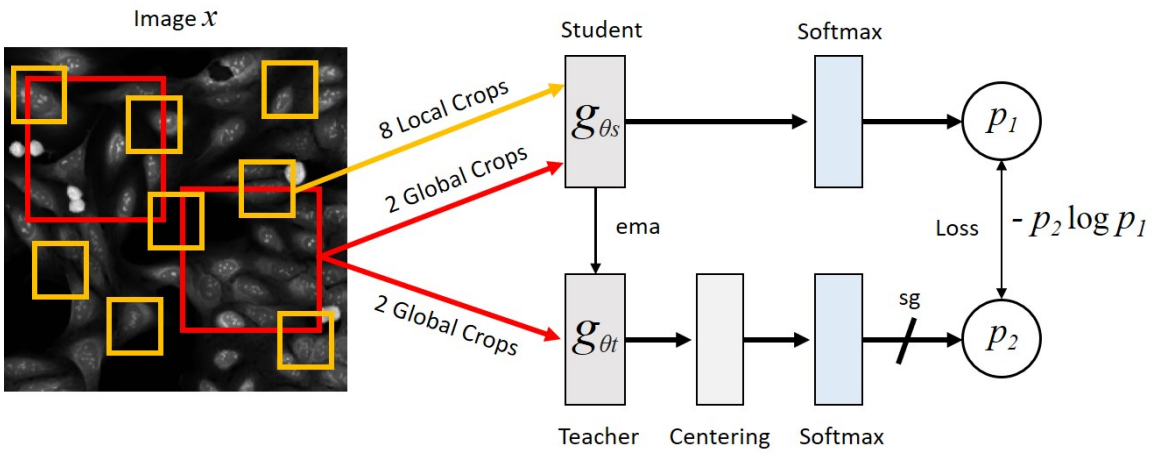


Fig. 5.1 A summary of self-distillation with no labels (DINO). Two randomly sampled global crops are shown in red, and eight randomly sampled local crops in yellow. The student network is fed all the crops, and the teacher is fed just the global crops. "sg" is the stop-gradient operator to prevent gradients propagating through the teacher, which is updated with the exponential moving average (ema) of the student weights. Adapted from Caron *et al.* [96].

where $m > 0$ is a rate parameter and B represents the batch size. The sharpening of model outputs is achieved by choosing a low value for the temperature τ_t in the softmax normalization of the teacher [96].

Network - ViT Small

The student and teacher networks $g = h \circ f$ consist of two components: the backbone f and projection head h. The projection head consists of a 3-layer multi-layer perceptron (MLP) with hidden dimension 2048 followed by L_2 normalization and a weight normalized fully connected layer with 384 dimensions.

While the DINO training scheme is compatible with many networks as the backbone f such as ResNet [132], the small ViT produced the best k-NN clustering results on ImageNet. In addition, ViTs are desirable for the segmentation and interpretability properties revealed in self-attention.

Hence, all the models trained in this chapter used the ViT small (ViT-S/8) network with the following parameters: number of transformer blocks: 12, channel dimension: 384, number of multi-head attention heads: 6, length of the token sequence: 785, total number of parameters (without counting the projection head): 21 million, and patch size: 8×8.

5.3 Methods

5.3.1 Dataset

The dataset used in this chapter is available at: https://bbbc.broadinstitute.org/BBBC021. The publicly available BBBC021 dataset [38] consists of MCF-7 breast cancer cells exposed to several chemical compounds for 24h. The cells are stained with three fluorescent labels: DNA, F-actin, and β-tubulin. In total there are 39,600 images (13,200 fields of view imaged as three channels), provided in TIFF format.

5.3.2 WS-DINO

While DINO has been effective in medical imaging [97], there are a few drawbacks for image-based profiling. Fine-tuning in an entirely self-supervised manner may be strongly influenced by the batch effect. One way to correct for the batch effect is to use batch information in the batch normalisation layers of a CNN [104]. However, ViT architectures do not use batch normalisation in the projection heads.

Hence, we were interested in developing a new method to incorporate metadata (Section 2.4.3) such as weak labels [77] (Section 3.2.1) which can be effective at both correcting for experimental batch effects and learning enhanced phenotypic representations. There is evidence for using metadata in self-supervised frameworks to improve representations in parallel fields such as medical imaging [213]. This chapter presents the first study with DINO for image-based profiling.

We propose a **W**eakly **S**upervised form of self-**di**stillation with **no** labels (WS-DINO) as an adaptation to the DINO algorithm. The aim of the approach is to retain the benefits of self-supervised learning with knowledge distillation while also incorporating the weak label information as part of the training.

We introduce the notation x_{i,y_i} to represent the i^{th} field of view of a fluorescent channel in the dataset with the weak label y_i - the treatment or compound. The superscript contains the crop information: g for global and l for local crops. When generating the sets of different views V_t (seen by teacher) and V_s (seen by student) for training, we enforce the constraint that the global and local crops are sampled from different images with the same weak label. We define the sets V_t and V_s for the randomly sampled ordered pair (i, j) where $i \neq j$ and $y_i = y_j$:

$$V_t = \{x^{g,1}_{i,y_i}, x^{g,2}_{i,y_i}\} \tag{5.4}$$

$$V_s = V_t \left\{ x^{l,k}_{j,y_j} : k \in \{1, ..., n\} \right\} \tag{5.5}$$

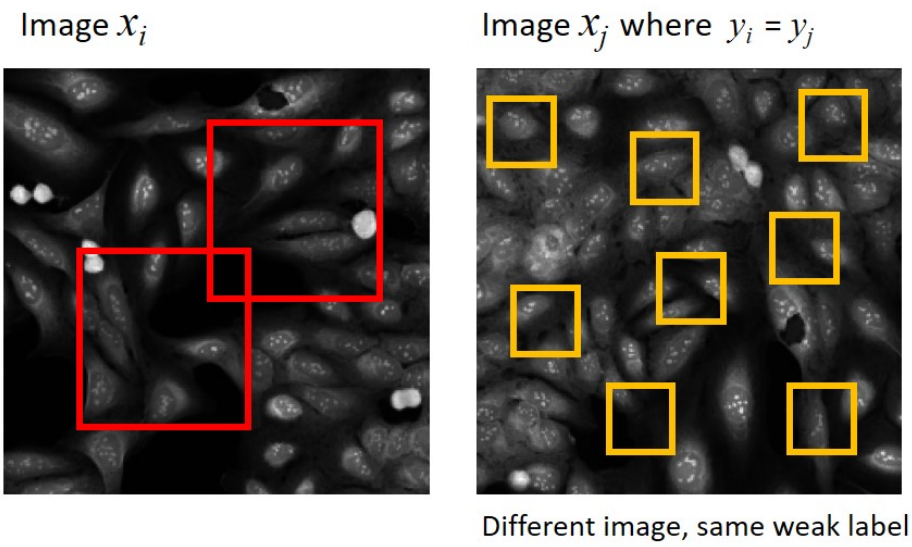


Fig. 5.2 WS-DINO sampling. Two global crops (red) are taken from an image x_i, and eight local crops are taken from a different image x_j with the same weak label $y_j = y_i$. This allows features to be learned for cells with the same treatment or compound across multiple plates, wells or experimental batches.

where the superscript details the k^{th} local crop of n total crops (default: $n = 8$). Sampling the local crops from a different image is in contrast to sampling random crops with different augmentations from the same image, which is the method of DINO. For WS-DINO we minimize the loss:

$$\min_{\theta_s} \sum_{x \in V_t} \sum_{\substack{x' \in V_s \\ x' \neq x}} H(P_t(x), P_s(x')) \tag{5.6}$$

where P is defined in Eqn. 5.1, θ_s parameterizes the student network and $H(a, b) = -a \log b$, as in the original construction. In this section we have presented the key details of our adaptation to the DINO algorithm from the original paper [96].

5.3.3 Data and augmentation

For training and evaluation, we used the labelled subset of the BBBC021 dataset [38], typically used for MOA evaluation [39]. This subset consists of 12 unique MOA, 38 unique compounds and 103 unique treatments (compound/concentration pairs) across 10 experimental batches. There are 1320 3-channel images in the dimethyl sulfoxide (DMSO) treated control group, and 2528 3-channel images with MOA annotations.

Several studies have used the annotated dataset to evaluate classical and machine learning strategies [39, 72, 73, 77, 98, 76, 75, 87, 78, 92].

Pre-processing

To remove systematic variations in pixel intensities across each field of view, we used a CellProfiler [43] pipeline to perform an illumination correction. A smoothing function with a filter size of 320 pixels was used to create an illumination correction function. Each image was corrected by dividing all pixel intensities by the illumination correction function. This methodology is consistent with best practice established by the JUMP Cell Painting Consortium [63]. After correction, images were resized using bicubic interpolation to a size of 640×512 pixels. A maximum pixel value cut-off of $10,000$ (to remove outlier values many standard deviations away from the mean) was enforced before normalizing each image to have a mean of 0 and standard deviation of 1. The scripts to replicate the pre-processing are available in our GitHub repository (Appendix).

Post-processing

Next we take the median feature embedding from four 224×224 crops around the centre of each image (equivalent to a 448×448 pixel centre crop split into four non-overlapping crops). To aid in correcting for batch effects (which can be significant in HCI screens) and to capture the range of variation of unperturbed cells, we applied typical variation normalization (TVN) [73] as a post-processing correction. TVN is a technique where a principal component analysis (PCA) transformation without dimensionality reduction (whitening) is learned using the DMSO embeddings (aggregated to one embedding per image from the median of four crops around the centre). The mean of each of the field-level embeddings is taken across a plate, followed by median aggregation to treatment level resulting in 103 embeddings - in line with the studies following Ando *et al.* [73].

5.3.4 Performance validation and evaluation

We evaluated the corrected and aggregated feature embeddings with two metrics ubiquitous in representation learning studies using the BBBC021 dataset: not-same-compound (NSC) matching [39] and not-same-compound-and-batch (NSCB) matching [73] with the MOA labels. NSC matching is a 1-Nearest-Neighbour (1-NN) match for each given treatment to the nearest neighbour in representation space which is **not**

of the same compound. Cosine distance was used as the distance measure. NSCB matching is the same method as NSC matching with the additional constraint to exclude treatments from the same compound **and** batch as the given treatment. A small number of treatments have a MOA label only present in a single batch and these were excluded in NSCB evaluation.

5.3.5 Training

For all the models the following variables are kept constant: network architecture (ViT-S/8 backbone with 3-layer multi-layer perceptron head), size and number of crops: 2 global 224×224 random resize crops (range $= 0.1 - 0.2$ of full size image) and 8 local 96×96 random resize crops (range $= 0.04 - 0.08$), teacher momentum $= 0.99$, and batch size $= 16$. There was no batch normalization, weight decay or gradient clipping. The optimizer was adamW [214]. We applied horizontal and vertical flips (with a probability of 0.5 per sample) as augmentations in training. Learning rate and temperature parameters are controlled with adaptive cosine schedules with linear warm-up. After 10 epochs of linear warmup, the learning rate $= 4 \times 10^{-6}$ and the teacher temperature $= 0.04$. The learning rate was decayed to 3×10^{-6} over 400 epochs, and the teacher temperature kept constant after warmup. All other parameters unless otherwise stated are default DINO parameters.

A separate model was trained for each of of the three channels. Sometimes for 3 fluorescent channels, RGB weights are used (e.g. R = DNA, B = β-tubulin etc.) however we found this to give significantly worse performance. There isn't any reason that RGB channels should correspond to (structurally different) fluorescent channels. Additionally, training a separte model for each channel makes our method easily adaptable to any number of channels (for example 5 Cell Painting channels). Every model was initialized with weights from DINO trained on RGB ImageNet [96] [102], with each grayscale image channel being converted to RGB. Embeddings of size 384 were extracted for each channel then concatenated to a size of 1152, followed by L2 normalization. We provide a full WS-DINO PyTorch implementation in our GitHub repository (Appendix).

Note on training and evaluation - transductive learning

We trained **and** evaluated our models on the annotated subset of the BBBC021 dataset which is in line with other unsupervised studies on this dataset. This is called

transductive learning [215] and is justifiable as none of the models had access to the MOA labels in training.

This does raise an interesting point about self-supervised learning in this context. Is it permissible to train and test on the same dataset? It seems to be a sin of machine learning, however we believe transductive learning with weak labels is valid in this setting. This is best demonstrated with an example scenario:

Imagine we have a new dataset from a high-content screen with unknown MOA/target relationships. We wish to use a model to learn features for clustering compounds and making predictions. We have two options:

1. We could use a model trained or fine-tuned on another dataset. This would be analogous to the train/test scenario in standard supervised learning (data splitting).

2. Since the weak labels are free information (i.e. always paired to the images in high-content screens), and the images we want to profile are also available as inputs, we can fine-tune the model on the entire new dataset we want to cluster (transductive learning). This will result in superior performance, and enables the model to learn cross-batch features in the dataset of interest.

The only reason to choose option 1. over option 2. is to save computational resource. Given WS-DINO can be fine-tuned to convergence on a single GPU in less than 24 hours, we do not consider this too costly - it is not significantly more resource than using a CellProfiler pipeline (which is also less automated). Furthermore, we propose simply using publicly available ImageNet weights if saving computational resource is of concern. Hence, the two most useful models for this setting are either transfer learning with weights from an enormous dataset, which should generalise well enough to unseen data, or to fine-tune transductively.

Further training details

The hyperparameters are fixed across all of the models we train. Many of the parameters are the default values from the original DINO implementation. We optimised the remaining parameters such as learning rate with a small subset of BBBC021 trained without weak labels. In our final models the batch size is limited by GPU memory - a batch size of 16 is small and most likely a larger batch size would improve performance. The models were trained with two GPUs on the AstraZeneca Scientific Computing Platform with a maximum allocation of 32G memory for each GPU.

Model	Weak Label	Best NSC/NSCB	Mean NSC/NSCB
DINO (finetuning)	None	92% / 95%	90% / 90%
WS-DINO (finetuning)	Treatment	92% / 90%	89% / 83%
	Compound	**98% / 96%**	96% / 93%
	MOA	100% / 100%	99% / 99%

Table 5.1 The results of training with NSC and NSCB scores of the best performing epoch as well as the mean NSC and NSCB scores between 50 and 250 epochs of training each model. Finetuing on BBBC021 labelled set.

We determined the best epoch by following the method of Janssens *et al.* [98]. We calculated NSC for each model and selected the epoch with the highest score. This method was chosen to allow a comparison of our model to other studies, however we note there are drawbacks to selecting best epoch with MOA information. In Table 5.1 the mean values of NSC and NSCB over 200 epochs of training are displayed for each model, and the results show improved performance for NSC and NSCB MOA matching was observed over multiple epochs of training using both compound and MOA as the weak label. The values are lower than the best values as this range contains both under- and over-fit models. We provide epoch data and training logs in our GitHub repository.

5.4 Results and Discussion

As a baseline we used the unsupervised DINO network [96]. First, we evaluated with ImageNet weights only (transfer learning). Next we fine-tuned the model on the BBBC021 annotated dataset without weak labels. We then trained WS-DINO with the same network and parameters. Two weak labels y were evaluated: treatment and compound. Additionally we trained with MOA as the label y, although this cannot be considered a weak label. The images in training were sampled using a weighted random sampler enforcing even sampling of images from each weak label group.

WS-DINO achieves state-of-the-art results on BBBC021 using the compound as the weak label, outperforming all known previous approaches using this dataset in NSC and NSCB MOA prediction (Table 5.3). The features from this model are plotted in a t-SNE plot in Fig. 5.3. It is notable that using treatment as the weak label does not improve upon the unsupervised approach (DINO). In this dataset all images of the same treatment are from the same batch, however this is not true for the different compounds

Model	Weak Label	NSC	NSCB
DINO with ImageNet weights only	None	91%	82%
DINO finetuned on BBBC021	None	92%	95%
WS-DINO finetuned on BBBC021	Treatment	92%	90%
	Compound	**98%**	**96%**
	MOA	100%	100%

Table 5.2 A summary of the results of our experiments. We propose that compound as a weak label is most effective as it provides cross-batch information to the feature learning. MOA is included as a strong label for proof of concept, but since we are trying to determine unkown MOA relationships it is not representative of real image-based profiling.

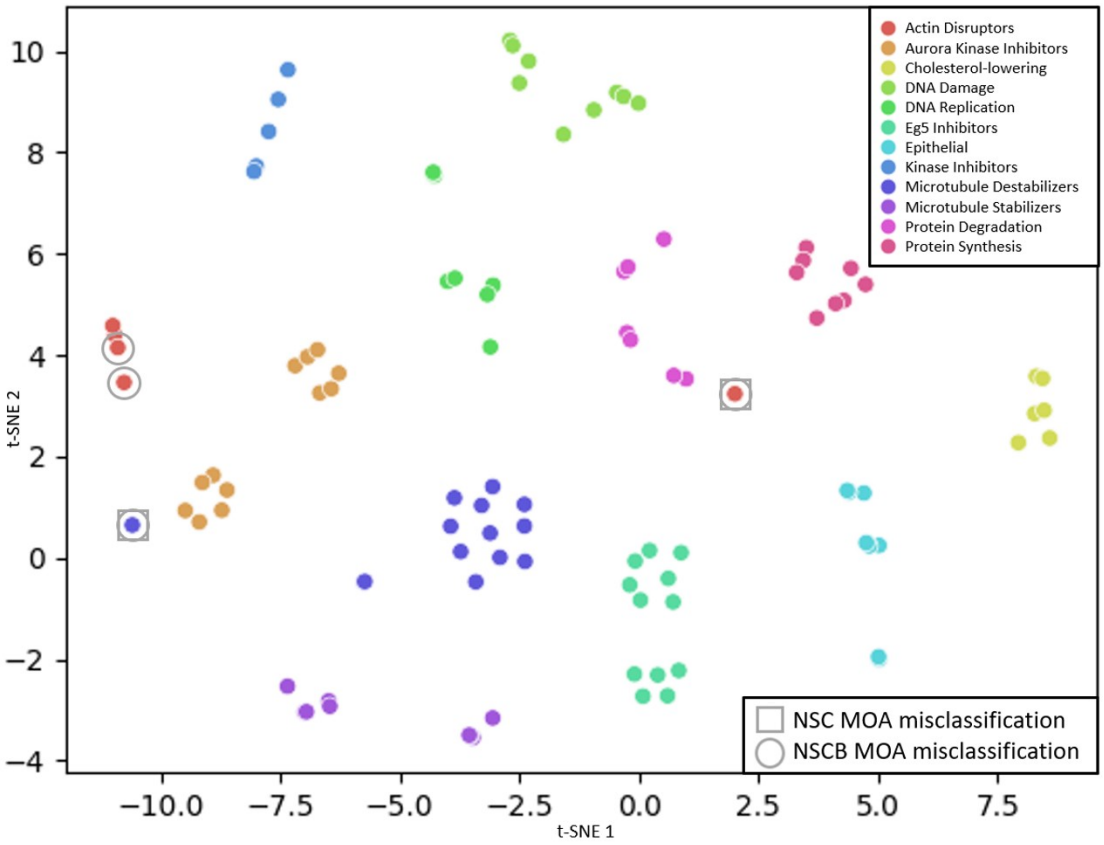


Fig. 5.3 Two-dimensional t-SNE plot of each aggregated treatment feature from 200 epochs of training WS-DINO with compound as the weak label: 98% NSC and 96% NSCB MOA classification.

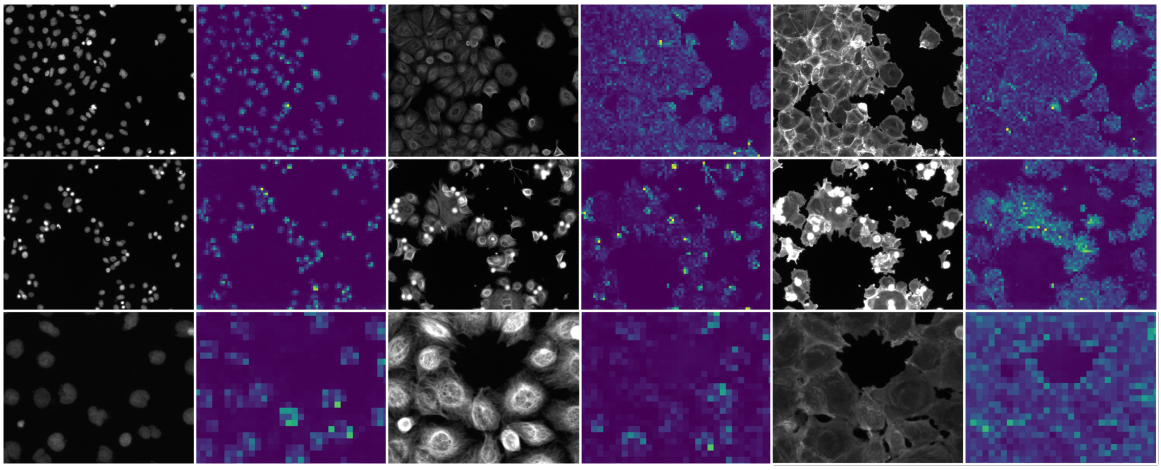


Fig. 5.4 Examples of BBBC021 images with their coupled self-attention maps. Left two columns: DNA channel. Middle two columns: β-tubulin channel. Right two columns: F-actin channel. The first two rows show the images at full size, and the bottom row displays a section of an image zoomed in by a scale factor of four. Produced from WS-DINO weights with compound as the weak label.

and MOAs which are expressed with different images across multiple batches. Hence we propose that WS-DINO may provide effective batch correction for datasets with weak label classes with images representing multiple batches.

We present self-attention maps (Fig. 5.4) which reveal the segmentation properties of the algorithm. Such visualisations increase confidence in the model by demonstrating the network is learning biologically and structurally meaningful features.

We use MOA as as *psuedo-weak* label as a proof-of-concept for our method. However, we propose future work to evaluate MOA prediction using datasets with partial MOA labels. One advantage of WS-DINO is that it is adaptable to datasets with some known MOA information. In practice, some MOA labels would be known in a drug screening dataset, and hence in training the weak label can be MOA where available, and compound when unavailable. We provide evidence this approach would be successful, however we suggest that this should be explored with further work.

The application of weakly supervised frameworks to high content imaging assays for the purpose of drug discovery could alleviate many bottlenecks in current analysis approaches. The performance of classical segmentation-based approaches depends heavily on pre-selected parameters for identification of cellular subregions. These pipelines can require significant re-optimisation to account for different experimental factors such as cell line, cell density or microscope selection and/or settings. Bypassing

Type	Method	Reference	Single Cell	NSC	NSCB
Weakly Supervised	**WS-DINO**	**This work**	**No**	**98%**	**96%**
	CNN with Mixup	Caicedo *et al.* 2018 [77]	Yes	95%	89%
Classical Features	CellProfiler	Singh *et al.* 2014 [76]	Yes	90%	85%
	Factor Analysis	Ljosa *et al.* 2013 [39]	Yes	94%	77%
Supervised	Multiscale-CNN	Godinez *et al.* 2017 [75]	No	93%	N/A
Unsupervised	Contrastive Learning	Perakis *et al.* 2021 [72]	Yes	96%	95%
	UMM Discovery	Janssens *et al.* 2020 [98]	No	97%	85%
	VAE+	Lafarge *et al.* 2019 [78]	Yes	93%	82%
	CytoGAN: LSGAN	Goldsborough *et al.* 2017 [92]	No	68%	N/A
Transfer Learning	Deep Metric Network	Ando *et al.* 2017 [73]	Yes	96%	95%
	Inception V3 Pretrained	Pawlowski *et al.* 2016 [87]	No	91%	N/A

Table 5.3 A comparison of our method to other selected studies using the BBBC021 annotated dataset. To the best of our knowledge we achieve the best performance in the literature (as of late 2022) with BBBC021. In particular, we highlight how WS-DINO performs significantly better than all existing methods which don't use single-cell segmentation as a pre-processing step.

the requirement for segmentation to yield single-cell crops removes these sources of variability and irreproducibility across and within datasets.

WS-DINO synthesises a powerful self-supervised framework with a way to implicitly incorporate the informative weak labels in learning phenotypic representations. Our method can contribute to accelerating drug discovery pipelines by clustering phenotypes in a biologically meaningful hierarchy. The proposed framework is general and the method is not specific to DINO. The sampling of pairs of images with the same weak label is a concept adaptable to other networks. Self-supervised methods are an active field of research and new algorithms have outperformed DINO on ImageNet classification, for example Masked Siamese Networks [216]. Future work could incorporate these studies which would be adaptable in a very similar way to WS-DINO. We will will revisit some of the ideas in this chapter with a different, larger dataset in Chatper 6.

The work presented in this chapter was released as a preprint in 2022 and accepted to *NeurIPS 2022 Workshop on Learning Meaninful Representations of Life*. The work was selected to be presented as a talk at the conference. Thank you to the co-authors of the publication.

Cross-Zamirski, J.O., Williams, G., Mouchet, E., Schönlieb, C.B., Turkki, R. and Wang, Y. Self-Supervised Learning of Phenotypic Representations from Cell Images with Weak Labels. In *NeurIPS 2022 Workshop on Learning Meaningful Representations of Life*. (2022). https://doi.org/10.48550/arXiv.2209.07819

Chapter 6

Class-Guided Image-to-Image Diffusion: Cell Painting from Brightfield Images with Class Labels

In this chapter we introduce and implement a model which combines image-to-image and label guided denoising diffusion probabilistic models. Image-to-image reconstruction problems with free or inexpensive metadata in the form of class labels appear often in biological and medical image domains. Existing text-guided or style-transfer image-to-image approaches do not translate to datasets where additional information is provided as discrete classes. We introduce and implement a model which combines image-to-image and class-guided denoising diffusion probabilistic models. To the best of the authors knowledge, the is the first known use of a diffusion model for high-content images, and in image-based profiling. We train our model on the publicly available JUMP-CP Target-2 dataset of microscopy images used for drug discovery, with and without incorporating metadata labels. We produce higher quality images than in the study presented in Chapter 4. By exploring the properties of image-to-image diffusion with relevant labels, we show that class-guided image-to-image diffusion can improve the meaningful content of the reconstructed images and outperform the unguided model in useful downstream tasks.

6.1 Introduction

Conditional denoising diffusion probabilistic models (DDPMs) [140, 139] are trained to learn a probability distribution capable of generating realistic samples from an

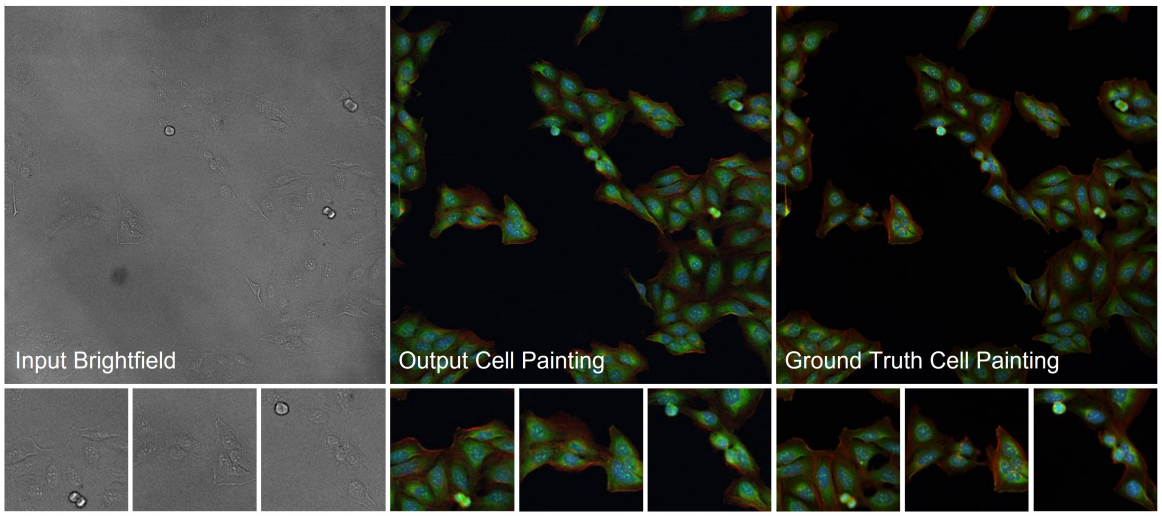


Fig. 6.1 A colour composite example of three channels from a test plate: red (AGP), green (ER) and blue (DNA).

input condition. These constructions typically fall into one of two categories: models conditional on an input image (image-to-image) [141] **or** models conditional on a class label [139, 217]. While many other diffusion models exist which incorporate natural language text encoders such as CLIP [218] (text-to-image) [180, 181], there has been much less attention on advancing models with both paired image **and** class label information. This can be attributed to a lack of generalist datasets which have both class labels and paired images, as this information can be expensive, sparse or narrow in application [219, 220].

Despite this, image-to-image problems with discrete metadata appear often in biological and medical image reconstruction. Examples of these inverse problems include PET reconstruction from MRI [221], predicting fluorescent labels from transmitted light microscopy [35], sparse-view CT reconstruction and artifact removal [222]. Through the nature of image acquisition there is often additional inexpensive *side* [221] or *weak label* [77] information which can be incorporated to guide the training of the inverse process towards the main task. For biological and medical datasets, class labels have been used in deep learning architectures to learn more faithful and generalisable representations [103, 104], and as extra information in image-to-image tasks [221].

These problems are dataset specific, and well-established databases of natural images [102, 223] and their associated labels are rarely analogous to the challenges presented by biological and medical datasets. These real-world datasets can have unique types of metadata labels, and application-specific ways to evaluate performance in downstream tasks. Using the predicted images in such tasks to quantify performance

may be more important and informative than benchmarking with metrics such as Fréchet Inception Distance (FID) [183] and structural similarity index (SSIM) [188]. Accurately capturing a distribution of images is complementary, if not subsidiary, to being able to differentiate between images and their features [9].

We investigate the utility of image-to-image diffusion with class labels using a subset of Target2 data generated as part of the the JUMP-CP effort [33] to predict Cell Painting [10] images from paired brightfield images (Chapter 4). We find that the quality of extracted morphological features from the predicted images, and their performance on downstream mechanism of action prediction and clustering tasks can be boosted with relevant labels. This type of approach may lead to increased clinical success of image-to-image methods in drug discovery [9] and related medical reconstruction tasks [224], as a way to guide the image generation with biologically informative class information. In this study we make the following contributions:

- We introduce and implement a general framework for class-guided image-to-image diffusion, our model building upon the *Palette* image-to-image framework [141] and guided diffusion [139] .
- We apply our model to the prediction of 5-channel Cell Painting fluorescent microscopy from 3-channel brightfield images, and show that incorporating label information can improve performance. We evaluate the images with extracted biological features and a transfer learning approach to simulate image-based profiling in a drug discovery pipeline.

6.2 Related work

Generative adversarial networks (GANs) [155] have been the prevailing method for image-to-image translation tasks since the introduction of *pix2pix* [165] in 2016. GAN based methods have been widely adopted in medical imaging for a variety of tasks [225] including PET denoising, PET-CT translation and correction of magnetic resonance motion artefacts [171]. GANs are used in cell microscopy for cross-modality prediction [226] and super resolution [144].

Other models used for reconstruction tasks include variational auto-encoders (VAEs) [157] and normalizing flows [227]. VAEs have been used to learn or approximate the joint distribution of multiple modalities [219], sometimes with a product or mixture of experts approach to combine the distributions [228]. Product of experts have also be used for multimodal conditional image synthesis with GANs [229]. Flow-based models

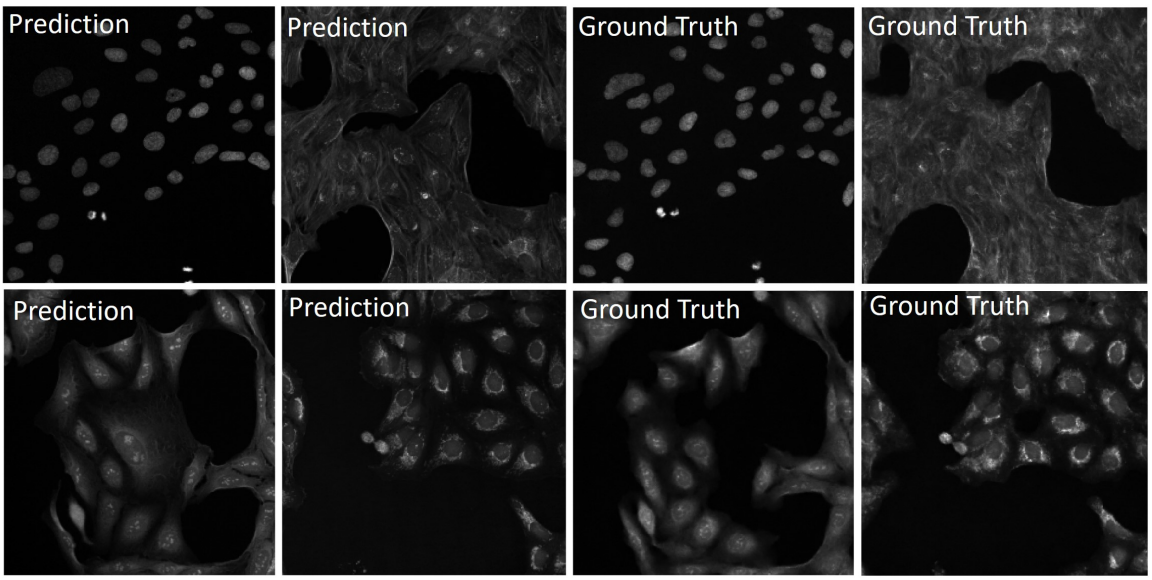


Fig. 6.2 Random examples of the predicted channels (diffusion model with Target as label in sampling) vs the ground truth. All the channels are included except the RNA channel, which is very similar to the ER channel.

for modality transfer (such as MRI to PET) have outperformed conditional GANs and VAEs while leveraging *side* information [221].

Diffusion models are growing in popularity in medical imaging and have been used predominantly for MRI and CT modalities in reconstruction problems [230]. Diffusion-based generative models can achieve state of the art image quality without suffering from problems such as mode collapse, training instability, or not allowing for likelihood estimation. By comparison, GANs can suffer from training instability and mode collapse [160] in addition to feature hallucinations which are particularly undesirable in medical applications [175]. VAEs do not produce high quality image samples and flow-based models have restrictions such as the requirement for invertibility of the network.

Weakly-supervised diffusion models have been used in medical imaging, notably in anomaly detection [231, 232]. However, these models are not strictly guided image-to-image models and instead use the difference between the ground truth and reconstructed image for anomaly detection. This method would not generalize beyond anomaly detection. *InstructPix2Pix* [233] combines a text-guided conditional diffusion model with an image-to-image framework using text based prompts. However, text encoders for style-transfer are not appropriate in datasets where metadata labels are discrete classes.

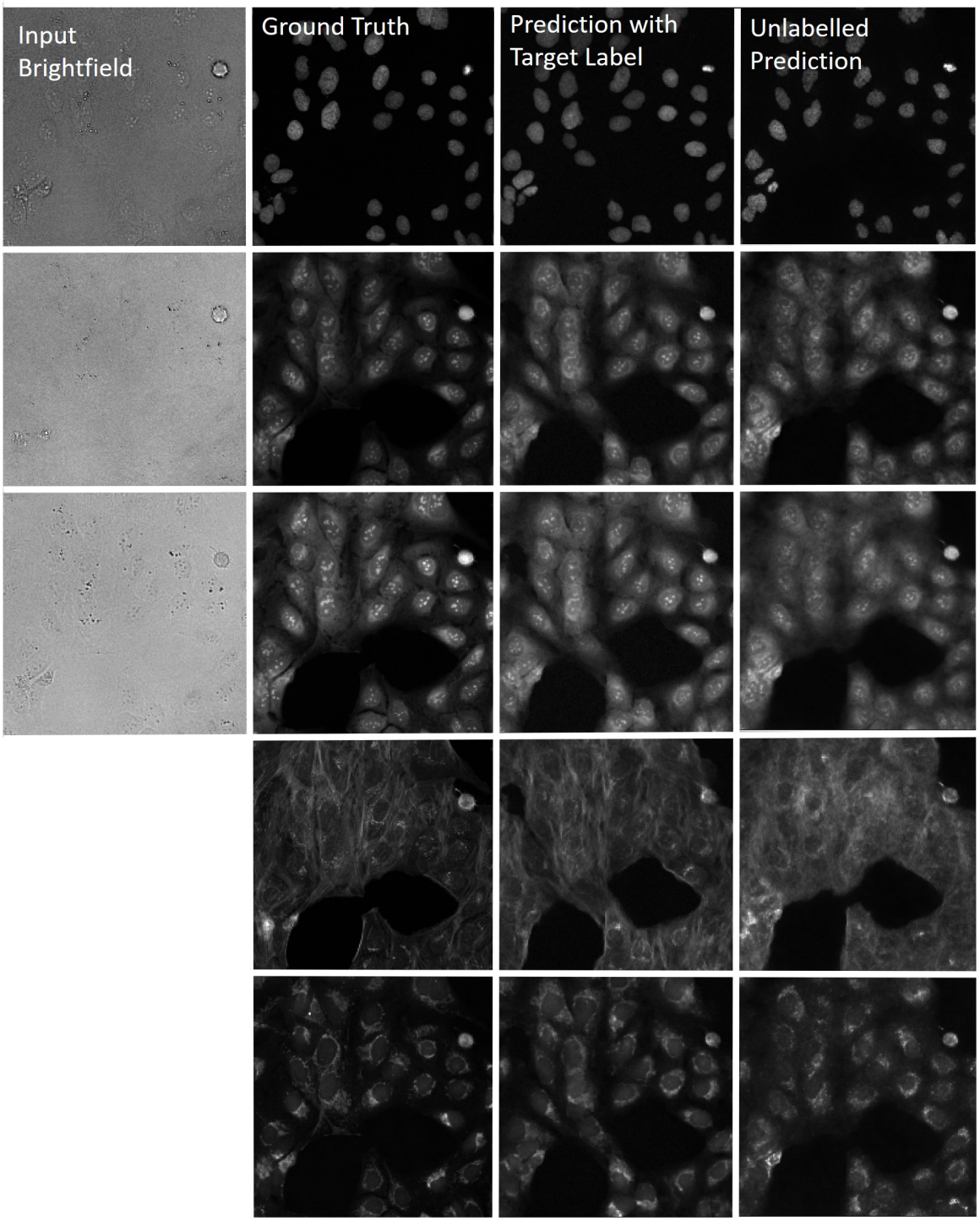


Fig. 6.3 An example of the predicted channels and their input brightfield vs the ground truth. The images are zoomed in by a scale factor of 4

Hence there is scope for developing a diffusion model for image reconstruction with discrete metadata. Examples of image-to-image problems with (often under-utilised) data include: prediction of fluorescent image channels from transmitted light images for drug discovery [35] where freely available weak labels include treatment and compound [77], as well as batch information. Experimental batch effects can be significant, and batch information has been integrated into a number of machine learning models in image-based profiling [104, 103, 73]. PET reconstruction from much cheaper MRI scans for Alzheimer's prediction also has inexpensive and relevant metadata such as patient age, sex, disease status and genotype which has been incorporated into improving image reconstruction quality [221].

To the best of our knowledge, this work is the first to use a diffusion model for fluorescent microscopy prediction. We build upon existing studies using deep learning to predict fluorescent labels [35] from transmitted light images such as brightfield, a cheaper and less invasive modality for imaging cells which can still capture meaningful information [53]. Specifically, we predict Cell Painting [10] image channels which capture rich cell morphology information which can be used in a variety of tasks in image-based profiling including bioactivity, cytotoxicity and mechanism of action prediction [9].

6.2.1 Image-to-image conditional diffusion

We base our model on the *Palette* framework for image-to-image diffusion from Saharia *et al.* [141]. Their model outperforms GANs on four tasks: colorization, inpainting, uncropping and JPEG restoration. *Palette* is a denoising diffusion probabilistic model [140] of the form $p(y \mid x)$ which is trained to predict the output image y conditional on the input image x. The noisy image $\widetilde{y}$ is given by:

$$\widetilde{y} = \sqrt{\gamma} y + \sqrt{1-\gamma}\epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \tag{6.1}$$

for Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ and noise level indicator γ. A neural network f_θ is trained to denoise $\widetilde{y}$ for a given x with the loss function:

$$\mathbb{E}_{(x,y)}\mathbb{E}_{\epsilon\sim\mathcal{N}(0,I)}\mathbb{E}_{\gamma}\left\| f_\theta(x, \underbrace{\sqrt{\gamma} y + \sqrt{1-\gamma}\epsilon}_{\widetilde{y}}, \gamma) - \epsilon \right\|_p^p \tag{6.2}$$

where p is the chosen norm (L_1 or L_2). Eq. (6.2) is the image-conditional version of L_{simple} from Ho *et al.* [140].

The reverse diffusion process is computed step-by-step as:

$$y_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}\left(y_t - \frac{1-\alpha_t}{\sqrt{1-\gamma_t}} f_\theta(x, y_t, \gamma_t)\right) + \sqrt{1-\alpha_t}\epsilon_t \tag{6.3}$$

for $t = T, \ldots, 1$ steps. The noise level indicator γ_t is a function of t, and α_t is the noise variance scale parameter (also timestep-dependent).

6.2.2 Conditional image synthesis

For conditional image synthesis with class labels, Dhariwal and Nichol [139] introduced two modifications to unconditional DDPM from Ho *et al.* [140]: adaptive group normalization (AdaGN) and classifier guidance (CG). AdaGN is a modification to the architecture which incorporates the class information into normalization layers in training, while classifier guidance exploits the gradients of a pre-trained classifier to guide the inference process (note: in this section we change $y \to k$ and $x \to y$ from the original paper to be consistent with the notation used in this chapter)

Adaptive group normalization

AdaGN is a layer used to incorporate the timestep and class embedding into the residual blocks following a group normalization operation [179]. It is defined as:

$$\text{AdaGN}(h, k) = k_s \text{GroupNorm}(h) + k_b \tag{6.4}$$

where $k = [k_s, k_b]$ is a linear projection of the timestep and class embedding, and h is the activations of the residual block after the first convolution. This layer can be incorporated in the absence of class labels with just the timestep embedding: AdaGN $= k_s \text{GroupNorm}(h)$.

Classifier guidance

Classifier guidance enables the use of class information in inference of the trained diffusion model. Sohl-Dickstein *et al.* [176] and Song *et al.* [178] showed this can be achieved using pre-trained classifier gradients to condition the sampling of the diffusion model. First, the classifier $p_\phi(k \mid y_t)$ is pre-trained to predict the class k from noisy images y_t.

The aim is to sample each transition from the distribution:

$$p_{\theta,\phi}(y_t \mid y_{t+1}, k) = Z p_\theta(y_t \mid y_{t+1}) p_\phi(k \mid y_t) \tag{6.5}$$

where $p_\theta(y_t \mid y_{t+1})$ is the unconditional reverse noising process and Z is a normalizing constant. Although it is intractable to sample from the distribution in Eq. (6.5), it can be approximated as a perturbed Guassian distribution [176]:

$$\log(p_\theta(y_t \mid y_{t+1}) p_\phi(k \mid y_t)) \approx \log p(z) + C, \tag{6.6}$$

$$z \sim \mathcal{N}(\mu + \Sigma g, \Sigma), \quad g = \nabla_{y_t} \log p_\phi(k \mid y_t)|_{y_t=\mu} \tag{6.7}$$

where $g = \nabla_{y_t} \log p_\phi(k \mid y_t)$ are the gradients of the classifier and C is a constant which can be ignored. In inference, this shifts the mean of the sampled Gaussian to guide the denoising process towards the given class label k. The relative weighting of the classifier guidance term can be scaled with a constant s (Algorithm 2).

6.3 Multi-modal conditional diffusion

In this section we present the major adaptations to the diffusion models presented in the previous section. In *Palette*, Saharia *et al.* [141] removed both classifier guidance and the class embedding of the AdaGN layer introduced by Dhariwal and Nichol [139]. In this section we re-introduce the class label k while retaining the conditional dependence on input image x. We redefine the input conditions for image and associated class label as (x_k, k).

We summarise the training scheme for class guided image-to-image diffusion in Algorithm 1, and the sampling scheme in Algorithm 2. Initially we change the network f_θ introduced in Eqs. (6.2) and (6.3) to have a dependence on k: $f_\theta = f_\theta(x_k, y_t, k, \gamma_t)$. Following Eqs. (6.3) and (6.7), we define each iteration of the reverse process of class-guided image-to-image diffusion to be computed as:

$$y_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(y_t - \frac{1-\alpha_t}{\sqrt{1-\gamma_t}} f_\theta(x_k, y_t, k, \gamma_t) \right) + \sqrt{1-\alpha_t} z \tag{6.8}$$

for $t = T, \ldots, 1$. Here:

$$z \sim \mathcal{N}(\mu + s\Sigma\nabla_{y_t} \log p_\phi(k \mid y_t), \Sigma) \quad \text{if} \quad t > 1, \quad \text{else} \quad z = 0 \tag{6.9}$$

Algorithm 1: Training the denoising model f_θ

repeat
  $(x_k, y_0, k) \sim p(x_k, y, k)$
  $\gamma \sim p(\gamma)$
  $\epsilon \sim \mathcal{N}(0, I)$
  Take a gradient descent step on
    $\nabla_\theta \left\| f_\theta(x_k, \sqrt{\gamma} y_0 + \sqrt{1-\gamma}\epsilon, k, \gamma) - \epsilon \right\|_p^p$
until converged

Algorithm 2: Classifier guided diffusion sampling, given a diffusion model $(\mu_\theta(x_t), \Sigma_\theta(x_t))$, classifier $p_\phi(k \mid y_t)$, and gradient scale s.

Input: class label k, input image x_k, gradient scale s
$y_T \sim \mathcal{N}(0, I)$
for $t = T, \ldots, 1$ **do**
  $z \sim \mathcal{N}(\mu + s\Sigma\nabla_{y_t} \log p_\phi(k \mid y_t), \Sigma)$ if $t > 1$, else $z = 0$
  $y_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(y_t - \frac{1-\alpha_t}{\sqrt{1-\gamma_t}} f_\theta(x_k, y_t, k, \gamma_t) \right) + \sqrt{1-\alpha_t} z$
end for
return y_0

where $p_\phi(k \mid y_t)$ is the pre-trained classifier as in Eq. (6.5). The f_θ dependence on k is achieved with the AdaGN layer. It is optional to use k in sampling, as the AdaGN layer does not need to see the label. We test this and find that although it is possible to exclude k in sampling, it is necessary to include for improved performance. It is also possible to sample without classifier guidance by setting $s = 0$.

6.3.1 Model architecture and training

The training objective follows the form of Eq. (6.2) for neural network f_θ which is trained to denoise $\widetilde{y}$ for a given (x_k, k) with the loss function:

$$\mathbb{E}_{(x,y)} \mathbb{E}_{\epsilon \sim \mathcal{N}(0,I)} \mathbb{E}_\gamma \left\| f_\theta(x_k, \underbrace{\sqrt{\gamma} y_0 + \sqrt{1-\gamma}\epsilon}_{\widetilde{y}}, k, \gamma) - \epsilon \right\|_p^p \tag{6.10}$$

where p is L_2 norm.

Our network is a U-Net (Section 3.1.2) architecture [140] which is based on the modified 256×256 class-conditional U-Net model used in *Palette* [139, 182]. This U-Net consists of a stack of BigGAN [168] residual layers and downsampling convolutions in the downsampling path, and a similar stack of residual layers and upsampling convolutions in the upsampling path, in addition to the typical skip connections.

Residual connections are rescaled by a factor $\frac{1}{\sqrt{2}}$ following previous works [178, 139]. There are additional attention (Section 3.1.3) layers at multiple resolutions (32×32, 16×16 and 8×8 pixels) and multiple attention heads. The projection timestep γ is embedded into each residual block.

The network is adapted to take images of size 512×512 with $(5 + 3)$ input channels and 5 output channels in order to fit the requirements of the brightfield and Cell Painting channels. The conditional image x_k is concatenated to an image of Gaussian noise with equal size and channel dimensions to the output image (in our case 8 channels and 512×512 pixels). Hence the process is denoising the noised input channels as in DDPM [140] (Section 3.3.2) with the additional image and class condition.

The classifier architecture of network $p_\phi(k \mid y_t)$ is the downsampling branch of the U-Net used in f_θ with an additional attention pool and an 8×8 output layer.

6.3.2 This model in the diffusion model landscape

We recall from Section 3.3.2 that training the standard DDPM is equivalent to training the function $\epsilon_\theta(y_t, t)$ to predict the noise component of the noisy sample y_t. Once this objective is trained with gradient descent, images can be sampled with:

$$y_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}\left(y_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}}\epsilon_\theta(y_t, t)\right) + \sqrt{1-\alpha_t}z \tag{6.11}$$

for $t = T, \ldots, 1$ iterations, starting from noise $y_T \sim \mathcal{N}(0, I)$. For $t = 0$, $z = 0$, otherwise z is normally distributed $z \sim \mathcal{N}(0, I)$. It is notable how this is very similar to class-guided image-to-image diffusion (Eq. (6.8)). There are two differences: the neural network has multiple dependencies $\epsilon_\theta(y_t, t) \rightarrow f_\theta(x_k, y_t, k, \gamma_t)$ and z becomes dependant on the classifier as in Eq. (6.9), as opposed to being sampled from the standard Gaussian.

Diffusion models (and more generally, generative models) are a very active and fast changing area of research. Our work is heavily related to Guided Diffusion [140] and *Palette* [141]. Related work uses natural language embeddings such as CLIP (Contrastive Language Image Pre-training) to incorporate multimodal information [218, 180, 181]. The main downside of these diffusion models is slow inference, and although alternative methods such as DDIMs (Denoising Diffusion Implicit Models) [178] exist, new models such as Consistency models [234] speed up inference significantly with seemingly no downside. Future work in image-based profiling should explore these models where appropriate for the dataset and task.

6.4 Experiments

6.4.1 Dataset

The dataset used in this chapter is available at: https://registry.opendata.aws/cellpainting-gallery/. We used a subset of one of the publicly-available JUMP Cell Painting [34] dataset cpg0000 [235] [33], available from the Cell Painting Gallery on the Registry of Open Data on AWS. 10 plates (experimental replicates) were chosen to ensure a variety of biological phenotypes were present. They contain pairs of compounds associated by the genes they target, in addition to 46 controls compounds with a variety of mechanisms. In total, every plate contains around 2000 images - each with 5 Cell Painting channels and 3 brightfield channels. These plates were screened regularly throughout data production to enable downstream assessment of connectivity of perturbations between batches of compounds screen. Every plate contains treated cells representing 290 perturbations, each with paired perturbation with a matching target (145 targets total).

U2-OS cells were incubated in 5µM compounds for 48 h, then fixed and stained according to the updated Cell Painting protocol [63]. Plates were imaged on a CellVoyager CV8000 (Yokogawa, Tokyo, Japan) with a water-immersion 20x objective (NA 1.0). Excitation and emission wavelengths were as follows for fluorescent channels: DNA (ex: 405nm, em: 445/45nm), ER (ex: 488nm, em: 525/50nm), RNA (ex: 488nm, em: 600/37nm), AGP (ex: 561nm, em: 600/37nm) and Mito (ex: 640nm, em: 676/29nm). The three brightfield images were acquired from different focal z-planes; within, 4µm above and 4µm below the focal plane. Images were saved as 16-bit .tiff files with 2×2 binning (998×998 pixels).

6.4.2 Pre-processing

To ensure that systematic variations in pixel intensity are not present in input images, we used a standardised CellProfiler [84] pipeline to perform illumination correction on all images. A smoothing function of filter size 249 pixels generated an illumination correction function per imaging channel per plate. The pixel intensities of all images were then divided by their respective correction function. This methodology is consistent with best practice established during the JUMP Cell Painting consortium [63]. After illumination correction, all images were re-sized to 512×512 pixels using bicubic interpolation. The images were all normalised to have a standard deviation of 1 and a mean of 0. A maximum pixel cutoff of 15 was enforced to exclude extreme outliers.

6.4.3 Model training

We trained each model using 9 training plates and evaluated on a single, unseen test plate. For each model this was done twice, learning weights for 2 different, randomly selected test plates (the same 2 plates for each model). This is equivalent to k-fold cross validation - although we trained 2 versions of each model rather than 10, as producing 10 full plates per model was not possible due to the computationally intensive nature of sampling DDPMs.

Using the full plates, we trained models with no labels (*Palette*), perturbation (pert) as a weak label, target as a label. The labels were included through the AdaGN layer. We compared using the labels in training and inference against using labels in training but not in inference through the AdaGN layer to test if the labels were required in sampling. Target as a label was included as a proof of concept of the method, but it is expensive information not freely available in a practical setting (compared to the perturbation which is free information). Classifier guidance was not used to generate entire plates due to the extreme computational demands (over 500 GPU hours per model, per plate). This training regime is equivalent to Algorithm 2 with $s = 0$.

The active subset was around one third of the full plate, and this allowed us to sample using classifier guidance with $s = 1$. Additionally, training with known active compounds would provide more meaningful class labels for the model. The classifier $p_\phi(k \mid y_t)$ was the downsampling branch of the U-Net with an additional output layer (as introduced by Ho *et al.* [140]). Training images were noised with the timestep dependent noise distribution, and the model was trained until the loss converged. The full training and sampling schemes are presented in Algorithms 1 and 2.

In training, the images were subject to random horizontal and vertical flips and 90 degree rotations each with probability $p = 0.5$. Models were trained until the loss appeared to stop decreasing, which was typically around $250,000$ iterations. Even though the quality of cellular structures appeared to improve beyond this, we found overfitting to be a problem for larger number of epochs as phantom structures appeared on the empty background. All models were trained with a batch size of 2 and the Adam optimizer with a learning rate of $8e^{-5}$. The linear noise schedule of $(10e^{-6}, 0.001)$ (as in *Palette*) with $T = 2000$ was used in training and inference. We provide the code and parameters to replicate these models in our GitHub repository. All the models were trained on the AstraZeneca Scientific Computing Platform (SCP) with 32GB GPUs. Total training time was around 24 hours on a single GPU, and sampling from the trained model was around 4 minutes per 5 channel image (increasing to 15 minutes with classifier guidance).

6.4.4 Post-processing

CellProfiler was used to extract features with the standard Cell Painting pipeline implemented and described in Chapter 4. The model outputted channels were re-normalised as in the pre-processing. CellProfiler [42] was used to segment nuclei, cells and cytoplasm, then extract morphological features from each of the channels. Single cell measurements of fluorescence intensity, texture, granularity, density, location and various other features were calculated as feature vectors. Features were aggregated using the median value per image.

The Pycytominer (Appendix B) package was used to normalise the cell-painting features generated for the synthetic images. The features derived for synthetic images generated by each model was normalized by using all the samples. All the features generated from the ground truth data were also used for the prediction feature selection operation to allow for a fair comparison. These included dropping na columns, variance thresholding, correlation thresholding and dropping blocklisted features. Approximately 650-700 cell-painting features were selected for each plate. Features were aggregated to the perturbation level, giving 290 features per plate.

In order to segregate the active perturbations from inactives, PCA was performed using 1262 Cell Painting features that remained after CellProfiler feature selection (from the ground truth images). The top 100 dimensions of the PCA were then used to evaluate the cosine-distances between all-pairs of data points (well). An average cosine-distance score against the negative DMSO controls across all replicates was used as a score to segregate out the actives from inactives using 1D C-kmeans clustering algorithm with $k = 3$ for 3 clusters. There were a total of 118 perturbations representing 59 targets selected for the active subset, with the remaining perturbations (inactives) showing no phenotypic divergence from negative controls. We provide visualisations of the dataset and the active subset in the Appendix.

6.4.5 Transfer learning with DINO

We used the self-supervised learning algorithm DINO [96] pre-trained with ImageNet [102] weights to profile the images with transfer learning, following the methodology presented in Chapter 5. The backbone of the network is a vision transformer (ViT-S/8) with a 3-layer multi-layer perceptron head, from which the embeddings are extracted. The median feature embedding was taken from four 224×224 crops around the centre of each image (equivalent to a 448×448 pixel centre crop split into four non-overlapping crops). Embeddings of size 384 were extracted for each channel then concatenated

to a size of 1920 for 5-channel Cell Painting (1152 for brightfield) followed by L_2 normalization. These feature representations, like the CellProfiler features, were then used for target prediction.

6.5 Results

6.5.1 Evaluation

We evaluated our models with image-level and feature-level metrics, which are presented in Table 6.1 (the entire plate) and Table 6.2 (the active subset). We compared Pearson correlation coefficient (PCC), Fréchet Inception Distance (FID) [183], structural similarity index measure (SSIM) [188] and mean-squared and mean-absolute error (MSE/MAE). The values in the tables were calculated by comparing the predicted images with the ground truth images for each model. The values presented are the mean values of all the images. Examples of the images are presented in Figures 6.1 and 6.4. We also compare the FID scores and feature values between the two ground truth plates as the limit of a perfect reconstruction (each plate is meant to be an experimental replication of the same cells and treatments). The feature-level metrics were chosen to be representative of downstream applications which would be performed with real Cell Painting images in a drug discovery pipeline. The metrics used are as follows:

NN matching / NN top 5

We searched the feature spaces of each plate - both CellProfiler (CP) and transfer learning (TL) spaces - for the nearest neighbours by cosine distance. The values reported in the tables are the total number of matching targets which are nearest neighbours in the feature space of the model or ground truth plate feature space (for both plates). We repeated this analysis but for each point searching for the 5 nearest neighbours, and reporting a match if one of the 5 perturbations shared a target with the chosen point. It is notable that NN matching for this dataset is very difficult. Chance would give 1 in 289 - around 0.3%. Hence why even the ground truth features struggle to get many matches.

Matching target distance (MTdist)

Since there hundreds of targets and perturbations in this dataset, even searching the top 5 nearest neighbours is not sufficient to evaluate the relationships between targets.

We propose the mean matching target distance (MTdist) as an informative metric. For each pair of perturbations sharing a matching target, the cosine distance is calculated between the points in feature space. The mean distance for all 290 perturbations (118 in the active subset) is presented for each model. Since the models feature spaces are normalised this should be a fair comparison between the models.

CellProfiler feature correlation (CPcor)

Following the methodology of our previous Cell Painting prediction studiy in Chapter 4, we correlated each model's CellProfiler features to the ground truth CellProfiler features, and report the mean value. We also correlate the features between the ground truth replicates as a baseline (0.569 for the whole plate and 0.615 for the active subset). We would not expect the model generating features from an unseen batch to exceed this value. We include a breakdown of the features by group and channel in the supplementary material, alongside two-dimensional t-SNE plots of the features.

Label									
Training	**Sampling**	**PCC ↑**	**FID ↓**	**SSIM ↑**	**MSE / MAE ↓**	**NN matches ↑** **CP / TL**	**NN Top 5 ↑** **CP / TL**	**MTdist ↓** **CP / TL**	**CPcor ↑**
None	None	**0.793**	3.54	**0.350**	**0.400** / **0.338**	6 / **8**	**23** / **25**	**0.886** / **0.0729**	**0.430**
Pert	None	0.760	**3.26**	0.267	0.465 / 0.380	**7** / 5	20 / **25**	0.910 / 0.0852	0.384
Pert	Pert	0.752	3.49	0.260	0.481 / 0.392	**7** / 3	22 / 22	0.919 / 0.0936	0.381
Target*	None	0.741	3.69	0.239	0.489 / 0.402	4 / 4	15 / 17	0.888 / 0.0834	0.283
Target*	Target*	0.745	3.86	0.228	0.495 / 0.408	**17** / **15**	**32** / **51**	**0.791** / 0.0836	0.327
GT Cell Painting		–	1.55†	–	–	12 / 13	31 / 28	0.868 / 0.0924	0.569†
GT Brightfield		–	–	–	–	– / 12	– / 28	– / (0.0551)	–

Table 6.1 Mean image and feature metrics for class-guided image-to-image models for two full plates, each generated with a different model. Note the brightfield feature space (3 channels) is a different size to the Cell Painting feature space (5 channels). *Target is not a freely available label and is included as a proof of concept. †We provide FID and CPcor values calculated between the two ground truth (GT) test plates, which are prepared and treated as identical replicates.

6.6 Discussion and conclusion

The purpose of this study was to explore how metadata in the form of discrete classes can be used to guide image-to-image translation tasks. DDPMs and other generative

Label AdaGN	Label CG	PCC ↑	SSIM ↑	MSE / MAE ↓	NN matches ↑ CP / TL	NN Top 5 ↑ CP / TL	MTdist ↓ CP / TL	CPcor ↑
None	None	**0.773**	0.294	**0.423** / 0.320	4 / 2	12 / 16	0.971 / 1.533	0.386
Pert	None	0.762	**0.379**	0.444 / **0.310**	6 / 4	18 / **18**	0.939 / **1.357**	**0.507**
Pert	Pert	0.752	0.338	0.463 / 0.330	**7** / **7**	**24** / **18**	**0.929** / 1.541	0.504
Target*	None	0.730	0.235	0.506 / 0.375	9 / 14	27 / **27**	0.883 / 1.405	0.404
Target*	Target*	0.696	0.202	0.573 / 0.408	**11** / 6	**31** / 21	**0.879** / 1.579	0.355
GT Cell Painting		–	–	–	9 / 13	21 / 26	0.919 / 0.233	0.615†
GT Brightfield		–	–	–	– / 16	– / 26	– / (1.148)	–

Table 6.2 The analysis of Table 1 is repeated for the active subset only. There are too few images in the active subset to calculate FID. *Target is not a freely available label and is included as a proof of concept. †We provide the CPcor value calculated between actives in the two ground truth (GT) test plates, which are prepared and treated as identical replicates.

models have been successful in achieving state of the art FID scores, however learning details which differentiate between images based on biology and structure is less studied when compared to generating realistic images which could have been sampled from a training distribution.

All the models achieved very low FID scores, and the values compare favourably to the values achieved by the GAN and U-Net models in Chapter 4, which were 18.21 and 20.19 respectively (unreported in the study). For entire plates (Table 6.1), incorporating labels through AdaGN generally reduced the performance of the pixel-level metrics, although using the perturbation as a label in training resulted in the lowest FID score. Some images from the labelled models had some background noise which was not present in the unlabelled model, and this is reflected in the image-level metrics. We present an example of this effect in the supplementary material. While it is possible that class labels can improve certain aspects of the images, they may also reduce the image quality by fitting to unwanted background noise if there are uninformative class labels or no signal to be found in the training set. This was particularly notable using target as the label, which moved matching targets closer in feature space, but reduced the faithfulness of the generated image. This effect is likely amplified in high-content microscopy images where over 50% of the pixels are irrelevant background with no cellular structures.

The results for the model trained with full plates of images (290 perturbations) suggest that the image-to-image model is capable of capturing strong phenotypic signals (true positives) but struggled with noisy, lower signal images (the inactives). We may have led the model astray with uninformative labels in training. To test this theory, we

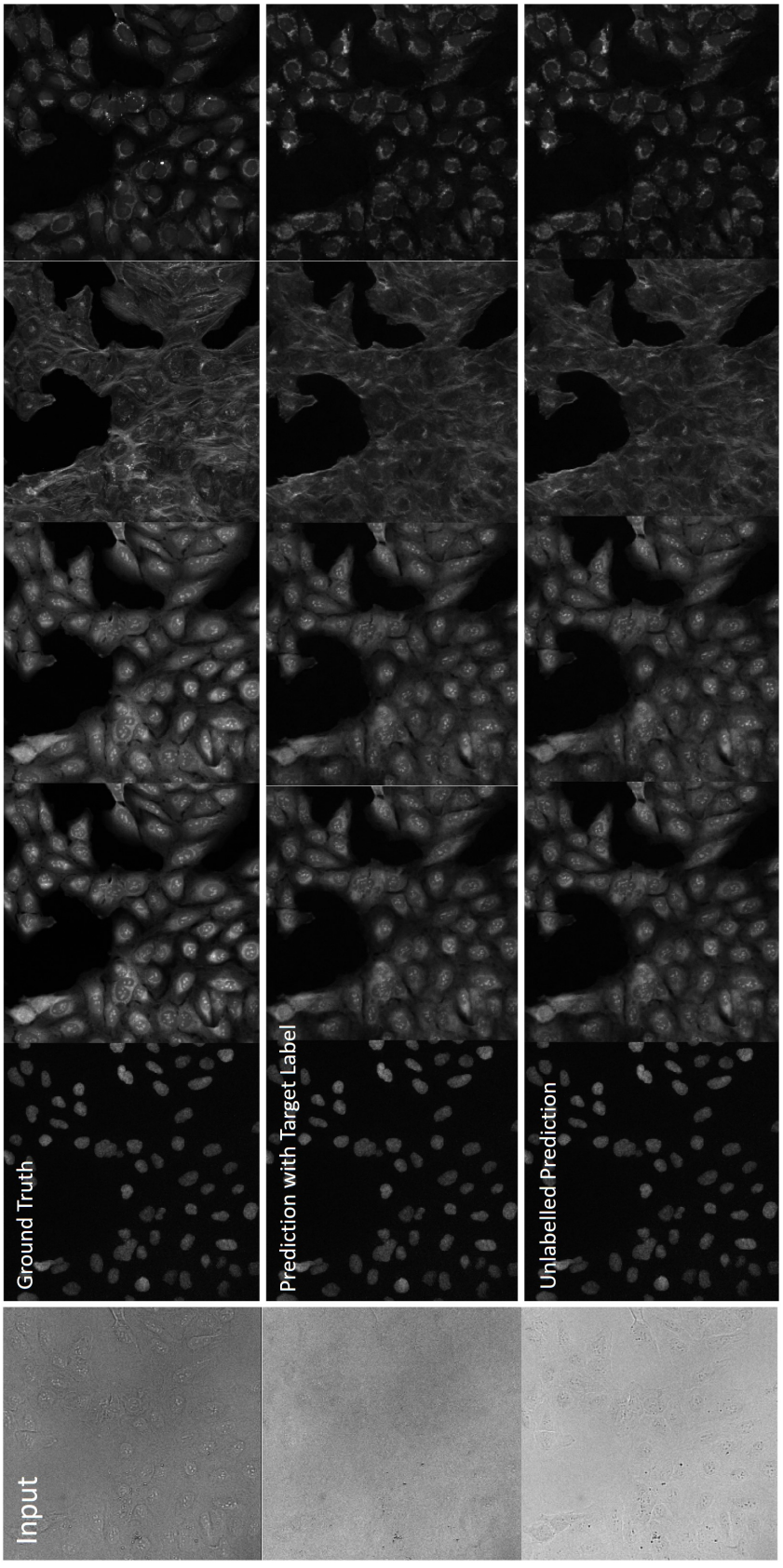


Fig. 6.4 A randomly chosen example of the predicted channels vs the ground truth, as well as the input brightfield channels. Columns left to right: Brightfield (input), DNA, RNA, ER, Mito, AGP.

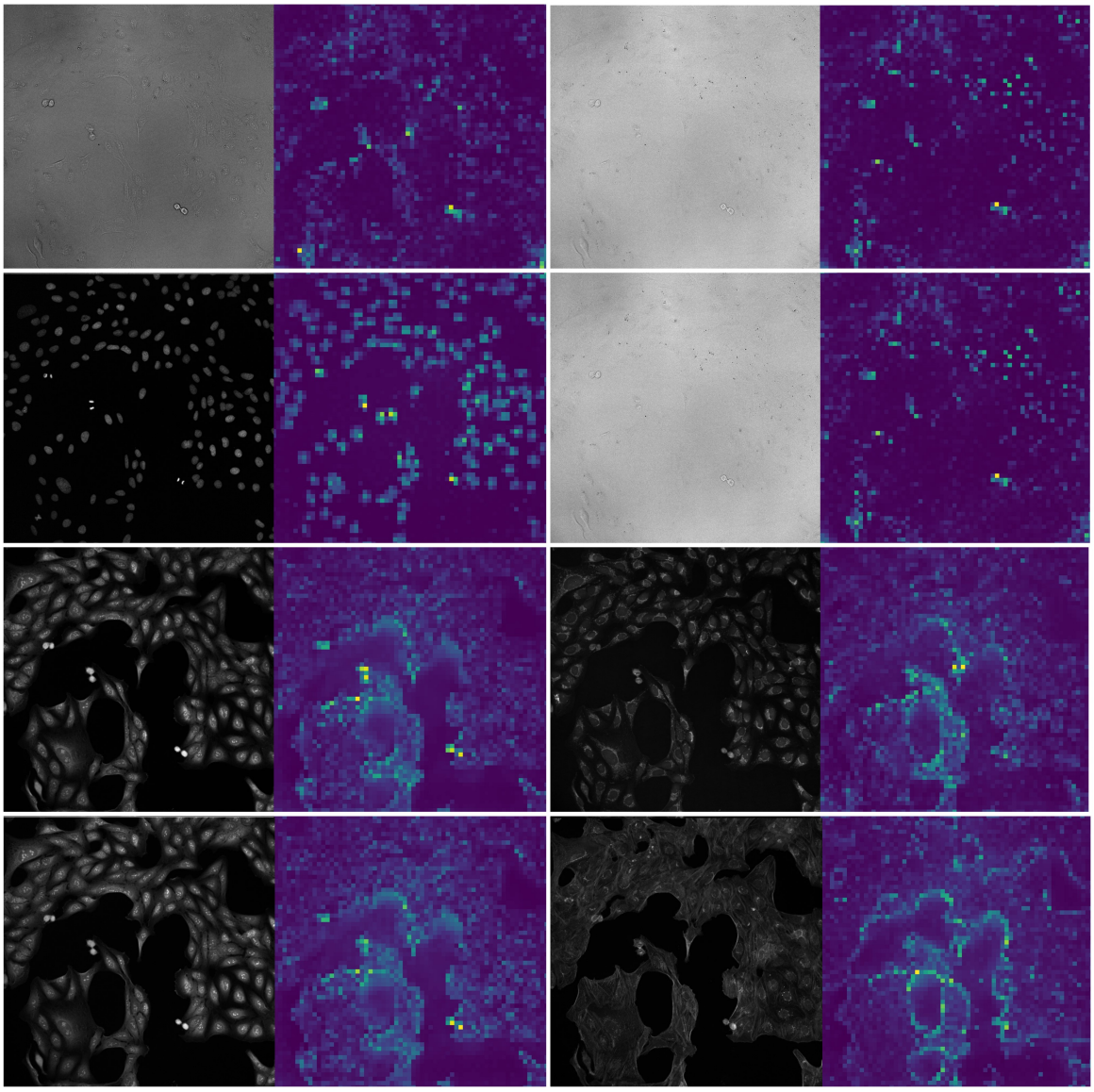


Fig. 6.5 An example of paired self-attention maps for ground truth (real) images with transfer learning (DINO) weights. Left column (top to bottom): Brightfield 1, DNA, RNA, ER. Right column (top to bottom): Brightfield 2, Brightfield 3, Mito, AGP.

repeated the analysis with the active subset (Table 6.2), which represents the images of cells with meaningful and quantifiable phenotypic differences from the control group (untreated cells). This resulted in a significant improvement over the unlabelled model (*Palette*) in SSIM, target matching and CellProfiler feature correlation. Furthermore, the pixel correlations and errors were not significantly reduced, so unlike when using the whole plate, there was less of a cost to the improved performance. Incorporating classifier guidance improved target matching but also at a small cost to image and feature quality. Our results show that class-guided image-to-image diffusion improves upon the naive model under well-chosen conditions, and highlight how crucial the quality of labels and training data is.

The values in Tables 6.1 were produced by models trained and tested on the whole plate, while Table 6.2 presents results from images trained with the smaller active subset. The smaller training set of the active subset reduced the quality of the unlabelled model. However, incorporating labels produced the highest correlations of features, and the highest SSIM even in the low training data regime. These effects were observed in both training splits. Very recently, Cell Painting datasets of immense scale with millions of images across thousands of compounds and over 50 batches have become public, and hold great promise for machine learning in drug discovery [236, 237]. The batch effect is a large part of this, and we explore the batch effect properties of our models in the supplementary material.

This study provides a valuable comparison of methods employing brightfield image channels as an input for image-based profiling. Recent studies have explored this under-utilised modality which may contain as much predictive power as fluorescent stained images [53]. Our results further reveal the potential of brightfield both as an input for cross modality prediction and as a competitive profiling modality in itself. This success may also be attributed to powerful, pretrained attention based architectures [96], which can overcome the traditional drawbacks of brightfield and are able to find meaningful structures from noisy images (we present self-attention maps of transfer learning with brightfield images in the supplementary material). Brightfield and transmitted light has traditionally been seen as less informative than fluorescent staining, but the limit of brightfield may be higher than previously thought. Furthermore, we have presented a way to use brightfield to generate full plates of model-generated Cell Painting, from which existing software can extract hand-crafted features for a greater level of interpretability.

In conclusion, we present a novel way to use discrete metadata to guide image-to-image translation. We predict unseen batches of Cell Painting from brightfield, and

Fig. 6.6 Examples of images and paired self attention maps for the ground truth vs diffusion model with label and without. Left to right: DNA, RNA, ER channels. In this example, the self-attention maps are more similar to the ground truth for the model incorporating the target label, suggesting using target information has improved the quality of the reconstruction in terms of useful information content. The remaining two channels are displayed in Fig. 6.7

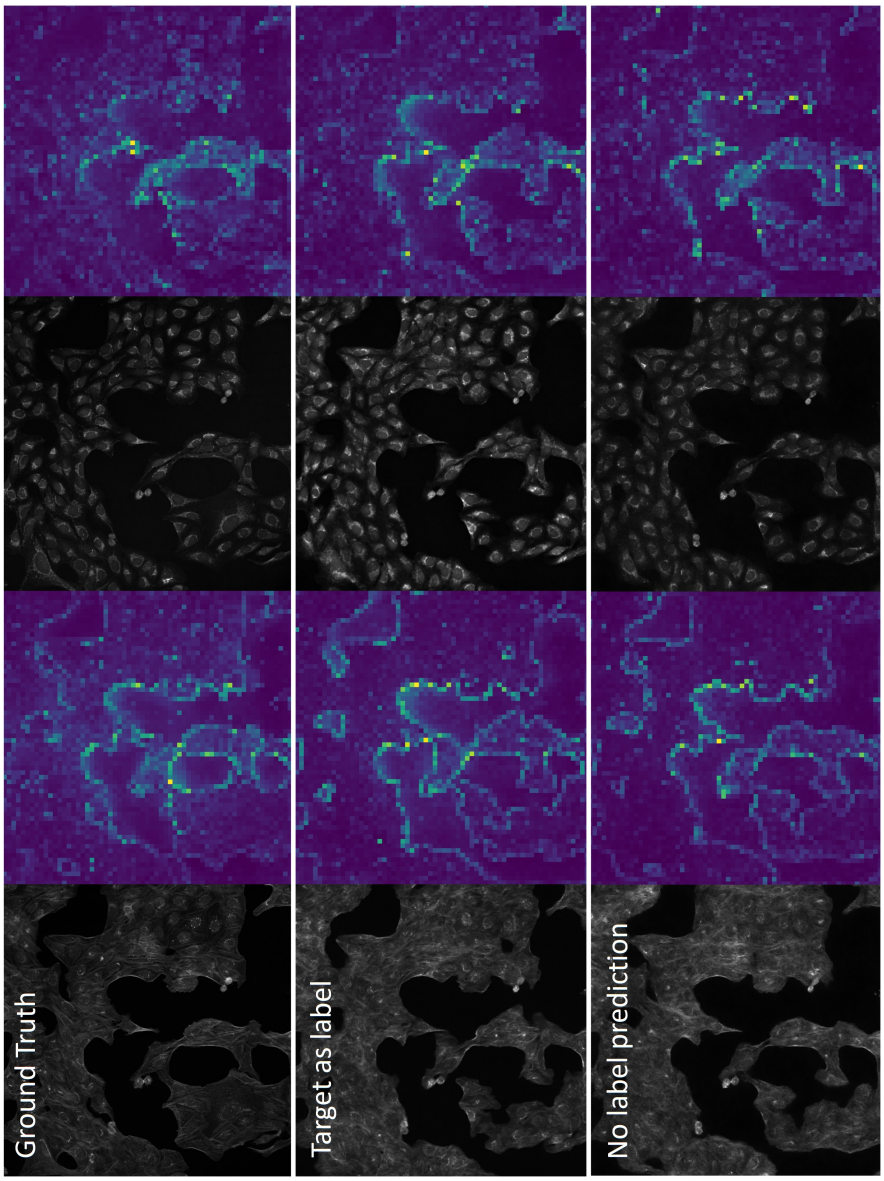


Fig. 6.7 A continuation of Fig. 6.6. Left to right: AGP, Mito.

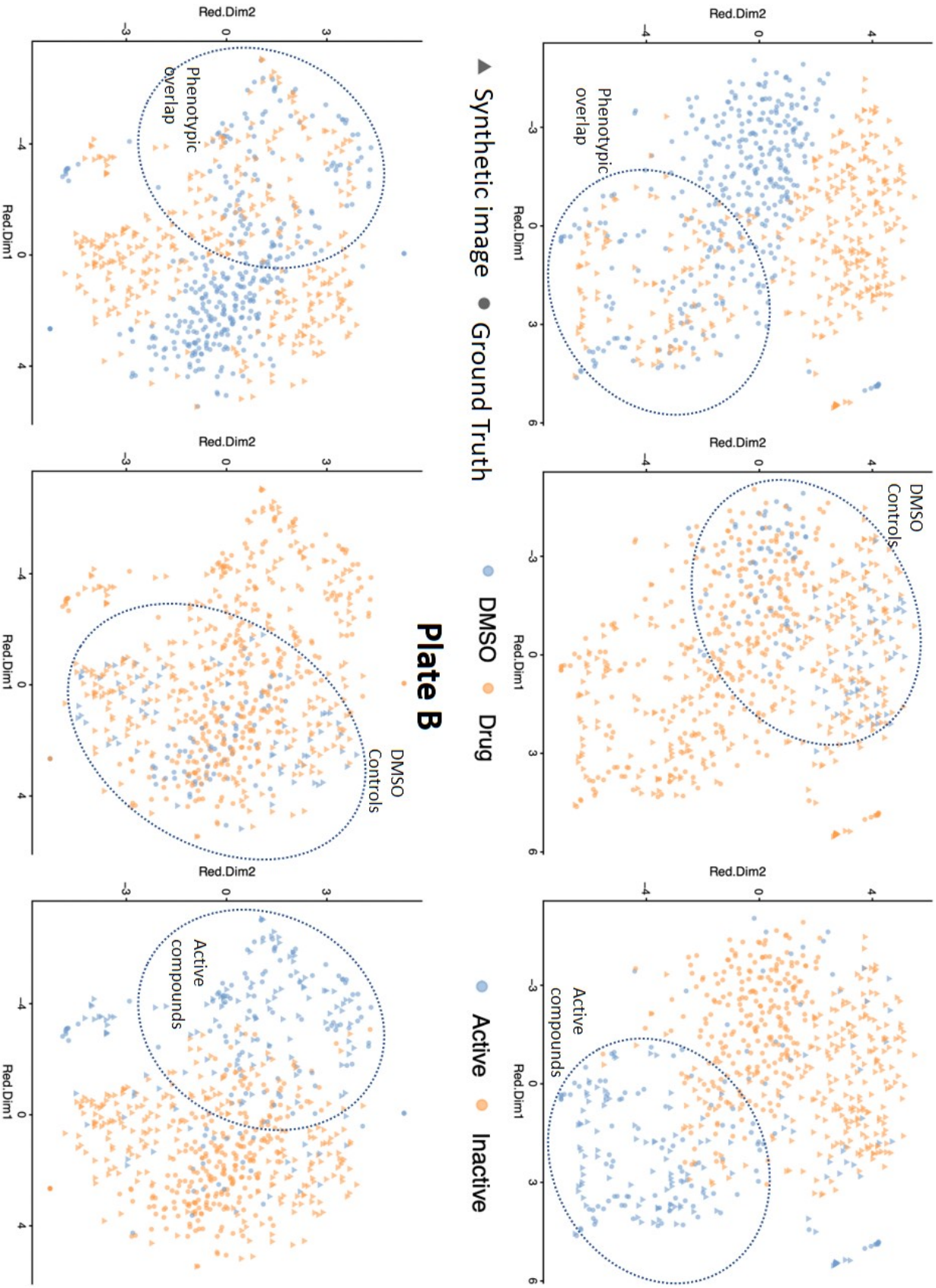


Fig. 6.8 t-SNE plots of both plates for the unlabelled model.

surpass the performance of previous methods in multiple metrics (Chapter 4). We perform image-based profiling predictions with the model predicted plates and achieve stronger results when using the freely available perturbation label with the active subset. This includes phenotypic feature correlations, SSIM and target matching, a common task in drug discovery. We propose our method could have impact in other biomedical fields to guide learning meaningful features and structures with multimodal data.

The work presented in this chapter was released as a preprint and submitted for publication in 2023.

Cross-Zamirski, J.O., Williams, G., Mouchet, E., Anand, P., Wang, Y. and Schönlieb, C.B. Class-guided image-to-image diffusion: Cell Painting from brightfield images with class labels. *arXiv Preprint* (2023). https://doi.org/10.48550/arXiv.2303.08863.

Chapter 7

Conclusions and Outlooks

7.1 Summary

We have introduced a number of new deep learning methods to the field of image-based profiling. In Chapter 4 we have presented label-free Cell Painting, and explored the utility of this approach. While it is not possible to reproduce real Cell Painting with very high accuracy, we have gone some way to revealing the potential of brightfield as modality, and also its limitations. In just two years since the study was performed, new deep learning models have surpassed GANs as the state-of-the-art in generative modelling (explored in Chapter 6). As computer vision becomes more accomplished, the future of label-free staining may have a significant impact on drug discovery pipelines by increasing the information content of assays, reducing apoptosis, phototoxicity and cytotoxicity, and allowing for imaging of live cells.

In Chapter 5 we have introduced an image-level method which is competitive with single-cell approaches to profile high-content images. Bypassing single-cell segmentation has a number of advantages including saving time, computational cost but also unlocking the ability to capture population level features and interactions between cells such as at the cell boundaries. We have used a self-supervised framework, which is one of the most powerful deep learning methods for learning feature representations, but also presented a new methodology to elegantly incorporate meaningful information. This information can also be used to force the network to learn cross-batch features, which can go some way towards overcoming the troublesome batch effect prevalent in image-based profiling.

Finally we have presented the first study to use a denoising diffusion probabilistic model to generate microscopy data. In our ambitious study we generate entire plates of synthetic Cell Painting data from unseen batches, and test this in a rigorous way

with a difficult but useful downstream task. Success on tasks such a predicting matching targets (or MOA) of perturbations or treatments is what will lead to new drug candidates. We propose that in the future, this may be possible with brightfield screens, likely incorporating carefully chosen metadata. This would accelerate and reduce the cost of drug discovery.

By working with attention mechanisms and generative models, our methods have produced images which are visually interpretable. These kind of methods may be more willingly adopted into drug discovery pipelines run by interdisciplinary teams of scientists from many backgrounds.

7.1.1 Limitations of this work

Generating a synthetic dataset is a study in itself. The models are complex, slow to implement and train, and sensitive to hyperparameters. However, once images are generated the researcher is essentially back to square one - the number of ways to evaluate the synthetic data is as limitless as image-based profiling itself. It was simply not practical to then implement an entirely new study with both the synthetic data and the real data - we instead chose representative ways to evaluate and compare the models and then stopped. In future studies there is potential to expand further and attempt to answer more questions of the data.

With this in mind, it always felt like we could have gone further with each of our studies. Each study is designed to mimic one particular component of the large and complex patchwork of a drug discovery pipeline. In this artificial context, defining the aim of the study is not always easy. There is really no way to know if AI will be successful unless implemented into 10-15 year long pipelines which are run to completion. In 10-15 years the landscape of deep learning and computer vision will be vastly different, and a great proportion of current work will be redundant.

7.2 Future work and challenges

The findings of this thesis may influence the direction of future work in the following ways:

- The brightfield modality deserves more attention as a serious candidate for image-based profiling applications.
- Although the studies released in last year have already hinted at this trend, incorporating metadata into computer vision architectures will likely become a

staple in image-based profiling. Future studies should explore the most informative and biologically sensitive ways to do this.

- Self-attention and the vision transformer mean that single-cell segmentation is no longer a necessary prerequisite for image-based profiling with deep learning.
- Deep generative models are extremely capable and will likely become the focus of many future studies. We anticipate the focus of these studies to be on the quality of the features and representations extracted from generative models.

References

[1] OJ. Wouters, M. McKee, and J. Luyten. Estimated research and development investment needed to bring a new medicine to market, 2009-2018. *JAMA*, 323(9):844–853, 2020.

[2] D. Sun, W. Gao, H. Hu, and S. Zhou. Why 90% of clinical drug development fails and how to improve it? *Acta Pharmaceutica Sinica B*, 2022.

[3] J. Vamathevan, D. Clark, P. Czodrowski, I. Dunham, E. Ferran, G. Lee, B. Li, A. Madabhushi, P. Shah, M. Spitzer, and S. Zhao. Applications of machine learning in drug discovery and development. *Nat Rev Drug Discov*, 18(6):463–477, 2019.

[4] A. Mullard. Machine learning brings cell imaging promises into focus. *Nature reviews. Drug discovery*, 18(9):653–655, 2019.

[5] D. Paul, G. Sanap, S. Shenoy, D. Kalyane, K. Kalia, and RK. Tekade. Artificial intelligence in drug discovery and development. *Drug discovery today*, 26(1):80, 2021.

[6] A. Bender and I. Cortés-Ciriano. Artificial intelligence in drug discovery: what is realistic, what are illusions? part 1: ways to make an impact, and why we are not there yet. *Drug discovery today*, 26(2):511–524, 2021.

[7] B. Bruntz. What to expect from ai-enabled drug discovery in 2023. *Drug Discovery & Development*, 2022. https://www.drugdiscoverytrends.com/ai-drug-discovery-in-2023/.

[8] J. Moffat, F. Vincent, J. Lee, J. Eder, and M. Prunotto. Opportunities and challenges in phenotypic drug discovery: an industry perspective. *Nat Rev Drug Discov*, 16:531–543, 2017.

[9] SN. Chandrasekaran, H. Ceulemans, JD. Boyd, and AE. Carpenter. Image-based profiling for drug discovery: due for a machine-learning upgrade? *Nature Reviews Drug Discovery*, 20:145–159, 2021.

[10] MA. Bray, S. Singh, H. Han, CT. Davis, B. Borgeson, C. Hartland, M. Kost-Alimova, SM. Gustafsdottir, CC. Gibson, and AE. Carpenter. Cell painting, a high-content image-based assay for morphological profiling using multiplexed fluorescent dyes. *Nature Protocols*, 11:1757–1774, 2016.

[11] JC. Caicedo, S. Cooper, F. Heigwer, S. Warchal, P. Qiu, C. Molnar, AS. Vasilevich, JD. Barry, HS. Bansal, O. Kraus, M. Wawer, L. Paavolainen, MD. Herrmann, M. Rohban, J. Hung, H. Hennig, J. Concannon, I. Smith, PA. Clemons, S. Singh, P. Rees, P. Horvath, RG. Linington, and AE. Carpenter. Data-analysis strategies for image-based cell profiling. *Nat Methods*, 14:849–863, 2017.

[12] Nature portfolio: Drug screening, 2022. https://www.nature.com/subjects/drug-screening.

[13] F. Vincent, P. Loria, M. Pregel, R. Stanton, L. Kitching, K. Nocka, R. Doyonnas, C. Steppan, A. Gilbert, T. Schroeter, and M-C. Peakman. Developing predictive assays: The phenotypic screening 'rule of 3'. *Science Translational Medicine*, 7(293), 2015.

[14] F. Sams-Dodd. Target-based drug discovery: is something wrong? *Drug Discovery Today*, 10(2):139–147, 2005.

[15] K. Sonehara and Y Okada. Genomics-driven drug discovery based on disease-susceptibility genes. *Inflamm Regener*, 41(8), 2021.

[16] JP. Hughes, S. Rees, SB. Kalindjian, and KL. Philpott. Principles of early drug discovery. *British journal of pharmacology*, 162(6):1239–1249, 2011.

[17] GE. Croston. The utility of target-based discovery. *Expert Opinion on Drug Discovery*, 12(5):427–429, 2017.

[18] DC. Swinney and J. Anthony. How were new medicines discovered? *Nature reviews Drug discovery*, 10(7):507–519, 2011.

[19] F. Vincent, A. Nueda, J. Lee, M. Schenone, M. Prunotto, and M. Mercola. Phenotypic drug discovery: recent successes, lessons learned and new directions. *Nat Rev Drug Discov*, 21:899–914, 2022.

[20] SJ. Warchal, A. Unciti-Broceta, and NO. Carragher. Next-generation phenotypic screening. *Future medicinal chemistry*, 8(11):1331–1347, 2016.

[21] C. Bock, P. Datlinger, F. Chardon, MA. Coelho, MB. Dong, KA. Lawson, T. Lu, L. Maroc, TM. Norman, B. Song, G. Stanley, S. Chen, M. Garnett, W. Li, J. Moffat, LS. Qi, RS. Shapiro, J. Shendure, JS. Weissman, and X. Zhuang. High-content crispr screening. *Nat Rev Methods Primers 2*, 8, 2022.

[22] X. Yang, L. Kui, M. Tang, D. Li, K. Wei, W. Chen, J. Miao, and Y. Dong. High-throughput transcriptome profiling in drug and biomarker discovery. *Frontiers in genetics 11*, 19, 2020.

[23] W. Zheng, N. Thorne, and JC. McKew. Phenotypic screens as a renewed approach for drug discovery. *Drug discovery today*, 18(21-22):1067–1073, 2012.

[24] P. Gautam, A. Jaiswal, T. Aittokallio, H. Al-Ali, and K. Wennerberg. Phenotypic screening combined with machine learning for efficient identification of breast cancer-selective therapeutic targets. *Cell chemical biology*, 26(7):970–979.e4., 2019.

[25] TH. Pham, Y. Qiu, J. Zeng, L. Xie, and P. Zhang. A deep learning framework for high-throughput mechanism-driven phenotype compound screening and its application to covid-19 drug repurposing. *Nat Mach Intell*, 3:247–257, 2021.

[26] KA. Giuliano, RL. DeBiasio, RT. Dunlay, A. Gough, JM. Volosky, GN. Pavlakis, and D. Lansing Taylor. High-content screening: A new approach to easing key bottlenecks in the drug discovery process. *J Biomol Screen*, 2:249–259, 1997.

[27] M. Bickle. The beautiful cell: high-content screening in drug discovery. *Anal Bioanal Chem*, 398:219–226, 2010.

[28] Cell Voyager CV8000 General Specifications, Yokogawa Electric Corporation, 2017. https://web-material3.yokogawa.com/GS80H01D01-01E.pdf.

[29] A. Chessel and RE. Carazo Salas. From observing to predicting single-cell structure and function with high-throughput/high-content microscopy. *Essays in biochemistry*, 63(2):197–208, 2019.

[30] F. Zanella, JB. Lorens, and W. Link. High content screening: seeing is believing. *Trends in biotechnology*, 28(5):237–245, 2010.

[31] P. Lang, K. Yeow, A. Nichols, and A. Scheer. Cellular imaging in drug discovery. *Nat Rev Drug Discov*, 5:343–356, 2006.

[32] V. Ljosa, KL. Sokolnicki, and AE. Carpenter. Annotated high-throughput microscopy image sets for validation. *Nat Methods*, 9:637, 2012.

[33] JUMP-Target, JUMP-Cell Painting Consortium, The Broad Institute, 2022. https://github.com/jump-cellpainting/JUMP-Target.

[34] Jump-cell painting consortium, joint undertaking in morphological profiling, 2021. Broad Institute https://jump-cellpainting.broadinstitute.org/.

[35] EM. Christiansen, SJ. Yang, DM. Ando, A. Javaherian, G. Skibinski, S. Lipnick, E. Mount, A. O'Neil, K. Shah, AK. Lee, P. Goyal, W. Fedus, R. Poplin, A. Esteva, M. Berndl, LL. Rubin, P. Nelson, and S. Finkbeiner. In silico labeling: Predicting fluorescent labels in unlabeled images. *Cell*, 173(3):792–803.e19, 2018.

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

[37] A. Bullen. Microscopic imaging techniques for drug discovery. *Nat Rev Drug Discov*, 7:54–67, 2008.

[38] PD. Caie, RE. Walls, A. Ingleston-Orme, S. Daya, T. Houslay, R. Eagle, ME. Roberts, and NO. Carragher. High-content phenotypic profiling of drug response signatures across distinct cancer cells. *Molecular cancer therapeutics*, 9(6):1913–1926, 2010.

[39] V. Ljosa, PD. Caie, R. Ter Horst, KL. Sokolnicki, EL. Jenkins, S. Daya, ME. Roberts, TR. Jones, S. Singh, A. Genovesio, PA. Clemons, NO. Carragher, and AE. Carpenter. Comparison of methods for image-based profiling of cellular morphological responses to small-molecule treatment. *Journal of biomolecular screening*, 18(10):1321–1329, 2013.

[40] DAPI (4',6-diamidino-2-phenylindole), ThermoFisher Scientific, 2022. https://www.thermofisher.com/uk/en/home/life-science/cell-analysis/fluorophores/dapi-stain.html.

[41] K. Im, S. Mareninov, MFP. Diaz, and WH. Yong. An introduction to performing immunofluorescence staining. *Methods in molecular biology (Clifton, N.J.)*, 1897:299–311, 2019.

[42] AE. Carpenter, TR. Jones, MR. Lamprecht, C. Clarke, IH. Kang, O. Friman, DA. Guertin, JH. Chang, RA. Lindquist, J. Moffat, Golland P., and DM. Sabatini. Cellprofiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol.*, 7:R100, 2006.

[43] C. McQuin, A. Goodman, V. Chernyshev, L. Kamentsky, BA. Cimini, KW. Karhohs, M. Doan, L. Ding, SM. Rafelski, D. Thirstrup, W. Wiegraebe, S. Singh, T. Becker, JC. Caicedo, and AE. Carpenter. Cellprofiler 3.0: Next-generation image processing for biology. *PLOS Biology*, 16(7):e2005970, 2018.

[44] V. Lulevich, Y-P. Shih, SH. Lo, and G-Y. Liu. Cell tracing dyes significantly change single cell mechanics. *The Journal of Physical Chemistry B*, 113(18):6511–6519, 2009.

[45] D. Broadwater, M. Bates, M. Jayaram, M. Young, J. He, AL. Raithel, TW. Hamann, W. Zhang, B. Borhan, RR. Lunt, and SY. Lunt. Modulating cellular cytotoxicity and phototoxicity of fluorescent organic salts through counterion pairing. *Sci Rep*, 9:15288, 2019.

[46] T. Haraguchi, T. Shimi, T. Koujin, N. Hashiguchi, and Y. Hiraoka. Spectral imaging fluorescence microscopy. *Genes to cells: devoted to molecular cellular mechanisms*, 7(9):881–887, 2002.

[47] J. Icha, M. Weber, JC. Waters, and C. Norden. Phototoxicity in live fluorescence microscopy, and how to avoid it. *BioEssays : news and reviews in molecular, cellular and developmental biology*, 39(8), 2017.

[48] M. Purschke, N. Rubio, KD. Held, and RW. Redmond. Phototoxicity of hoechst 33342 in time-lapse fluorescence microscopy. *Photochem. Photobiol. Sci.*, 9:1634–1639, 2010.

[49] A. Kiepas, E. Voorand, F. Mubaid, PM. Siegel, and CM. Brown. Optimizing live-cell fluorescence imaging conditions to minimize phototoxicity. *Journal of cell science*, 133(4):jcs242834, 2020.

[50] RM. Levenson and JR. Mansfield. Multispectral imaging in biology and medicine: Slices of life. *Cytometry Part A*, 69A(8):748–758, 2006.

[51] R. Ali, M. Gooding, T. Szilágyi, B. Vojnovic, M. Christlieb, and M. Brady. Automatic segmentation of adherent biological cell boundaries and nuclei from brightfield microscopy images. *Machine Vision and Applications*, 23:607–621, 2012.

[52] MAS. Ali, O. Misko, S-O. Salumaa, M. Papkov, K. Palo, D. Fishman, and L. Parts. Evaluating very deep convolutional neural networks for nucleus segmentation from brightfield cell microscopy images. *SLAS DISCOVERY: Advancing the Science of Drug Discovery*, 26(9):1125–1137, 2021.

[53] A. Gupta, PJ. Harrison, H. Wieslander, J. Rietdijk, JC. Puigvert, P. Georgiev, C. Wählby, O. Spjuth, and I-M. Sintorn. Is brightfield all you need for mechanism of action prediction? *BioRxiv preprint*, 2022. https://doi.org/10.1101/2022.10.12.511869.

[54] A. Sauvat, G. Cerrato, J. Humeau, M. Leduc, O. Kepp, and G. Kroemer. High-throughput label-free detection of dna-to-rna transcription inhibition using brightfield microscopy and deep neural networks. *Computers in Biology and Medicine*, 133:104371, 2021.

[55] C. Ounkomol, S. Seshamani, MM. Maleckar, F. Collman, and GR. Johnson. Label-free prediction of three-dimensional fluorescence images from transmitted-light microscopy. *Nature Methods*, 15(11):917–920, 2018.

[56] S. Imboden, X. Liu, BS. Lee, MC. Payne, CJ. Hsieh, and NYC. Lin. Investigating heterogeneities of live mesenchymal stromal cells using ai-based label-free imaging. *Sci Rep*, 11:6728, 2021.

[57] S. Helgadottir, B. Midtvedt, J. Pineda, A. Sabirsh, CB. Adiels, S. Romeo, D. Midtvedt, and G. Volpe. Extracting quantitative biological information from bright-field cell images using deep learning. *Biophysics Reviews*, 2(3):031401, 2021.

[58] H. Kobayashi, C. Lei, Y. Wu, A. Mao, Y. Jian, B. Guo, Y. Ozeki, and K. Goda. Label-free detection of cellular drug responses by high-throughput bright-field imaging and machine learning. *Sci Rep*, 7:12454, 2017.

[59] A. Pahl and S. Sievers. The cell painting assay as a screening tool for the discovery of bioactivities in new chemical matter. *Methods Mol Biol*, 1888:115–126, 2019.

[60] M-A. Trapotsi, E. Mouchet, G. Williams, T. Monteverde, K. Juhani, R. Turkki, F. Miljković, A. Martinsson, L. Mervin, KR. Pryde, E. Müllers, I. Barrett, O. Engkvist, A. Bender, and K. Moreau. Morphological profiling enables high-throughput screening for proteolysis targeting chimera (protac) phenotypic signature. *ACS Chemical Biology*, 12(7):1733–1744, 2022.

[61] C. Willis, J. Nyffeler, and J. Harrill. Phenotypic profiling of reference chemicals across biologically diverse cell types using the cell painting assay. *SLAS discovery: advancing life sciences R&D*, 25(7):755–769, 2020.

[62] MH. Rohban, S. Singh, X. Wu, JB. Berthet, MA. Bray, Y. Shrestha, X. Varelas, JS. Boehm, and AE. Carpenter. Systematic morphological profiling of human gene and allele function via cell painting. *eLife*, 6:e24060, 2017.

[63] BA. Cimini, SN. Chandrasekaran, M. Kost-Alimova, L. Miller, A. Goodale, B. Fritchman, P. Byrne, S. Garg, N. Jamali, DJ. Logan, HB. Concannon, C-H. Lardeau, E. Mouchet, S. Singh, SH. Abbasi, P. Aspesi Jr, JD. Boyd, T. Gilbert, D. Gnutt, S. Hariharan, D. Hernandez, G. Hormel, K. Juhani, M. Melanson, L. Mervin, T. Monteverde, JE. Pilling, A. Skepner, SE. Swalley, A. Vrcic, E. Weisbart, G. Williams, A. Yu, B. Zapiec, and AE. Carpenter. Optimizing the cell painting assay for image-based profiling. *BioRxiv preprint*, 2022. https://doi.org/10.1101/2022.07.13.499171.

[64] JC. Caicedo, J. Arevalo, F. Piccioni, MA. Bray, CL. Hartland, X. Wu, AN. Brooks, AH. Berger, JS. Boehm, AE. Carpenter, and S. Singh. Cell painting predicts impact of lung cancer variants. *Molecular biology of the cell*, 33(6):ar49, 2022.

[65] K. Heiser, PF. McLean, CT. Davis, B. Fogelson, HB. Gordon, P. Jacobson, B. Hurst, B. Miller, RW. Alfa, BA. Earnshaw, and ML. Victors. Identification of potential treatments for covid-19 through artificial intelligence-enabled phenomic analysis of human cells infected with sars-cov-2. *BioRxiv preprint*, 2020.

[66] J. Nyffeler, C. Willis, R. Lougee, A. Richard, K. Paul-Friedman, and JA. Harrill. Bioactivity screening of environmental chemicals using imaging-based high-throughput phenotypic profiling. *Toxicology and applied pharmacology*, 389:114876, 2020.

[67] PJ. Hajduk, T. Gerfin, JM. Boehlen, M. Häberli, D. Marek, and SW. Fesik. High-throughput nuclear magnetic resonance-based screening. *Journal of medicinal chemistry*, 42(13):2315–2317, 1999.

[68] L-P. Li, B-S. Feng, J-W. Yang, CL. Chang, Y. Bai, and H-W. Liu. Applications of ambient mass spectrometry in high-throughput screening. *Analyst, The Royal Society of Chemistry*, 138(11):3097–3103, 2013.

[69] LM. Mayr and D. Bojanic. Novel trends in high-throughput screening. *Current Opinion in Pharmacology*, 9(5):580–588, 2009.

[70] GP. Way, H. Spitzer, P. Burnham, A. Raj, F. Theis, S. Singh, and AE. Carpenter. Image-based profiling: a powerful and challenging new data type. *Pacific Symposium on Biocomputing*, 27:407–411, 2022.

[71] C. Scheeder, F. Heigwer, and M. Boutros. Machine learning and image-based profiling in drug discovery. *Current opinion in systems biology*, 10:43–52, 2018.

[72] A. Perakis, A. Gorji, S. Jain, K. Chaitanya, S. Rizza, and E. Konukoglu. Contrastive learning of single-cell phenotypic representations for treatment classification. *arXiv preprint*, 2021. https://doi.org/10.1007%2F978-3-030-87589-3_58.

[73] DM. Ando, CY. McLean, and M. Berndl. Improving phenotypic measurements in high-content imaging screens. *BioRxiv preprint*, page 161422, 2017. https://doi.org/10.1101/161422.

[74] OZ. Kraus, JL. Ba, and BJ. Frey. Classifying and segmenting microscopy images with deep multiple instance learning. *Bioinformatics*, 32(12):i52–i59, 2016.

[75] WJ. Godinez, I. Hossain, SE. Lazic, JW. Davies, and X. Zhang. A multi-scale convolutional neural network for phenotyping high-content cellular images. *Bioinformatics (Oxford, England)*, 33(13):2010–2019, 2017.

[76] S. Singh, MA. Bray, TR. Jones, and AE Carpenter. Pipeline for illumination correction of images for high-throughput microscopy. *Journal of microscopy*, 256(3):231–236, 2014.

[77] JC. Caicedo, C. McQuin, A. Goodman, S. Singh, and AE. Carpenter. Weakly supervised learning of single-cell feature embeddings. *Proceedings. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 9309–9318, 2018.

[78] MW. Lafarge, JC. Caicedo, AE. Carpenter, JP. Pluim, S. Singh, and M. Veta. Capturing single cell phenotypic variation via unsupervised representation learning. *Proceedings of The 2nd International Conference on Medical Imaging with Deep Learning, PMLR*, 2019.

[79] MH. Rohban, HS. Abbasi, S. Singh, and AE Carpenter. Capturing single-cell heterogeneity via data fusion improves image-based profiling. *Nat. Commun.*, 10:2082, 2019.

[80] N. Moshkov, M. Bornholdt, S. Benoit, C. McQuin, M. Smith, A. Goodman, R. Senft, Y. Han, M. Babadi, P. Horvath, BA. Cimini, AE. Carpenter, S. Singh, and JC. Caicedo. Learning representations for image-based profiling of perturbations. *bioRxiv preprint*, 2022.

[81] GP. Way, M. Kost-Alimova, T. Shibue, WF. Harrington, S. Gill, F. Piccioni, T. Becker, H. Shafqat-Abbasi, WC. Hahn, AE. Carpenter, F. Vazquez, and Singh S. Predicting cell health phenotypes using image-based morphology profiling. *Molecular biology of the cell*, 32(9):995–1005, 2021.

[82] F. Fuchs, G. Pau, D. Kranz, O. Sklyar, C. Budjan, S. Steinbrink, T. Horn, A. Pedal, W. Huber, and M. Boutros. Clustering phenotype populations by genome-wide rnai and multiparametric imaging. *Molecular systems biology*, 6(1):370, 2010.

[83] T. Stoeger, N. Battich, MD. Herrmann, Y. Yakimovich, and L. Pelkmans. Computer vision for image-based transcriptomics. *Methods (San Diego, Calif.)*, 85:44–53, 2015.

[84] CellProfiler Published Pipelines, 2022. https://cellprofiler.org/published-pipelines.

[85] G. Montavon, W. Samek, and KR. Müller. Methods for interpreting and understanding deep neural networks. *Digital signal processing*, 73:1–15, 2018.

[86] AL. Beam, AK. Manrai, and M. Ghassemi. Challenges to the reproducibility of machine learning models in health care. *Jama*, 323(4):305–306, 2020.

[87] N. Pawlowski, JC. Caicedo, S. Singh, AE. Carpenter, and A. Storkey. Automating morphological profiling with generic deep convolutional networks. *BioRxiv preprint*, 2016. https://doi.org/10.1101/085118.

[88] AX. Lu, OZ. Kraus, S. Cooper, and AM. Moses. Learning unsupervised feature representations for single cell microscopy images with paired cell inpainting. *PLoS Comput Biol*, 15(9):e1007348, 2019.

[89] L. Taylor and G. Nitschke. Improving deep learning with generic data augmentation. *In 2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1542–1547, 2018.

[90] C. Sommer, c. Straehle, U. Köthe, and FA. Hamprecht. Ilastik: Interactive learning and segmentation toolkit. *2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 230–233, 2011.

[91] J. Xu, D. Zhou, D. Deng, J. Li, C. Chen, X. Liao, G. Chen, and PA. Heng. Deep learning in cell image analysis. *Intelligent Computing*, 2022.

[92] P. Goldsborough, N. Pawlowski, JC. Caicedo, S. Singh, and AE. Carpenter. Cytogan: Generative modeling of cell images. *BioRxiv preprint*, 2017. https://doi.org/10.1101/227645.

[93] C. Doersch and A. Zisserman. Multi-task self-supervised visual learning. *arXiv preprint*, 2017. https://doi.org/10.48550/arxiv.1708.07860.

[94] R. Hadsell, Chopra S., and LeCun Y. Dimensionality reduction by learning an invariant mapping. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1735–1742, 2006.

[95] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A Joulin. learning of visual features by contrasting cluster assignments. *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[96] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A Joulin. Emerging properties in self-supervised vision transformers. *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[97] T. Truong, S. Mohammadi, and M. Lenga. How transferable are self-supervised features in medical image classification tasks? *Proceedings of Machine Learning for Health, in Proceedings of Machine Learning Research*, 158:54–74, 2021.

[98] R. Janssens, X. Zhang, A. Kauffmann, A. de Weck, and EY. Durand. Fully unsupervised deep mode of action learning for phenotyping high-content cellular images. *Bioinformatics (Oxford, England)*, page btab497 Advance online publication, 2021.

[99] Y. Ji, M. Cutiongco, BS. Jensen, and K. Yuan. Cp2image: generating high-quality single-cell images using cellprofiler representations. *In NeurIPS 2022 Workshop on Learning Meaningful Representations of Life*, 2022.

[100] S. Li, S. Besson, C. Blackburn, M. Carroll, RK. Ferguson, H. Flynn, K. Gillen, R. Leigh, D. Lindner, M. Linkert, and WJ. Moore. Metadata management for high content screening in omero. *Methods*, 96:27–32, 2016.

[101] N. Moshkov, T. Becker, K. Yang, P. Horvath, V. Dancik, BK. Wagner, PA. Clemons, S. Singh, AE. Carpenter, and JC. Caicedo. Predicting compound activity from phenotypic profiles and chemical structures. *bioRxiv preprint*, 2022.

[102] J. Deng, W. Dong, R. Socher, LJ. Li, L. Kai, and FF. Li. Imagenet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[103] S. Wang, M. Lu, N. Moshkov, JC. Caicedo, and BA. Plummer. Anchoring to exemplars for training mixture-of-expert cell embeddings. *arXiv preprint*, 2021.

[104] A. Lin and A. Lu. Incorporating knowledge of plates in batch normalization improves generalization of deep learning for microscopy images. *Machine Learning in Computational Biology*, pages 74–93, 2022.

[105] M-A. Trapotsi, LH. Mervin, AM. Afzal, N. Sturm, O Engkvist, IP. Barrett, and A. Bender. Comparison of chemical structure and cell morphology information for multitask bioactivity predictions. *Journal of Chemical Information and Modeling*, 61(3):1444–1456, 2021.

[106] GP. Way, T. Natoli, A. Adeboye, L. Litichevskiy, A. Yang, X. Lu, JC. Caicedo, BA. Cimini, K. Karhohs, DJ. Logan, MH. Rohban, M. Kost-Alimova, K. Hartland, M. Bornholdt, SN. Chandrasekaran, M. Haghighi, E. Weisbart, S. Singh, A. Subramanian, and AE. Carpenter. Morphology and gene expression profiling provide complementary information for mapping cell state. *Cell Systems*, 13(11):911–923.e9, 2022.

[107] YL. Chow, S. Singh, AE. Carpenter, and GP. Way. Predicting drug polypharmacology from cell morphology readouts using variational autoencoder latent space arithmetic. *PLoS computational biology*, 18(2):p.e1009888, 2022.

[108] O. Méndez-Lucio, PAM. Zapata, J. Wichard, D. Rouquié, and DA. Clevert. Cell morphology-guided de novo hit design by conditioning generative adversarial networks on phenotypic image features. *chemrxiv preprint*, 2020.

[109] S. Jenkinson, F. Schmidt, LR. Ribeiro, A. Delaunois, and JP. Valentin. A practical guide to secondary pharmacology in drug discovery. *Journal of Pharmacological and Toxicological Methods*, 105:106869, 2020.

[110] S. Dara, S. Dhamercherla, SS. Jadav, CH. Babu, and MJ. Ahsan. Machine learning in drug discovery: a review. *Artificial Intelligence Review*, 55(3):1947–1999, 2022.

[111] X. Zeng, X. Tu, Y. Liu, X. Fu, and Y. Su. Toward better drug discovery with knowledge graph. *Current opinion in structural biology*, 72:114–126, 2022.

[112] M. Elbadawi, S. Gaisford, and AW. Basit. Advanced machine-learning techniques in drug discovery. *Drug Discovery Today*, 26(3):769–777, 2021.

[113] J. Meyers, B. Fabian, and N. Brown. De novo molecular design and generative models. *Drug Discovery Today*, 26(11):2707–2715, 2021.

[114] ZH. Zhou. Machine learning. *Springer Nature*, 2021.

[115] Y. LeCun, Y. Bengio, and G Hinton. Deep learning. *Nature*, 521:436–444, 2015.

[116] CM. Bishop. Neural networks and their applications. *Review of scientific instruments*, 65(6):1803–1832, 1994.

[117] OI. Abiodun, A. Jantan, AE. Omolara, KV. Dada, NA. Mohamed, and H. Arshad. State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11):e00938, 2018.

[118] R. Kitchin. The data revolution: Big data, open data, data infrastructures and their consequences. *Sage*, 2014.

[119] CA. Mack. Fifty years of moore's law. *IEEE Transactions on semiconductor manufacturing*, 24(2):202–207, 2011.

[120] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[121] A.F. Agarap. Deep learning using rectified linear units (relu). *arXiv preprint*, 2018. https://doi.org/10.48550/arXiv.1803.08375.

[122] AL. Maas, AY. Hannun, and AY. Ng. Rectifier nonlinearities improve neural network acoustic models. *In Proc. icml*, 30(1):3, 2013.

[123] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[124] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, AN. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[125] D. Rumelhart, G. Hinton, and R. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[126] S. Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[127] D.P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[128] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, and A. Desmaison. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[129] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, and M. Kudlur. TensorFlow: a system for Large-Scale machine learning. *In 12th USENIX symposium on operating systems design and implementation (OSDI 16)*, pages 265–283, 2016.

[130] K. O'Shea and R. Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

[131] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[132] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[133] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint*, 2020. https://doi.org/10.48550/arXiv.2010.11929.

[134] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *Lecture Notes in Computer Science*, vol 9351 Springer, Cham, 2015.

[135] CM. Hyun, HP. Kim, SM. Lee, S. Lee, and JK. Seo. Deep learning for undersampled mri reconstruction. *Physics in medicine and biology*, 63(13):135007, 2018.

[136] KH. Jin, MT. McCann, E. Froustey, and M. Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 26(9):4509–4522, 2017.

[137] F. Isensee, PF. Jaeger, S Kohl, J. Petersen, and KH. Maier-Hein. nnu-net: a self-configuring method for deep learningbased biomedical image segmentation. *Nature Methods*, 18(2):203–211, 2021.

[138] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[139] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *arXiv preprint*, 2021. https://arxiv.org/abs/2105.05233.

[140] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *arXiv preprint*, 2020. https://arxiv.org/abs/2006.11239.

[141] C. Saharia, W. Chan, H. Chang, CA. Lee, J. Ho, T Salimans, DJ. Fleet, and M. Norouzi. Palette: Image-to-image diffusion models. *arXiv preprint*, 2021. https://arxiv.org/abs/2111.05826.

[142] A. Steiner, A. Kolesnikov, X. Zhai, R. Wightman, J. Uszkoreit, and L. Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021.

[143] R. Shao, Z. Shi, J. Yi, PY. Chen, and CJ. Hsieh. On the adversarial robustness of vision transformer. *arXiv preprint arXiv:2103.15670*, 2021.

[144] H. Zhang, I. Goodfellow, D. Metaxas, and A Odena. Self-attention generative adversarial networks. *In International conference on machine learning*, pages 7354–7363, 2019.

[145] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, and A. Dosovitskiy. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34:12116–12128, 2021.

[146] C. Matsoukas, JF. Haslum, M. Söderberg, and K. Smith. Is it time to replace cnns with transformers for medical images? *arXiv preprint*, 2021. https://doi.org/10.48550/arxiv.2108.09038.

[147] S. Spiegel, I. Hossain, C. Ball, and X Zhang. Metadata-guided visual representation learning for biomedical images. *BioRxiv preprint*, page 725754, 2019.

[148] J. Simm, G. Klambauer, A. Arany, M. Steijaert, JK. Wegner, E. Gustin, V. Chupakhin, YT. Chong, J. Vialard, P. Buijnsters, I. Velter, A. Vapirev, S. Singh, AE. Carpenter, R. Wuyts, S. Hochreiter, Y. Moreau, and H. Ceulemans. Repurposing high-throughput image assays enables biological activity prediction for drug discovery. *Cell chemical biology*, 25(5):611–618.e3, 2018.

[149] D. Hendrycks, M. Mazeika, S. Kadavath, and D. Song. Using self-supervised learning can improve model robustness and uncertainty. *Advances in neural information processing systems*, 32, 2019.

[150] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. *In International conference on machine learning*, pages 1597–1607, 2020.

[151] A. Jaiswal, AR. Babu, MZ. Zadeh, D. Banerjee, and F. Makedon. A survey on contrastive self-supervised learning. *Technologies*, 9(1), 2020.

[152] SJ. Pan and Q. Yang. A survey on transfer learning. *in IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[153] OZ. Kraus, BT. Grys, J. Ba, Y. Chong, BJ. Frey, C. Boone, and BJ. Andrews. Automated analysis of high-content microscopy data with deep learning. *Molecular systems biology*, 13(4):924, 2017.

[154] SBZ. Hua, AX. Lu, and AM. Moses. Cytoimagenet: A large-scale pretraining dataset for bioimage transfer learning. *arXiv preprint arXiv:2111.11646*, 2021.

[155] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S Ozair, A. Courville, and Y Bengio. Generative adversarial nets. *Adv. Neural Inf. Process. Syst*, 27:2672–2680, 2014.

[156] X. Chen, N. Mishra, M. Rohaninejad, and P. Abbeel. Pixelsnail: An improved autoregressive generative model. *In International Conference on Machine Learning*, pages 864–872, 2018.

[157] DP. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint*, 2013. https://arxiv.org/abs/1312.6114.

[158] I. Kobyzev, SJ. Prince, and MA. Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE transactions on pattern analysis and machine intelligence*, 43(11), 2020.

[159] M. Arjovsky and L. Bottou. Towards principled methods for training generative adversarial networks. *arXiv preprint*, 2017.

[160] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and AA. Bharath. Generative adversarial networks: An overview. *in IEEE Signal Processing Magazine*, 35(1):53–65, 2018.

[161] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. *International Conference on Machine Learning, ICM*, 2017.

[162] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and AC. Courville. Improved training of wasserstein gans. *In Advances in Neural Information Processing Systems (NIPS)*, pages 5769–5779, 2017.

[163] F. Buggenthin, C. Marr, M. Schwarzfischer, PS. Hoppe, O. Hilsenbeck, T. Schroeder, and FJ. Theis. An automatic method for robust and fast cell detection in bright field images from high-throughput microscopy. *BMC Bioinformatics*, 14:297, 2013.

[164] M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv*, 2014. https://doi.org/10.48550/arxiv.1411.1784.

[165] P. Isola, JY. Zhu, T. Zhou, and AA. Efros. Image-to-image translation with conditional adversarial networks. *In Proc. IEEE Conference on Computer Vision and Pattern Recognition*, page 1125–1134, 2017.

[166] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila. Analyzing and improving the image quality of stylegan. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020.

[167] AH. Bermano, R. Gal, Y. Alaluf, R. Mokady, Y. Nitzan, O. Tov, O. Patashnik, and D. Cohen-Or. State-of-the-art in the architecture, methods and applications of stylegan. *In Computer Graphics Forum*, 41(2):591–611, 2022.

[168] A. Brock, J. Donahue, and K. Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint*, 2018.

[169] CB. Jin, H. Kim, M. Liu, W. Jung, S. Joo, E. Park, YS. Ahn, IH. Han, JI. Lee, and X. Cui. Deep ct to mr synthesis using paired and unpaired data. *Sensors*, 19(10):2361, 2019.

[170] V. Sandfort, K. Yan, PJ. Pickhardt, and RM. Summers. Data augmentation using generative adversarial networks (cyclegan) to improve generalizability in ct segmentation tasks. *Sci Rep*, 9:16884, 2019.

[171] K. Armanious, C. Jiang, M. Fischer, T. Küstner, T. Hepp, K. Nikolaou, S. Gatidis, and B. Yang. Medgan: Medical image translation using gans. *Computerized medical imaging and graphics*, 79:101684, 2020.

[172] D. Martinelli. Generative machine learning for de novo drug discovery: A systematic review. *Computers in Biology and Medicine*, page 105403, 2022.

[173] WW. Qian, C. Xia, S. Venugopalan, A. Narayanaswamy, M. Dimon, GW. Ashdown, J. Baum, J. Peng, and DM. Ando. Batch equalization with a generative adversarial network. *Bioinformatics*, 36:i875–i883, 2020.

[174] S. Arora, A. Risteski, and Y. Zhang. Do gans learn the distribution? some theory and empirics. *In International Conference on Learning Representations*, 2018.

[175] JP. Cohen, M. Luck, and S. Honari. Distribution matching losses can hallucinate features in medical image translation. *In International conference on medical image computing and computer-assisted intervention*, pages 529–536, 2018. Springer, Cham.

[176] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *In International Conference on Machine Learning. PMLR*, pages 2256–2265, 2015.

[177] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

[178] Y. Song, J. Sohl-Dickstein, DP. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint*, 2020. https://arxiv.org/abs/2011.13456.

[179] Y. Wu and K. He. Group normalization. *arXiv preprint*, 2018. https://doi.org/10.48550/arxiv.1803.08494.

[180] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint*, 2022. https://arxiv.org/abs/2204.06125.

[181] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, SKS. Ghasemipour, BK. Ayan, SS. Mahdavi, RG. Lopes, T. Salimans, J. Ho, DJ. Fleet, and M. Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint*, 2022. https://doi.org/10.48550/arxiv.2205.11487.

[182] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *arXiv preprint*, 2021.

[183] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems, 30*, 2017.

[184] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[185] JA. O'Reilly and F. Asadi. Pre-trained vs. random weights for calculating fréchet inception distance in medical imaging. *13th Biomedical Engineering International Conference (BMEiCON)*, pages 1–4, 2021.

[186] Z. Wang, AC. Bovik, HR. Sheikh, and EP. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[187] J. Snell, K. Ridgeway, R. Liao, BD. Roads, MC. Mozer, and RS. Zemel. Learning to generate images with perceptual similarity metrics. *In 2017 IEEE International Conference on Image Processing (ICIP)*, pages 4277–4281, 2017.

[188] A. Horé and D. Ziou. Image quality metrics: Psnr vs. ssim. *20th International Conference on Pattern Recognition*, pages 2366–2369, 2010.

[189] K. Pearson. Vii. mathematical contributions to the theory of evolution.—iii. regression, heredity, and panmixia. *Philosophical Transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character*, 187:253–318, 1896.

[190] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image superresolution using a generative adversarial network. *In Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 105–114, 2017.

[191] T. Nguyen, V. Bui, A. Thai, V. Lam, C. Raub, LC. Chang, and G. Nehmetallah. Virtual organelle self-coding for fluorescence imaging via adversarial learning. *J Biomed Opt*, 25(9):096009, 2020.

[192] U. Sara, M. Akter, and M. Uddin. Image quality assessment through fsim, ssim, mse and psnr—a comparative study. *Journal of Computer and Communications*, 7:8–18, 2019.

[193] Z. Yu, Q. Xiang, J. Meng, C. Kou, Q. Ren, and Y. Lu. Retinal image synthesis from multiple-landmarks input with generative adversarial networks. *Biomedical engineering online,*, 18(1):62, 2019.

[194] L. Ma, R. Shuai, X. Ran, W. Liu, and C. Ye. Combining dc-gan with resnet for blood cell image classification. *Med Biol Eng Comput*, 58:1251–1264, 2020.

[195] Z. Zhang, Q. Liu, and Y. Wang. Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):749–753, 2018.

[196] TC. Wang, MY. Liu, JY. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[197] Z. Zhou, MM. Rahman Siddiquee, N. Tajbakhsh, and J. Liang. Unet++: A nested u-net architecture for medical image segmentation. *In Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*, pages 3–11, 2018.

[198] P. Schober, C. Boer, and LA. Schwarte. Correlation coefficients: Appropriate use and interpretation. *Anesthesia and analgesia*, 126(5):1763–1768, 2018.

[199] GP. Way. Blocklist features - cell profiler. figshare. dataset., 2020. https://doi.org/10.6084/m9.figshare.10255811.v3.

[200] MM. Mukaka. Statistics corner: A guide to appropriate use of correlation coefficient in medical research. *Malawi Med J.*, 24(3):69–71, 2012.

[201] L. McInnes, J. Healy, N. Saul, and L. Großberger. Umap: uniform manifold approximation and projection. *J. Open Source Softw.*, 3:861, 2018.

[202] Umap: Uniform manifold approximation and projection for dimension reduction, 2018. Leland McInnes Revision 23b789e0 https://umaplearn.readthedocs.io/en/latest/.

[203] H. Wieslander, A. Gupta, E. Bergman, E. Hallström, and PJ. Harrison. Learning to see colours: Biologically relevant virtual staining for adipocyte cell images. *PloS one*, 16(10):e0258546, 2021.

[204] V. Ghodrati, J. Shao, M. Bydder, Z. Zhou, W. Yin, KL. Nguyen, Y. Yang, and P. Hu. Mr image reconstruction using deep learning: evaluation of network structure and loss functions. *Quantitative imaging in medicine and surgery*, 9(9):1516–1527, 2019.

[205] JF. Pambrun and R. Noumeir. Limitations of the ssim quality metric in the context of diagnostic imaging. *Proc. IEEE Int. Conf. Image Process. (ICIP)*, page 2960–2963, 2015.

[206] S. Cheng, S. Fu, YM. Kim, W. Song, Y. Li, Y. Xue, J. Yi, and L. Tian. Single-cell cytometry via multiplexed fluorescence prediction by label-free reflectance microscopy. *Sci. Adv.*, 7(3):eabe0431, 2021.

[207] Y. Liu, H. Yuan, Z. Wang, and S. Ji. Global pixel transformers for virtual staining of microscopy images. *in IEEE Transactions on Medical Imaging*, 39(6):2256–2266, 2020.

[208] A. Pratapa, M. Doron, and JC Caicedo. Image-based cell phenotyping with deep learning. *Current opinion in chemical biology*, 65:9–17, 2021.

[209] F. Iorio, R. Bosotti, E. Scacheri, V. Belcastro, P. Mithbaokar, R. Ferriero, L. Murino, R. Tagliaferri, N. Brunetti-Pierri, A. Isacchi, and D. di Bernardo. iscovery of drug mode of action and drug repositioning from transcriptional responses. *Proceedings of the National Academy of Sciences of the United States of America*, 104(33):14621–14626, 2010.

[210] E. Weissler, T. Naumann, T. Andersson, R. Ranganath, O. Elemento, Y. Luo, DF. Freitag, J. Benoit, MC. Hughes, F. Khan, P. Slater, K. Shameer, M. Roe, E. Hutchison, SH. Kollins, U. Broedl, Z. Meng, JL. Wong, L. Curtis, E. Huang, and M. Ghassemi. The role of machine learning in clinical research: transforming the future of evidence generation. *Trials*, 22:537, 2021.

[211] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint*, 2015. https://doi.org/10.48550/arxiv.1503.02531.

[212] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[213] YNT. Vu, R. Wang, N. Balachandar, C. Liu, AY. Ng, and P. Rajpurkar. Medaug: Contrastive learning leveraging patient metadata improves representations for chest x-ray interpretation. *In Machine Learning for Healthcare Conference*, pages 755–769, 2021.

[214] I. Loshchilov and F Hutter. Decoupled weight decay regularization. *ICLR*, 2019.

[215] A. Arnold, R. Nallapati, and WW. Cohen. A comparative study of methods for transductive transfer learning. *In Seventh IEEE international conference on data mining workshops (ICDMW 2007)*, pages 77–82, 2007.

[216] M. Assran, M. Caron, I. Misra, P. Bojanowski, F. Bordes, P. Vincent, A. Joulin, M. Rabbat, and N Ballas. Masked siamese networks for label-efficient learning. *arXiv preprint*, 2022. https://doi.org/10.48550/arXiv.2204.07141.

[217] C. Saharia, J. Ho, W. Chan, T. Salimans, DJ. Fleet, and M. Norouzi. Image super-resolution via iterative refinement. *arXiv preprint*, 2021. https://arxiv.org/abs/2104.07636.

[218] A. Radford, JW. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, and G. Krueger. Learning transferable visual models from natural language supervision. *In International conference on machine learning*, pages 8748–8763, 2021.

[219] M. Wu and N. Goodman. Multimodal generative models for scalable weakly-supervised learning. *Advances in Neural Information Processing Systems*, 31, 2018.

[220] F. Zhan, Y. Yu, R. Wu, J. Zhang, and S. Lu. Multimodal image synthesis and editing: A survey. *arXiv preprint*, 2022. https://arxiv.org/abs/2112.13592.

[221] H. Sun, R. Mehta, HH. Zhou, Z. Huang, SC. Johnson, V. Prabhakaran, and V. Singh. Dual-glow: Conditional flow-based generative model for modality transfer. *In Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10611–10620, 2019.

[222] Y. Song, L. Shen, L. Xing, and S. Ermon. Solving inverse problems in medical imaging with score-based generative models. *arXiv preprint*, 2021. https://arxiv.org/abs/2111.08005.

[223] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[224] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, Y. Shao, W. Zhang, B. Cui, and M-H. Yang. Diffusion models: A comprehensive survey of methods and applications. *arXiv preprint*, 2022. https://arxiv.org/abs/2209.00796.

[225] X. Yi, E. Walia, and P. Babyn. Generative adversarial network in medical imaging: A review. *Medical image analysis*, 58:101552, 2019.

[226] C. Belthangady and LA. Royer. Applications, promises, and pitfalls of deep learning for fluorescence image reconstruction. *Nat Methods*, 16:1215–1225, 2019.

[227] DP. Kingma and P. Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems, 31*, 2018.

[228] Y. Shi, B. Paige, and P. Torr. Variational mixture-of-experts autoencoders for multi-modal deep generative models. *Advances in Neural Information Processing Systems, 32*, 2019.

[229] X. Huang, A. Mallya, TC. Wang, and MY. Liu. Multimodal conditional image synthesis with product-of-experts gans. *In European Conference on Computer Vision*, pages 91–109, 2022. Springer, Cham.

[230] A. Kazerouni, EK. Aghdam, M. Heidari, R. Azad, M. Fayyaz, I. Hacihaliloglu, and D. Merhof. Diffusion models for medical image analysis: A comprehensive survey. *arXiv preprint*, 2022. https://arxiv.org/abs/2211.07804.

[231] P. Sanchez, A. Kascenas, X. Liu, AQ. O'Neil, and SA. Tsaftaris. What is healthy? generative counterfactual diffusion for lesion localization. *arXiv preprint*, 2022. https://arxiv.org/abs/2207.12268.

[232] J. Wolleb, F. Bieder, R. Sandkühler, and PC. Cattin. Diffusion models for medical anomaly detection. *arXiv preprint*, 2022. https://arxiv.org/abs/2203.04306.

[233] T. Brooks, A. Holynski, and AA. Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint*, 2022. https://arxiv.org/abs/2211.09800.

[234] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

[235] SN. Chandrasekaran, BA. Cimini, A. Goodale, L. Miller, M. Kost-Alimova, N. Jamali, J. Doench, B. Fritchman, A. Skepner, M. Melanson, J. Arevalo, JC. Caicedo, D. Kuhn, D. Hernandez, J. Berstler, H. Shafqat-Abbasi, D. Root, S. Swalley, S. Singh, and AE. Carpenter. Three million images and morphological profiles of cells treated with matched chemical and genetic perturbations. *bioRxiv*, 2022.

[236] M. Sypetkowski, M. Rezanejad, S. Saberian, O. Kraus, J. Urbanik, J. Taylor, B. Mabey, M. Victors, J. Yosinski, AR. Sereshkeh, and I. Haque. Rxrx1: A dataset for evaluating experimental batch correction methods. *arXiv preprint arXiv:2301.05768*, 2023.

[237] MM. Fay, O. Kraus, M. Victors, L. Arumugam, K. Vuggumudi, J. Urbanik, K. Hansen, S. Celik, N. Cernek, G. Jagannathan, and J. Christensen. Rxrx3: Phenomics map of biology. *bioRxiv*, pages 2023–02, 2023.

Appendix A

Additional Results

In this section we present figures from the three experimental studies which were not included in the main body of the thesis.

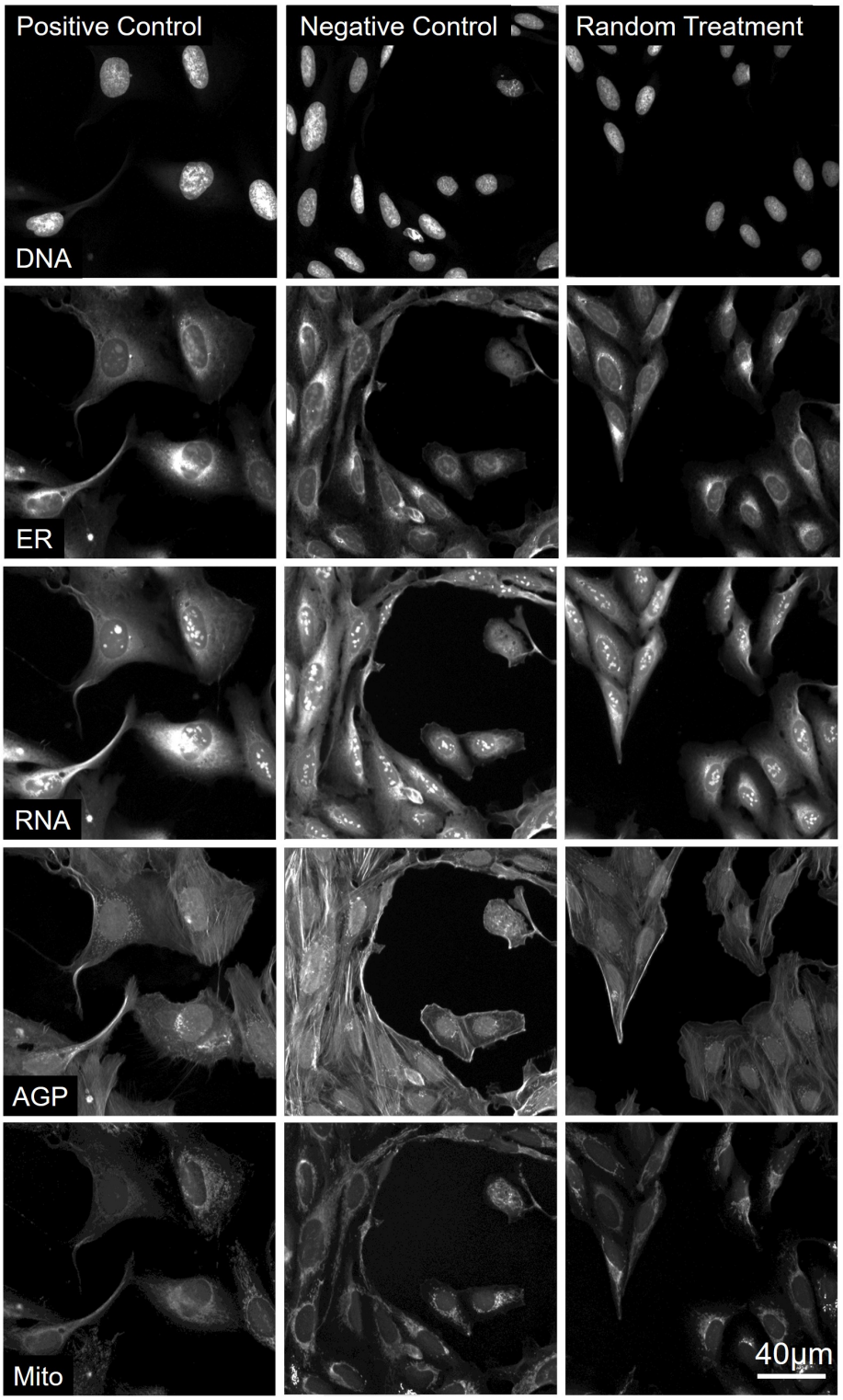


Fig. A.1 Typical example of different treatment groups in Cell Painting image channels from the dataset used in Chapter 4: positive control, negative control and random treatment. Five channels are displayed for each example to highlight visual differences between the treated cells, notably the increased size and sparsity of the cells in the positive control group.

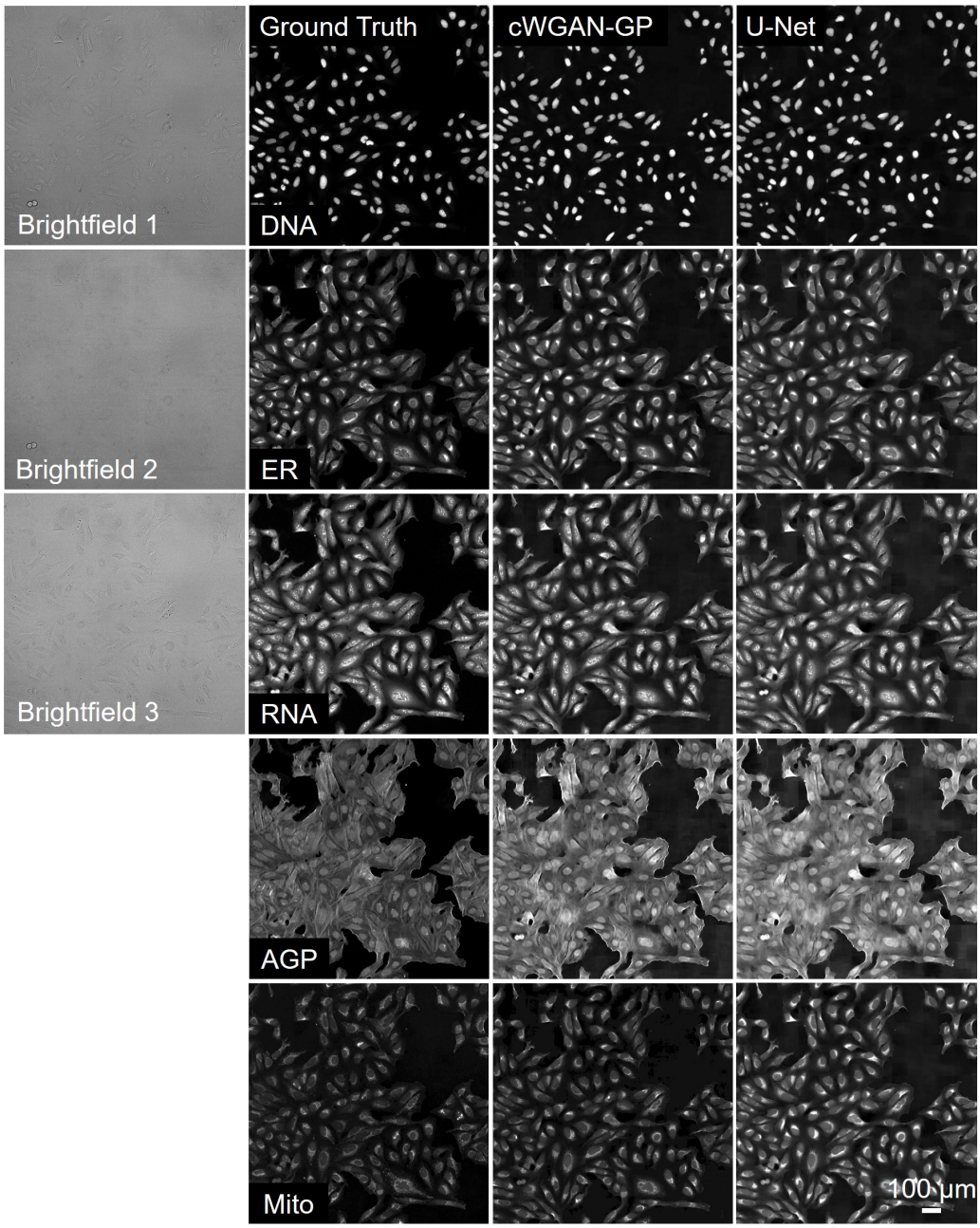


Fig. A.2 A typical example of brightfield, ground truth fluorescent, and predicted channels from the Chapter 4 test dataset for the U-Net and cWGAN-GP models. The images are as they are used in the CellProfiler analysis (998 x 998 pixels), representing a full field of view. Images are independently contrasted for visualization

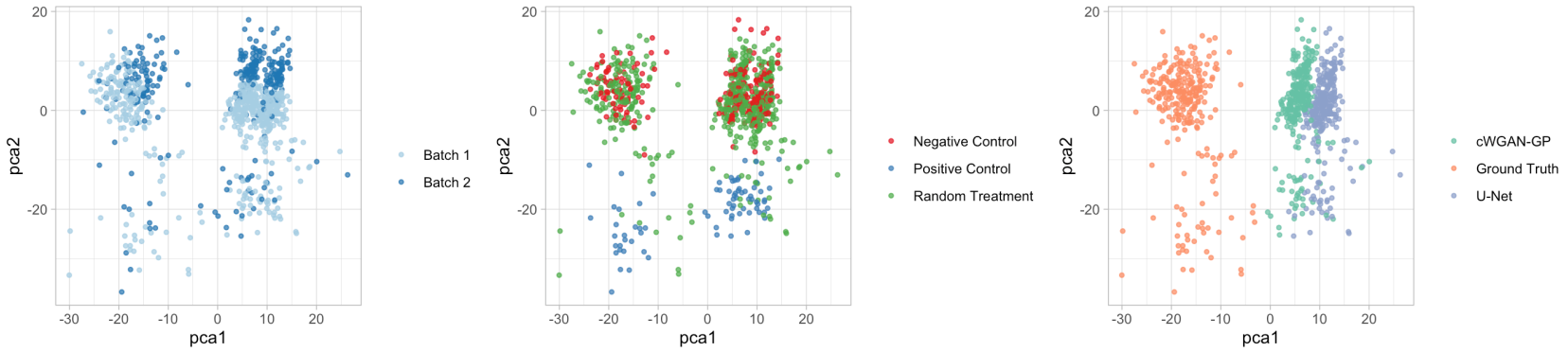


Fig. A.3 PCA is used to replicate the UMAP results (Fig. 4.7), showing that that both models can reproduce the separation between treatments and batches seen in the ground truth features. cWGAN-GP lies closer to the ground truth than U-Net, although the two models sit much closer to each other in feature space than to the ground truth. This plot was made with Riku Turrki.

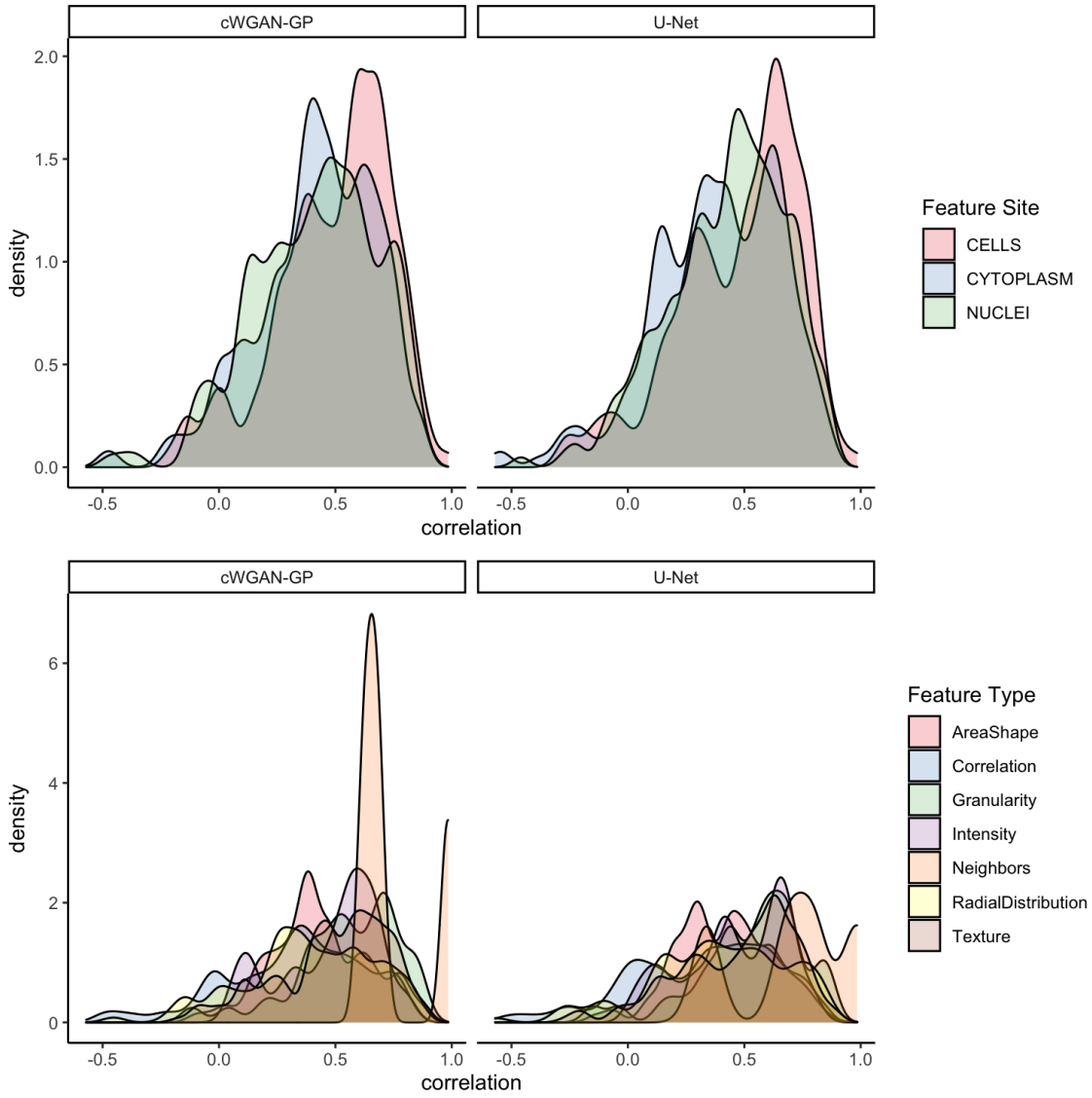


Fig. A.4 Density plots of feature correlation to ground truth for both models (U-Net and cWGAN-GP) in Chapter 4 by Feature Site and Feature Type (for all features after feature selection). This plot was made with Riku Turkki.

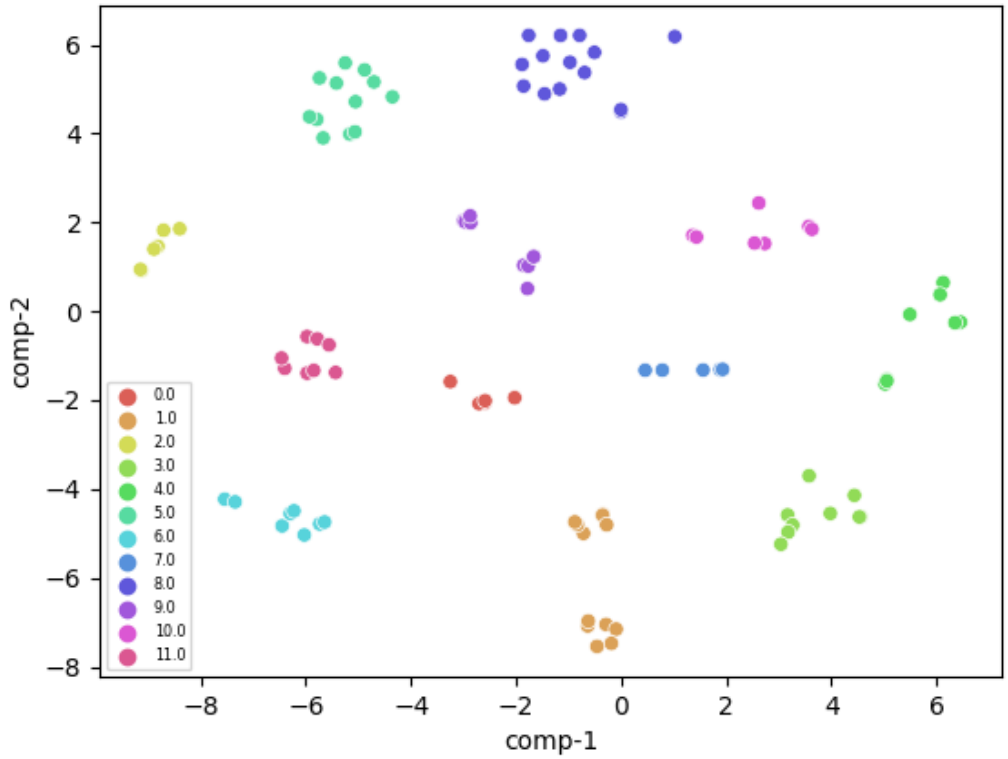


Fig. A.5 Two-dimensional t-SNE plot of each aggregated treatment feature training WS-DINO with MOA as the weak label: 100% NSC and 100% NSCB MOA classification.

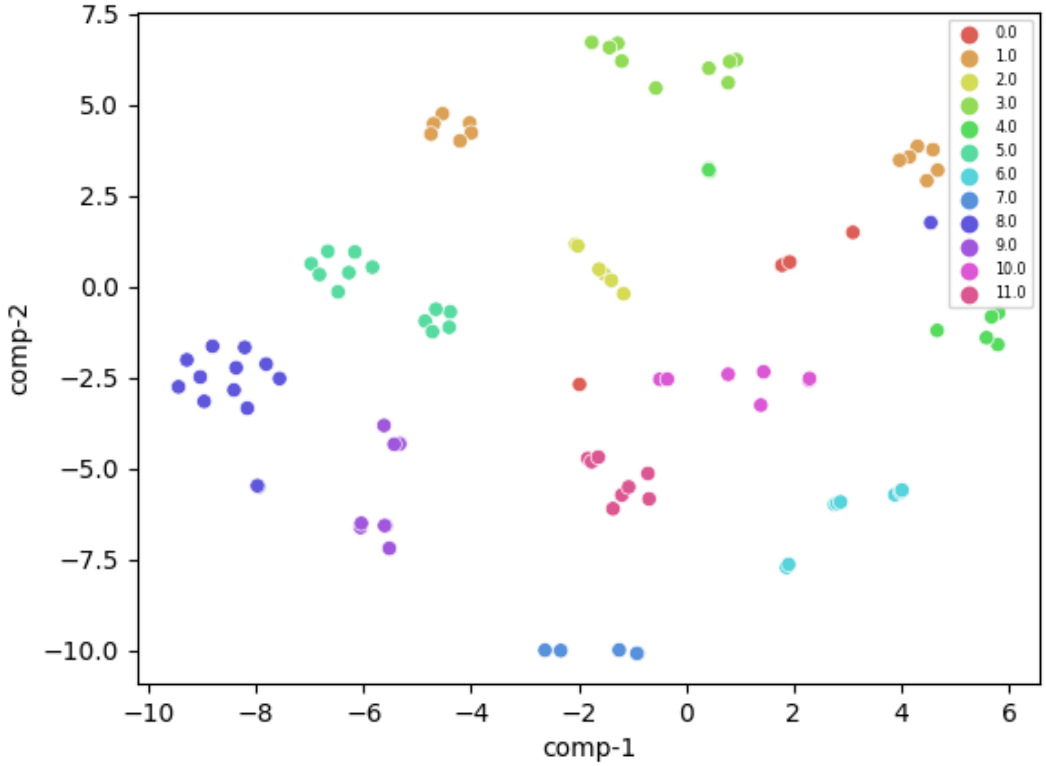


Fig. A.6 Two-dimensional t-SNE plot of each aggregated treatment feature training WS-DINO with treatment as the weak label: 92% NSC and 90% NSCB MOA classification.

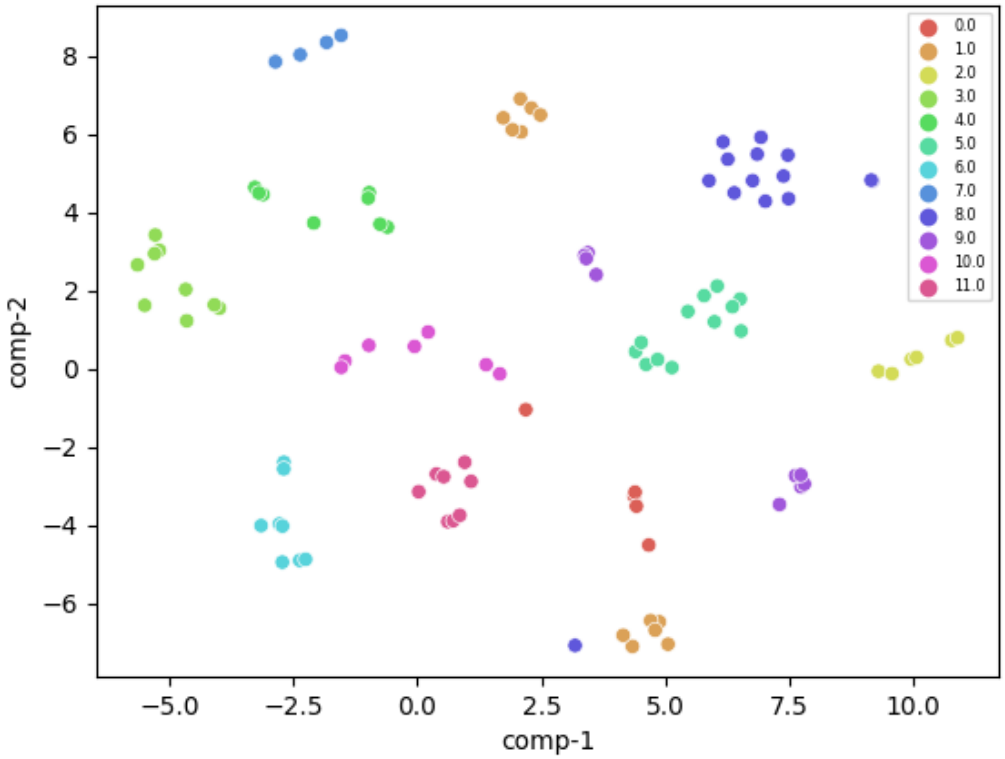


Fig. A.7 Two-dimensional t-SNE plot of each aggregated treatment feature training DINO with no labels: 92% NSC and 90% NSCB MOA classification.

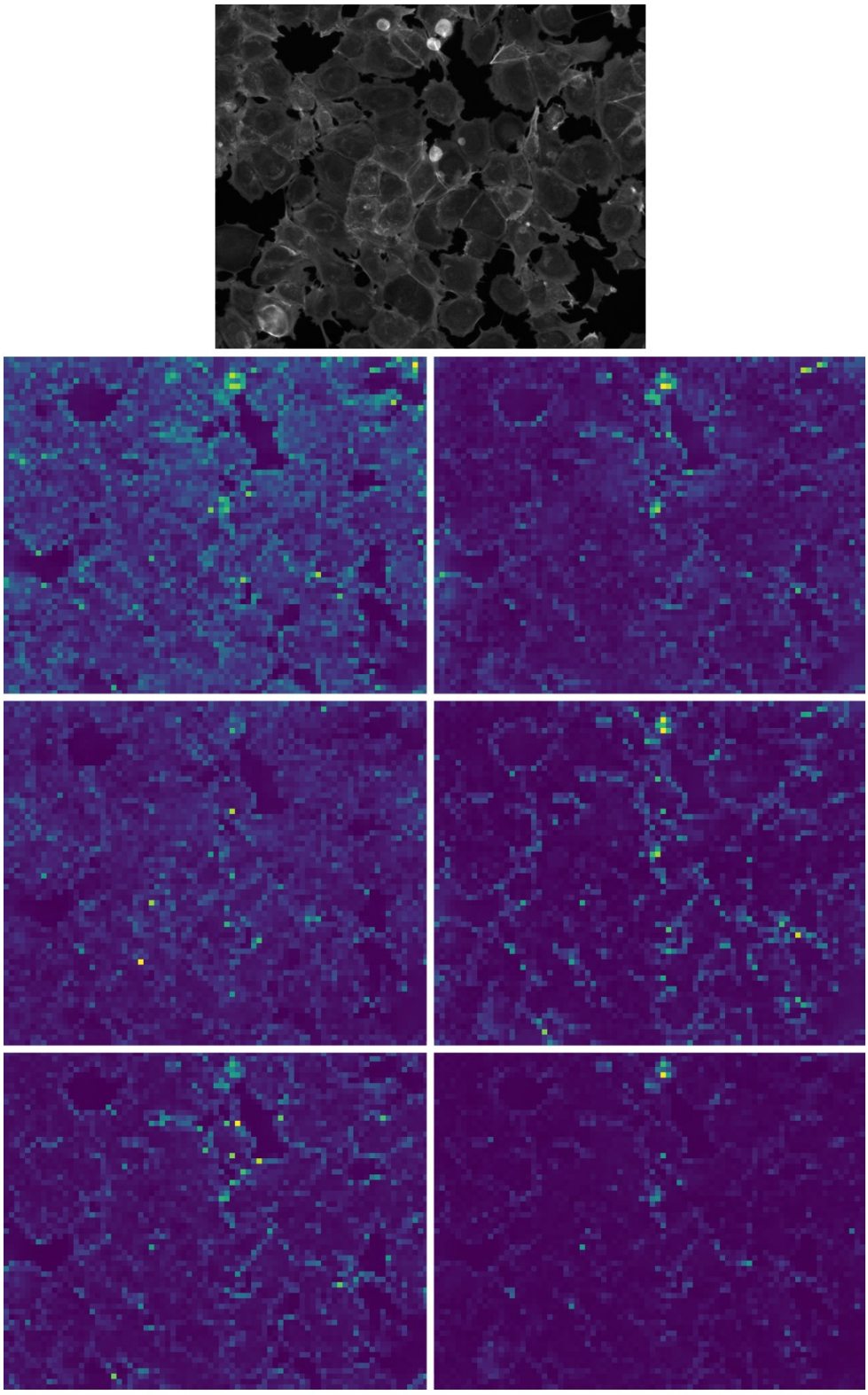


Fig. A.8 Multi-head self attention example with an image from the F-actin channel of the BBBC021 dataset. The ViT backbone has six attention heads which can all be visualised - in other figures in this thesis we typically only display one due to space constraints.

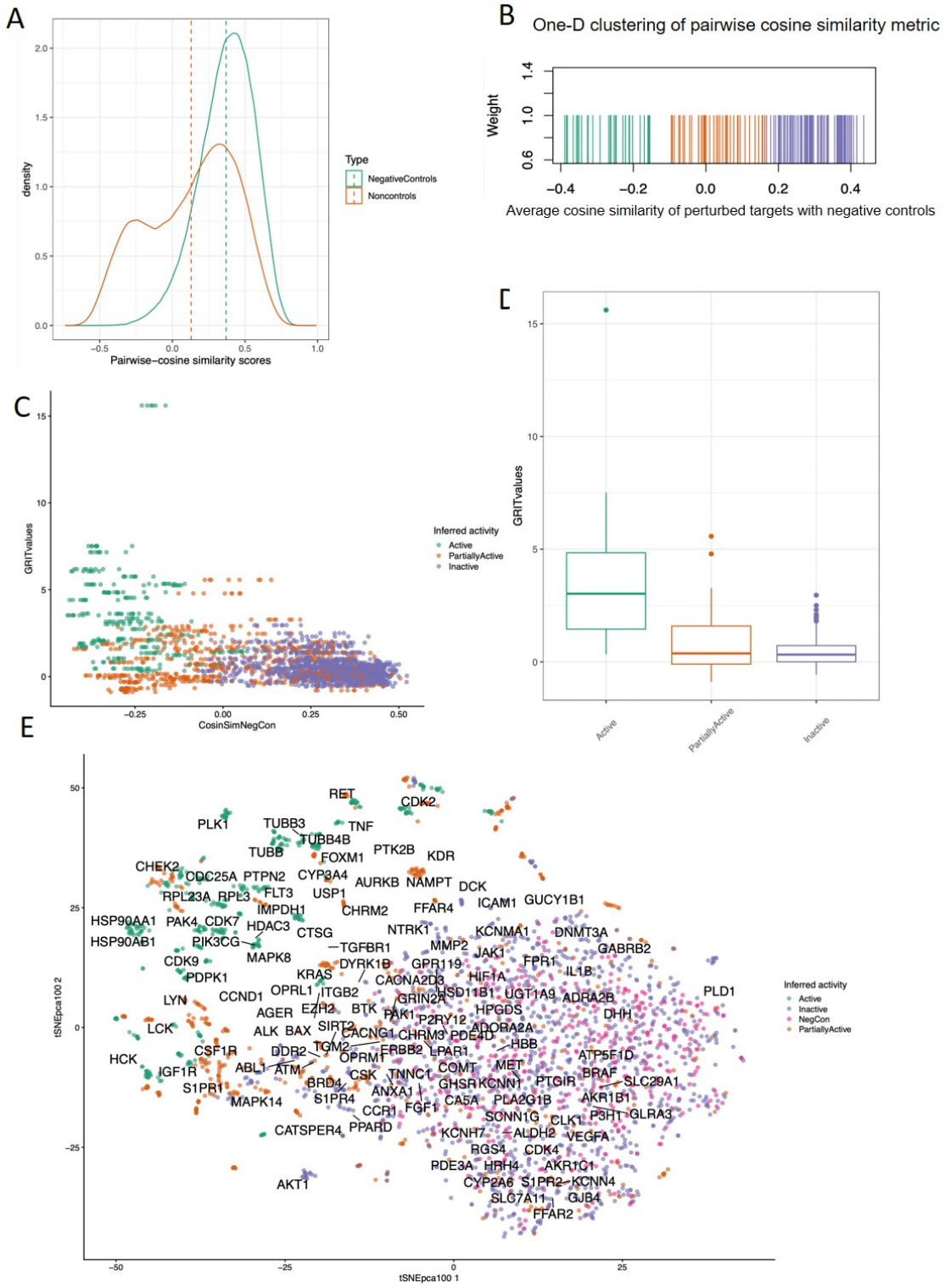


Fig. A.9 Inferring the activity from CellProfiler features with the Target-2 dataset. (A) Distribution of all pairwise cosine similarity scores derived from top 100 PCA dimensions across the negative controls and the drug treatments. (B) One-dimensional K-means clustering of the average cosine similarity metric computed between targets and negative controls. (C) Scatter plot of the GRIT values computed for each target and the corresponding cosine similarity metric calculated from negative controls. (D) Box plot depicting varying values of GRIT scores across inferred target activity. (E) The t-stochastic neighbor embedding reduced dimensional plot of all the 10 TARGET-2 plates colored based on the inferred target activity. This plot was made with Praveen Anand.

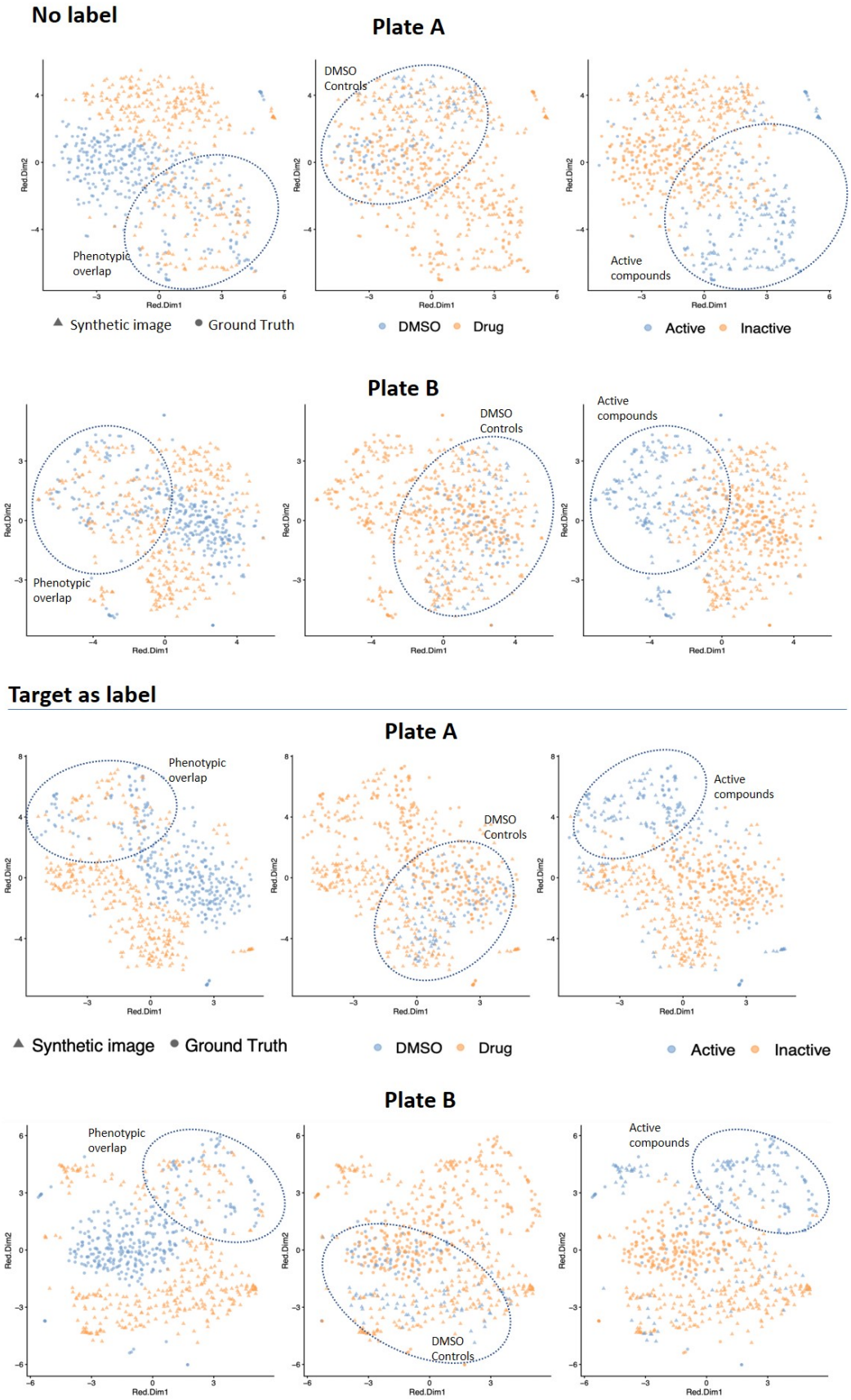


Fig. A.10 A comparison of the t-SNE plots comparing the real and predicted CellProfiler features for the image-to-image diffusion model with and without labels.

Appendix B

Resources

Almost all the deep learning models and data analysis pipelines used in this thesis were implemented in python. We provide the code used for the models used in all three experimental chapters in pubic GitHub repositories. There is additional material related to the studies and their publications available in these repositories, including some raw data and large supplementary tables, training logs etc.

Label-Free Prediction of Cell Painting from Brightfield Images

https://github.com/crosszamirski/Label-free-prediction-of-Cell-Painting-from-brightfield-images

Self-Supervised Learning of Phenotypic Representations from Cell Images with Weak Labels

https://github.com/crosszamirski/WS-DINO

Class-guided image-to-image diffusion: Enhanced Cell Painting from brightfield images with weak labels

https://github.com/crosszamirski/guided-I2I*

*As this is unpublished work at the time of submission, this repository may not be public yet.

I would like to express my gratitude to the owners and contributors of the public repositories which were extremely useful for this work:

https://github.com/pytorch/pytorch

https://github.com/facebookresearch/dino

https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix

https://github.com/Janspiry/Palette-Image-to-Image-Diffusion-Models

https://github.com/openai/guided-diffusion

https://github.com/cytomining/pycytominer

https://github.com/jump-cellpainting/JUMP-Target