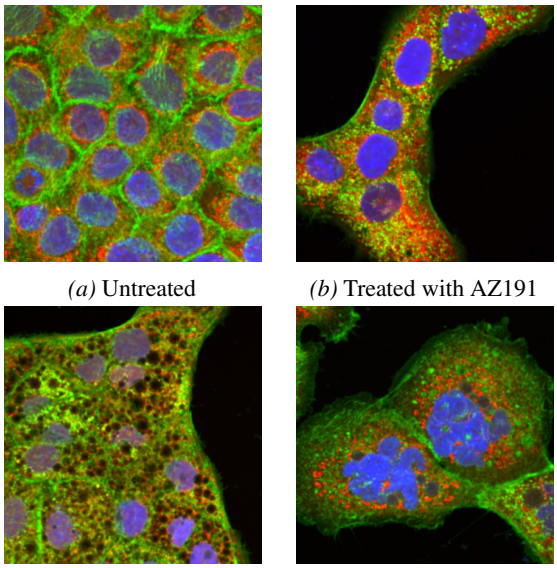
Asymmetric Contrastive Objectives for Efficient Phenotypic Screening

Luke Nightingale [1] **Joseph Tuersley** [1] **Scott Warchal** [1] **Andrea Cairoli** [1] **Jacob Howes** [1] **Cameron Shand** [1]
Andrew J. Powell [1 2] **Darren V.S. Green** [2] **Amy Strange** [1] **Michael Howell** [1]

Abstract

Phenotypic screening experiments produce many microscope images of cells under diverse perturbations, with biologically significant responses often subtle or difficult to identify visually. A central challenge is to extract image representations that distinguish activity from controls and group phenotypically similar perturbations. In this work we propose new adaptations of contrastive loss functions that incorporate experimental metadata as learned class vectors, and a geometrically inspired variant, called SPC, where class vectors are confined to the unit sphere and updated only by attractive terms (allowing more overlap of phenotypically similar classes). The approach is tested on two popular benchmarking datasets, BBBC021 and RxRx3-core; and we also evaluate performance on uncurated screens of HaCaT cells to gauge effectiveness in a realistic use-case scenario. We find we outperform prior methods across the three datasets and on a wide array of metrics measuring phenotype grouping, biological recall, drug-target interaction and mechanism-of-action inference. We also show we maintain this improved performance compared to models over 10x larger in parameter count, and that SPC can be used as an effective fine-tuning technique. The method is easy to implement and is well suited to settings with limited data or compute resources.



(a) Untreated *(b)* Treated with AZ191 *(c)* Treated with APY0201 *(d)* Treated with AMG900

Figure 1. HaCaT cells can exhibit visible cellular changes on treatment. Green channel: plasma membrane & F-actin; red channel: mitochondria; blue channel: DNA.

1. Introduction

High-content screening (HCS) is a high-throughput approach to drug discovery (Bickle, 2010; Bray et al., 2016). To facilitate testing many biological samples in parallel, experiments are performed with arrayed multi-well plates, each well holding a population of cells. The cells are given perturbants and then subsequently imaged with a microscope. In this study a perturbation means either a chemical treatment consisting of a small molecule at a specified concentration, or a CRISPR gene knockout. Cells can be fluorescently tagged for proteins or structures such as nucleus and mitochondria, and these form a sequence of image channels. The data output of an experiment is a large number of multi-channel images accompanied by metadata indicating the perturbation, plate and well of origin. Some molecules also have a known mechanism-of-action (MoA). MoA describes the pharmacological interaction which is believed to underlie the effect of a compound. MoA induced phenotypes are often subtle and require an algorithm to distinguish, but in some cases are visible by eye (Figure 1). Alongside perturbed samples, each plate includes controls expected to elicit no response, typically DMSO in compound screens and non-targeting CRISPR reagents for genetic screens.

A common way to quantitatively analyse cellular images is by extracting handcrafted morphological features using tools such as CellProfiler (Carpenter et al., 2006), followed

[1]The Francis Crick Institute, United Kingdom [2]GlaxoSmithKline, United Kingdom. Correspondence to: Luke Nightingale <luke.nightingale@crick.ac.uk>.

Preprint. January 29, 2026.

by aggregation and dimensionality reduction for downstream analysis. With the advent of highly accurate deep learning algorithms, there has been growing interest in replacing handcrafted features with learned image representations derived from data (Ando et al., 2017; Caicedo et al., 2018; Lafarge et al., 2019; Krentzel et al., 2023). These approaches aim to produce embeddings that uncover phenotypic variation that is difficult to capture with predefined measurements and support tasks such as treatment clustering, outlier detection, and MoA inference.

Recent work has focused on training large models, vision transformers, on increasing dataset sizes. These come in several flavours, typically adaptations of either masked autoencoders (Kraus et al., 2024; Kenyon-Dean et al., 2024) or DINO (Yao et al., 2024; Haslum et al., 2024; Doron et al., 2023; Kim et al., 2025). Also popular has been contrastive learning based around augmentations (Perakis et al., 2021; Bushiri Pwesombo et al., 2025; Huang et al., 2025) or multimodality (Sanchez-Fernandez et al., 2023; Fradkin et al., 2024; Lu et al., 2025; Yu et al., 2025). Contrastive learning naturally frames the learning problem in terms of proximity for semantically similar samples, which fits well the underlying phenotype alignment goal of HCS.

A classic approach is weakly supervised learning (WSL) (Caicedo et al., 2018; Moshkov et al., 2024), often considered the most performant method for small datasets (Kraus et al., 2024). WSL works by training a treatment classifier and then extracting embeddings from the encoder. Identically treated cells exhibit variation in shape and texture leading to substantial heterogeneity within a population (Figure 1). One goal of leveraging treatment labels is to limit how much the model responds to this variation and instead learn features more representative of inter-treatment differences. However since the objective of a classifier is to distinguish classes it can also separate treatments of the same phenotype, or learn other unwanted correlations (Sypetkowski et al., 2023), raising the question of whether there is a more effective way to introduce metadata.

This study considers simple injections of metadata labels into the conceptually appealing contrastive losses, and proposes a variant called Spherical Phenotype Clustering (SPC) with a modified update mechanism allowing more class overlap. We show in contrast to prevailing large scale trends, that the described metadata injections allow contrastive losses trained on limited data to achieve state-of-the-art performance in several highly regarded microscopy benchmarks, and that this extends to the case of uncurated screens. We further show that the method can compete with models 10x or 100x larger in parameter count despite the vastly smaller number of training images.

2. Spherical Phenotype Clustering

As discussed in the introduction, a conventional way to incorporate metadata during training is with WSL, which is a treatment classifier where the encoder is then used to extract embeddings. We summarize the main components as follows:

- A dataset $\{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N\}$ where each $\boldsymbol{x}_i$ is an image and N is the total number of images in the dataset. For each image there is an associated label y_i, an integer value between 1 and the total number of classes C.
- A parameterized encoder function f (the model) that takes an image and outputs a D dimensional embedding vector, $\boldsymbol{z}_i = f(\boldsymbol{x}_i)$. In this work we use either a ResNet18 (He et al., 2016) or, for fine-tuning, a vision transformer (Dosovitskiy et al., 2020).
- A final linear layer that maps embedding vectors to class predictions comprising a weight matrix $W \in \mathbb{R}^{D\times C}$ and bias term $\boldsymbol{b} \in \mathbb{R}^C$, so that the predicted class is given by the maximum element of the vector $W^T\boldsymbol{z}_i + \boldsymbol{b}$.

From now on, we let dataset indexed quantities such as $\boldsymbol{x}_i$ instead be indexed by minibatch. During training the function f is optimized by repeatedly sampling a minibatch of M images from the dataset and updating parameters based on the gradient of the cross entropy loss function

$$l_i = -\log \frac{\exp(W_{y_i}^T \boldsymbol{z}_i + \boldsymbol{b}_{y_i})}{\sum_{j=1}^{C} \exp(W_j^T \boldsymbol{z}_i + \boldsymbol{b}_j)}, \tag{1}$$

where W_j indicates the jth column of W and can be viewed as a prototype embedding for class j. In this and the following expressions we have omitted that the loss is averaged over batch samples, so that the actual loss is $L = (1/M)\sum_{i=1}^{M} l_i$.

Since treatments are always known when conducting a biological experiment, the usefulness of f in HCS is not in the direct ability to perform treatment classification, but rather being able to cluster similar treatments to deduce relationships between them. Following training, images are converted to vectors using f and then aggregated (by computing the mean) to create a well or treatment-level profile, which is a vector of the same dimension as $\boldsymbol{z}_i$. The profiles are then compared with distance based metrics to assess how effectively they cluster according to a biologically meaningful label such as MoA. This is the sense in which the approach is described as weakly supervised - the metadata provides a weak label which helps to learn the biological grouping a perturbation belongs to. To give a concrete example, if we had a dataset with multiple mTOR inhibitor

compounds, then each is a class with many associated images, but they belong to the same biological mechanism, mTOR inhibitor; so a prominent mTOR cluster raises confidence new molecules with unknown mechanism would be correctly grouped.

2.1. Contrastive options for incorporating experimental metadata

The contrastive loss in popular algorithm SimCLR is called NTXent (Chen et al., 2020a). We define first the following similarity function:

$$\mathrm{sim}(\boldsymbol{u}, \boldsymbol{v}) = \exp\left(\frac{1}{\tau}\frac{\boldsymbol{u}\cdot\boldsymbol{v}}{|\boldsymbol{u}||\boldsymbol{v}|}\right),$$

which is the exponential of the cosine similarity multiplied by the reciprocal of temperature τ. And we further define an augmented version of an image in the dataset as $\boldsymbol{x}_i^a$ and the corresponding embedding $\boldsymbol{z}_i^a = f(\boldsymbol{x}_i^a)$. The expression for NTXent can then be written as

$$l_i = -\log \frac{\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_i^a)}{\sum_{j=1}^{M}\left[\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j^a) + \mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_{j\neq i})\right]},$$

where the notation $\boldsymbol{z}_{j\neq i}$ indicates the ith term is not included in the summation. This loss function encourages the embedding for an image to be closer to its augmentation than to embeddings for other images sampled from the dataset.

There is substantial heterogeneity in the cells of HCS images that isn't relevant to discovering relationships between perturbations. For example changes in orientation of cells, technical noise across wells, shape and texture differences that are typical of variation in control samples. A fully unsupervised model would learn a lot of that variation. An autoencoder must remember it to reconstruct the entire image in detail and contrastive models will pick up on it as an effective way to make sure an image and its augmentation is similar. We here adapt contrastive learning so that in addition to the instance-instance comparisons, we have centroid-like class prototypes that constrain samples to be clustered nearby. The idea is to "forget" information related to variation inside of the class represented by those prototypes and produce an embedding space more representative of inter-treatment variation.

In conventional contrastive learning, images are typically compared to their augmentations because without prior knowledge there isn't a better way to define a positive sample pair (Wang & Isola, 2020). Considering the class vectors used in WSL as a kind of sample representing a treatment prototype, we use them to replace augmentations in the following altered NTXent:

$$l_i = -\log \frac{\mathrm{sim}(\boldsymbol{z}_i, W_{y_i})}{\sum_{j=1}^{M}\left[\mathrm{sim}(\boldsymbol{z}_i, W_{y_j}) + \mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_{j\neq i})\right]}. \quad (2)$$

Compared with Eq. (1) this formulation differs in the use of the cosine similarity, the presence of instance-instance terms and a summation that now runs over the elements in a minibatch rather than over the classes.

A summation running over M rather than C elements is of interest in HCS where the number of classes in a dataset can be very large. For example, if they had decided to use weakly supervised learning with wells as targets, a recent work (Kraus et al., 2024) would be training on the equivalent of $C = 93 \cdot 10^6$ classes. More generally the number of classes grows linearly with the size of the dataset. If we assume a (fixed) k images per well, the number of wells is N/k. Treatments similarly grow with the size of the dataset, but are usually performed with multiple replicates r, so that we would have N/kr treatment classes. Example values of k and r would be 160 and 12 for the HaCaT JUMP dataset described below.

The other contrastive loss function we consider is SupContrast (Khosla et al., 2020)

$$l_i = \frac{-1}{\alpha}\log \sum_{p\in P(i)} \frac{\mathrm{sim}(\boldsymbol{z}_i, W_{y_p}) + \mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_{p\neq i})}{\sum_{j=1}^{M}\left[\mathrm{sim}(\boldsymbol{z}_i, W_{y_j}) + \mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_{j\neq i})\right]},$$

where $\alpha = 2|P(i)| - 1$, $P(i)$ is the set of batch indices labelled with class y_i and we have again adapted the original expression by inserting W.

The normalized embeddings of the cosine similarity lie on a unit $(D-1)$-sphere. Optimizing over this space implicitly means learning angular relationships between embeddings. One argument given over the softmax in Eq. (1) is that it is undesirable to use a radially sensitive optimization objective during training when downstream test metrics based on cosine distance ignore the radial component of an embedding (Wang et al., 2017). This is the case in HCS, where cosine distance is used extensively in nearest-neighbour derived metrics and low dimensional plots generated with t-SNE or UMAP. In other work it is observed that when a class is tightly clustered on a hypersphere, the embeddings lie on a polar cap, which is linearly separable from the rest of the embeddings (Wang & Isola, 2020). The reciprocal of the temperature, $1/\tau$, has sometimes been viewed as a radial hyperparameter, controlling the amount of surface area available for clustering (Ranjan et al., 2017).

To distinguish the effect of spherical embeddings and temperature from other aspects of the contrastive losses, we

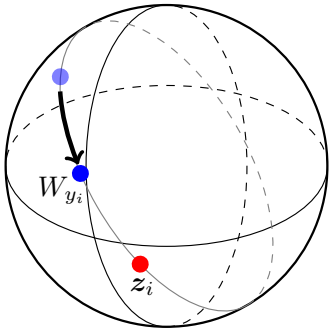


Figure 2. Geometric origin of class updates

also consider using an exact softmax with temperature and normalized vectors

$$l_i = -\log \frac{\text{sim}(\boldsymbol{z}_i, W_{y_i})}{\sum_{j=1}^{C} \text{sim}(\boldsymbol{z}_i, W_j)}, \tag{3}$$

which is known as the L_2-constrained Softmax Loss (Ranjan et al., 2017; Wang et al., 2017)

2.2. Spherical update paths

It has been observed in previous work (Draganov et al., 2024; Wang et al., 2017; Zhang et al., 2020) that loss functions on spheres of the type considered above can exhibit convergence issues related to the unnormalized embeddings failing to lie on the unit sphere. In (Draganov et al., 2024) for example, it is shown that the gradient is orthogonal to a sphere, and then proven that the intuitive expectation finite step sizes along this gradient would lead to continually increasing embedding size holds and can slow convergence. It is also not clear that model parameters, which in the case of a ResNet are primarily filter kernel weights, and class vectors which play a role as embedding centroids, should be optimized under the same mechanism.

Motivated by this we introduce Spherical Phenotype Clustering (SPC) in which we will optimize class vectors by moving them along shortest spherical paths (geodesics) of the hypersphere. In order to define this trajectory we need a point for the class vector W_{y_i} to move towards, and the obvious choice is $\boldsymbol{z}_i$, a model generated embedding for class y_i. Even with random initialization of W, optimization of the model quickly makes the $\{\boldsymbol{z}_i\}$ non-random. We now observe that a closer study of this mechanism yields the following result:

The tangent to the shortest spherical path between W_{y_i} and $\boldsymbol{z}_i$ is a negative multiple of the gradient of the cosine similarity loss function.

A derivation of this relationship is given in Appendix A. Expanding the logarithm in NTXent Eq. (2) we have

$$\begin{aligned} l_i &= -\log \frac{\text{sim}(\boldsymbol{z}_i, W_{y_i})}{\sum_{j=1}^{M} [\text{sim}(\boldsymbol{z}_i, W_{y_j}) + \text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_{j\neq i})]} \\ &= -\frac{1}{\tau} \frac{W_{y_i}^T \boldsymbol{z}_i}{|W_{y_i}^T||\boldsymbol{z}_i|} + \log \sum_{j=1}^{M} [\text{sim}(\boldsymbol{z}_i, W_{y_j}) + \text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_{j\neq i})] \end{aligned}$$

The first term is the cosine similarity loss multiplied by a constant factor, referred to as the attractive component of the loss because it decreases as W_{y_i} and $\boldsymbol{z}_i$ draw close. Similarly, the second term has a repulsive effect. The observation that the gradient of the attractive component of the loss is proportional to the tangent of the geodesic allows us to unpack several differences between SPC and training with the NTXent of Section 2.1:

1. The arc path is confined to a specified radius, whereas gradient based NTXent updates move off the sphere.

2. A different learning rate and optimizer for class vectors. For example momentum may be desirable for model parameters but distort the spherical update path.

3. Repulsive terms are included in the model optimization objective, but not class vector updates.

Item 3 has a naive appeal in the clustering context. Many classes (treatments) would be expected to be very similar, and leaving out repulsive terms would allow them to pack together so that similar classes exhibit a greater degree of overlap. Conversely including the repulsive terms may lead to a model that separates treatments of the same MoA.

That the attractive component of NTXent is proportional to the cosine loss allows us to implement SPC by alternating two coupled updates on each minibatch. First, we update the encoder parameters using the full NTXent objective in Eq. (2): each sample embedding $z_i = f(x_i)$ is pulled toward its class vector W_{y_i} (attraction) while being pushed away from other class vectors W_{y_j} in the minibatch and from other instance embeddings $z_{j\neq i}$ (repulsion). Second, we update the class vectors using only the attractive term, so that each class vector is moved toward embeddings of samples from its class without any repulsive forces from other classes. Concretely, we perform a no-momentum SGD step on the cosine objective $L(W) = -\sum_{i=1}^{M} W_{y_i}^T \boldsymbol{z}_i / |W_{y_i}^T||\boldsymbol{z}_i|$, followed by for each class c the projection $W_c \leftarrow W_c/|W_c|$ to keep class vectors on the unit sphere. This "no-repulse" class update allows phenotypically similar classes to overlap in embedding space while the encoder is still trained with repulsive terms to prevent trivial collapse.

Table 1. BBBC021 ablation results, mean ± s.d. over three training runs. mAP score is reported with both well and treatment profiles. NTXent and SupContrast refer to the altered versions incorporating metadata described in Section 2.1. Note that removing repulsive terms from model updates (Cosine loss row) results in trivial collapse.

Loss function	mAP (well)	mAP (treatment)	NSC	NSCB
Softmax (WSL)	.790 ± .012	.905 ± .009	.967 ± .005	.920 ± .006
L_2-Softmax	.803 ± .001	.882 ± .002	.932 ± .016	.923 ± .021
NTXent	**.892 ± .011**	.931 ± .007	**.974 ± .005**	**.956 ± .010**
SupContrast	.882 ± .008	**.935 ± .006**	**.974 ± .005**	.952 ± .006
Cosine	.284 ± .010	.402 ± .012	.207 ± .056	.188 ± .063
Confine to sphere & split optimizers				
NTXent	.890 ± .009	.934 ± .010	.970 ± .016	.949 ± .012
NTXent w/o repulse (SPC)	**.924 ± .003**	**.959 ± .006**	**.980 ± .000**	**.967 ± .010**
SupContrast	.894 ± .004	.932 ± .016	.975 ± .020	.961 ± .007
SupContrast w/o repulse	.900 ± .008	.948 ± .004	.974 ± .014	.956 ± .010

3. Results

We report results on three datasets: classic benchmark dataset BBBC021, two new screens of HaCaT cells and RxRx3-core. After training, all embedding vectors for evaluation images originating from a given well are mean averaged to create a single well-level embedding profile. Similarly, embeddings for a perturbation are averaged to create a perturbation-level profile. For the ResNet models we also apply a domain adaptation technique to the batch norm layers. Refitting batch norm layer statistics during evaluation has been known for some time in computer vision to be an effective way to reduce data distribution shift (Li et al., 2018; Xiao & Snoek, 2024; Liang et al., 2025) - in our case we use it to mitigate the effects of plate variation. The technique has been applied previously in HCS (Lin & Lu, 2022; Sypetkowski et al., 2023; Kraus et al., 2024). Except for RxRx3-core, we do not perform postprocessing.

Two metrics we work with are k-nearest neighbours with $k = 1$ (1-NN) and mean average precision (mAP). In both cases we use cosine distance as the underlying distance metric. In this context, the precision and recall values for mAP are calculated at different thresholds of distance from a sample. See (Kalinin et al., 2024) for a detailed discussion of mAP in HCS. Unless otherwise stated, in the following sections both 1-NN and mAP use MoA as ground truth labels. All models are trained using perturbation (treatment or gene KO ID) as class, and results are mean averaged over three training runs.

3.1. BBBC021

BBBC021 (Caie et al., 2010; Ljosa et al., 2012) is comprised of MCF-7 cells treated with 113 small molecules at eight different concentrations. Among the full dataset there is a curated subset of 103 treatments annotated with MoA based on visibility of phenotypes and prior literature. Most prior work has focused on this portion of the data, which we also use for our own benchmarking.

We follow (Caicedo et al., 2018; Moshkov et al., 2024) in partitioning cells so that 60% form training data, and we use the other 40% for evaluation. The traditional way to report performance on BBBC021 has been under the NSC (Not-Same-Compound) and NSCB (Not-Same-Compound-or-Batch) scores on treatment-level profiles. These are both 1-NN based metrics. In the NSC score, distances are only calculated between samples that have different compound labels, so that high scores require alignment between different compounds with the same MoA. NSCB additionally requires that neighbours come from a different batch.

We begin by examining the effect of replacing the conventional WSL loss function with alternatives discussed in Section 2.1 - a regular softmax, the L_2 variant, and the two contrastive options. For the loss functions with a temperature hyperparameter we test three values, $\tau = 0.1, 0.25, 1$ and in all cases find $\tau = 0.25$ to give the best performance on MoA based metrics. We train a softmax based loss with and without the bias term but find limited difference, and report without the bias term in Table 1 for a consistent setup. Details of the other variants are given in Appendix C.

We see (Table 1) a substantial metric gain using either of the contrastive objectives. In particular the well aggregated mAP score is ≈ 0.1 higher than in the non-contrastive case, and the treatment-level mAP finds an increase > 0.03. Normalizing vectors and including the hyperparameter τ does not, by itself, result in improved metrics.

On the second half of Table 1 we find simply confining to a sphere and using two optimizers does not improve performance but with the exclusion of repulsive terms it does, particularly for NTXent which is strictly speaking the case where our earlier derivation was valid. From here we continue with the NTXent w/o repulse setup (SPC).

Table 2. BBBC021: comparison with previous work

Study	Type	NSC	NSCB
(Ljosa et al., 2013)	CellProfiler / Factor Analysis	94%	77%
(Singh et al., 2014)	CellProfiler	90%	85%
(Lafarge et al., 2019)	Autoencoder	92.9%	82.2%
(Ando et al., 2017)	Transfer Learning	96%	95%
(Lin & Lu, 2022)	Transfer Learning / BEN	96%	88%
(Janssens et al., 2021)	DeepCluster	97%	85%
(Perakis et al., 2021)	SimCLR	97%	94%
(Caicedo et al., 2018)	WSL	97%	86%
(Cross-Zamirski et al., 2022)	WS-DINO (treatment)	92%	90%
Ours	SPC	**98.0%**	**96.7%**

Methods which use treatment metadata during training have described themselves as weakly supervised (Caicedo et al., 2018) or unsupervised (Janssens et al., 2021). In Table 2 we compare SPC to other results using metadata up to treatment level. SPC gives improved NSC and NSCB scores comparative to extant methods. Comparing with the closest analogue, the treatment classifier (Caicedo et al., 2018), we find $+1\%$ NSC and $+10.2\%$ NSCB.

3.2. HaCaT datasets

To explore a more challenging, real-world scenario we evaluate on two new cell painting screens, where the resultant images have not been curated. We adopt HaCaT cells, a spontaneously immortalised human keratinocyte cell line, and select two perturbant modalities: small molecules and genetic CRISPR knockouts. The JUMP-MoA small molecule library was initially developed by the Joint Undertaking for Morphological Profiling (JUMP) Cell Painting Consortium and aims at generating a large publicly available cell painting dataset (Chandrasekaran et al., 2023). This library comprises 90 compounds, 4 replicates per plate, covering 47 distinct MoA classes (Cimini et al., 2023), and thus serves as a suitable choice to elicit a broad range of cellular phenotypes. For CRISPR perturbations we selected 207 genes anticipated to produce distinct phenotypes across the key cell painting channels, including those linked to mitochondrial function, Golgi apparatus, F-actin, and nuclear division/cell cycle. Chemical screening was performed at three compound concentrations (0.1, 1.0 and 10 uM) with three replicate plates per concentration, while the genetic screen was performed with 5 replicate plates. Images were acquired in 4 channels at 40X magnification, 10 fields of view per well.

Training and evaluation are carried out with three different approaches to test splits. In the first case, cells, we use a 60:40 image split similarly to the BBBC021 split. Then we hold back plates, one of each concentration for JUMP, and two of the five for CRISPR. Finally we hold back perturbations, for JUMP keeping back all 3 plates of 1uM treatments, and for CRISPR holding back a random 40% of KOs as well as 40% of control wells. In this way we verify the model can learn similarity between perturbations not seen during training.

We compare with CellProfiler, SimCLR and WSL. CellProfiler v4.2.5 is applied to segment cellular objects and measure approximately 1500 morphological features. This was followed by aggregating to a well median, feature selection, and robust z-scoring to the plate DMSO control median and median-absolute-deviation values. WSL and SimCLR are trained with the same hyperparameter configuration as SPC. On these datasets we find it advantageous to train SPC with $\tau = 0.1$. We report results in Table 3. Numbers for the JUMP dataset are a better estimate of model performance because we randomized well positions to account for the confounding effects of plate layout and ran multiple treatments for each MoA. For CRISPR we have only the KO label, so results reduce to perturbation metrics and furthermore we used a typical CRISPR screen layout with each perturbation represented by a single well per plate rather than four randomized positions. Nonetheless, in all cases SPC results in a clear improvement.

3.2.1. Fine-tuning DINOv2 models

In the section on BBBC021, we compared with WS-DINO (Cross-Zamirski et al., 2022). WS-DINO uses WSL as a fine-tuning method for DINO trained vision transformers. We now ask the question of whether we outperform WSL as a fine-tuning mechanism, and also if there is any improvement over the trained-from-scratch ResNet18 models. To do so we work with the cell holdout JUMP HaCaT dataset, and leverage pretrained DINOv2 (Oquab et al., 2023) models. LoRA (Hu et al., 2022) allows training of a low-dimensional subspace of the linear layers inside a transformer, helping to avoid overfit on smaller datasets. We train ViT/B DINOv2 models for WSL and SPC, applying LoRA with subspace

Table 3. HaCaT cells: 1-NN/mAP scores on well profiles, mean ± s.d. over three training runs.

Method	Screen	Unseen cells		Unseen plates		Unseen perturbations	
		1-NN	mAP	1-NN	mAP	1-NN	mAP
CellProfiler	JUMP	.327	.115	.271	.113	.330	.196
SimCLR	JUMP	.249 ± .004	.078 ± .001	.192 ± .007	.080 ± .000	.321 ± .046	.141 ± .000
WSL	JUMP	.482 ± .008	.164 ± .003	.398 ± .004	.152 ± .000	.461 ± .010	.240 ± .010
SPC	JUMP	**.496 ± .002**	**.207 ± .001**	**.420 ± .004**	**.194 ± .002**	**.525 ± .002**	**.298 ± .004**
CellProfiler	CRISPR	.550	.197	.522	.246	.755	.243
SimCLR	CRISPR	.403 ± .003	.098 ± .000	.388 ± .004	.116 ± .001	.598 ± .001	.163 ± .001
WSL	CRISPR	.626 ± .003	.326 ± .002	.545 ± .009	.318 ± .003	.772 ± .003	.370 ± .009
SPC	CRISPR	**.714 ± .014**	**.450 ± .004**	**.584 ± .017**	**.410 ± .013**	**.804 ± .008**	**.483 ± .011**

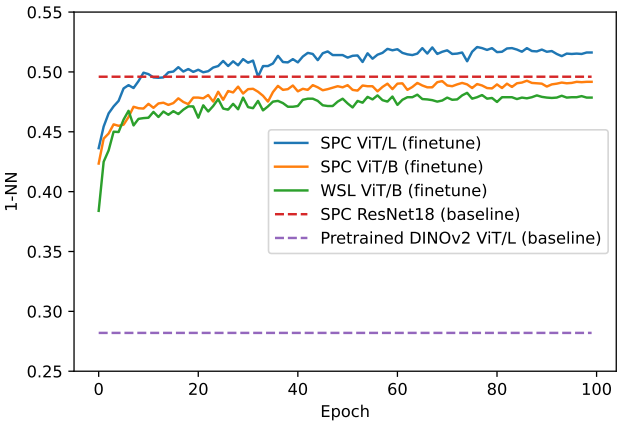


Figure 3. Performance of fine-tuned ViT models against baseline SPC and pretrained DINOv2.

dimension $r = 16$ to both. SPC outperforms WSL and both converge without overfit (Figure 3). Increasing to a ViT/L enables a small performance improvement over the fully trained ResNet18.

3.3. RxRx3-core

Among the vision transformer methods that have been investigated for HCS recently, one of the most celebrated has been OpenPhenom (Kraus et al., 2024). In (Kraus et al., 2024) it is shown by progressively training on larger and larger supersets of the RxRx3 HUVEC cell dataset, performance of a masked autoencoder (He et al., 2022) scales accordingly on the biological database relationship recall metric proposed in (Celik et al., 2022), and can outperform WSL. In a follow up article, the OpenPhenom group released to the community a subset of RxRx3 called RxRx3-core, intended as a benchmark dataset (Kraus et al., 2025). RxRx3-core comprises 222,601 images of 736 CRISPR KOs and 1,674 compounds over 8 concentrations. They evaluate on this dataset both the publicly available OpenPhenom model as well as the larger Phenom-1, and the ∼2B parameter Phenom-2 introduced in (Kenyon-Dean et al., 2024). OpenPhenom is trained on the full RxRx3 dataset, which is approximately 160x the size of RxRx3-core and contains additional perturbations as well as many more crops from the same cell populations (same source image). Phenom-1 and Phenom-2 are trained on larger datasets around 40x and 7x the size of RxRx3 respectively.

We train SPC on RxRx3-core, and evaluate two zero-shot prediction tasks using the evaluation code provided by (Kraus et al., 2025), including applying the same PCA-CS postprocessing step. Both tasks use perturbation profiles. The first is the biological recall metric by (Celik et al., 2022) mentioned above, where for each gene it is calculated the proportion of known relationships which are recalled by examining the top and bottom 5% of cosine distances to other perturbation profiles. The second metric is a drug-target interaction score using average precision to estimate how effectively the model ranks cosine similarity of a compound's known target genes against a set of genes selected at random.

Results are presented in Table 4. We find higher performance than Phenom-1 which is over 25x as large and trained on the full RxRx3, as well as CellProfiler, OpenPhenom and the recent contrastive methods in the second part of the table. Phenom-2 is a private model trained on a very large collection of HUVEC cells. Even with access to this model and compute, in many scenarios it would be unrealistic to be able to train on so many cells of one type, leaving a model which can perform effectively with limited data such as SPC as the best choice.

4. Related Work

Contrastive learning is widely used in unsupervised representation learning. Popular realizations include SimCLR (Chen et al., 2020a) and MoCo v2 (Chen et al., 2020b). Various methods implement image clustering in a contrastive setting. This can be explicit as with Contrastive Clustering

Table 4. RxRx3-core metrics for biological recall and drug-target interaction. Each biological recall column is a different database. Highest score **bolded**, and second highest underlined. *: Data from (Kraus et al., 2025). **: Data from (Huang et al., 2025). †: Mean over three training runs, individual runs reported in Appendix C.

Method	Parameters	Biological recall				Drug-target interaction	
		CORUM	HuMAP	Reactome	StringDB	AP	Z-score↑
Random embeddings*	-	.107	.111	.107	.115	.214 ± .003	0.00
CellProfiler*	-	.361	.444	.160	.330	.276 ± .018	3.34
OpenPhenom-S/16*	25M	.300	.352	.158	.281	.274 ± .017	3.89
Phenom-1*	307M	.395	.482	.188	.349	.290 ± .017	4.35
Phenom-2*	1,860M	**.486**	**.553**	.197	**.415**	**.307 ± .015**	**6.04**
ViT-WSL**	22M	.249	.290	.148	.242	.259 ± .013	3.37
MolPhenix**	36M	.262	.306	.142	.241	-	-
CLOOME**	25M	.328	.406	.135	.278	-	-
CellCLIP**	1,477M	.354	.416	.145	.307	.257 ± .015	2.81
CWA-MSN**	22M	.386	.447	.158	.327	.267 ± .015	3.55
SPC-ResNet18 (ours) †	11M	.428	.498	**.204**	.382	.298 ± .016	4.31

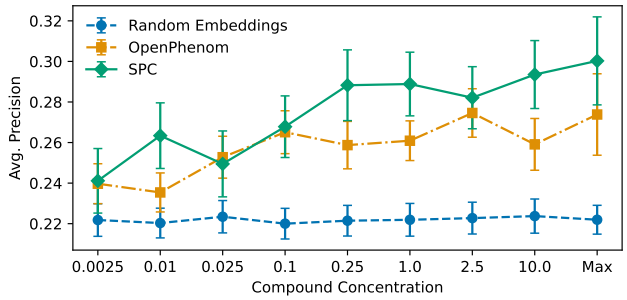


Figure 4. Mean average precision of OpenPhenom and SPC at different concentrations, with ± s.d. over 100 random seeds of negative samples.

(Li et al., 2021) or implicit in the sense that the technique involves refining models that are trained with contrastive learning (Van Gansbeke et al., 2020; Niu et al., 2022). Applying contrastive learning in the labelled data setting is a motivating theme of SupContrast (Khosla et al., 2020). A study of augmentations in contrastive learning with cell microscopy as an application was conducted in (Bendidi et al., 2023).

The L_2-softmax loss, and extensions involving margins and subclusters, have been studied in facial recognition (Ranjan et al., 2017; Wang et al., 2017; 2018). In these works it is typically shown that the explicit spherical geometry outperforms a conventional softmax in image classification tasks. One variant, ArcFace (Deng et al., 2019), was previously considered in HCS (Yang et al., 2022). HCS and facial recognition share a link in both being supplied with large numbers of classes with limited representation. Approximating a softmax with a contrastive loss in the large class setting has been found useful or necessary in the past (Mnih & Kavukcuoglu, 2013; Wu et al., 2018). The approach in (Wu et al., 2018) shows contrastive losses to be effective in the limit of each image having its own class vector representation, in this case the $> 10^6$ images of ImageNet.

Our approach is related to spherical interpolation which has a history in computer graphics (Shoemake, 1985). NTXent, our loss function of choice, was named in (Chen et al., 2020a) but appears in earlier work (Wu et al., 2018; Sohn, 2016) sometimes being referred to as NCELoss.

5. Conclusion

A notable feature distinguishing consumer images from those generated in HCS is some level of semantic labelling always being present in the form of experimental metadata. Our work demonstrates this metadata can support contrastive objectives that outperform WSL and other popular methods on small datasets. By avoiding training with MoA and injecting only the chemical or genetic content of a well, we inherit the desirable property of WSL that the approach is viable in any arrayed screen regardless of how well annotated the content. The results on RxRx3-core indicate that modest computational resources can produce models that compete with or outperform current large scale microscopy machine learning, even when the data samples are drawn from many different experimental batches. This is of significant value to organisations participating in drug discovery or biological research.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning for scientific image analysis, specif-

ically representation learning for high-content biological microscopy. The contributions are methodological and are intended to support research applications such as phenotypic screening and exploratory biological analysis. The work does not involve human-identifiable data, is not intended for clinical or decision-making use, and does not introduce ethical concerns beyond those already well understood for the application of machine learning in biological research. We therefore do not anticipate significant societal or ethical consequences specific to this work, beyond its contribution to established scientific workflows.

References

Ando, D. M., McLean, C. Y., and Berndl, M. Improving phenotypic measurements in high-content imaging screens. *BioRxiv*, 2017. 2, 6, 13

Arevalo, J., Su, E., Ewald, J. D., van Dijk, R., Carpenter, A. E., and Singh, S. Evaluating batch correction methods for image-based cell profiling. *Nature Communications*, 15(1):6516, 2024. 17

Bendidi, I., Bardes, A., Cohen, E., Lamiable, A., Bollot, G., and Genovesio, A. No free lunch in self supervised representation learning. *arXiv preprint arXiv:2304.11718*, 2023. 8

Bickle, M. The beautiful cell: high-content screening in drug discovery. *Analytical and bioanalytical chemistry*, 398:219–226, 2010. 1

Bray, M.-A., Singh, S., Han, H., Davis, C. T., Borgeson, B., Hartland, C., Kost-Alimova, M., Gustafsdottir, S. M., Gibson, C. C., and Carpenter, A. E. Cell painting, a high-content image-based assay for morphological profiling using multiplexed fluorescent dyes. *Nature protocols*, 11 (9):1757–1774, 2016. 1, 18

Bushiri Pwesombo, D., Beese, C., Schmied, C., and Sun, H. Semisupervised contrastive learning for bioactivity prediction using cell painting image data. *Journal of Chemical Information and Modeling*, 65(2):528–543, 2025. 2

Caicedo, J. C., McQuin, C., Goodman, A., Singh, S., and Carpenter, A. E. Weakly supervised learning of single-cell feature embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9309–9318, 2018. 2, 5, 6

Caie, P. D., Walls, R. E., Ingleston-Orme, A., Daya, S., Houslay, T., Eagle, R., Roberts, M. E., and Carragher, N. O. High-content phenotypic profiling of drug response signatures across distinct cancer cells. *Molecular cancer therapeutics*, 9(6):1913–1926, 2010. 5

Carpenter, A. E., Jones, T. R., Lamprecht, M. R., Clarke, C., Kang, I. H., Friman, O., Guertin, D. A., Chang, J. H., Lindquist, R. A., Moffat, J., et al. Cellprofiler: image analysis software for identifying and quantifying cell phenotypes. *Genome biology*, 7:1–11, 2006. 1

Celik, S., Huetter, J.-C., Melo, S., Lazar, N., Mohan, R., Tillinghast, C., Biancalani, T., Fay, M., Earnshaw, B., and Haque, I. S. Biological cartography: Building and benchmarking representations of life. In *NeurIPS 2022 Workshop on Learning Meaningful Representations of Life*, 2022. 7

Chandrasekaran, S. N., Ackerman, J., Alix, E., Ando, D. M., Arevalo, J., Bennion, M., Boisseau, N., Borowa, A., Boyd, J. D., Brino, L., et al. Jump cell painting dataset: morphological impact of 136,000 chemical and genetic perturbations. *bioRxiv*, pp. 2023–03, 2023. 6

Chandrasekaran, S. N., Cimini, B. A., Goodale, A., Miller, L., Kost-Alimova, M., Jamali, N., Doench, J. G., Fritchman, B., Skepner, A., Melanson, M., et al. Three million images and morphological profiles of cells treated with matched chemical and genetic perturbations. *Nature Methods*, pp. 1–8, 2024. 17

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020a. 3, 7, 8

Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020b. 7

Cimini, B. A., Chandrasekaran, S. N., Kost-Alimova, M., Miller, L., Goodale, A., Fritchman, B., Byrne, P., Garg, S., Jamali, N., Logan, D. J., et al. Optimizing the cell painting assay for image-based profiling. *Nature Protocols*, pp. 1–44, 2023. 6, 18

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. *arXiv preprint arXiv:2209.07819*, 2022. 6

Deng, J., Guo, J., Xue, N., and Zafeiriou, S. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699, 2019. 8

Doron, M., Moutakanni, T., Chen, Z. S., Moshkov, N., Caron, M., Touvron, H., Bojanowski, P., Pernice, W. M., and Caicedo, J. C. Unbiased single-cell morphology with self-supervised vision transformers. *bioRxiv*, pp. 2023–06, 2023. 2

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

Draganov, A., Vadgama, S., and Bekkers, E. J. The hidden pitfalls of the cosine similarity loss. *arXiv preprint arXiv:2406.16468*, 2024. 4

Fradkin, P., Azadi Moghadam, P., Suri, K., Wenkel, F., Bashashati, A., Sypetkowski, M., and Beaini, D. How molecules impact cells: Unlocking contrastive phenomolecular retrieval. *Advances in Neural Information Processing Systems*, 37:110667–110701, 2024. 2

Haslum, J. F., Matsoukas, C., Leuchowius, K.-J., Müllers, E., and Smith, K. Metadata-guided consistency learning for high content images. In *Medical Imaging with Deep Learning*, pp. 918–936. PMLR, 2024. 2

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. 2

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022. 7

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022. 6

Huang, P.-J., Liao, Y.-H., Kim, S., Park, N., Park, J., and Shin, D. Efficient cell painting image representation learning via cross-well aligned masked siamese network. *arXiv preprint arXiv:2509.19896*, 2025. 2, 8

Janssens, R., Zhang, X., Kauffmann, A., de Weck, A., and Durand, E. Y. Fully unsupervised deep mode of action learning for phenotyping high-content cellular images. *Bioinformatics*, 37(23):4548–4555, 2021. 6

Kalinin, A. A., Arevalo, J., Vulliard, L., Serrano, E., Tsang, H., Bornholdt, M., Rajwa, B., Carpenter, A. E., Way, G. P., and Singh, S. A versatile information retrieval framework for evaluating profile strength and similarity. *bioRxiv*, 2024. 5

Kenyon-Dean, K., Wang, Z. J., Urbanik, J., Donhauser, K., Hartford, J., Saberian, S., Sahin, N., Bendidi, I., Celik, S., Fay, M., et al. Vitally consistent: Scaling biological representation learning for cell microscopy. *arXiv preprint arXiv:2411.02572*, 2024. 2, 7

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020. 3, 8

Kim, V., Adaloglou, N., Osterland, M., Morelli, F. M., Halawa, M., König, T., Gnutt, D., and Marin Zapata, P. A. Self-supervision advances morphological profiling by unlocking powerful image representations. *Scientific Reports*, 15(1):4876, 2025. 2, 13

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 15

Kraus, O., Kenyon-Dean, K., Saberian, S., Fallah, M., McLean, P., Leung, J., Sharma, V., Khan, A., Balakrishnan, J., Celik, S., et al. Masked autoencoders for microscopy are scalable learners of cellular biology. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11757–11768, 2024. 2, 3, 5, 7

Kraus, O., Comitani, F., Urbanik, J., Kenyon-Dean, K., Arumugam, L., Saberian, S., Wognum, C., Celik, S., and Haque, I. S. Rxrx3-core: Benchmarking drug-target interactions in high-content microscopy. *arXiv preprint arXiv:2503.20158*, 2025. 7, 8, 13, 18

Krentzel, D., Shorte, S. L., and Zimmer, C. Deep learning in image-based phenotypic drug discovery. *Trends in Cell Biology*, 33(7):538–554, 2023. 2

Lafarge, M. W., Caicedo, J. C., Carpenter, A. E., Pluim, J. P., Singh, S., and Veta, M. Capturing single-cell phenotypic variation via unsupervised representation learning. In *International Conference on Medical Imaging with Deep Learning*, pp. 315–325. PMLR, 2019. 2, 6

Li, Y., Wang, N., Shi, J., Hou, X., and Liu, J. Adaptive batch normalization for practical domain adaptation. *Pattern Recognition*, 80:109–117, 2018. 5, 13

Li, Y., Hu, P., Liu, Z., Peng, D., Zhou, J. T., and Peng, X. Contrastive clustering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 8547–8555, 2021. 8

Liang, J., He, R., and Tan, T. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, 133(1):31–64, 2025. 5

Lin, A. and Lu, A. Incorporating knowledge of plates in batch normalization improves generalization of deep learning for microscopy images. In *Machine Learning in Computational Biology*, pp. 74–93. PMLR, 2022. 5, 6, 13

Ljosa, V., Sokolnicki, K. L., and Carpenter, A. E. Annotated high-throughput microscopy image sets for validation. *Nature methods*, 9(7):637–637, 2012. 5

Ljosa, V., Caie, P. D., Ter Horst, R., Sokolnicki, K. L., Jenkins, E. L., Daya, S., Roberts, M. E., Jones, T. R., Singh, S., Genovesio, A., et al. Comparison of methods for image-based profiling of cellular morphological responses to small-molecule treatment. *Journal of biomolecular screening*, 18(10):1321–1329, 2013. 6, 13

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 15

Lu, M., Weinberger, E., Kim, C., and Lee, S.-I. Cell-CLIP - learning perturbation effects in cell painting via text-guided contrastive learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=W3JnXa9mW2. 2

Mnih, A. and Kavukcuoglu, K. Learning word embeddings efficiently with noise-contrastive estimation. *Advances in neural information processing systems*, 26, 2013. 8

Moshkov, N., Bornholdt, M., Benoit, S., Smith, M., McQuin, C., Goodman, A., Senft, R. A., Han, Y., Babadi, M., Horvath, P., et al. Learning representations for image-based profiling of perturbations. *Nature communications*, 15(1):1594, 2024. 2, 5

Niu, C., Shan, H., and Wang, G. Spice: Semantic pseudo-labeling for image clustering. *IEEE Transactions on Image Processing*, 31:7264–7278, 2022. 8

Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 6

Orton, R. J., Sturm, O. E., Vyshemirsky, V., Calder, M., Gilbert, D. R., and Kolch, W. Computational modelling of the receptor-tyrosine-kinase-activated mapk pathway. *Biochemical Journal*, 392(2):249–261, 2005. 15

Perakis, A., Gorji, A., Jain, S., Chaitanya, K., Rizza, S., and Konukoglu, E. Contrastive learning of single-cell phenotypic representations for treatment classification. In *Machine Learning in Medical Imaging: 12th International Workshop, MLMI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, September 27, 2021, Proceedings 12*, pp. 565–575. Springer, 2021. 2, 6

Ranjan, R., Castillo, C. D., and Chellappa, R. L2-constrained softmax loss for discriminative face verification. *arXiv preprint arXiv:1703.09507*, 2017. 3, 4, 8

Sanchez-Fernandez, A., Rumetshofer, E., Hochreiter, S., and Klambauer, G. Cloome: contrastive learning unlocks bioimaging databases for queries with chemical structures. *Nature Communications*, 14(1):7339, 2023. 2

Shoemake, K. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pp. 245–254, 1985. 8, 13

Singh, S., Bray, M.-A., Jones, T., and Carpenter, A. Pipeline for illumination correction of images for high-throughput microscopy. *Journal of microscopy*, 256(3):231–236, 2014. 6, 13

Sohn, K. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29, 2016. 8

Sypetkowski, M., Rezanejad, M., Saberian, S., Kraus, O., Urbanik, J., Taylor, J., Mabey, B., Victors, M., Yosinski, J., Sereshkeh, A. R., et al. Rxrx1: A dataset for evaluating experimental batch correction methods. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4285–4294, 2023. 2, 5

Van der Maaten, L. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 15

Van Gansbeke, W., Vandenhende, S., Georgoulis, S., Proesmans, M., and Van Gool, L. Scan: Learning to classify images without labels. In *European conference on computer vision*, pp. 268–285. Springer, 2020. 8

Wang, F., Xiang, X., Cheng, J., and Yuille, A. L. Normface: L2 hypersphere embedding for face verification. In *Proceedings of the 25th ACM international conference on Multimedia*, pp. 1041–1049, 2017. 3, 4, 8

Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., and Liu, W. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5265–5274, 2018. 8

Wang, T. and Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pp. 9929–9939. PMLR, 2020. 3

Wei, J.-H., Zhang, Z. C., Wynn, R. M., and Seemann, J. Gm130 regulates golgi-derived spindle assembly by activating tpx2 and capturing microtubules. *Cell*, 162(2): 287–299, 2015. 15

Wu, Z., Xiong, Y., Stella, X. Y., and Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 8

Xiao, Z. and Snoek, C. G. Beyond model adaptation at test time: A survey. *arXiv preprint arXiv:2411.03687*, 2024. 5

Yang, S., Shen, T., Fang, Y., Wang, X., Zhang, J., Yang, W., Huang, J., and Han, X. Deepnoise: signal and noise disentanglement based on classifying fluorescent microscopy images via deep learning. *Genomics, Proteomics & Bioinformatics*, 20(5):989–1001, 2022. 8

Yao, H., Hanslovsky, P., Huetter, J.-C., Hoeckendorf, B., and Richmond, D. Weakly supervised set-consistency learning improves morphological profiling of single-cell images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6978–6987, 2024. 2

Yu, Y., Tenenholtz, N., Mackey, L., Wei, Y., Alvarez-Melis, D., Amini, A. P., and Lu, A. X. Causal integration of chemical structures improves representations of microscopy images for morphological profiling. *arXiv preprint arXiv:2504.09544*, 2025. 2

Zhang, D., Li, Y., and Zhang, Z. Deep metric learning with spherical embedding. *Advances in Neural Information Processing Systems*, 33:18772–18783, 2020. 4

A. Correspondence between cosine similarity loss and spherical arc

In the notation that follows $\hat{\boldsymbol{z}}$ denotes a unit vector in the direction of $\boldsymbol{z}$. A convenient expression for the path along a spherical arc is provided by graphics algorithm SLERP (Shoemake, 1985). SLERP gives a spherically interpolated point between two vectors $\boldsymbol{w}$, $\boldsymbol{z}$ on a unit sphere, with $t \in [0, 1]$ indicating what proportion of distance along the arc the interpolated point should be

$$\boldsymbol{r}(t) = \frac{\sin[(1-t)\Omega]}{\sin\Omega}\hat{\boldsymbol{w}} + \frac{\sin[t\Omega]}{\sin\Omega}\hat{\boldsymbol{z}}, \tag{A.1}$$

where Ω is the angle between $\hat{\boldsymbol{w}}$ and $\hat{\boldsymbol{z}}$. It is a well known fact of high dimensional geometry that on an n-sphere, just as in the case of an ordinary sphere in three dimensional space, the shortest path between two points lies on a great circle. Great circles are defined by the intersection of the spherical surface with a 2-plane passing through the origin. On specifying a 2-plane using two points on the arc, the SLERP formula can be confirmed with straightforward geometry (Figure 5).

As described in the main article, the path we are interested in is the one starting at a class vector $\boldsymbol{w}$ and ending at a model embedding vector $\boldsymbol{z} = f(\boldsymbol{x})$ where image $\boldsymbol{x}$ has the same class as $\boldsymbol{w}$.

Proposition. *The tangent to the shortest spherical path between $\boldsymbol{w}$ and $\boldsymbol{z}$ is a negative multiple of the gradient of the cosine similarity loss function.*

The definition of the cosine similarity is $\cos\Omega = \boldsymbol{z}^T\boldsymbol{w}/|\boldsymbol{z}||\boldsymbol{w}|$ where Ω is the angle between $\boldsymbol{z}$ and $\boldsymbol{w}$. The cosine loss is the negative of this quantity. Computing the gradient of the cosine loss we get

$$\begin{aligned}\nabla_{\boldsymbol{w}}\left(-\frac{\boldsymbol{z}^T\boldsymbol{w}}{|\boldsymbol{z}||\boldsymbol{w}|}\right) &= -\frac{\boldsymbol{z}}{|\boldsymbol{z}||\boldsymbol{w}|} + \frac{\boldsymbol{z}^T\boldsymbol{w}}{|\boldsymbol{z}||\boldsymbol{w}|^3}\boldsymbol{w} \\ &= -\frac{\boldsymbol{z}}{|\boldsymbol{z}||\boldsymbol{w}|} + \frac{|\boldsymbol{z}||\boldsymbol{w}|\cos\Omega}{|\boldsymbol{z}||\boldsymbol{w}|^3}\boldsymbol{w} \\ &= -\frac{1}{|\boldsymbol{w}|}\left(\hat{\boldsymbol{z}} - \cos\Omega\,\hat{\boldsymbol{w}}\right). \end{aligned} \tag{A.2}$$

Treating Eq. (A.1) as the path of a point over the sphere, we compute its tangent vector by taking the derivative with respect to t.

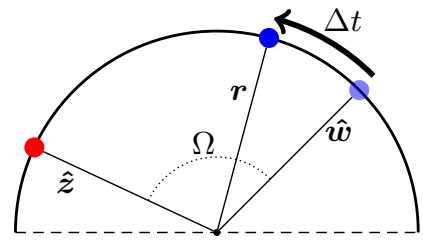


Figure 5. Verifying the SLERP formula

$$\begin{aligned}\left.\frac{d\boldsymbol{r}}{dt}\right|_{t=0} &= \frac{d}{dt}\left\{\frac{\sin[(1-t)\Omega]}{\sin\Omega}\hat{\boldsymbol{w}} + \frac{\sin[t\Omega]}{\sin\Omega}\hat{\boldsymbol{z}}\right\}\Bigg|_{t=0} \\ &= \frac{\Omega}{\sin\Omega}\Big(-\cos[(1-t)\Omega]\hat{\boldsymbol{w}} + \cos[t\Omega]\hat{\boldsymbol{z}}\Big)\Big|_{t=0} \\ &= \frac{\Omega}{\sin\Omega}\Big(-\cos\Omega\,\hat{\boldsymbol{w}} + \hat{\boldsymbol{z}}\Big),\end{aligned}$$

which is proportional to the negative of the gradient in Eq. (A.2) when $\Omega/\sin\Omega \geq 0$. This is the case since we use the smaller arc on a great circle giving $0 \leq \Omega \leq \pi$. □

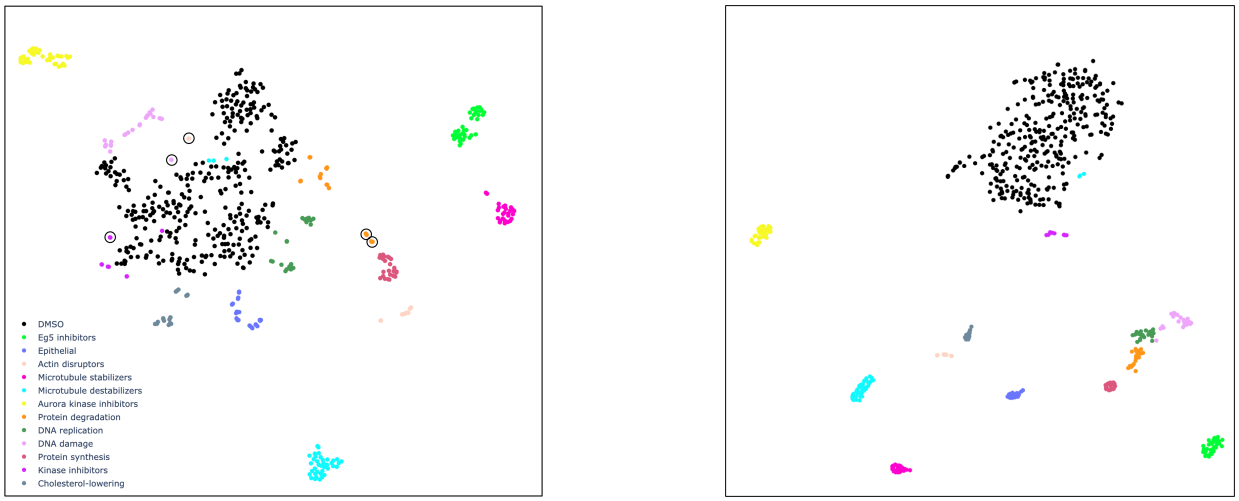
B. Pre-processing and training details

We perform similar BBBC021 preprocessing to that in (Ando et al., 2017) including using the illumination correction functions (Singh et al., 2014), log transforming data and making use of the single cell crops defined in (Ljosa et al., 2013), but we rescale channels independently. To preprocess the HaCaT datasets we resized each image to 512x512 pixels and generated 16 crops under a 4x4 tiling scheme (see Figure 7). Since we have 10 fields per well the 4x4 tiling results in each well having 160 associated images of shape 128x128x4. We then filter the dataset by removing any tiles with very limited cellular content. We do this by calculating an intensity threshold in the nuclear channel over the whole dataset (10% of the maximum minus minimum intensities, post outlier clipping). Tiles where less than 5% of the pixels pass the threshold value are removed from the dataset. We now rescale each channel independently to lie in a 0 to 1 range after first clipping outlier intensities on a per channel basis. The outlier clipping thresholds we use are the 0.01 and 99.9 percentiles as in (Kim et al., 2025). Images are converted to uint8 for storage. RxRx3-core is largely already pre-processed - we tile the dataset 2x2 to generate four 256x256 crops as in (Kraus et al., 2025).

Our implementation of domain adaptation is similar to previous work (Li et al., 2018; Lin & Lu, 2022) but we do not make any changes to sampling during training. We use controls to compute the batch normalization statistics for BBBC021 and the HaCaT datasets, but for RxRx3-core

Table 5. Extended BBBC021 results

Method	Well profile				Treatment profile		
	MoA 1-NN	Replicate 1-NN	Replicate mAP	MoA mAP	MoA mAP	NSC	NSCB
Loss function variants							
(WSL) Softmax	0.995	0.818	0.669	0.790	0.905	96.7%	92.0%
Softmax w/ bias	0.994	0.812	0.657	0.792	0.911	94.4%	92.7%
L_2-Softmax ($\tau = 0.1$)	0.992	**0.931**	**0.847**	0.666	0.789	82.2%	78.6%
L_2-Softmax ($\tau = 0.25$)	0.988	0.701	0.496	0.803	0.882	93.2%	92.3%
L_2-Softmax ($\tau = 1.0$)	0.797	0.566	0.10	0.292	0.388	40.1%	30.7%
NTXent ($\tau = 0.1$)	0.989	0.709	0.495	0.824	0.882	91.2%	91.3%
NTXent ($\tau = 0.25$)	0.995	0.692	0.457	0.892	0.931	97.4%	95.6%
NTXent ($\tau = 1.0$)	0.981	0.646	0.373	0.841	0.858	91.2%	79.7%
SupContrast ($\tau = 0.1$)	0.992	0.697	0.462	0.826	0.894	93.8%	92.0%
SupContrast ($\tau = 0.25$)	0.993	0.686	0.453	0.882	0.935	97.4%	95.2%
SupContrast ($\tau = 1.0$)	0.971	0.633	0.345	0.785	0.822	91.2%	76.8%
Cosine	0.680	0.517	0.096	0.284	0.402	20.7%	18.8%
Confine to sphere & split optimizers ($\tau = 0.25$, $lr = 0.1$)							
NTXent	0.996	0.693	0.460	0.890	0.934	97.0%	94.9%
(SPC) NTXent w/o repulse	0.996	0.669	0.425	**0.924**	**0.959**	**98.0%**	**96.7%**
SupCon	0.993	0.700	0.476	0.894	0.932	97.5%	96.1%
SupCon w/o repulse	0.992	0.662	0.409	0.900	0.948	97.4%	95.6%
SGD learning rate ablations							
SPC ($lr = 0.01$)	0.995	0.668	0.421	0.907	0.947	97.4%	96.7%
SPC ($lr = 1$)	0.994	0.646	0.393	0.909	0.939	97.7%	96.3%
No batch norm domain adaption							
WSL	0.998	0.867	0.753	0.708	0.880	87.7%	92.0%
SPC	**1.000**	0.720	0.537	0.903	0.918	88.0%	96.0%



(a) L_2-Softmax ($\tau = 0.1$) *(b)* SPC ($\tau = 0.25$)

Figure 6. High performance on replicate matching doesn't necessarily improve biologically meaningful clustering. Circled in the L_2-Softmax plot are instances where all treatment replicates are tightly packed, but detached from a main MoA cluster. By contrast SPC MoA clusters are well defined, even if there is internal overlap of treatments. Plots generated with TSNE on evaluation well profiles.

where controls are not consistent across modalities we revert to the usual random sampling from batch.

All ResNet18 models used an embedding dimension 384. For BBBC021 we use the 3-channel augmentation scheme RandAugment with magnitude set to 12. For the multi-channel HaCaT and RxRx3-core data we use three standard augmentations, RandomResizedCrop with scale (0.8, 1.0), RandomHorizontalFlip with probability 0.5, and GaussianBlur with kernel size 9, sigma 5, probability 0.5. We use the torchvision package to implement augmentations. Im-

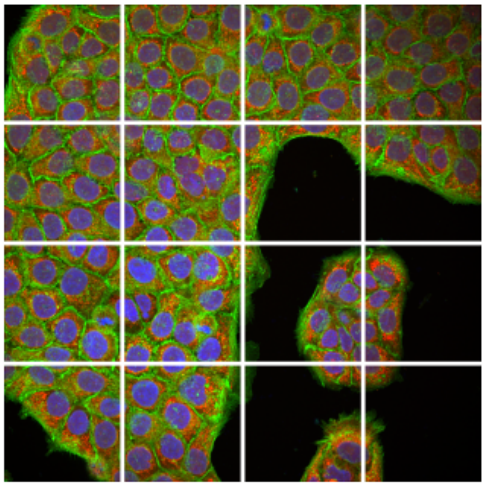


Figure 7. Raw 1080x1080 HaCaT images are resized to 512x512 and then tiled.

ages in uint8 form are standardized by converting to float32, subtracting 127.5 from each pixel and then dividing by 255. BBBC021 and HaCaT were trained with the Adam optimizer (Kingma & Ba, 2014) set to initial learning rate of 0.0003 and no weight decay, a batch size of 16 and SPC SGD learning rate 0.1. RxRx3-core was trained with the same hyperparameter configuration as the HaCaT dataset, except we increase the batch size to 512 so that training completes in a reasonable amount of time. Vision transformers were trained with AdamW (Loshchilov & Hutter, 2017) initial learning rate 0.0003, weight decay 0.05 and betas (0.9, 0.95), SPC SGD initial learning rate 1, a cosine annealing scheduler with 5 warmup epochs, embedding dimension 768, a batch size of 256 and $\tau = 0.1$. Training runs were terminated after 100 epochs without early stopping.

The majority of training and evaluation was performed on a 32GB memory NVIDIA V100 GPU with a 20-core Intel Xeon Gold 6148 CPU and 256GB RAM. Vision transformer and RxRx3-core training runs were performed on an NVIDIA H100 GPU.

C. Further results

C.1. BBBC021

We collect in Table 5 extended results for BBBC021, including the temperature ablation study. Replicate metrics quantify the extent to which the closest samples in a well profile's neighbourhood are other wells of the same treatment. An interesting phenomenon emerges of replicate metrics falling on increase of MoA scores. Replicate based metrics penalize a sample being next to a different compound of the same MoA, while MoA metrics require the treatment to be close to other compounds of the same MoA. In support of the arguments we made in the main article, we find the exact softmax functions including the L_2 version generally better at separating a treatment from the rest of the dataset, but less effective at creating biological clusters of similar treatments. To some extent this is visible qualitatively (Figure 6). Table 5 also shows that ablating the domain adaption and SGD learning rate components of SPC leads to the same conclusions reached in the main article, that a contrastive objective and dropping repulsive terms improves biological clustering.

C.2. HaCaT datasets

C.2.1. Qualitative observations

We now train a model on both the JUMP and CRISPR datasets, using the cell train splits of each dataset, and investigate the evaluation profiles for relationships between perturbations. A t-SNE (Van der Maaten & Hinton, 2008) of well profiles (Figure 8) shows consistent clustering of multiple replicates for a given treatment, and also local structure such as concentration gradients and multiple compounds of the same MoA in adjacent mini-clusters. Some cluster arrangements violate MoA. In one we find three compounds marked with different MoA: PD-198306 (MAP kinase inhibitor), neratinib (EGFR inhibitor), selumetinib (MEK inhibitor). An explanation is offered by all three lying on the MAPK/ERK signalling pathway (Orton et al., 2005). Inspection of the images gives marginal visual confirmation (Figure 9). The circumstance is clearer if we apply TGF-beta stimulation to the cells. Under these conditions the cells develop a characteristic staining pattern in the mitochondrial (red) channel. These observations suggest the algorithm is able to detect subtle cellular changes that visual inspection would agree with under more extreme conditions, and also illustrate the difficulty in effectively quantifying performance on datasets that are not curated in the manner of BBBC021.

Aurora kinase inhibitor compounds interfere with chromosome segregation during mitosis and have a very clear visual phenotype (Figure 1d). Examining the t-SNE we find clustering nearby CRISPR KOs of Aurora A kinase (AURKA) and ANLN. ANLN is also involved in mitosis, presenting a similar phenotype consisting of enlarged, multinucleated cells. To confirm the t-SNE is a faithful representation of model derived distances, we use perturbation profiles to compute the cosine distances between aurora kinase inibitor AMG900 and all perturbations in both datasets (Figure 10). The three unhighlighted perturbations closer than AURKA are CRISPR KOs of CCNA2, CDK1 and GOLGA2 which are closely related genes also involved in cell cycle and mitosis. For example GOLGA2 activates AURKA through its interaction with TPX2 (Wei et al., 2015). It is seen that we are able to effectively deduce which gene KOs have

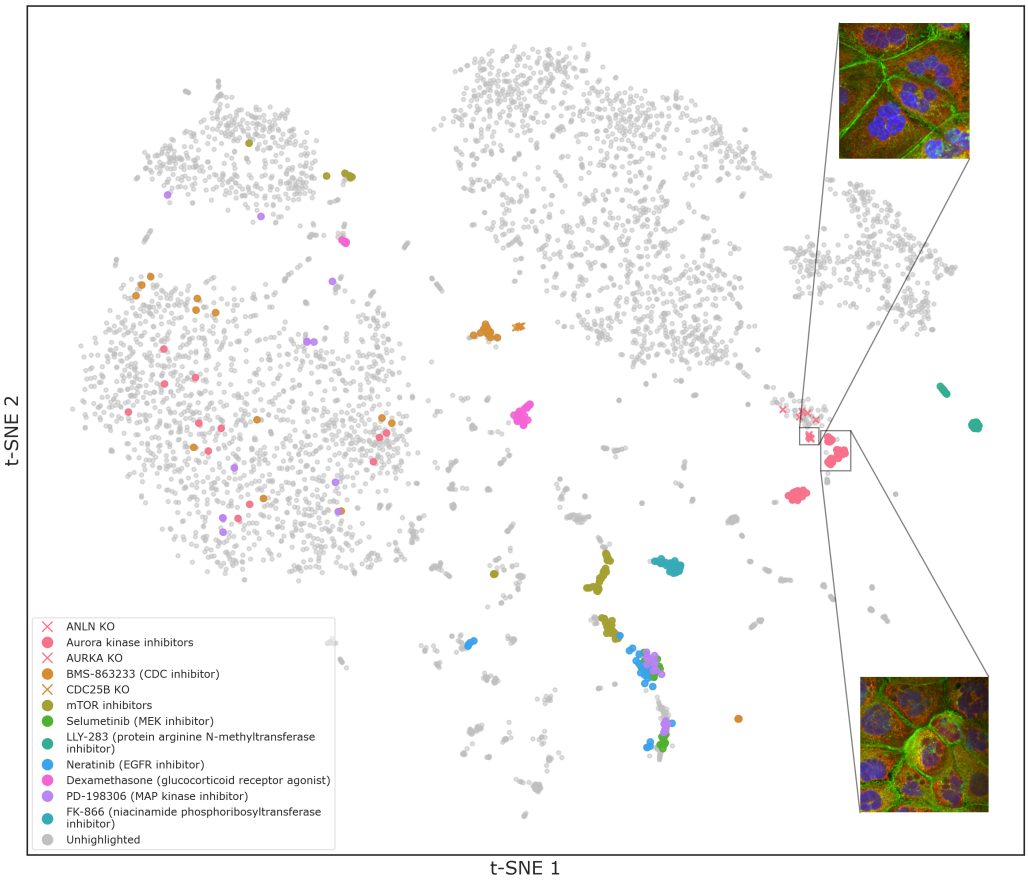


Figure 8. SPC aligns Aurora kinase inhibitor molecules with associated gene KOs, among other phenotypic clusters. t-SNE of well profiles, with several arrangements highlighted. Highlighted unclustered wells are typically low concentration treatments. Insets: ANLN KO and AMG900.

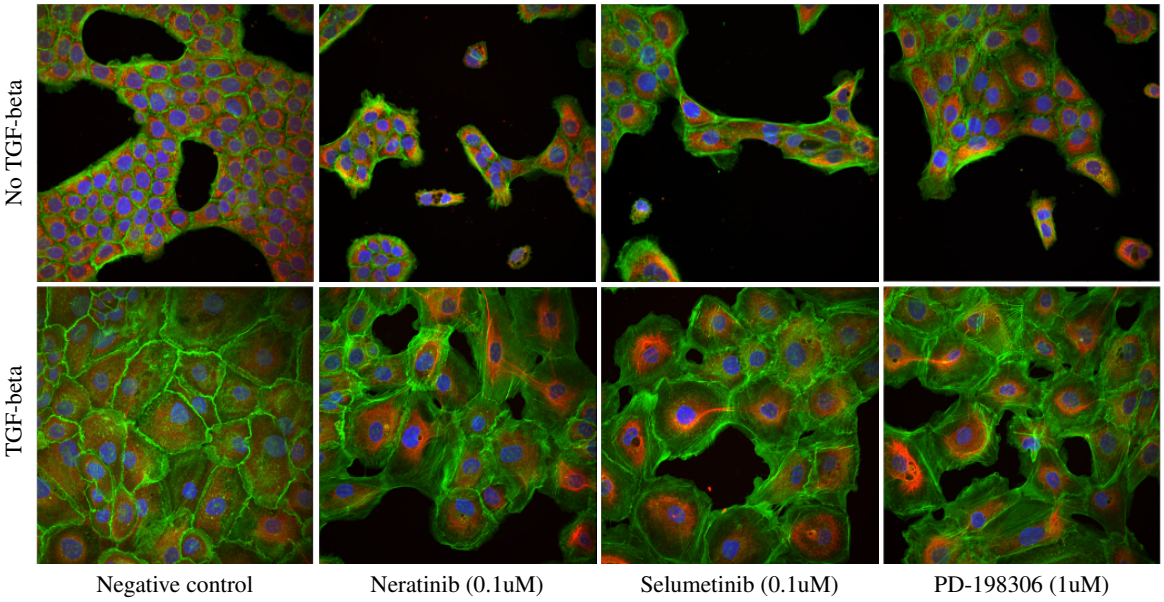


Figure 9. Visual similarity between three compounds with different MoA that SPC identified as similar, and negative controls for comparison. Cells are also shown with TGF-beta stimulation where the mitochondrial response becomes prominent.

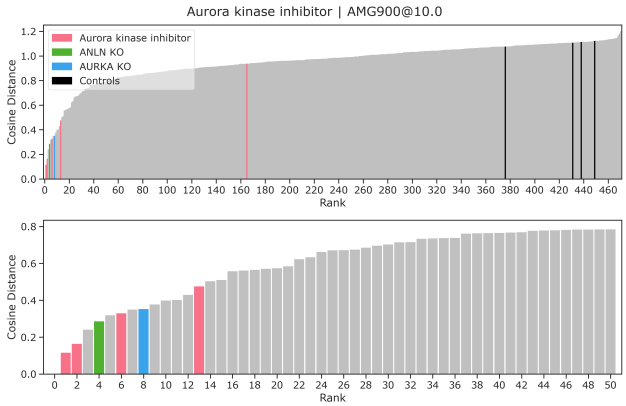
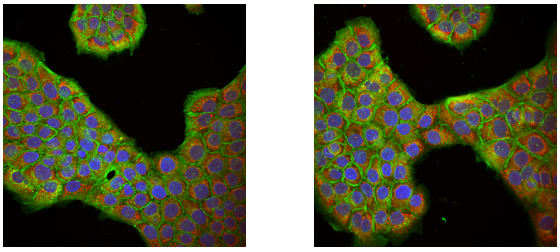


Figure 10. Ordered cosine distances between perturbation profiles and AMG900. First row: all profiles, second row: closest 50.



(a) Untreated *(b)* Dexamethasone

Figure 12. Glucocorticoid receptor agonist dexamethasone is difficult to distinguish from control samples by inspecting images.

similar effects to the AMG900 molecule. This type of tight clustering between chemical and genetic perturbations is an open challenge, and more generally even a small improvement in the ability to match chemicals to known phenotypic landmarks can greatly impact drug discovery by enabling the assignment of unknown chemical entities to a specific mechanism of action, target, or biological function (Chandrasekaran et al., 2024).

We also find convincing qualitative clustering of the glucocorticoid receptor agonist dexamethasone. Dexamethasone is an example of a compound for which it is challenging to see any visually discernible effect (Figure 12), and furthermore we have found limited success extracting it with CellProfiler. Quantitatively the mAP for this MoA is 0.14 using CellProfiler and 0.64 for the model of Figure 8 illustrating that our deep learning method can identify phenotypes that might otherwise be missed.

C.2.2. Out-of-distribution generalization and batch effects

Considering the extent to which SPC can make inferences about concepts that were not included in the optimization objective as metadata, we first note treatment information does not directly encode MoA or, for RxRx3-core, the relationships among genes and their interactions with molecules. Strong results on these tasks demonstrates that we've effectively learnt meaningful biological relationships that weren't injected during training.

Table 6. Out-of-distribution generalization on HaCaT screens.

Method	Training data	Eval data	1-NN	mAP
WSL	CRISPR	JUMP	.335	.090
SPC	CRISPR	JUMP	.429	.119
WSL	JUMP	CRISPR	.510	.179
SPC	JUMP	CRISPR	.550	.227

Table 7. SPC batch effect quantification with HaCaT cells.

Dataset	Graph conn.	Silhouette	KBET
JUMP 1 screen	.450	.817	.652
JUMP 2 screens	.450	.806	.600

Nonetheless, it is also of interest to see how SPC generalizes to classes and batches not present during training. In the main article we reported for the HaCaT screens results with holdout plates and perturbations. We now take this a step further and report results training on one screen, JUMP or CRISPR, and evaluating on the other. This allows us to assess performance on a perturbation modality and experimental batch not seen during training (the CRISPR and JUMP screens were performed five months apart). Table 6 results show that SPC, unlike WSL, continues to outperform CellProfiler and SimCLR on an unseen perturbation modality (compare cell holdout column in Table 3).

These results, along with main article results on BBBC021 and RxRx3-core (which contains 180 experimental batches and 1744 plates) indicate a robustness to batch effects. We add a quantification of batch effects with a second HaCaT JUMP screen which has been prepared and imaged under the same conditions as the one described in the main article, except four months later, allowing us to isolate batch effects from other differences in screen content. Employing three metrics measuring batch effects (see (Arevalo et al., 2024)), graph connectivity, silhouette batch and KBET, we take plate as the batch variable and report scores in Table 7. It is seen that there is relatively little increase in batch effect metrics by incorporating a second experimental batch.

C.3. RxRx3-core

Table 8 reports results for individual SPC training runs. We also repeat the comparison with the exact softmax approaches tested on BBBC021 including the L_2 variety, where again SPC shows stronger performance. More details on the metrics used in Table 4 and Table 8, including the

Table 8. Further RxRx3-core results on biological database recall and drug-target interaction.

Method	Biological recall					Drug-target interaction	
	CORUM	HuMAP	Reactome	SIGNOR	StringDB	AP	Z-score↑
L_2-Softmax($\tau = 0.01$)	.418	.468	.184	.189	.365	.295 ± .017	3.93
L_2-Softmax($\tau = 0.1$)	.408	**.501**	.189	.189	.364	.268 ± .017	2.48
WSL	.414	.494	.195	.189	.371	.287 ± .019	3.16
SPC (mean)	**.428**	.498	**.204**	**.198**	**.382**	**.298 ± .016**	**4.31**
SPC training run 1	.423	.511	.221	.200	.386	.298 ± .017	4.09
SPC training run 2	.443	.497	.200	.195	.385	.298 ± .015	4.45
SPC training run 3	.420	.487	.193	.200	.376	.297 ± .017	4.40

definition of the Z-score, are given in (Kraus et al., 2025).

D. HaCaT screening

We followed the standard cell painting assay protocol (Bray et al., 2016; Cimini et al., 2023), except for optimisations of dye concentrations to achieve optimal staining in HaCaT cells. For our staining procedures, MitoTracker was added live to cells, followed by cell fixation, permeabilisation and staining for the following subcellular compartments: nuclear DNA (Hoechst 33342), endoplasmic reticulum (concanavalin Alexa-488), RNA and nucleoli (syto-14), Golgi apparatus and plasma membrane (wheat germ agglutinin Alexa-555), and F-actin (phalloidin Alexa-568). Eight chemical substitutions to JUMP-MoA were made due to intellectual property constraints and two due to supply issues. We treated JUMP for 3 days. CRISPR screening was performed using 10 nM guide RNA, treating for 5 days to ensure maximum gene knockdown efficiency. Images were acquired using an Opera Phenix Plus high content microscope, and were max projected per channel.