

# Bridging Generalization Gaps in High Content Imaging Through Online Self-Supervised Domain Adaptation

Johan Fredin Haslum<sup>1,2,3,\*†</sup> Christos Matsoukas<sup>1,2,3,†</sup> Karl-Johan Leuchowius<sup>3</sup> Kevin Smith<sup>1,2</sup>

<sup>1</sup> KTH Royal Institute of Technology, Stockholm, Sweden <sup>2</sup> Science for Life Laboratory, Stockholm, Sweden <sup>3</sup> AstraZeneca, Gothenburg, Sweden

## Abstract

High Content Imaging (HCI) plays a vital role in modern drug discovery and development pipelines, facilitating various stages from hit identification to candidate drug characterization. Applying machine learning models to these datasets can prove challenging as they typically consist of multiple batches, affected by experimental variation, especially if different imaging equipment have been used. Moreover, as new data arrive, it is preferable that they are analyzed in an online fashion. To overcome this, we propose CODA, an online self-supervised domain adaptation approach. CODA divides the classifier's role into a generic feature extractor and a task-specific model. We adapt the feature extractor's weights to the new domain using cross-batch self-supervision while keeping the task-specific model unchanged. Our results demonstrate that this strategy significantly reduces the generalization gap, achieving up to a 300% improvement when applied to data from different labs utilizing different microscopes. CODA can be applied to new, unlabeled out-of-domain data sources of different sizes, from a single plate to multiple experimental batches.

## 1. Introduction

High Content Imaging (HCI) plays a pivotal role in modern drug discovery and development, being used throughout preclinical drug discovery cascades. It can capture detailed phenotypic responses of cells treated with compounds or genetic perturbants, and reveal complex sub-cellular processes. Machine learning can help analyze HCI data to unveil biological correlations, reveal mode-of-action, predict compound bioactivity, and predict toxicities [20, 26, 30, 31, 36]. Recent advances in ML for HCI have helped accelerate screening of compound libraries, enhanced data interpretation, and enabled novel therapeutic

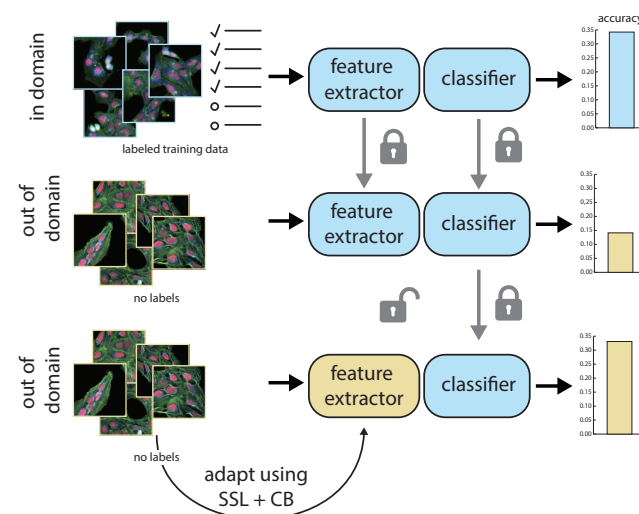


Figure 1. Illustration of the proposed approach. CODA divides the role of a classifier into two separate entities: one model designed to extract generic features and a subsequent task-specific model that operates on these features to accomplish the given task. We then adapt the feature extractor when testing on a new source using self-supervision, while leaving the task-specific model untouched. The result is a model that can be easily adapted to new out-of-domain data as it arrives, seeing a significant boost in performance, without the need for any labels.

insights [18]. However, several challenges hinder its full potential.

In particular, the generalization gap presents significant challenges to HCI and drug discovery. Discrepancies caused by variations in experimental conditions, apparatus, biological noise, and the presence of random or systematic errors can impede model performance. Limited ability of standard ML models to adapt or transfer across HCI settings results in reduced predictive accuracy [15]. To fully harness the power of machine learning in HCI, there is a need for robust and adaptable models that can generalize effectively across different contexts and conditions without

\*Corresponding author: Johan Fredin Haslum <jhaslum@kth.se>

†These authors contributed equally to this work.

compromising performance.

Recently, ten pharmaceutical companies, six supporting technology companies, and two non-profit partners formed the JUMP-CP (Joint Undertaking in Morphological Profiling) initiative to generate phenotypic response data for over 116,750 unique compounds, over-expression of 12,602 genes, and knockout of 7,975 genes using CRISPR-Cas9, all in human osteosarcoma cells (U2OS) [3]. The dataset is estimated to be 115 TB in size and captures 1.6 billion cells and their single-cell profiles using the Cell Painting assay [1,6]. A subset of the compounds profiled by the consortium were profiled across all the participating labs, using the different equipment setups available in the individual labs. This dataset offers a unique opportunity to develop and test domain adaptation methods for HCI.

In this study, we leverage the newly-released JUMP-CP data to develop and validate a new approach for online self-supervised domain adaptation (SSDA), Cross-batch Online Domain Adaptation (CODA) which uses cross-batch self-supervision to adapt a feature extractor to incoming out-of-domain data. By using the setup illustrated in Figure 1, where the model is separated into an adaptable feature extractor and a frozen task-specific classifier, we are able realize huge improvements, up to 300%, when the model is applied to data from different labs or different microscopes. Crucially, this can be done without the need for any labels for the out-of-domain data.

We test our approach using data from different institutions in the JUMP-CP data repository – training CODA using data from a source institution and performing SSDA to successfully adapt to the other institutions without access to any labels. Our contributions can be summarized as follows:

- Propose CODA, a self-supervised domain adaptation method, enabling online adaptation of a model trained on a single HCI data source to other out-of-domain sources (*e.g.*, different institution or microscope) – demonstrating its applicability to a variety of real-world experimental settings.
- Introduced ODA as an alternative approach when cross-batch consistency learning is not feasible, resulting in a slight performance drop from CODA but significant performance improvements over supervised methods.
- Conducted an extensive experimental validation on diverse subsets of data from the JUMP-CP repository, showcasing the robustness of the proposed approaches to variations in acquisition and apparatus, and verified the effectiveness of CODA in aligning the feature extractor to the target domain.

The code to reproduce our experiments can be found at <https://github.com/cfredinh/coda>.

## 2. Related Work

The problem of distributional shifts between the training and test sets is widely recognized to degrade performance in various domains [5,9,14,22]. To mitigate these effects, traditional strategies involve gathering more data or employing sophisticated augmentation techniques to incorporate test distribution-like data into the training domain [19,38]. However, these approaches may not always be feasible, as anticipating the expected domain shifts during testing is not always possible. HCI data faces similar challenges due to domain shifts [4,15]. Experimental batches in HCI data exhibit high homogeneity within themselves but have limited overlap with other batches due to inherent biological noise and variations in experimental setups. These variations are commonly referred to as *batch effects* in HCI, representing undesirable domain shifts resulting from biological noise and difficult-to-control experimental conditions.

Research in the field of addressing distribution shifts focuses on two main directions: Domain Generalization (DG) and Domain Adaptation (DA). DG aims to learn domain-invariant features from one or multiple source domains during training, using techniques focused on identifying domain-invariant features [5,12,15,27]. DA, on the other hand, leverages data from parts or the entire target domain during training, allowing for supervised or unsupervised alignment of features to handle distribution shifts [28,29,32,37]. However, anticipating all possible distribution shifts during training is impractical, resulting in limited generalization capabilities across test domains. Consequently, performance cannot be guaranteed for unknown test domains.

The concept of updating model weights *online* has recently gained attention, with [33] introducing a pre-text task for weight updates, followed by [13] using a image reconstruction task. The underlying principle in most such approaches is that either only a sample or the full test set can be used to align the test domain without relying on and data associated with the primary task. This approach has shown clear performance gains in the natural image domain *e.g.* [13,33]. Although these methods individually update weights for each sample, they are suboptimal for feature extraction in HCI data, as confirmed by our own experiments and recent findings [21]. Considering the nature of HCI data, which is often grouped into subsets such as wells, plates, or batches, adapting feature extraction strategies to these groups becomes an appealing and efficient option.

While test time domain adaptation has shown success in the Natural Imaging domain, its application in the medical and biomedical imaging domain, specifically in tasks like medical image segmentation [23] and image reconstruction [10,17], remains limited. Notably, there is a lack of research directly addressing domain shifts in medical and biomedical classification tasks, and particularly for HCI

data, where these distributional shifts overwhelmingly dominate the learning signal.

### 3. Methods

In this study, we address the challenge of generalization gaps caused by domain shifts in new sources of High Content Imaging (HCI) data. To tackle this problem, we propose a novel self-supervised domain adaptation (SSDA) strategy called CODA which is able to deal with the unique challenges associated with HCI data.

Building upon the recent work of [13], we adopt a dual-model approach, which separates the model into a feature extractor and a classifier (Figure 1). Within our framework, the feature extractor is trained in a self-supervised manner and produces features which are then processed by the classifier to solve the classification task (Figure 2). Then, when unlabeled data from a new domain is encountered, one can update only the feature extractor using self-supervision to adapt to the new domain. This allows the model to adapt the feature extractor to the new domain while preserving the ability of the classifier to make correct predictions.

However, as demonstrated in this study, directly applying this design yields very poor performance in HCI data. This is because the biological signals of interest are overshadowed by acquisition and experimental artifacts (see Table 2). To overcome this obstacle, we made adaptations inspired by [15] to modify the SSDA so that it becomes agnostic to these distracting artifacts. This allows it to learn features that better distinguish the biological signal of interest and pass them on to the frozen classifier.

**Baseline** The primary baselines we utilize in this study involve the supervised learning of a standalone Vision Transformer (ViT) model on HCI data, as commonly used. Once trained, we directly apply this model to the target dataset.

**The dual model** In our approach, we utilize a vision transformer (ViT) model to learn meaningful representations from input patches. However, the standard ViT lacks the ability to differentiate generic low-level features from task-specific ones. To address this, we take inspiration from Test-Time Training (TTT) [13, 33], which separates the model into a feature extractor and a classifier. As seen in Figure 2, the feature extractor learns generic representations through self-supervised training, while the classifier focuses on solving the specific task. In our study, we employ DINO [2], a consistency-based method, instead of reconstruction based Masked Autoencoders (MAE) [13]. This is motivated by the subpar performance of MAE in HCI data [21] instead using DINO which has show better performance than other SSL approaches in HCI [15, 21]. After the feature extractor is trained, it generates features that are used by the classification model.

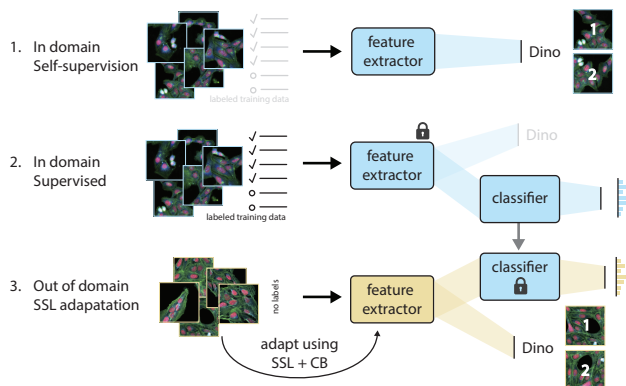


Figure 2. The training strategy and deployment of the dual model in CODA. In the first step, a labeled data source is used to pretrain a feature extractor using self-supervision. In step 2, a classifier is appended to the feature extractor and supervised training is performed on the same data source. In step 3, the model is adapted online to a new unlabeled out-of-domain data source by using cross-batch consistency learning [15] and self-supervision [2] to adapt the feature extractor while keeping the weights of the classifier frozen.

**Self-supervised domain adaptation** When a model is deployed in a new domain, the performance can be severely impacted by distribution shifts that alter the data representation, particularly in the case of HCI data, as discussed previously. These shifts primarily stem from intrinsic properties of the data rather than task-specific characteristics. In our problem the task remains constant, it is the appearance of the data that changes. Therefore, the domain adaptation method should focus on producing unbiased features that can be consumed by the classifier for the task at hand.

Adapting the features of a monolithic model to a new domain can be challenging due to the entanglement of low-level and high-level features, impacting both generic and task-specific representations. However, employing a dual model system allows for updating the feature extractor independently, while preserving the task-specific portion of the network. Inspired by this insight, we adopt the on-the-fly feature extractor update approach introduced by [13, 33].

Figure 2 illustrates the process. First, a labeled data source is utilized to pretrain the feature extractor through self-supervised learning (step 1). Subsequently, a classifier is appended to the feature extractor, and supervised training is conducted on the same data source (step 2). Finally, to adapt the model to a new unlabeled out-of-domain data source, using self-supervision techniques, specifically DINO [2] are employed, allowing the feature extractor to be updated while keeping the classifier weights frozen (step 3). By employing this approach, the classification model remains unaffected, while the feature extractor is adapted to extract generic representations suitable for the task, agnostic to the peculiarities of the data source.

Table 1. Imaging settings and data volume for the studied sources.

Source	Description	Objective	Batches	Plates	Images
S3	Opera Phoenix, widefield, laser	20X/1	13	25	85,409
S5	CV8000, confocal, laser	20X/0.75	23	24	82,256
S8	ImageExpress Micro, confocal, LED	20X/0.75	4	4	13,824
S11	Operetta, widefield, LED	20X/1	4	7	23,373
S7	CV7000, confocal, laser	20X/0.75	7	7	24,192
S10	CV8000, confocal, laser	20X/0.75	6	6	13,812

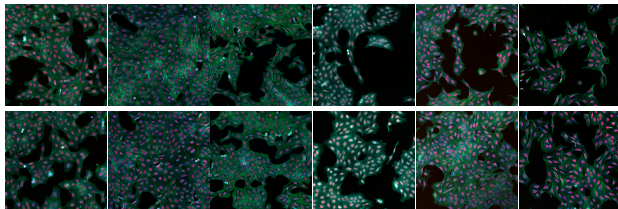


Figure 3. Example images from the six sources, showing the similarity between images of the same source but different compounds, highlighting the inherent variability and batch effects.

**Cross-batch consistency learning** In High Content Imaging (HCI), the data collection process is characterized by discrete experimental batches, leading to distribution shifts caused by variations in experimental conditions, capturing settings, and time points. These shifts are commonly referred to as *batch effects*. Ideally, each HCI image should capture only the biological effects of the treatment and no batch effects. However, in practice, batch effects often dominate the data, causing SSL methods to prioritize these confounding factors over the relevant biological signals. As a result, SSL methods tend to model batch effects rather than the desired biological signals, leading to suboptimal performance [15].

To address this challenge, Haslum et al. [15] propose a solution called Cross-Domain Consistency Learning (CBCL), which builds upon the principles of consistency-based SSL methods. CDCL leverages the concept of consistent representation between pairs of images that share the same treatment but come from different domains. The underlying assumption is that when the network is presented with two images of the same treatment but from different batches, the shared signal of interest should be the biological signal rather than the batch effects. We adopt this strategy in the Self-Supervised in domain and adaptation step (steps 1 and 3 in Figure 2) to mitigate the influence of batch effects and enhance the robustness of the self-supervised feature extractor.

**Test Time Training** Beyond the supervised baseline, we also include a recent Online Domain Adaptations approach, Test-Time Training (TTT) [13]. TTT used a dual-model setup with a MAE [16] as a feature extractor with a classification model stacked on top of it, these are trained in

sequence. At test time the FE is update for each test image individually, see [13] for more details.

## 4. Experimental Setup

In this section, we describe the datasets, task used and the implementations details of the methods described in the previous section 3. Starting with the task, we focus on Mechanism-of-Action (MoA) prediction. The MoA of a compound describes how a substance produces a pharmacological effect, often involving determining which target, such as proteins or enzymes, the substance interacts with. Understanding the MoA can help in predicting potential drug interactions, side effects, and can help guide the design of new, more effective drugs or therapies.

**Dataset** We conducted experiments using different subsets of the JUMP-CP Cell Painting High Content Imaging set [3]. The JUMP-CP dataset encompasses a wide range of compound and genetic perturbations that were imaged using an optimized version of the Cell Painting assay [1,6]. This dataset was generated through collaborations between multiple institutions, making it an ideal choice for studying domain shifts due to its diverse origins and the inclusion of various microscope types and settings. See Figure 3 for image examples.

For our analyses, we selected a subset of perturbations from all participating institutions of the JUMP-CP initiative. This subset comprised 302 unique compounds that were imaged across 15 different centers, coming from the JUMP-CP TARGET2 plates. It encompassed a total of 120 experimental batches and covered 141 distinct plates. As targets for our study, we aim to predict the Mechanism of Action (MoA) information associated with the compounds, which was obtained from the Drug Repurposing Hub [7]. Among the compounds, 135 had single MoA labels, representing 54 unique MoA types. With the goal of predicting the MoA of each the compounds, we treat the problem as a multi-class classification task.

Our main experiments focus on a subset of the data from four (anonymized) partners within the JUMP-CP consortium [3]. This subset consists of images captured using different microscopes and microscope types, with variations in objectives used. These four sources were selected as they were the largest subsets of data from each of the microscope types, providing the most diverse set of data sources, see top of Table 1 for details. Two additional sources were also used for auxiliary testing, see Table 1. These sources use similar microscope setups to the ones used in S5, allowing for comparison between similar image acquisition settings. Additional details for the sources used in this work can be found in Appendix A.2.

Note that in this work we use a subset of images with known MoA labels. The raw image data were prepared us-



ing standard illumination correction and intensity outlier removal followed by downscale to half the original size and compressed, using *DeepProfiler* [25]. Finally, we reduce the image channels from five to the three most informative channels, based in observations by [34].

**Implementation details** Throughout this study, we employ DEIT-S models [35], initialized with pretrained weights from IMAGENET [11]. For all supervised models, we utilize cross-entropy loss for the MoA task. Our training process includes a linear warmup phase of 3 epochs, during which the learning rate gradually increases until it reaches a value of  $10^{-4}$ . Subsequently, we employ a step-wise learning rate reduction strategy, reducing the learning rate by a factor of 10 each time the validation loss and accuracy metric reach a saturation point.

To train the dual model, we follow a two-step approach depicted in Figure 2. First, we perform a self-supervised step using DINO, adhering to the default settings described in [2] with slight variations. This involves training for 300 epochs, using a learning rate of  $10^{-4}$ , linear warm-up for 10 epochs, followed by cosine annealing. We use an exponential moving average of 0.996, following the augmentation strategy from [21]. When incorporating cross-batch examples, the same setup is used, with the exception of how the augmented samples are combined, using one global and three local views from each images in the sampled pair. The pairs are sampled based on metadata, with the requirement that the images are of the same treatment, but come from distinct batches. In the second step, the supervised task-specific step, we stack a DEIT-S model on top of the frozen feature extractor trained in the first step. Passing the tokens of the feature extractor into the task-specific model by removing the embedding layer and replacing it with a linear layer. The training strategy for this step remains consistent with the supervised baseline approach described earlier.

Finally, the third step is adaptation using DINO, with or without CB sampling, to adapt the feature extractor to new out-of-domain data. This is done following the exact same strategy as in the second step described above, unless otherwise stated, but applied to the test set images, crucially not relying on any labels related to the primary task.

## 5. Experiments and Results

In previous sections, we discussed the significant impact of distribution shifts in HCI. In this section, we demonstrate how dramatically performance degrades when applying a standalone model to a new HCI source.

We measured performance using Accuracy as the label proportions are maintained across sources. We further report the F1 scores in Table 4 in the Appendix. We first assess the generalization and adaptation capabilities of standalone DEIT models, which serve as our baselines. Next,

Table 2. Generalization performance across target sources (Acc.).

Source	Target						Model type (Set trained)
	S3	S5	S8	S11	S7	S10	
S3	39.0 ± 0.4	13.5 ± 1.3	15.6 ± 0.9	11.4 ± 1.0	19.5 ± 0.8	12.0 ± 0.6	Supervised
	35.6 ± 0.2	9.2 ± 1.2	9.9 ± 0.1	7.1 ± 0.5	14.8 ± 0.5	9.5 ± 0.8	Dual-model-DINO
	40.5 ± 0.4	14.3 ± 0.2	12.8 ± 1.1	9.0 ± 0.9	17.9 ± 0.5	10.8 ± 0.7	Dual-model-CB
	-	8.2 ± 1.4	9.0 ± 1.0	9.0 ± 2.8	10.8 ± 3.3	7.6 ± 1.6	TTT
	-	24.4 ± 0.6	25.4 ± 0.4	20.5 ± 1.3	27.2 ± 0.6	17.3 ± 1.0	ODA
	-	33.3 ± 0.3	33.9 ± 0.2	30.4 ± 0.6	36.7 ± 0.3	26.7 ± 0.1	CODA
	7.8 ± 0.2	35.9 ± 0.3	13.5 ± 0.5	5.9 ± 0.4	11.7 ± 1.0	12.7 ± 0.2	Supervised
	5.2 ± 0.6	36.2 ± 0.6	13.6 ± 0.5	5.2 ± 0.7	9.4 ± 0.4	11.9 ± 0.6	Dual-model-DINO
	6.0 ± 1.0	36.6 ± 0.7	14.9 ± 0.9	5.3 ± 1.0	9.1 ± 0.5	12.6 ± 0.6	Dual-model-CB
	6.6 ± 1.6	-	10.4 ± 1.8	5.9 ± 1.0	9.2 ± 2.2	9.0 ± 1.2	TTT
13.9 ± 1.1	-	25.5 ± 0.4	14.9 ± 1.0	23.4 ± 0.7	20.4 ± 0.7	ODA	
26.7 ± 1.3	-	28.5 ± 1.1	21.2 ± 0.7	27.6 ± 0.6	28.1 ± 0.5	CODA	
S5	9.5 ± 0.6	11.8 ± 1.4	31.0 ± 0.5	10.3 ± 1.5	18.3 ± 0.4	12.0 ± 0.7	Supervised
	7.0 ± 0.7	12.0 ± 0.8	31.2 ± 0.9	8.4 ± 0.4	13.0 ± 0.4	9.7 ± 0.3	Dual-model-DINO
	8.6 ± 0.5	12.0 ± 0.4	31.5 ± 1.9	9.3 ± 0.5	13.3 ± 0.4	9.7 ± 0.6	Dual-model-CB
	5.1 ± 0.6	4.6 ± 0.5	-	5.2 ± 0.5	3.3 ± 0.2	4.7 ± 0.6	TTT
	14.1 ± 0.7	19.0 ± 0.6	-	18.2 ± 0.3	21.8 ± 0.8	17.2 ± 0.7	ODA
	20.1 ± 0.4	16.5 ± 0.9	-	20.4 ± 1.0	23.9 ± 1.1	19.9 ± 0.8	CODA
	6.4 ± 1.0	6.6 ± 1.4	9.4 ± 0.7	32.4 ± 0.6	7.9 ± 1.2	6.9 ± 1.3	Supervised
	4.7 ± 1.2	5.2 ± 1.4	6.6 ± 0.6	30.1 ± 0.3	4.1 ± 0.5	4.4 ± 0.7	Dual-model-DINO
	4.5 ± 0.4	4.7 ± 0.7	6.9 ± 1.6	31.6 ± 0.8	4.4 ± 1.0	4.4 ± 1.0	Dual-model-CB
	4.7 ± 0.2	4.9 ± 0.3	7.8 ± 1.4	-	7.0 ± 0.0	5.6 ± 0.6	TTT
10.6 ± 0.2	14.8 ± 0.9	17.5 ± 1.1	-	17.7 ± 1.3	11.0 ± 0.3	ODA	
21.5 ± 0.5	16.5 ± 1.4	22.1 ± 0.9	-	23.8 ± 1.6	11.9 ± 1.2	CODA	

we deploy the dual model and evaluate its performance *with* and *without* adapting the feature extractor on the test source. Finally, we incorporate online self-supervised domain adaptation to address batch effects and assess its effectiveness. The results of our main experiments can be found in Table 2 and Figure 7.

In our experiments, we consider the following models and baselines:

- **Supervised** A standalone DEIT-S trained in a supervised fashion on the source data, applied to domain-shifted target data.
- **Dual-model-DINO** A DEIT-S feature extractor with a DEIT-S classifier stacked on top. The feature extractor is self-supervised with DINO and the classifier is trained in a supervised manner, with the feature extractor being frozen, both on the source data.
- **Dual-model-CB** The same as above, but in addition to DINO we use cross-batch image pair sampling, as described in section 3.
- **ODA** Online domain adaptation – this is the same as Dual-modal-DINO but the feature extractor is adapted to the target data using self-supervision.
- **TTT** Similar to ODA but using MAE instead of DINO and updating the feature extractor one image at a time.
- **CODA** Cross-batch Online Domain Adaptation – this is the same as Dual-modal-CB but the feature extractor is adapted to the target data.

**Baseline performance** We begin our evaluation with the standalone DEIT-S classification models. When the models are trained and evaluated on the same data source, as shown in Table 2, the performance ranges from 30.1 to 40.5 with a mean of 36.5% when including all sources, in terms of MoA

accuracy. This represents the situation when there is no domain shift and we have access to labels. When these models are applied to other splits (introducing a domain shift and no access to labels), a significant drop in performance is observed, with values ranging from 5.9 to 19.5 (first row of each non-diagonal element of Table 2 and the first bar in Figure 7 in Appendix). The substantial domain shifts cause the models to be reduced in accuracy by up to 84%.

**Performance of the dual model** Replacing the baseline with the dual model yields a performance similar to that of the standalone model on the in-domain data. Somewhat surprisingly, these models exhibit lower performance compared to the standalone models when applied to out-of-domain sources. The accuracy ranges from 4.7 to 14.8, as indicated in the second row of each non-diagonal element in Table 2 (and the second bar in Figure 7 in the Appendix).

**Performance when updating the feature extractor** Both the standalone and dual models fail to generalize to new HCI data sources. However, the situation changes dramatically when we employ online domain adaptation (ODA) on the feature extractor of the dual model when we apply it to new sources. As illustrated Table 2 (fifth row of each non-diagonal element of the source split) and the fourth bar in Figure 7 in the Appendix we observe a substantial performance improvement compared to both the standalone model and the dual model without updated feature extractor. The out-of-domain MoA accuracy for ODA ranges from 10.6 to 27.2 (a mean increase of  $174.8\% \pm 32.5$  over the baseline) surpassing the performance of the standalone model – although still not reaching the level achieved when testing and evaluating within the domain.

**Performance when employing cross-batch learning** Introducing cross-batch consistency learning (CBCL) for self-supervision of the dual model’s feature extractor brings additional improvements in both in-domain and out-of-domain scenarios. CODA combines CDCL with online domain adaptation (ODA), yielding an enormous performance boost over the baseline as shown in Figure 7 in the Appendix (last bar) and the last row of Table 2. CODA yields MoA accuracy ranging from 11.9 to 36.7, in many cases, the out-of-domain performance is comparable to the performance achieved when training and testing are conducted within the same domain – in some cases even exceeding that (e.g. S3→S8 with CODA yields 33.9 while the S8→S8 baseline is 31.0). The average performance boost of CODA over the baseline is  $232.6\% \pm 63.6$ . CDCL is also helpful when used without online domain adaptation, although it provides less of a performance boost than ODA. As seen in row three of the non-diagonals in Table 2 and the third bar in Figure 7 in the Appendix.

**MAE Performance** The idea of using separate models for feature extraction and classification, followed by feature extraction alignment was inspired by [13], where they use

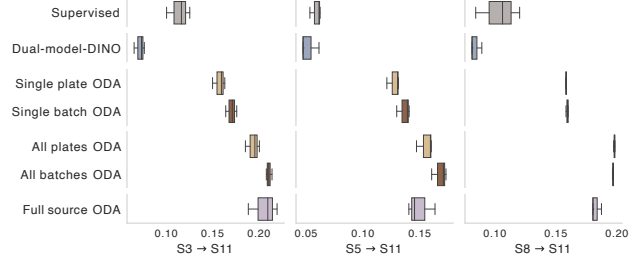


Figure 4. *Effect of Data Granularity. (Accuracy)* Investigating the performance impact when using ODA on different subsets (plate, batch, source) of the target domain. *Single plate/ batch* denotes the scenario where ODA is applied on a single plate or batch and then evaluated on the full target source. *All plate/ batch* denotes the scenario where ODA is applied on a single plate or batch and tested on the same subset.

MAE [16] for self-supervision. However, the performance of MAE in HCI data have been shown to be inferior to DINO, while also it does not allow for training with cross-batch examples [21]. For completeness, we report in Table 2, Table 3 and in the Appendix the results when using MAE instead of DINO. In domain MAE performs slightly worse than DINO. However, in the ODA and TTT setup MAE fails to approach DINO performance – even without CB training.

## 5.1. Analysis and Ablation Studies

In this section, we delve into further analysis and conduct ablation studies to gain a deeper understanding of the performance of CODA and the other models introduced in our experiments.

**Generalization across similar microscopes** In order to gain further insights into the generalization capabilities of the different models, we conducted experiments using two additional data sources, S7 and S10, as test sets. These two sources share more similarities in terms of microscope setup with S5, compared with S3, S8, and S11. If the microscope setup was the primary factor influencing generalization, we would expect to observe improved performance as the similarity (domain distances) between the sources increases [24]. However, upon examining the results in Table 2, we found no significant performance improvements for models trained on S5 and tested on S7 and S10 (S5→S7 and S5→S10). Surprisingly, the performance in S5→S8 was actually higher, despite the use of distinct microscope types and illumination methods. This suggests that the impact of source-to-source variability on the model’s generalization performance cannot be solely attributed to differences in imaging settings – but rather to small differences in the protocol, reagents, or the environment.

**Effect of Data Granularity** In our main experiments, we investigated the effectiveness of adapting the feature extractor to the full test source dataset, resulting in significant performance benefits, as shown in Table 2 and Figure 7. How-

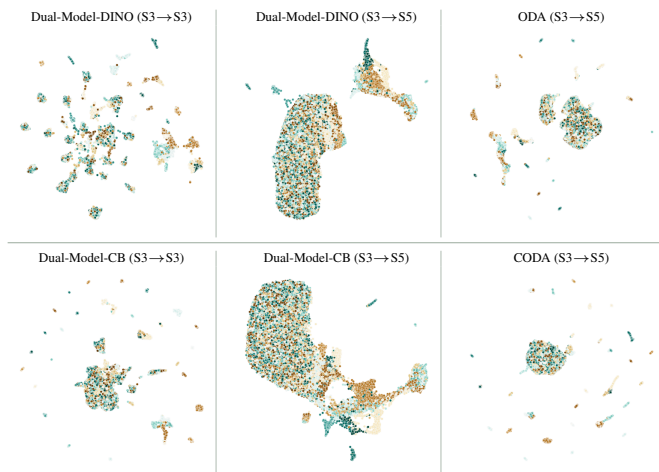


Figure 5. *Feature embeddings*. UMAP visualization of the feature space, showing the impact of distribution shift and the effects of feature extractor alignment. The first column shows the features in S3, column two and three shows S5.

ever, in real drug discovery scenarios, data is generated in separate experimental batches, introducing batch variability. In order to process the data as it arrives, an online approach is desirable. Moreover, we expect batch-level variability to be particularly pronounced, as it involves controlling confounding environmental variables such as incubation time, reagent concentration, and device usage. Therefore, applying ODA at the batch level may prove advantageous compared to applying it across multiple batches.

To assess the performance of ODA and the baselines at different granularities, we conducted transfers from S3→S11, S5→S11, and S8→S11, while varying the level of granularity at which the models were applied: plate, batch, and source. S11 consists of seven distinct plates belonging to four batches, and we applied ODA, along with the baseline models, separately for each plate and batch, resulting in a total of 11 settings (4 batches and 7 plates). We used the same setup as described earlier, with the mini-batch size reduced to 64 to accommodate the smaller dataset size when training on individual batches and plates. The results are illustrated in Figure 4.

Overall, ODA proves beneficial at all three levels of granularity (source, batch, and plate). Interestingly, aligning per batch yields the best performance, while plate-level alignment is slightly better or on par with full source training. This observation supports the known variability between batches, affirming the advantage of aligning within a group of similar variability. Additionally, since the same number of iterations is performed regardless of whether ODA is applied at the source or batch level, there is no additional cost associated with applying it to smaller subsets. In fact, it can be considered preferable as it facilitates easier parallelization.

Table 3. Generalization performance (Accuracy) from S3 to S7 and S10, between DINO and MAE.

Model type	S3 → S7	S3 → S10
Dual-model-MAE	14.2 ± 0.8	8.9 ± 0.5
Dual-model-DINO	<b>14.8 ± 0.5</b>	<b>9.5 ± 0.8</b>
ODA-MAE	17.8 ± 1.1	11.1 ± 0.4
ODA	<b>27.2 ± 0.6</b>	<b>17.3 ± 1.0</b>

**ODA using only a single plate or batch** While aligning the feature space at the batch level proved to be the most effective non-cross-batch strategy, aligning features for each new batch or plate can be time-consuming (although it is still relatively low in time and cost compared to the experimental and imaging pipeline). To evaluate the feasibility of aligning the feature extractor when working with subsets of the data, we repeated the experiment described in the previous paragraphs. However, this time we evaluated each of the models on the full source dataset, rather than only on the subset it was aligned on. That is, we performed ODA on a single plate or batch, and applied this model to the rest of the out-of-domain data. The results of this experiment are shown in Figure 4. We observed a slight but noticeable drop in performance when models were evaluated on the full source dataset compared to when they were evaluated only on the subset they were aligned on. Nevertheless, the overall benefits of ODA were still maintained, as even aligning with a random plate led to substantial performance improvements over the supervised baseline.

**How ODA/CODA changes the feature space** To gain a deeper understanding of how online alignment using ODA or CODA affects the feature space, we conducted a thorough analysis. We visualize the feature space of the dual model in Figure 5 and performed Centered Kernel Alignment (CKA) [8] analysis pre- and post-alignment in Figure 6.

In Figure 5 we provide a UMAP of the embeddings of the CLS token to visualize the feature space. When transitioning from the in-domain to the out-of-domain setting (left and middle column), we observe a significant shift in the feature space for both the Dual-Model-Dino and Dual-Model-CB. In the in-domain setting (S3→S3, left panels), clear structures are present. Note that Dual-Model-DINO (S3→S3) is over-clustered because it has picked up on batch effects, an undesirable property. When applied to out-of-domain data (without adaptation, in the middle panels) S3→S5 the models struggle to distinguish any mechanisms-of-action. After applying ODA or CODA (right panels S3→S5), the structure in the feature space is restored, allowing for better differentiation of MoA classes.

We further examined feature similarity within the feature extractor across different layers using CKA analysis. The results (Figure 6) revealed noticeable differences in feature representation pre- and post-alignment. This indicates that

weights throughout the feature extractor are updated during alignment, particularly for high-level features. It is possible that biological variations between source setups are challenging to adapt to using low-level features. Given the strong predictive performance of ODA and CODA models on out-of-domain data, as well as the good fit of the adapted features to the pre-trained task model, we focused on comparing the feature similarity between CODA before and after alignment, particularly for high-level features. We compare how closely aligned the pre- and post-aligned features resemble those learned in the in-domain setting. Figure 6 (top right) demonstrates a clear diagonal correlation between the post-alignment model and its S5 Dual-model-CB counterpart, suggesting that CODA training successfully learns features similar to those learned in the in-domain setting. When examining the CLS feature similarity (Figure 6 bottom right), the trend is clearer, and the strength of the similarity seems to be somewhat maintained, even at the high-level features. Interestingly, without aligning the feature extractor, the CLS representations remained similar across different depths of the feature extractor (Figure 6 bottom center), indicating that no high-level features are distinguishable in the new domain. This suggests that the domain shift is so severe that the learned features are no longer useful, potentially explaining why performance degradation is seen without alignment.

## 6. Discussion

Our empirical findings consistently support our initial expectations and highlight the limitations of standalone models in generalizing to new High Content Imaging (HCI) sources. In contrast, the dual model, with its adaptive capabilities, demonstrates a significant ability to reduce generalization gaps. The dual model’s bifurcated structure, consisting of separate feature extraction and task-specific components, facilitates a more straightforward adaptation to domain shifts. This design allows the task-specific features to remain intact while effectively adapting the feature extractor to new data characteristics.

The incorporation of CDCL into the self-supervised feature extraction process further strengthens the model’s ability to mitigate batch effects, enhancing its overall robustness. CDCL ensures consistent representations across different batches, leading to substantial improvements in the model’s generalization capabilities.

Our work highlights the critical importance of the training methodology employed in instructing the feature extractor via self-supervision. With the increasing development of innovative self-supervised methods, we anticipate the emergence of more advanced domain adaptation strategies in the near future. These strategies are expected to effectively address the current generalization gap in HCI, enabling more efficient and robust applications in this field.

Aside from the performance benefits demonstrated in

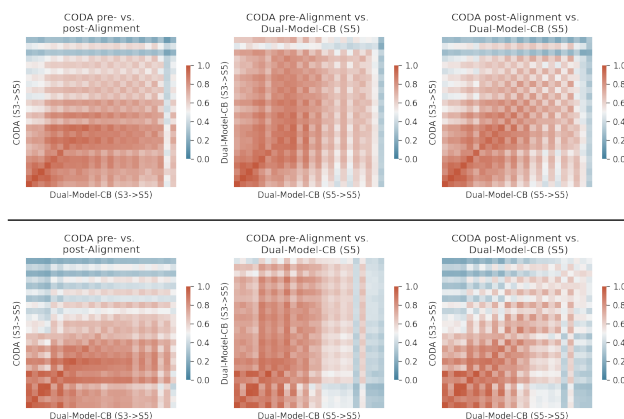


Figure 6. CKA. Feature similarities in S5 for all tokens (top) and CLS tokens (bottom) of the feature extractor. (left) CODA before and after adaptation. (middle) CODA before adaptation vs. a Dual-Model-CB trained in S5. (right) CODA after adaptation vs. a Dual-Model-CB trained in S5.

our work, it is worth noting that methods like CODA that can adapt online to domain shifts are of critical importance in HCI and drug discovery, where sources of variation are high and unpredictable. As such, online domain adaptation methods like CODA will be essential in mitigating these sources of variation that are ultimately beyond our control.

## 7. Conclusion

Our findings emphasize the limitations of standalone models when applied to novel High Content Imaging (HCI) data sources and highlight the effectiveness of the dual model approach in reducing generalization gaps. The dual model’s bifurcated structure, comprising self-supervised feature extraction and task-specific components, enhances adaptability to new domain shifts. Furthermore, the integration of Cross-Domain Consistent Learning enhances the model’s robustness and consistency across different batches, thereby improving its generalization capabilities. The advancement of sophisticated self-supervised methods is expected to drive progress in online domain adaptation strategies, ultimately addressing the prevailing generalization gap in high content imaging. Overall, our method offers a viable strategy to mitigate batch effects and distribution shifts caused by differences in experimental settings and apparatus, leading to improved generalization performance in the HCI domain.

**Acknowledgements.** This work was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP). We acknowledge the use of Berzelius computational resources provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre. We would also like to thank the AWS Open Data Sponsorship Program for sponsoring the JUMP-CP data storage.



## References

- [1] Mark-Anthony Bray, Shantanu Singh, Han Han, Chadwick T. Davis, Blake Borgeson, Cathy L. Hartland, Maria Kost-Alimova, Sigrún Margrét Gústafsdóttir, Christopher C. Gibson, and Anne E Carpenter. Cell painting, a high-content image-based assay for morphological profiling using multiplexed fluorescent dyes. *Nature Protocols*, 11:1757–1774, 2016. [2](#), [4](#)
- [2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. [3](#), [5](#)
- [3] Srinivas Niranj Chandrasekaran, Jeanelle Ackerman, Eric Alix, D. Michael Ando, John Arevalo, Melissa Bennion, et al. Jump cell painting dataset: morphological impact of 136,000 chemical and genetic perturbations. *bioRxiv*, 2023. [2](#), [4](#), [I](#)
- [4] Srinivas Niranj Chandrasekaran, Hugo Ceulemans, Justin D. Boyd, and Anne E Carpenter. Image-based profiling for drug discovery: due for a machine-learning upgrade? *Nature Reviews. Drug Discovery*, 20:145 – 159, 2020. [2](#)
- [5] Sungha Choi, Sanghun Jung, Huiwon Yun, Joanne Taery Kim, Seungryong Kim, and Jaegul Choo. Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11575–11585, 2021. [2](#)
- [6] Beth A. Cimini, Srinivas Niranj Chandrasekaran, Maria Kost-Alimova, Lisa Miller, Amy Goodale, Briana Fritchman, et al. Optimizing the cell painting assay for image-based profiling. *bioRxiv*, 2022. [2](#), [4](#)
- [7] Steven M. Corsello, Joshua A. Bittker, Zihan Liu, Joshua Gould, Patrick McCarren, Jodi E. Hirschman, Stephen E. Johnston, Anita Vrcic, Bang Wong, Mariya Khan, Jacob K. Asiedu, Rajiv Narayan, C. C. Mader, Aravind Subramanian, and Todd R. Golub. The drug repurposing hub: a next-generation drug library and information resource. *Nature Medicine*, 23:405–408, 2017. [4](#)
- [8] Corinna Cortes, Mehryar Mohri, and Afshin Rostamizadeh. Algorithms for learning kernels based on centered alignment, 2014. [7](#)
- [9] Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey. *ArXiv*, abs/1702.05374, 2017. [2](#)
- [10] Mohammad Zalbagi Darestani, Jiayu Liu, and Reinhard Heckel. Test-time training can close the natural distribution shift performance gap in deep learning based compressed sensing. In *International Conference on Machine Learning*, 2022. [2](#)
- [11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. [5](#)
- [12] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *Neural Information Processing Systems*, 2019. [2](#)
- [13] Yossi Gandelsman, Yu Sun, Xinlei Chen, and Alexei A. Efros. Test-time training with masked autoencoders. *ArXiv*, abs/2209.07522, 2022. [2](#), [3](#), [4](#), [6](#)
- [14] Robert Geirhos, Carlos R. Medina Temme, Jonas Rauber, Heiko H. Schütt, Matthias Bethge, and Felix Wichmann. Generalisation in humans and deep neural networks. *ArXiv*, abs/1808.08750, 2018. [2](#)
- [15] Johan Fredin Haslum, Christos Matsoukas, Karl-Johan Leuchowius, Erik Mullers, and Kevin Smith. Metadata-guided consistency learning for high content images. *ArXiv*, abs/2212.11595, 2022. [1](#), [2](#), [3](#), [4](#)
- [16] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. [4](#), [6](#)
- [17] Yufan He, Aaron Carass, Lianrui Zuo, Blake E. Dewey, and Jerry L Prince. Autoencoder based self-supervised test-time adaptation for medical image analysis. *Medical image analysis*, 72:102136, 2021. [2](#)
- [18] Katie Heiser, Peter F McLean, Chadwick T Davis, Ben Fogelson, Hannah B Gordon, Pamela Jacobson, Brett Hurst, Ben Miller, Ronald W Alfa, Berton A Earnshaw, et al. Identification of potential treatments for covid-19 through artificial intelligence-enabled phenomic analysis of human cells infected with sars-cov-2. *BioRxiv*, pages 2020–04, 2020. [1](#)
- [19] Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *ArXiv*, abs/1807.01697, 2018. [2](#)
- [20] Markus Hofmarcher, Elisabeth Rumetshofer, Djork-Arne Clevert, Sepp Hochreiter, and Gunter Klambauer. Accurate prediction of biological assays with high-throughput microscopy images and convolutional networks. *Journal of chemical information and modeling*, 59(3):1163–1171, 2019. [1](#)
- [21] Vladislav Kim, Nikolaos Adaloglou, Marc Osterland, Flavio M. Morelli, and Paula A. Marin Zapata. Self-supervision advances morphological profiling by unlocking powerful image representations. *bioRxiv*, 2023. [2](#), [3](#), [5](#), [6](#)
- [22] Suhyeon Lee, Hongje Seong, Seongwon Lee, and Euntae Kim. Wildnet: Learning domain generalized semantic segmentation from the wild. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9926–9936, 2022. [2](#)
- [23] Xiaofeng Liu, Fangxu Xing, Chao Yang, Georges El Fakhri, and Jonghye Woo. Adapting off-the-shelf source segmenter for target medical image segmentation. *Medical image computing and computer-assisted intervention : MICCAI ... International Conference on Medical Image Computing and Computer-Assisted Intervention*, 12902:549–559, 2021. [2](#)
- [24] Christos Matsoukas, Johan Fredin Haslum, Moein Sorkhei, Magnus Söderberg, and Kevin Smith. What makes transfer learning work for medical images: feature reuse & other factors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9225–9234, 2022. [6](#)
- [25] Nikita Moshkov, Michael Bornholdt, Santiago Benoit, Matthew Smith, Claire McQuin, Allen Goodman, Re-

- becca A. Senft, Yu Han, Mehrtash Babadi, Peter Horvath, Beth A. Cimini, Anne E. Carpenter, Shantanu Singh, and Juan C. Caicedo. Learning representations for image-based profiling of perturbations. *bioRxiv*, 2022. 5
- [26] Jo Nyffeler, Clinton Willis, Felix R. Harris, M.J. Foster, Bryant Chambers, Megan Culbreth, Richard E. Brockway, Sarah Davidson-Fritz, Daniel Dawson, Imran Shah, Katie Paul Friedman, Dan Chang, Logan J. Everett, John F. Wambaugh, Grace Patlewicz, and Joshua A. Harrill. Application of cell painting for chemical hazard evaluation in support of screening-level chemical assessments. *Toxicology and Applied Pharmacology*, 468:116513, 2023. 1
- [27] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *European Conference on Computer Vision*, 2018. 2
- [28] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1406–1415, 2018. 2
- [29] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, and Kate Saenko. Universal domain adaptation through self supervision. *Advances in neural information processing systems*, 33:16282–16292, 2020. 2
- [30] Christopher J Schulze, Walter M Bray, Marcos H Wonerhmann, Joshua Stuart, R Scott Lokey, and Roger G Linington. “function-first” lead discovery: mode of action profiling of natural product libraries using image-based screening. *Chemistry & biology*, 20(2):285–295, 2013. 1
- [31] Jaak Simm, Günter Klambauer, Adam Arany, Marvin Steijaert, Jörg Kurt Wegner, Emmanuel Gustin, Vladimir Chupakhin, Yolanda T Chong, Jorge Vialard, Peter Buijnsters, et al. Repurposing high-throughput image assays enables biological activity prediction for drug discovery. *Cell chemical biology*, 25(5):611–618, 2018. 1
- [32] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV Workshops*, 2016. 2
- [33] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pages 9229–9248. PMLR, 2020. 2, 3
- [34] Maciej Sypetkowski, Morteza Rezanejad, Saber Saberian, Oren Kraus, John Urbanik, James Taylor, Ben Mabey, Mason Victors, Jason Yosinski, Alborz Rezazadeh Sereshkeh, et al. Rrx1: A dataset for evaluating experimental batch correction methods. *arXiv preprint arXiv:2301.05768*, 2023. 5
- [35] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *ArXiv*, abs/2012.12877, 2020. 5
- [36] Gregory P Way, Maria Kost-Alimova, Tsukasa Shibue, William F Harrington, Stanley Gill, Federica Piccioni, Tim Becker, Hamdah Shafqat-Abbasi, William C Hahn, Anne E Carpenter, et al. Predicting cell health phenotypes using image-based morphology profiling. *Molecular biology of the cell*, 32(9):995–1005, 2021. 1
- [37] Xiangyu Yue, Zangwei Zheng, Shanghang Zhang, Yang Gao, Trevor Darrell, Kurt Keutzer, and Alberto Sangiovanni Vincentelli. Prototypical cross-domain self-supervised learning for few-shot unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13834–13844, 2021. 2
- [38] Stephan Zheng, Yang Song, Thomas Leung, and Ian J. Goodfellow. Improving the robustness of deep neural networks via stability training. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4480–4488, 2016. 2