Deep Morphological Profiling for Chemical Perturbation Analysis in the CPJUMP1 Dataset

Beini Wang

April 1, 2025

Abstract

1 Introduction

Recent advances in microscopy have enabled large-scale screening of chemical and genetic perturbations making changes on the phenotypes of cells (Chandrasekaran et al., 2024; Feldman et al., 2022). Chandrasekaran et al. (2024) provided the CPJUMP1 dataset which contains images of cells treated by different chemical compounds that provides a benchmark for evaluating methods that measure perturbation similarities and learn effective representations. With the advancement of statistical learning, supervised methods like machine learning with crafted features, deep learning with multiple data source, and deep unsupervised methods using embeddings from variational autoencoders (VAEs) can learn richer, data-driven representations and construct more effective models Pawlowski et al. (2016); Kensert et al. (2019); Ong et al. (2023); Wong et al. (2023); Tang et al. (2024); Stossi et al. (2024); Palma et al. (2025).

The goal of this project is to study how cells respond to different drug treatments using morphological profiles with a focus on evaluating, comparing, and validating the performance between traditional machine learning and deep learning methods using extracted morphological profiles and corresponding cell images on classification tasks. To achieve this goal, we split the project into 4 parts. Firstly, we extract morphological profiles from the cell images using existing Python packages (scikit-image) and generate additional cell segmentation images using *CellPose*. Secondly, we build models to predict the drug treatments from given images using supervised machine learning and deep learning methods with cell images and extract morphological profiles. Then, we discover biologically similar treatments by clustering on the treatment level using unsupervised methods with extracted morphological profiles and embeddings from Variational Autoencoders (VAE). After these analyses, we validate both the goodness of profiles and embeddings using Mean average precision (mAP) (Chandrasekaran et al., 2024) and the goodness of the clustering using external morphological features from *CellProfiler*.

The rest of the report is organized in the following way. Section 2 introduces the dataset used, and the packages/software used for image feature extraction. Section 3 discusses the supervised and unsupervised learning techniques used throughout this project. Section 4 summarizes the results of these analyses and compares the performance of different methodologies. Section 5 concludes this report and provide possible next steps for futures studies.

2 Dataset and Feature Extraction

Throughout this report, we use a subset of the CPJUMP1 dataset (plate `BR00116991`), which includes 2,867 grayscale images (Chandrasekaran et al., 2024). Each image is accompanied by its records in the metadata, which specifies the chemical compound, coordinates and site positions.

To ensure a better performance for downstream analysis, we use *Cellpose* to do the cell segmentation and generate additional cell mask images. Cellpose was run offline, and the resulting masks were matched to the original images via filename prefixes. Each mask was validated by visual inspection for quality and accuracy. These images use different coloring to distinguish cells which complements the identifiability of original images. A typical image generated by *Cellpose* is shown in Figure 1.

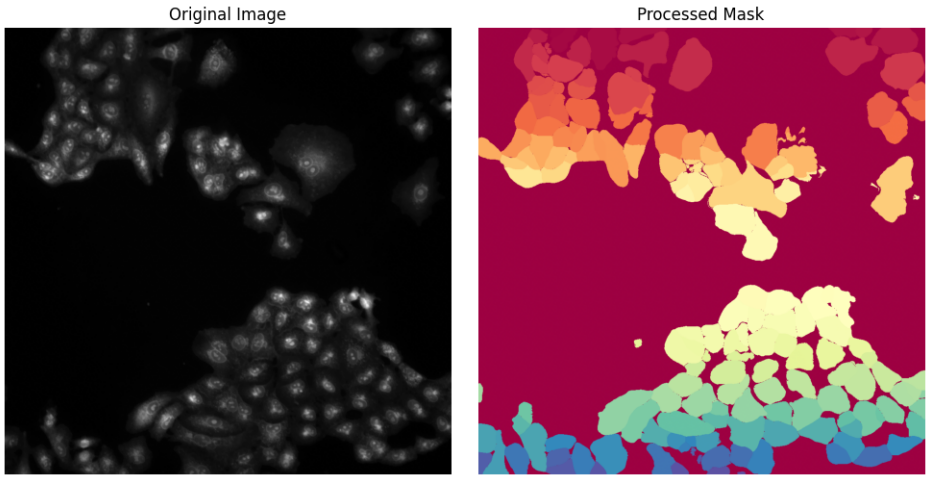

Figure 1: A typical image generated by *Cellpose*.

In addition, we extract morphological features using the software *CellProfiler*, and the following modules: IdentifyPrimaryObjects, IdentifySecondaryObjects, IdentifyTertiaryObjects, MeasureObjectSizeShape, MeasureObjectIntensity, MeasureTexture, ExportToSpreadsheet. Not all the features in the final spreadsheets are used. We extract 308 morphological features containing the characteristics of the images (those starting with mean_). These features will be used to verify the consistency of discovered clusters/representations (more in Section 4).

To collect more information for classification tasks, we extract region-based features (*regionprops* in the scikit-image package) from the original images. These features include area, eccentricity, mean intensity, etc. Then, they are aggregated with the corresponding image for classification.

All code, including data loading, feature extraction, model training, and analysis scripts, is available at Colab Link. Supplementary materials and additional resources are provided at GitHub Repository Link.

3 Methods

3.1 Treatment Classification

The first task after data preparation and feature extraction is the multiclassification of treatments based on cell images. For this task, we use two different models, the traditional machine learning

method (Random Forest), and the deep learning method (Convolution Neural Network).

For the random forest model, we apply two different feature engineering approaches. The baseline model uses only the basic statistics (mean, std, min, and max pixel values) from each image. Then, an enhanced model is developed using information in both the original images and masked (segmented) images from *CellPose*. For each (Image, Mask) pair, we use the *regionprops* function from the scikit-image package to compute the area, eccentricity, perimeter, mean intensity, and solidity. To ensure that all labels are represented in both the training and test sets, we used a stratified split with an 80-20 train-test ratio for both models.

For the Convolutional Neural Network (CNN), we construct a model with two convolutional layers and one fully connected layer. The baseline model is trained using the original grayscale images, and the enhanced model combines the original grayscale images with the segmentation-masked images from *CellPose*. The model was trained using label-encoded treatment classes with an early stopping mechanism based on validation loss. The dataset is split into 70% training, 15% validation, and 15% test sets.

3.2 Treatment Clustering

In this part of the analysis, we use clustering methods to find if there is any treatment that has a similar effect on cell morphology. We apply two different methods for image embedding in this task. The first method uses the same extracted features for each (Image, Mask) pair as in Section 3.1. The second method uses the latent embedding from a Variational AutoEncoder (VAE). A VAE is first trained by concatenating the original grayscale images with their corresponding masked image from *CellPose*. After training, each image pair is passed through the encoder, and the resulting mean vector from the latent space is extracted and used for the clustering task.

For both methods, the clustering is on the treatment level and the extracted features are aggregated for each treatment. We perform KMeans with various k (2 to 15) and choose the best k by silhouette score. The clustering results are visualized using both UMAP and t-SNE. Details can be found in Section 4.

3.3 Extracted Feature Quality

To evaluate how well representations recover images from the same treatment, we use mean Average Precision (mAP) from Chandrasekaran et al. (2024). mAP measures how well similar samples are ranked based on their pairwise cosine similarity. For each representation vector, average precision is computed by ranking all other samples and checking how well the model retrieves its true ranking. The final score is the mean of these average precision scores across all samples.

Similar to Chandrasekaran et al. (2024), we check the mAP for different well positions under two cases. In the first case, we did not distinguish the well position, data from both similar and different well positions are pooled together. For the second analysis, we split the data into similar well positions and different well positions. These two categories are defined using the 10th percentile of non-zero distances among all pairs of cells.

3.4 Clustering Validation

To validate the clustering results using the latent embedding from VAE, we use the extracted features from *CellProfiler*. We compute the cosine similarities among all pairs of images using these extracted features both within groups and between groups. Then we use the Mann-Whitney U test to compare if the similarities between groups are significantly larger than those within groups.

4 Results

4.1 Treatment Classification

Random Forest. The baseline model using original images achieves an overall accuracy of 0.12 across all categories on the test set, while the enhanced model using additional masked images improves the accuracy to 0.18. The weighted average precision, recall, and f1-score are 0.06, 0.12, 0.08 and 0.08, 0.18, 0.11, respectively. There are two possible reasons that may lead to such a performance. First, we only have around 2,000 training samples compared to the large number of treatment types. Secondly, we have a huge class imbalance issue where one of the treatment types (DMSO) takes up to around 20% of the total samples.

Convolutional Neural Network. Both the baseline (single-channel) CNN and the enhanced (grayscale + mask) CNN achieved a similar performance. The overall accuracy is 0.18 and 0.16, respectively. The weighted average precision, recall, and f1-score are 0.03, 0.18, 0.06 and 0.02, 0.16, 0.04, respectively.

The slight degradation using the additional masked images shows that Cellpose mask as an additional channel may theoretically improve the coverage of classes, however due to the small sample size and large number of features, the model may not fully capturing the association between features and the treatment type. We also see that the treatment type DMSO is predicted more often than other types which is due to a similar reason described in the random forest model.

A summary of results for baseline and enhanced random forest and CNN can be found in Table 1. As we can see form the table, CNNs are not as good as random forests as CNNs typically require large datasets to effectively learn spatial patterns in image data, especially when trained from scratch. With only around 2,000 training samples, CNNs are prone to overfitting and may not generalize well. On the other hand, Random Forest can perform relatively well on small datasets since it relies on decision trees and are robust to overfitting when properly pruned.

Table 1: Performance comparison of Random Forest and CNN models

Model	**Version**	**Accuracy**	**Precision / Recall / F1**
Random Forest	Baseline (grayscale only)	0.12	0.06 / 0.12 / 0.08
Random Forest	Enhanced (grayscale + mask)	**0.18**	**0.08 / 0.18 / 0.11**
CNN	Baseline (grayscale only)	**0.18**	0.03 / 0.18 / 0.06
CNN	Enhanced (grayscale + mask)	0.16	0.02 / 0.16 / 0.04

4.2 Treatment Clustering

Clustering based on Extracted Features from Original and CellPose Images. The extracted features are aggregated at the treatment level before passed into the KMeans algorithm Using the silhouette method, the best number of clusters is $k = 2$. The t-SNE and UMAP projections indicated two main clusters with no strong substructure. Many treatments ended up in one major cluster, with a smaller secondary cluster. Figure 2 and 3 present the clustering result using extracted features.

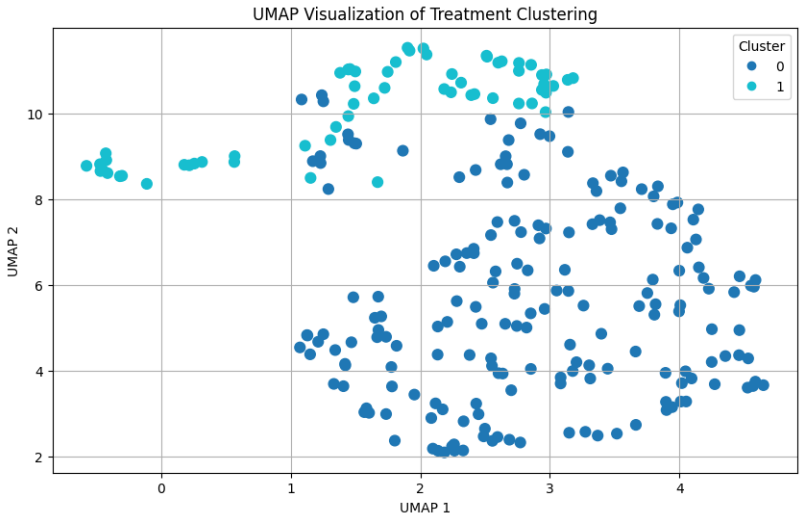

Figure 2: Visualization of clustering with features from original and CellPose images using UMAP.

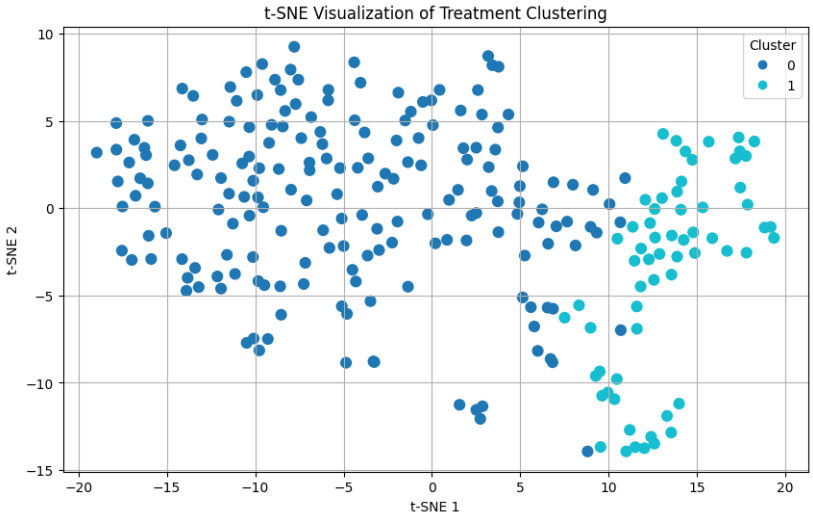

Figure 3: Visualization of clustering with features from original and CellPose images using t-SNE.

Clustering based on VAE Embeddings. The VAE is trained with early stopping and maximum 30 epochs. The latent dimension is set to be 32. Similar to the extracted features, each perturbation embeddings are aggregated at treatment level before passing into the KMeans algorithm. Again the silhouette method suggests $k = 2$. Figure 5 and **??** show the clustering result using VAE embeddings. We have two clusters of treatments, which could possible indicating that morphological changes might split between "low effect / small morphological difference" and "higher effect / distinct morphological difference" classes.

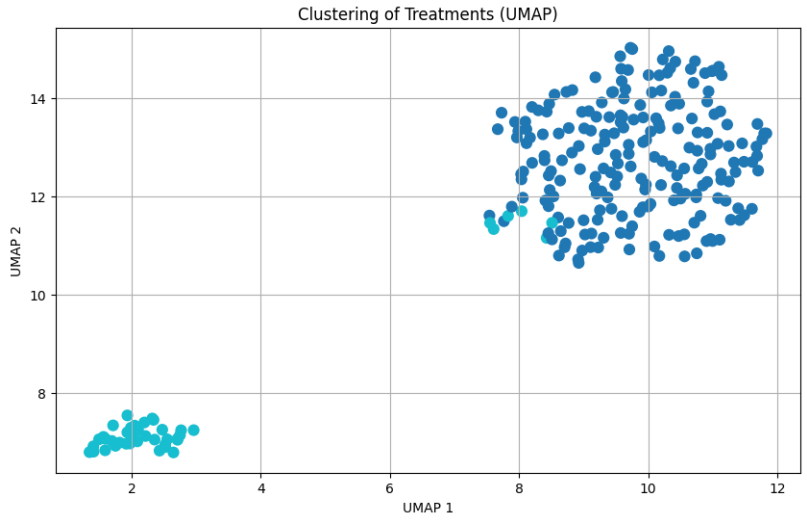

Figure 4: Visualization of clustering with features from VAE embeddings using UMAP.

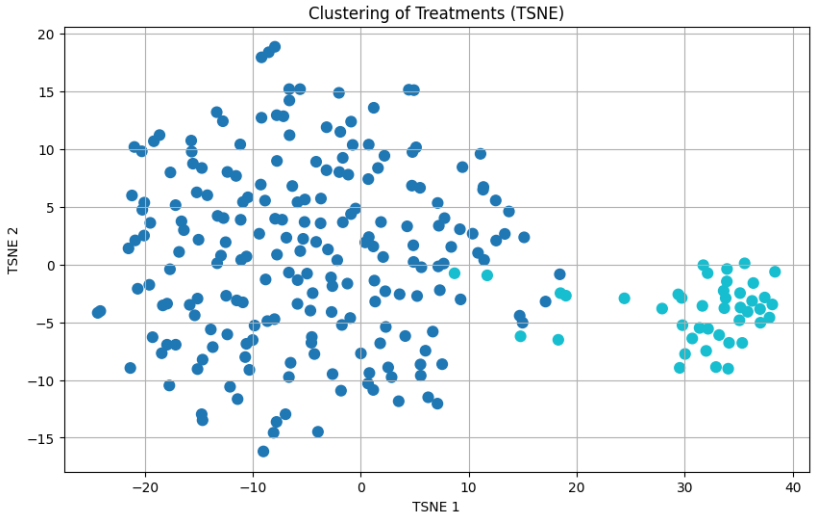

Figure 5: Visualization of clustering with features from VAE embeddings using t-SNE.

4.3 Mean Average Precision (mAP) Analysis for Extracted Feature Quality

In this section, we present the results for the method described in Section 3.3. We evaluate the quality of extracted features from both the (Image, Mask) pair and the latent embeddings from VAE. Here we calculate the cosine similarity score on the image level rather than the treatment level. When we do not distinguish the well position, the mAPs are 0.0548 and 0.0434 for the extracted features and VAE embeddings, respectively. When we separate the similar position from different positions, the mAPs for similar positions are 0.0256 and 0.0174, respectively. The mAPs for different positions are 0.0515 and 0.0378, respectively. The results are summarized in Table 2.

Table 2: Mean Average Precision (mAP) scores for image-level similarity using extracted features and VAE embeddings

Evaluation Setting	**Extracted Features**	**VAE Embeddings**
No Well Position Distinction	0.0548	0.0434
Same Position Only	0.0256	0.0174
Different Position Only	0.0515	0.0378

From the results, the mAP scores are not very high. This is expected because even visually similar images can belong to different treatments, as shown in the clustering. Therefore, low mAP does not imply poor features - it may capture true biological similarity across compounds.

4.4 Biological Validation with CellProfiler

In this section, we use the *CellProfiler*, a popular and widely-used software to extract features from the images and use them to validate the clustering results in the previous section. Figure 6 and 7 are the boxplot for the two selected features for the 5 most frequent compounds.

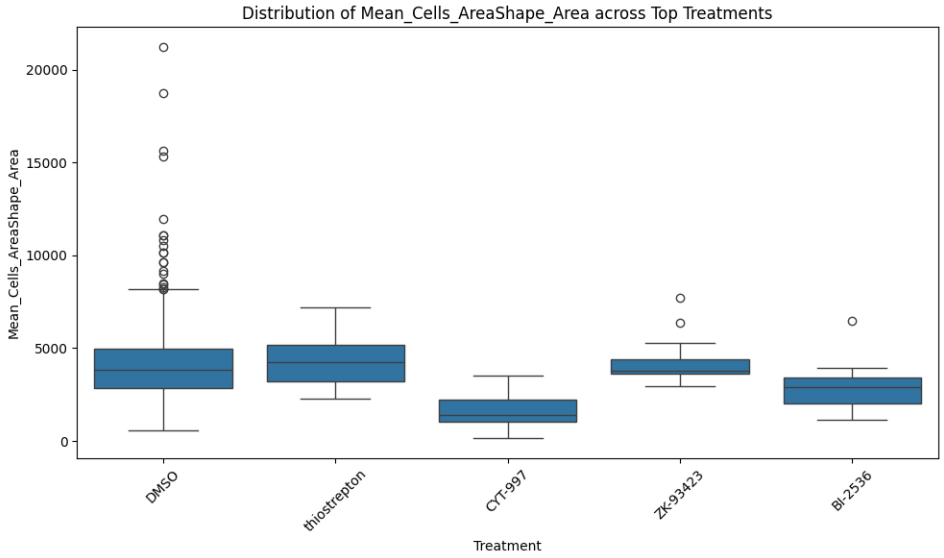

Figure 6: Visualization of average cell shape area for the 5 most frequent compounds.

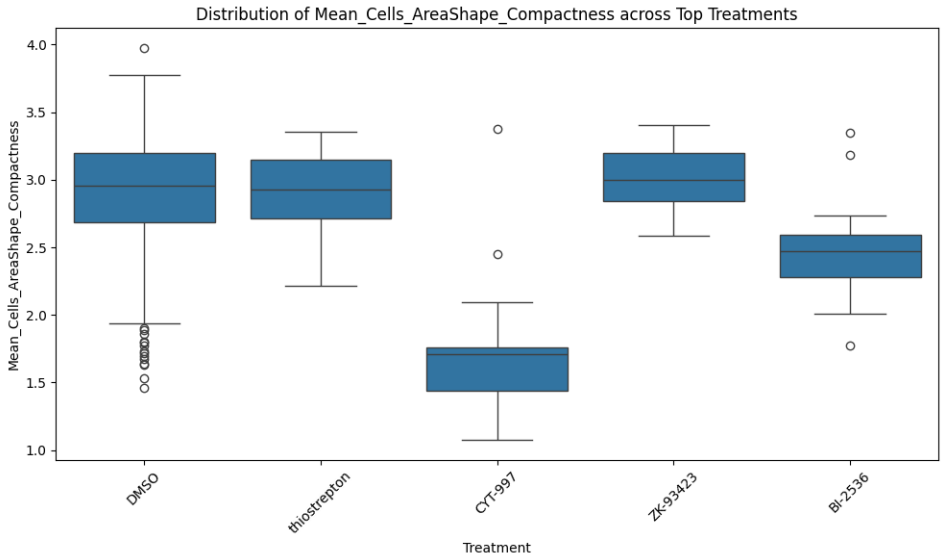

Figure 7: Visualization of average cell shape compactness for the 5 most frequent compounds.

As we can see from the plots, different treatments share a different feature profile and potentially, we can use these feature profiles at the treatment level to validate the goodness of our clustering algorithm.

As described in Section 2, we aggregate these features at the treatment level and compute the pairwise cosine similarity within and between the clusters. Figure 8 and 9 provide the visualization of the similarity scores for the two cases. We can see the difference are clear, indicating the goodness of the clustering results. The p-value for the Mann-Whitney U test is $p \ll 10^{-4}$.

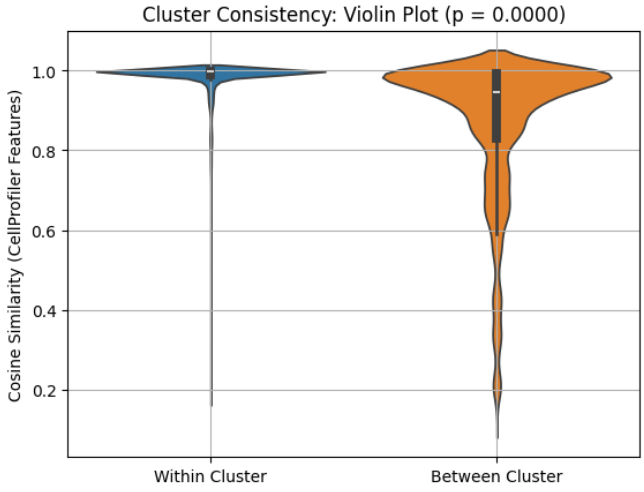

Figure 8: Violin plot for the similarity scores between and within the clusters.

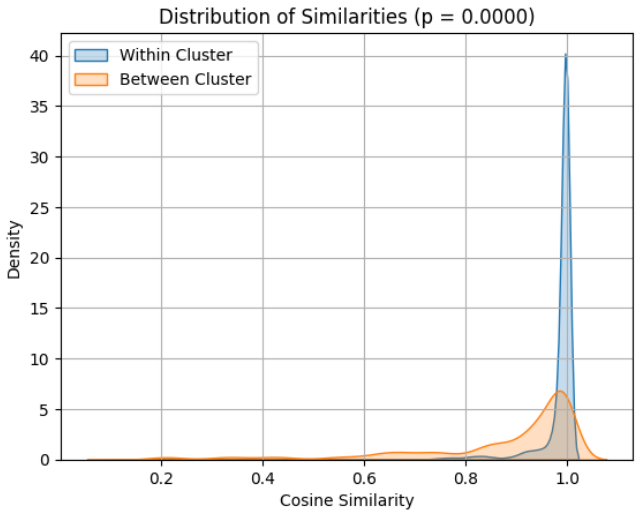

Figure 9: Boxplot for the similarity scores between and within the clusters.

5 Conclusion

Our study demonstrates that predicting drug treatments from single-cell images is hard when we have hundreds of compounds and morphological phenotypes. Although classification metrics like accuracy remained low, especially for small sampled treatments, augmenting grayscale features with Cellpose segmentation provided some improvement. A CNN approach also struggled, indicating the complexity of morphological differences among treatments.

Yet, clustering via regionprops or VAE embeddings showed that many treatments can make same change on cells, corroborated by CellProfiler morphological similarities. The low replicate mAP suggests morphological changes can be small and variable, and well-position artifacts complicate retrieval.

these findings shows that while simple classification is challenging for large chemical libraries, combining segmentation-based features, deep embeddings, and external morphological references offers biologically meaningful insights into how different perturbations make change on the cells. This can provide significance insight for future drug development.

5.1 Future Directions

For future work, there are some directions could improve the robustness and interpretability of drug treatment classification and clustering. First, oversampling techniques can be used to deal with class imbalance, which can make sure that rare treatments can also contribute to model training. Image augmentation methods could improve generalization by exposing models to a broader variety of morphological patterns. Using pre-trained models, particularly those trained on biological image datasets, might provide a stronger feature backbone. Finally, integrating gene pathway information with morphological features may strengthen the biological validity of define clusters and support some more hypotheses.

References

Chandrasekaran, S. N., Cimini, B. A., Goodale, A., Miller, L., Kost-Alimova, M., Jamali, N., Doench, J. G., Fritchman, B., Skepner, A., Melanson, M., Kalinin, A. A., Arevalo, J., Haghighi, M., Caicedo, J. C., Kuhn, D., Hernandez, D., Berstler, J., Shafqat-Abbasi, H., Root, D. E., Swalley, S. E., Garg, S., Singh, S., and Carpenter, A. E. (2024). Three million images and morphological profiles of cells treated with matched chemical and genetic perturbations. *Nature Methods*, 21(6):1114–1121.

Feldman, D., Funk, L., Le, A., Carlson, R. J., Leiken, M. D., Tsai, F., Soong, B., Singh, A., and Blainey, P. C. (2022). Pooled genetic perturbation screens with image-based phenotypes. *Nature Protocols*, 17(2):476–512.

Kensert, A., Harrison, P. J., and Spjuth, O. (2019). Transfer Learning with Deep Convolutional Neural Networks for Classifying Cellular Morphological Changes. *SLAS Discovery*, 24(4):466–475.

Ong, K., Cai, X., Marur, V., Soloveva, V., Mueller, U., and Chen, A. (2023). Deep learning-based rapid macrophage cell detection and localization in high-content microscopy screening. In Tomaszewski, J. E. and Ward, A. D., editors, *Medical Imaging 2023: Digital and Computational Pathology*, page 46, San Diego, United States. SPIE.

Palma, A., Theis, F. J., and Lotfollahi, M. (2025). Predicting cell morphological responses to perturbations using generative modeling. *Nature Communications*, 16(1):505.

Pawlowski, N., Caicedo, J. C., Singh, S., Carpenter, A. E., and Storkey, A. (2016). Automating Morphological Profiling with Generic Deep Convolutional Networks.

Stossi, F., Singh, P. K., Marini, M., Safari, K., Szafran, A. T., Rivera Tostado, A., Candler, C. D., Mancini, M. G., Mosa, E. A., Bolt, M. J., Labate, D., and Mancini, M. A. (2024). SPACe: an open-source, single-cell analysis of Cell Painting data. *Nature Communications*, 15(1):10170.

Tang, Q., Ratnayake, R., Seabra, G., Jiang, Z., Fang, R., Cui, L., Ding, Y., Kahveci, T., Bian, J., Li, C., Luesch, H., and Li, Y. (2024). Morphological profiling for drug discovery in the era of deep learning. *Briefings in Bioinformatics*, 25(4):bbae284.

Wong, D. R., Logan, D. J., Hariharan, S., Stanton, R., Clevert, D.-A., and Kiruluta, A. (2023). Deep representation learning determines drug mechanism of action from cell painting images. *Digital Discovery*, 2(5):1354–1367.