APPLICATION OF DEEP LEARNING ALGORITHMS TO SINGLE-CELL SEGMENTATION AND PHENOTYPIC PROFILING

Summary of the PhD Thesis

Nikita Moshkov

Doctoral School of Interdisciplinary Medicine,
Faculty of Medicine, University of Szeged

Academic supervisors:
Attila Kertész-Farkas, Ph.D.
Peter Horvath, Ph.D.
Juan C. Caicedo, Ph.D.

Szeged
2022

1 Introduction

1.1 The relevance of research

Advancements in high-throughput technologies made it possible to automatically and objectively analyze even on scales as large as millions and billions of cells, thus we have an opportunity to perform high-throughput experiments with single cells and then perform analysis with computational methods, applicable for the obtained type of data and try to make biological sense out of this data.

Different types of data (or data modalities) can allow us to inspect the state of each particular cell from different perspectives. One of the practical tasks, where all the possible information can be useful to make decisions, is drug discovery, especially in personalized medicine. The biggest challenge is to accurately and cost-effectively combine and use the existing expensive treatment modalities.

Here we focus mostly on the imaging data and one of the first steps of the image-based analysis of single cells is *cell or nucleus segmentation* – classification of each pixel as a background or foreground (semantic segmentation), or determining if the pixel belongs to a specific object (instance segmentation). In recent years this field has been emerging by adopting and creating deep learning algorithms for this task, bringing significant improvements [1].

The segmentation might be followed by the identification of biological phenotypes through the quantification of cell morphology, variation of which might show, for instance, differences between treated and not treated cells in drug screening experiments [2]. The phenotypes can be described by feature-vectors, also called *profiles* and the process of the extraction is called profiling and morphological profiling is also might be referred to as *image-based profiling* [3].

1.2 Publications

Papers related to the research topic:

- **Moshkov N.**, Mathe B., Kertesz-Farkas A., Hollandi R., Horvath P. Test-time augmentation for deep learning-based cell segmentation on microscopy images. Scientific Reports. 2020. Vol. 10, 5068. Q1 journal, IF 3.998 (2020). DOI: `https://doi.org/10.1038/s41598-020-61808-3`
- Hollandi R.*, **Moshkov N.***, Paavolainen L., Tasnadi E., Piccinini F., Horvath P. Nucleus segmentation: towards automated solutions. Trends in Cell Biology. 2022. Q1 journal, IF 20.808 (2021). DOI: `https://doi.org/10.1016/j.tcb.2021.12.004`
- Hollandi R., Diosdi A., Hollandi G., **Moshkov N.**, Horvath P. AnnotatorJ: an ImageJ plugin to ease hand-annotation of cellular compartments. Molecular Biology of the Cell. 2020 Vol. 31. № 20. P. 2157-2288. Q1 journal, IF 3.791 (2020). DOI: `https://doi.org/10.1091/mbc.E20-02-0156`

Preprints related to the research project:

- **Nikita Moshkov**, Tim Becker, Kevin Yang, Peter Horvath, Vlado Dancik, Bridget K. Wagner, Paul A. Clemons, Shantanu Singh, Anne E. Carpenter, Juan C. Caicedo. Predicting compound activity from phenotypic profiles and chemical structures bioRxiv 2020.12.15.422887, DOI: `https://doi.org/10.1101/2020.12.15.422887`
- **Nikita Moshkov**, Michael Bornholdt, Santiago Benoit, Matthew Smith, Claire McQuin,

Allen Goodman, Rebecca Senft, Yu Han, Mehrtash Babadi, Peter Horvath, Beth A. Cimini, Anne E. Carpenter, Shantanu Singh, Juan C. Caicedo. Learning representations for image-based profiling of perturbations bioRxiv 2022.08.12.50378, DOI: `https://doi.org/10.1101/2022.08.12.503783`

Conferences, related to the research project:

- HEPTECH AIME19 AI & ML (2019). Test-time augmentation for deep learning-based cell segmentation on microscopy images (poster). Link: `https://indico.wigner.hu/event/1058/contributions/2542/`

2 Nucleus segmentation: towards automated solutions

The article related to this section [1].

Personal contributions:

- Literature review of the segmentation and preprocessing methods (reflected in the text of the article, supplementary materials).
- Literature review of existing nuclei databases (reflected in the text of the article, supplementary materials).
- Formulate the conclusions (reflected in the text of the article).
- Contributions to implementation of the online software supplied with the article.
- Participated in correspondence with the reviewers.

The field of nucleus segmentation was developing over the last few years with the help of deep learning. Practitioners started to use widely deep learning-based segmentation methods, especially after the DSB 2018 challenge [4], which clearly showed the superiority of deep learning-based methods over the classical ones. Besides, the computational resources have become more affordable, and the methods tend to be more user-friendly by providing guides for the tools and sometimes by providing graphical user interfaces. The review is aimed to provide an overview of the methods and datasets related to nuclei segmentation and guide practitioners in the field.

The main result of the review is the raising of concerns and questions about the current state of the field. The first concern is related to the lack of diversity of existing datasets in terms of microscopy modalities. Turns out most of those openly published annotated datasets are either for H&E images or fluorescent images. Other microscopy modalities (e.g, DIC (differential interference contrast), light-sheet or phase contrast) are poorly represented in publicly available datasets. Besides, the size of the published datasets also matters, most of the datasets do not contain many objects and images.

Another point is a call for a solution to the common challenges in nuclei segmentation (touching, overlapping and irregularly shaped nuclei). Current deep learning methods are able to partially address those challenges, but more progress is desired. Both novel model architectures and high-scale training datasets might positively impact in this regard.

The real problem, which is on the surface, but rarely discussed, is the lack of a unified approach for the evaluation of nuclei segmentation methods. After inspecting all the methods eventually presented in the review, it has become clear that the evaluation methods and the datasets don't overlap. Even though there are datasets that are supposed to be the standards, different subsets of the test sets are getting used in different articles. The problem could

be solved by discussions inside the community and enforcing the standards. Two candidate platforms to host such standardized tests could be Kaggle and BIAFLOWS [5].

The last conclusion of the paper is that the field could try to move towards the general models which can segment nuclei from images of diverse modalities. Some models are already capable of doing this, though with a limited amount of modalities.

This review goes with the assistant tool for nuclei segmentation method selection (called *unbiased*) which is available online at GitHub Pages `https://biomag-lab.github.io/microscopy-tree/`. It is supposed to help in choosing potentially useful methods based on microscopy modality, the dimensionality of images and potential challenges in the data of interest.

3 AnnotatorJ: an ImageJ plugin to ease hand annotation of cellular compartments

The article related to this section [6].

Personal contributions:

- Contributions to the implementation of the AnnotatorJ codebase (integration of Keras models).

To train a single-cell (nuclei) segmentation based on deep learning, annotated data is needed. To train more robust models, bigger datasets are desired, but manual annotation is an expensive process as it requires a significant amount of time and effort from biology experts. To make the annotation process faster and more accurate, a plugin AnnotatorJ [6] for ImageJ/FIJI [7] (the software for bioimage analysis) was developed which combines single-cell identification with deep learning and manual annotation.

The main feature of AnnotatorJ is a contour assistant. Contour assistant uses the pre-trained U-Net model to predict the area covered by the object of interest. After that, the user can refine the contours of the object if needed.

To make trained models compatible with ImageJ/Fiji, which is developed in Java, we used the library DL4J and ND4J (`http://deeplearning4j.org/`). AnnotatorJ is openly available at `https://github.com/spreka/annotatorj`.

4 Test-time augmentation for deep learning-based cell segmentation on microscopy images

The article related to this section [8].

Personal contributions:

- Prepared the experiments.
- Conduct the experiments.
- Interpretation of the results.
- Text of the article.
- Correspondence with the reviewers.

Deep learning-based nuclei segmentation heavily relies on manually annotated data, which in most cases is annotated by domain experts. To increase the amount of training data and train more robust models, data augmentation [9] has become a common technique in deep

learning. Data augmentation is frequently used in the case of diverse or limited datasets, which is often the case in the field of nuclei and cell segmentation.

While the usual data augmentation approach is performed during the training time, the idea of another approach, test-time augmentation (TTA) (Figure 1) is to perform predictions on the original and the augmented versions of the data samples and then merge the predictions. The experiments with test-time augmentation were conducted in the setting of the nucleus segmentation task.

4.1 Test-time augmentation

The pipeline of test-time augmentation includes four steps:

1. Augmentation of the original image.
2. Inference of original and augmented versions of the image.
3. Dis-augmentation: if the original image was flipped or rotated, the transformation should be reverted to the original orientation to allow further correct merging of the predictions.
4. Final merging: this step is different for Mask R-CNN and U-Net and discussed further.

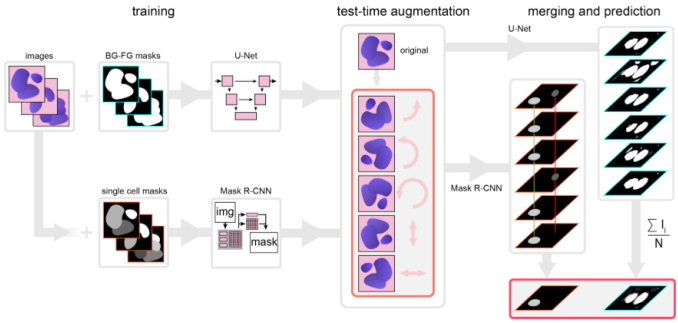


Figure 1: Proposed test-time augmentation techniques. Input: Run inference on several augmented instances of the same test images with trained models. To merge predictions, pixel-wise majority voting used for U-Net, object matching and majority voting used for Mask R-CNN. The source of the figure [8].

For U-Net predictions step (4) is straightforward, just sum and average all the dis-augmented probability maps. The resulting probability map is then converted to a binary mask by thresholding (0.5) which is further used for evaluation of the segmentation (Figure 1, right).

Mask R-CNN, as an instance segmentation framework, requires more post-processing. Here, each object is processed separately: for each detected object the majority voting is done. Before majority voting the object alignment should be done: the objects from the predictions of original and augmented versions of the input image are checked if those can be considered the same object. In this setup, two objects (each from different versions of the input image) are considered to be the same object if the intersection over union (IoU, also known as Jaccard Index) between them is at least 0.5. If the same detected object is present in the majority of the predictions, then it will be included in the final prediction mask. The mask of the included object is corrected by majority voting on the pixel level.

4.2 Results

Test-time augmentation improved the performance for all the train-test splits on average, if used together with Mask R-CNN models. The mean gain in the mAP_{DSB} metric is between 0.01 and 0.02. In some test examples, test-time augmentation could change the prediction quality by a large margin (see examples in Figure 2).

Test-time augmentation combined with the method [10] (the best performing method for the DSB 2018 test set according to the Kaggle scoreboard at the time of publishing of the paper [8]) further increases the performance by 0.011 in mAP_{DSB}.

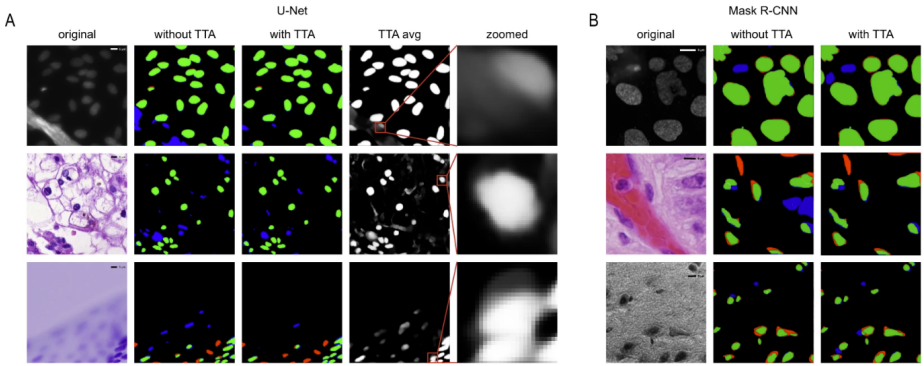


Figure 2: Comparison of predictions with and without TTA on example images. A. U-Net. First column: original image, the second: predictions without TTA, the third: predictions with TTA. Colors: false negative predictions (red), true positive (green), and false positives (blue). The fourth column – averaged TTA predictions before thresholding and the fifth: zoomed insets from the previous column. Rows are example images. B. Mask R-CNN. Columns are as first three in A, rows are example images. The source of the figure [8].

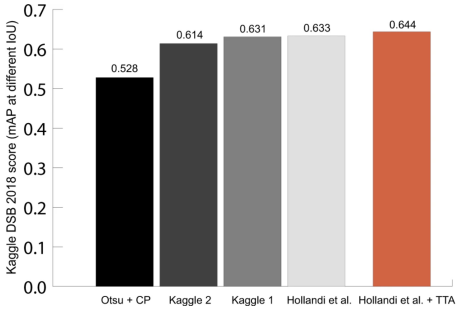


Figure 3: DSB 2018 Stage 2 test scores for different methods, compared to [10] + TTA. The source of the figure [8].

5 Learning representations for image-based profiling of perturbations

The article related to this section [11].

Personal contributions:

- Prepared the experiments.
- Contributions to implementation of DeepProfiler software.
- Conduct the experiments.
- Data analysis and figures based on the results of experiments.
- Contributions to the text of the article and interpretation of the results.

5.1 Introduction

Phenotypic drug discovery is based on observations of drug effects on treated subjects, here, we consider single-cells. This problem not only requires significant wet lab efforts but also computational approaches to process the output data. CellProfiler [12] is the standard approach to extract representations of single-cells. It produces features which are human-readable and their usefulness was proven in different downstream tasks [13]. To extract even more biologically relevant representations of cells from images using deep learning in this work the attempt is to train deep learning models directly on images of single-cells with weakly-supervised learning (WSL) approach.

Now, a systematic evaluation of three large-scale Cell Painting public datasets is conducted. Those datasets contain thousands of perturbations, hundreds of plates, and millions of single cells. The tested representations are extracted by pre-trained models and models trained in a weakly-supervised setting and compared against classical features. To run training and feature extraction experiments, the publicly available tool *DeepProfiler* was developed.

The current best practices found for making deep learning methods improve the quality of downstream analysis. For interpretation of the obtained results with trained models and reasoning about challenges, a causal modeling framework is used [14] [15].

5.2 DeepProfiler

The first outcome of the project is pipeline called DeepProfiler, which is used to conduct experiments. It helps to train weakly-supervised models and extract representations of single-cells from high-throughput imaging experiments. Besides training and feature extraction, DeepProfiler has additional features for image compression and extraction of single-cell crops from full-sized images into separate image sets. The workflow is shown in Figure 4. The framework is implemented in Tensorflow [16] (for both versions 1 and 2). The source code, documentation and discussions are available on the GitHub page (`https://github.com/cytomining/DeepProfiler/`).

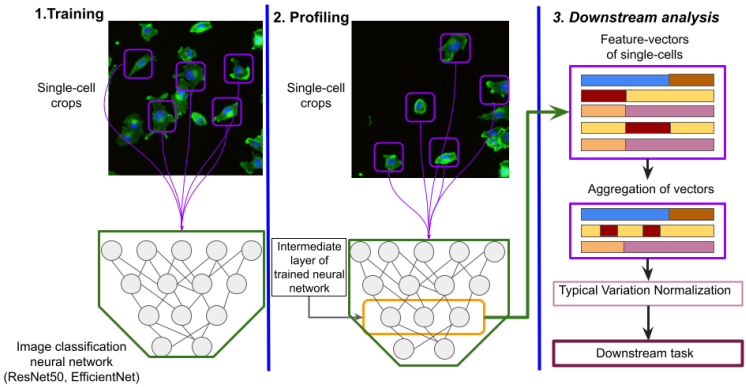


Figure 4: Typical usage of DeepProfiler. 1. Perform training of image classification network 2. Use the trained model to extract the representations 3. Use the representations for downstream analysis tasks. Steps 1 and 2 in the image are performed with DeepProfiler, step 3 is a user preference. The microscopy images used from the BBBC021 dataset [17].

5.3 Combined Cell Painting dataset

Another produced resource along with the project is a *combined Cell Painting dataset*. It was created to expand the potential feature-space both with biological and technical variation the treatments resulting into a strong phenotypes were collected from five Cell Painting datasets describe in the main text of the thesis. Three of those five datasets are used as benchmarks. Strong treatment is defined as one to produce a phenotype which is different to a phenotype of untreated cells. Resulting dataset contains 8.3 million single cells from 232 plates, 488 treatments and 2 types of negative controls.

5.4 Experimental setup

5.4.1 Experiments with pre-trained models

In this approach, pre-trained on ImageNet dataset [18]. As pre-trained networks require 3-channel input, each of the channels is replicated three times and sent to the model separately. As an input, single-cell crops of size 128×128 were used. The preprocessing includes resizing to 224×224 and min-max normalization to have a final input in the range $[-1, 1]$. The features were extracted from the *block6a_activation* layer. For each channel, the output dimensionality is 672 features, thus the full feature vector for the cell is 3360.

5.4.2 Experiments with weakly supervised learning

Training and the following feature extraction were conducted with DeepProfiler. The inputs are pre-cropped images of single-cells, saved as a stripe of five channels and reshaped during training. Class auto-balancing is done in each epoch of training. For all datasets, the parameters were: categorical cross-entropy loss, batch-size 32, a constant learning rate of 0.005 with SGD

optimizer, augmentations on, no label smoothing and 30 epochs. The models are initialized with ImageNet pre-trained weights.

Two setups for splitting the data to training and validation were used:

- Leave-plates-out - the single-cells from one subset of plates are used for training, and from another for validation.
- Leave-cells-out - the single-cells from each plate and each well are used both in training and validation, approximately 60% of cells from each well are used in training, 40% in validation.

Using trained models, features were extracted from *block6a_activation* layer (feature vector size is 672).

5.5 Causal relations in screening experiments

By applying different treatments to cells, biologists are trying to perturb their state and observe the response. The causal graph for that kind of experiment includes four variables: treatments T, images O, phenotypes Y and batch-effects C. In causality modeling terms, those are interventions, observations, outcomes and confounders respectively. T and O are observed variables, while Y and C are latent variables. The goal is to learn Y, a multidimensional representation of treatment, which could be used in the further downstream task. To be useful in the downstream analysis task, Y should encode biologically relevant representation, though the reality is that technical variation, the batch-effects C affect all other elements of this causal model. C affects images by technical variation in the image acquisition process, treatments by plate-layout design and phenotypes by environmental conditions. The relations are shown in the graph (Figure 5).

Treatment is expected to be the main cause to change in the phenotype of the cell. To extract the representation of phenotypic outcome, WSL is used with the pretext task of treatment classification. The representations extracted from the intermediate layers of CNNs encode all visual variation, in this case, both batch-effects and phenotypes. WSL together with batch correction would help to disentangle phenotypic variation from technical.

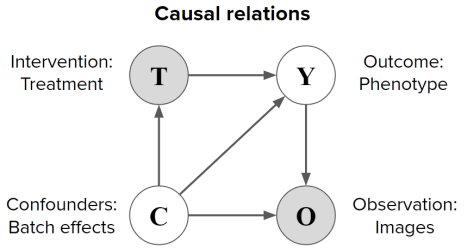


Figure 5: Causal model for screening experiment. T stands for treatments (interventions), O for images (observations), Y for phenotypes (outcomes) and C for batch-effects (confounders). The source of the figure [11].

5.6 Results and observations

The subsection discusses the results obtained with WSL on the combined Cell Painting dataset *CNN Cell Painting* model and models trained on the benchmark datasets. Pre-trained model on ImageNet (also referred to as *CNN ImageNet*) dataset and classical features extracted with CellProfiler serve as baselines.

5.6.1 Learned representations sharpen biological features

CNN Cell Painting model performs better in quantitative evaluation than both baselines in the evaluation task (Figure 6, cyan points). That was expected as manually engineered features might miss some information and the ImageNet model is trained on a completely different domain and not optimized for the images of cells. The models trained only on the corresponding benchmark datasets did not show a consistent improvement in their performance against the baselines (Figure 6, green points).

CNN ImageNet demonstrates similar or lower performance compared to CellProfiler features (Figure 6, yellow and pink points).

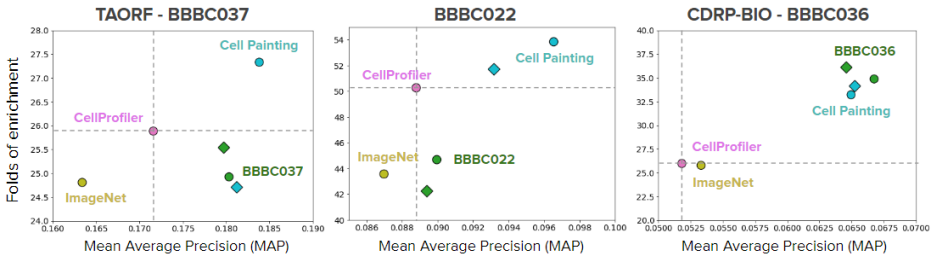


Figure 6: Quantitative performance of feature representations for three benchmark datasets in two metrics: mean average precision (X-axis) folds of enrichment (Y-axis). On the plot, the baselines are CellProfiler (pink) and CNN ImageNet (yellow), trained models: CNN Cell Painting model (cyan), trained on corresponding benchmark dataset (green). Leave-cells-out training-validation scheme shown with circles and leave-plates-out with diamonds. The source of the figure [11].

5.6.2 WSL learns both the phenotypes and the batch-effects

Different validation schemes leave-plates-out and leave-cells-out (see Experimental setup) help to understand the information contained in features learned from Cell Painting images. In leave-cells-out validation scheme the model as access to the full distribution of biological variation (treatments T) and technical variation (batch-effects C), yet with leave-plates-out scheme, the model still has access to the full distribution of biological variation, but only to a part of technical variation.

Major performance difference was observed in the pretext classification task for two validation schemes. In leave-cells-out setup, the trained CNN can accurately classify single-cells from both training and validation sets, while in leave-plates-out setup, the trained model completely fails to classify single-cells in validation set. Nonetheless, two models trained with different

validation schemes demonstrate similar performance in the downstream task (Figure 6). This observation leads to a conclusion that WSL models try to take advantage of any information that can explain the link between the images and treatments, including batch-effects.

5.6.3 Learning with strong phenotypes improves performance in the biological task

As in the previous section it was observed that controlling the distribution of confounding factors C does not change the downstream performance, now it is time to explore what happens if the phenotypic distribution Y is restricted. The intuition is that WSL minimizes an error in the pretext task by exploiting confounding factors to correctly classify treatments with a weak phenotypic response. Such treatments might have a stronger technical signal rather than a biologically relevant phenotypic signal.

WSL training only on strong treatments only in benchmark datasets was evaluated in leave-plates-out training-validation scheme. The results demonstrate minor performance improvement against training on full datasets (Figure 7, blue points).

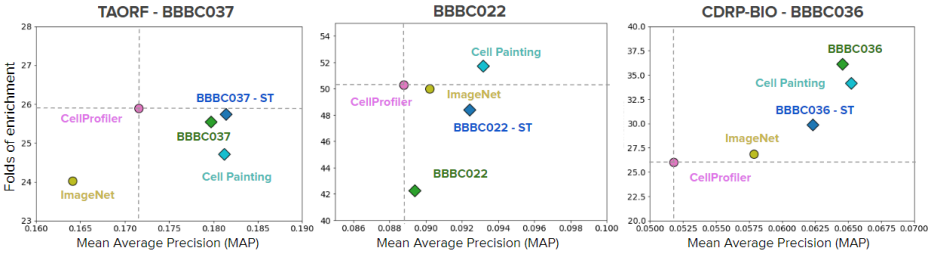


Figure 7: Quantitative performance of feature representations for three benchmark datasets in two metrics: mean average precision (X-axis) folds of enrichment (Y-axis). On the plot the baselines are CellProfiler (pink) and CNN ImageNet (yellow), trained models: CNN Cell Painting model (cyan), trained on corresponding benchmark dataset (green), trained on strong treatments from corresponding benchmark dataset (blue). All training experiments used leave-plates-out training-validation scheme. The source of the figure [11].

5.6.4 Diverse experimental conditions result in improved representations

The combined Cell Painting dataset was created to maximize both phenotypic (Y) and technical (C) variation by combining the treatments with the strongest resulting phenotypes from five datasets. Training on this dataset consistently improves performance over other approaches (Figure 6, cyan points), which means that this model can disentangle Y and C more efficiently. The most important outcome is that this model was trained once and could be used at all benchmarks without additional training.

5.6.5 Batch-correction is a crucial post-processing step

The role of batch-correction (see Batch correction using sphering transform in the thesis) is to reduce the impact of confounding technical factors C. It is crucial for all representations

tested: classical features, features extracted with pre-trained and trained CNNs. Mean average precision improves up to 90% versus raw features (see Figure 8). Also, using the effect of batch-correction can be observed qualitatively. It does not mean that the batch-effects are eliminated and further research is needed to learn how to disentangle technical and biological information in representations.

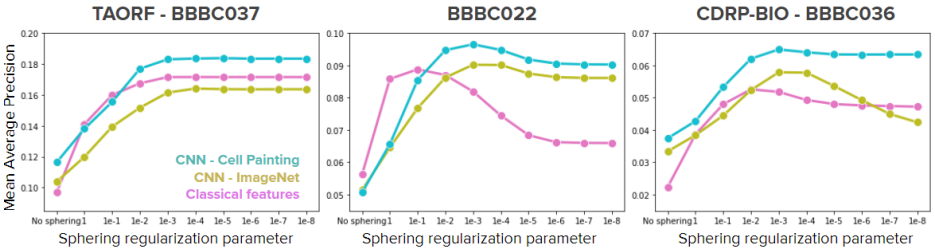


Figure 8: Mean average precision for sphering with different regularization parameters (smaller regularization term, more correction applied) for three datasets. For each dataset CellProfiler features (pink), ImageNet CNN (yellow) and Cell Painting CNN (cyan) are evaluated. The source of the figure [11].

6 Predicting compound activity from phenotypic profiles and chemical structures

The pre-print related to this section [19].

Personal contributions:

- Prepared the experiments.
- Conduct the experiments.
- Data analysis and figures based on the results of experiments.
- Contributions to the text of the article and correspondence with the reviewers.

Drug discovery is an expensive and very slow process, there are too many theoretically possible compounds to test in a real physical experiment. Even though pharmaceutical companies may afford to test millions of compounds in their experiments, this only covers a small fraction of possible compounds. Besides, to test those compounds the expensive phenotypic assay systems are used. Finally, this process is time-consuming and requires the time of experts to run the assays.

In this project, the aim is to evaluate the predictive power of the representations of chemical structures, cell morphology profiles and gene expression profiles, to predict assay outcomes computationally at a large scale. The hypothesis is that the predictive capabilities of those data sources are complementary and those data sources could be used together to further increase the success rate of the drug screening process. Besides, the basic data fusion techniques are tested, although it is not the focus of the project and this question might be investigated further.

6.1 Experiments and results

The experiments were conducted for several train-test split approaches. All the train test approaches share the same idea that we want to predict assays-compound interaction for compounds that are distinct relative to training data. From the practical perspective, there is little value in searching for similar chemical structures for the one with known activity. The closest train-test split to such a real-world scenario is a scaffold-based split (for 5-fold cross-validation) achieved with Bemis-Murcko clustering [20] [21]. Splits based on morphological and gene expression features were constructed using same-size K-Means clustering in their feature-spaces.

Our results show that morphology could accurately predict the largest number of assays with the median $AUROC > 0.9$ over cross-validation splits (28 for morphology, 19 for gene expression and 16 for chemical structures). Although, for lower AUROC thresholds (0.7) chemical structures tie with morphology. Interestingly, all three modalities share zero well-predicted assays and each pair of modalities share a few common well-predicted assays, which means that different data sources contain significantly complementary information.

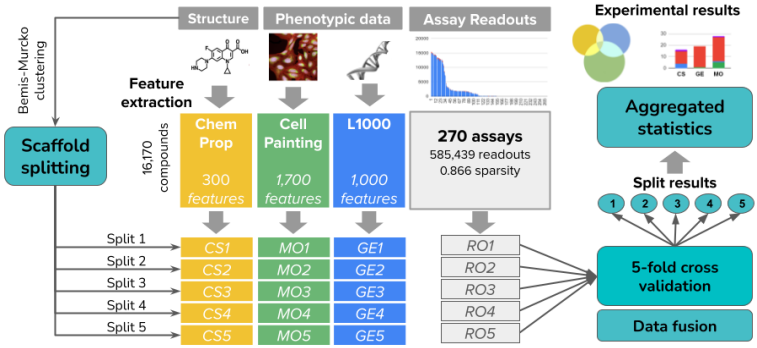
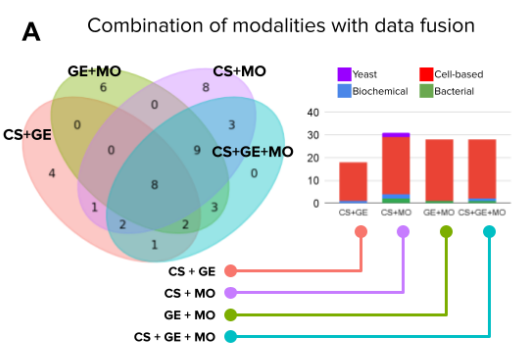


Figure 9: Illustration of experimental setup. The source of the figure [19].

Not only one modality can be used for predicting the assay-compound interaction. To combine modalities into a single predictor, two approaches were used: a) ***Early fusion*** - the feature vectors are concatenated into a single vector and used as an input for the neural network. b) ***Late fusion*** - for each modality the separate model is trained and then the prediction scores are aggregated, using the maximum probability among predictions for each compound-assay pair.

According to experiments, early data fusion did not provide any additional performance. Results for individual modalities did show that they do not share many well-predicted assays in common, and when the feature vectors are combined, additional noise to the assays is introduced, as assays can be well predicted by one modality but cannot be predicted by another. Late fusion works better in practice, though the performance gain is minor at best (Figure 10). The fusion approaches in the demonstrated tests are quite simple and more investigation for more effective fusion techniques is needed. As an additional metric, retrospective performance was measured. It is a simulation of the best possible data fusion. In this analysis, know the predictions are known in advance. Usage of fused with individual modalities can give 7-17% of

performance boost (Figure 10).



B Performance of individual modalities

	CS	GE	MO
# assays AUROC > 0.9	16	19	28

C Performance of data fusion by cross-validation (predictive)

	CS+MO	CS+GE	GE+MO	CS+GE+MO
# Median AUC > 0.9	31	18	28	28

D Retrospective analysis of predictors with AUROC > 0.9 (oracle counts)

	CS★MO	CS★GE	GE★MO	CS★GE★MO
Single	41	33	41	52
Plus fusion	44	37	48	57

Figure 10: Accurately predicted assays (median AUROC over splits is higher than 0.9). A. Venn diagram of accurately predicted assays using late fusion (left), bar plots show the distribution of accurately predicted assay types with late fusion (right). B. Number of accurately predicted assays per individual modality. C. Number of accurately predicted assays for combined modalities with the use of late fusion. Counts for median and mean AUROC over splits. D. Number of accurately predicted assays for retrospective analysis. "Single" is a simple union of the accurately predicted assays with individual modalities. "Plus fusion" is a union of accurately predicted assays with individual modalities plus the combined late fusion predictor. The source of the figure [19].

References

[1] Reka Hollandi, Nikita Moshkov, Lassi Paavolainen, Ervin Tasnadi, Filippo Piccinini, and Peter Horvath. Nucleus segmentation: towards automated solutions. *Trends Cell Biol.*, January 2022.

[2] Ben T Grys, Dara S Lo, Nil Sahin, Oren Z Kraus, Quaid Morris, Charles Boone, and Brenda J Andrews. Machine learning and computer vision approaches for phenotypic profiling. *J. Cell Biol.*, 216(1):65–71, January 2017.

[3] Mark-Anthony Bray, Shantanu Singh, Han Han, Chadwick T Davis, Blake Borgeson, Cathy Hartland, Maria Kost-Alimova, Sigrun M Gustafsdottir, Christopher C Gibson, and Anne E Carpenter. Cell painting, a high-content image-based assay for morphological profiling using multiplexed fluorescent dyes. January 2016.

[4] Juan C Caicedo, Allen Goodman, Kyle W Karhohs, Beth A Cimini, Jeanelle Ackerman, Marzieh Haghighi, Cherkeng Heng, Tim Becker, Minh Doan, Claire McQuin, Mohammad Rohban, Shantanu Singh, and Anne E Carpenter. Nucleus segmentation across imaging experiments: the 2018 data science bowl. *Nat. Methods*, 16(12):1247–1253, December 2019.

[5] Ulysse Rubens, Romain Mormont, Lassi Paavolainen, Volker Bäcker, Benjamin Pavie, Leandro A Scholz, Gino Michiels, Martin Maška, Devrim Ünay, Graeme Ball, Renaud Hoyoux, Rémy Vandaele, Ofra Golani, Stefan G Stanciu, Natasa Sladoje, Perrine Paul-Gilloteaux, Raphaël Marée, and Sébastien Tosi. BIAFLOWS: A collaborative framework to reproducibly deploy and benchmark bioimage analysis workflows. *Patterns (N Y)*, 1(3):100040, June 2020.

[6] Réka Hollandi, Ákos Diósdi, Gábor Hollandi, Nikita Moshkov, and Péter Horváth. AnnotatorJ: an ImageJ plugin to ease hand annotation of cellular compartments. *Mol. Biol. Cell*, 31(20):2179–2186, September 2020.

[7] Johannes Schindelin, Ignacio Arganda-Carreras, Erwin Frise, Verena Kaynig, Mark Longair, Tobias Pietzsch, Stephan Preibisch, Curtis Rueden, Stephan Saalfeld, Benjamin Schmid, Jean-Yves Tinevez,

Daniel James White, Volker Hartenstein, Kevin Eliceiri, Pavel Tomancak, and Albert Cardona. Fiji: an open-source platform for biological-image analysis. *Nat. Methods*, 9(7):676–682, June 2012.

[8] Nikita Moshkov, Botond Mathe, Attila Kertesz-Farkas, Reka Hollandi, and Peter Horvath. Test-time augmentation for deep learning-based cell segmentation on microscopy images. *Sci. Rep.*, 10(1):5068, March 2020.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, May 2017.

[10] Reka Hollandi, Abel Szkalisity, Timea Toth, Ervin Tasnadi, Csaba Molnar, Botond Mathe, Istvan Grexa, Jozsef Molnar, Arpad Balind, Mate Gorbe, Maria Kovacs, Ede Migh, Allen Goodman, Tamas Balassa, Krisztian Koos, Wenyu Wang, Juan Carlos Caicedo, Norbert Bara, Ferenc Kovacs, Lassi Paavolainen, Tivadar Danka, Andras Kriston, Anne Elizabeth Carpenter, Kevin Smith, and Peter Horvath. NucleAIzer: A parameter-free deep learning framework for nucleus segmentation using image style transfer. *Cell Syst.*, 10(5):453–458.e6, May 2020.

[11] Nikita Moshkov, Michael Bornholdt, Santiago Benoit, Claire McQuin, Matthew Smith, Allen Goodman, Rebecca Senft, Yu Han, Mehrtash Babadi, Peter Horvath, Beth A. Cimini, Anne E. Carpenter, Shantanu Singh, and Juan C Caicedo. Learning representations for image-based profiling of perturbations. *bioRxiv*, 2022.

[12] Anne E Carpenter, Thouis R Jones, Michael R Lamprecht, Colin Clarke, In Han Kang, Ola Friman, David A Guertin, Joo Han Chang, Robert A Lindquist, Jason Moffat, Polina Golland, and David M Sabatini. CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol.*, 7(10):R100, October 2006.

[13] Mohammad H Rohban, Hamdah S Abbasi, Shantanu Singh, and Anne E Carpenter. Capturing single-cell heterogeneity via data fusion improves image-based profiling. *Nat. Commun.*, 10(1):2082, May 2019.

[14] Donald B Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *J. Educ. Psychol.*, 66(5):688–701, October 1974.

[15] Fredrik Johansson, Uri Shalit, and David Sontag. Learning representations for counterfactual inference. In Maria Florina Balcan and Kilian Q Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 3020–3029, New York, New York, USA, 2016. PMLR.

[16] Tensorflow Developers. TensorFlow, 2021.

[17] Peter D Caie, Rebecca E Walls, Alexandra Ingleston-Orme, Sandeep Daya, Tom Houslay, Rob Eagle, Mark E Roberts, and Neil O Carragher. High-content phenotypic profiling of drug response signatures across distinct cancer cells. *Mol. Cancer Ther.*, 9(6):1913–1926, June 2010.

[18] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009.

[19] Nikita Moshkov, Tim Becker, Kevin Yang, Peter Horvath, Vlado C Dancik, Bridget K Wagner, Paul C Clemons, Shantanu Singh, Anne E Carpenter, and Juan C Caicedo. Predicting compound activity from phenotypic profiles and chemical structures. December 2020.

[20] G W Bemis and M A Murcko. The properties of known drugs. 1. molecular frameworks. *J. Med. Chem.*, 39(15):2887–2893, July 1996.

[21] Sebastian G Rohrer and Knut Baumann. Maximum unbiased validation (MUV) data sets for virtual screening based on PubChem bioactivity data. *J. Chem. Inf. Model.*, 49(2):169–184, February 2009.