Master's Thesis

Embedding optimization with deep neural networks for clustering image-based flow cytometry data

Department of Plant Sciences
Ludwig-Maximilians-Universität München

Dinesh Reddy Palli

Munich, July 31$^{\text{st}}$, 2023

`DineshPalliMasterThesis/Figures/sigillum.png`

Submitted in partial fulfillment of the requirements for the degree of M. Sc.
Supervised by Prof. Dr Fabian Theis & PD Dr Michael Menden

सत्त्वानुरूपा सर्वस्य श्रद्‌धा भवति भारत।

श्रद्‌धामयोऽयं पुरुषो यो यच्छ्रद्‌धः स एव स।।

You are what you believe you are.

“If I have seen further it is by standing on the shoulders of Giants”

- **Sir Issac Newton**

Contents

List of Figures

List of Tables

1 Abstract

Advances in cell-sorting enable collection of high-dimensional images capturing cell morphological diversity. However analyzing these datasets to identify distinct cell phenotypes poses challenges. The high-dimensional data generated by the Image-enabled Cell Sorter requires an embedding that enables the formation of biologically relevant clusters. This study evaluates distance metric learning and deep neural networks for clustering of classical image features and features extracted from cell images respectively. We assess if classical image features contain sufficient information to separate cell types and states on *Salmonella*, phytoplankton and cell cycle datasets. While providing preliminary separations, hand-engineered features lack complexity to capture nuanced morphological patterns denoting variations. We check pretrained deep convolutional networks, evaluating the performance, resulted in inferior performance relative to distance metric learning. We trained scDINO network on our data to evaluate the network's performance, which resulted in marginal improvement in its performance. The clustering performance was evaluated with cluster purity. By far, distance metric learning on classical image features proved to perform better at forming clusters of biological relevance, compared to the deep learning networks. However, by training vision transformers end-to-end on datasets, we demonstrate deep learning's potential to learn specialized features when optimized on relevant data. We are optimistic that with rigorous tuning, deep neural networks can uncover new biology from high-dimensional imaging datasets in an automated manner.

2 Introduction

2.1 Background

In biomedical sciences, understanding the connection between genetic variation and phenotypic diversity is of paramount importance. This relationship forms the foundation for deciphering the principles of cellular function, identifying the causes and markers of diseases, and is crucial for biological engineering and synthetic biology applications. High-throughput screening approaches are highly desirable in these contexts, as they allow for the efficient identification of variants exhibiting the properties of interest from heterogeneous populations of cells [?]. Methods that physically separate cells based on measurable characteristics have a wide range of usecases in both research and clinical applications, including cellular therapies. Techniques such as microfluidics, filters, and centrifugation can identify and physically separate cells from a heterogeneous population according to intrinsic characteristics like size, shape, and deformability [?]. This technique, when combined with high-throughput screening approaches, provide a powerful toolset for our understanding of genetic variation and its impact on phenotypic diversity [?]. One of the most popular technique in sorting cells based on their characteristics, is Fluorescent Assisted Cell Sorting. FACS is based on flow cytometry [?].

2.2 Flowcytometry & Fluorescence Microscopy

Flow cytometry is a technique used to analyze the physical and chemical properties of particles, usually cells as they flow through a laser beam ??. Flow cytometry uses the light properties scattered from cells or particles for identification or quantitative measurement of physical properties [?]. Labels, dyes, and stains can be used for multi-parametric analysis including size, shape, and the expression of specific proteins or other cellular components. The principle behind flow cytometry is the use of laser light to excite fluorescently labeled particles, which then emit light at specific wavelengths ??. The emitted light is collected and quantified using detectors, providing information on the characteristics of each cell or particle. Flow cytometry can sort cells based on their physical or chemical characteristics [?]. This is useful for isolating specific cell types from a heterogeneous population of cells, or for preparing cells for further analysis.

Fluorescence microscopy is a technique that uses fluorescence to generate an image of a sample ??. It is often used in biology to study the distribution of molecules in cells and tissues. The basic principle of fluorescence microscopy is that when light of a specific wavelength is shone on a fluorophore, the fluorophore absorbs the light and emits light of a longer wavelength ??. This emitted light is used to generate images of the samples [?].

DineshPalliMasterThesis/Figures/flow-cytometer.png

Figure 1: **Diagrammatic representation of a typical flow cytometer** illustrating the integration of the fluidic, optical, and electronic systems within the cytometer. The fluidic system guides the sample particles in a stream, the optical system illuminates the sample and collects the emitted light, and the electronic system converts the collected light into digital data for further analysis. [**?**]

DineshPalliMasterThesis/Figures/454px-Fluorescence_Microscopy_01.pn

Figure 2: **Working principle of a fluorescence microscope** A fluorescence microscope uses a light source to excite fluorescent molecules in a sample. The emitted light is then filtered and focused to create an image. The image shows a diagram of a fluorescence microscope. The light source is a mercury lamp, which emits a broad spectrum of light. The excitation filter only allows light of a certain wavelength to pass through. The dichroic mirror reflects the excitation light away and allows the fluorescent light to pass through. The emission filter only allows light of a certain wavelength to pass through. The objective lens focuses the fluorescent light onto the detector, which converts it into an image.

2.3 Image-Enabled Cell Sorter

Flow cytometry and fluorescence microscopy are commonly used techniques in biological and biomedical research. While flow cytometry allows for the rapid and high-throughput isolation of cells based on limited number of parameters on low-dimensional parameter space, it lacks the subcellular resolution needed to study processes such as protein trafficking, cellular signaling, or protein mislocalization during disease [?]. On the other hand, fluorescence microscopy provides high-resolution imaging of cellular morphology and protein localization, but it is not capable of quickly isolating cells with specific phenotypes [?]. Combining the spatial resolution of fluorescence microscopy with the sorting capabilities of flow cytometry has the potential to revolutionize experimental approaches by enabling the rapid identification and isolation of cells with specific sub-cellular phenotypes [?].

Daniel Schraivogel et al. presented a system that integrates flow cytometry and microscopy, operates at high speeds suitable for genetic screening and the analysis of short-lived dynamic phenomena, and can be used in non-specialized laboratories [?]. This is referred to as image-enabled cell sorting (ICS) ??. Traditional flow cytometry is able to distinguish between three stages of the cell cycle (G1, G2/mitosis, and S phase), but is unable to differentiate between cells in different stages of mitosis [?]. This was made possible with ICS ??.

Classical image features are the set of spatial image parameters extracted in real-time from each image channel. These included maximum intensity, defined as the highest pixel value, which was used to measure changes in Golgi integrity [?]. Radial moment, representing the mean-square distance of a signal from its centroid, helped distinguish cells with single versus multiple nuclei ??. Eccentricity, calculated from the side scatter image as the ratio of principal component magnitudes, enabled separating round from elongated cells. Size, measured by thresholding pixel counts, quantified nuclei and nucleoli size [?]. Finally, correlation, using the Pearson score between two channels, effectively captured co-localization of proteins like nuclear versus cytoplasmic RelA. Together, these basic spatial and intensity measurements comprised the key classical image features that empowered analysis and sorting of complex phenotypes [?]. The authors demonstrated their utility across diverse applications from profiling protein distributions to isolating mitotic stages ?? ??. These established image features formed a critical starting point in establishing ICS' capabilities.

DineshPalliMasterThesis/Figures/ICS_working.png

Figure 3: Schematic representation of the ICS optical and flow hardware components, showing the excitation and emission beam paths. The excitation beam path utilizes acousto-optic deflectors (AOD) to split a single laser beam into an array of beamlets with different optical frequencies and angles. Another AOD adjusts the optical frequency of a reference beam, which overlaps with the beamlet array. These overlapping beams intersect the flow cell (FC) of a cuvette sorter. The emission beam path involves generating images from digitized signals, including light loss, forward scatter (FSC), side scatter (SSC), and four fluorescent channels. Example images demonstrate the labeling of HeLa cells expressing GalNAcT2-green fluorescent protein (GFP) with CD147 PE-CF594 and DRAQ5 nuclear dye. Grayscale images display FSC, SSC, and light loss. The components used include beam splitter (BS), mirror (M), objective (Obj), deflection plates (DP), obscuration bar (OB), pinhole (P), lens (L), band pass (BP), photomultiplier tube (PMT), and photodiode (PD). Scale bar: 20 mm [**?**].

2.4 High-Dimensional Data from ICS Requires Nonlinear Embedding for clustering

Flow cytometry data typically involves gating strategy - a step where a series of gates are defined to subset, distinguish and refinement of a cellular population of interest. Manual gating typically involves two-dimensional plots for cell-type marker intensity and uses hierarchical gates to identify cell populations [**?**]. The method of using 2D plots for analyzing flow cytometry data is sufficient for experiments with fewer parameters. But the problem arises when experiments have more measured parameters [**?**]. The main issue is that 2D plots do not scale up well. When there are more cell markers, the number of plots grows exponentially. For example, an experiment with 18 markers would need 153 different 2D plots to show all combinations. That many plots become too complex to comprehend and interpret [**?**].

ICS produces high-dimensional single-cell data for which high-dimensional data analysis like dimensionality reduction, feature selection and clustering is needed, contradictory to manual gating in conventional flow cytometry as two-dimensional plots are unable to depict the complex high-dimensional structure of the data generated by the ICS [**?**]. The features from the images, produced by the ICS, requires an embedding that encodes visual features and patterns in a biologically meaningful way that allows machine learning models to understand and analyze images, on which clustering can be done. In machine learning, embedding refers to the process of representing complex, high-dimensional data in a lower-dimensional space, typically as continuous vectors. The goal of embedding is to capture the important characteristics and relationships of the data in a more compact and meaningful form.

The embedding should produce features that capture meaningful variations related to the clustering task. Irrelevant or noisy features can degrade cluster quality. More informative features that better separate the clusters will improve clustering accuracy. Features that overlap across groups are less useful. Features that are intuitive and provide insight into the biological meaning of the clusters aid interpretability of the clusters. Features that are difficult to interpret obfuscate clustering results. Minimizing correlation and redundancy between features can clarify cluster characteristics. Highly correlated features blur cluster boundaries, leading to less distinguition between classes of biological interest. the embedding should extract a concise, informative set of features that are tuned for the specific clustering problem. This enhances separation, cohesion, and interpretability of the resulting clusters. The key is learning an embedding space tailored to our biologically relevant clustering task.

DineshPalliMasterThesis/Figures/DataProcessingPipelineICS.png

Figure 4: **Overview of the ICS low-latency data-processing pipeline, highlighting the steps involved.** Photodetectors generate pulses with high-frequency modulations that encode the image waveform. Fourier analysis is employed to reconstruct the image from the modulated pulse. An image-processing pipeline extracts image features, while a pulse-processing pipeline derives additional features from the pulse. Real-time sort classification electronics utilize these features to classify particles and make sort decisions. The resulting sort decision is used to selectively charge the droplets, as indicated by the dotted gray line in panel A [**?**].

2.5 Unsupervised Machine Learning offers An Automated and Unbiased Approach for Cell Population Profiling

Cytometry is a common tool used by researchers for profiling cell populations in biological samples. This data can offer insights into the cellular composition of healthy tissues and also reveal how various cell subsets alter under disease conditions. A variety of machine-learning techniques have been developed to annotate known cell populations and to identify new cell subsets from the high-dimensional data generated by cytometry experiments.

Unsupervised machine learning techniques sort cells into groups based on their similarities, as determined solely by cytometry data, without the need for external information. Several generic unsupervised methods can be directly applied to cytometry data. Representation learning - transforming the data into an informative lower-dimensional space preserving relevant variation. Common unsupervised approaches like principal component analysis (PCA) or autoencoders can be used for representation learning on cytometry data. The learned representations can then be utilized for downstream tasks like clustering or classification.

For revealing the relationships between classes of biological interest, clustering methods are applicable. These include popular clustering methods like K-means and hierarchical clustering, probability-based methods like Gaussian mixture models, and density-based methods like HDBSCAN which can be applied to learned representations. This facilitate inspecting the intrinsic data structure to identify distinct cell subpopulations. Unsupervised methods offer several benefits when it comes to enumerating cell populations in high-dimensional space. They allow for an unbiased approach, which is unachievable with manual gating methods, and facilitate the automation of cell population identification with minimal human intervention.

2.6 Deep Learning-based Embedding for Unsupervised Cell-Type and State Variation Analysis in High-Dimensional Image Data

In this study, our goal is to develop an embedding capable of extracting distinctive features from cell images that capture meaningful variations in cell type and state which are biologically relevant. This was aimed to be achieved by the application of distance metric learning and deep learning to enable biologically meaningful clustering in high-dimensional space which can be applied to (multiple) datasets from different experiments.

In addressing the problem at hand, a question arises: do classical image features possess the necessary information to effectively cluster the data into clusters of biological relevance? Should the response be affirmative, the subsequent step involves assessing biologically meaningful clustering resulted from the former.

Distance metric learning is a machine learning technique used to learn a distance metric between pairs of data points in a high-dimensional space. The goal of employing distance metric learning in this study is to learn a distance function that accurately reflects the similarity or dissimilarity between the data points, such that similar points are close to each other and dissimilar points are far apart in the learned space. This can be useful for a variety of tasks, such as classification, clustering, and retrieval, where the distance metric plays an important role. Some popular distance metric learning algorithms include large margin nearest neighbor (LMNN), neighborhood component analysis (NCA) and metric learning for kernel regression (MLKR).

Neighborhood Components Analysis (NCA) is a probabilistic method that learns a linear transformation of the feature space to improve classification accuracy of a nearest-neighbor classifier. It aims to make points from the same class closer together while separating different classes [**?**].

Large Margin Nearest Neighbors (LMNN) learns a Mahalanobis distance metric to improve k-nearest neighbor classification. It tries to ensure each point's k-nearest neighbors belong to the same class while enforcing a large margin between different classes [**?**].

Metric Learning for Kernel Regression (MLKR) learns a custom distance metric to improve kernel regression performance. It iteratively predicts each training point using the other points and optimizes the metric to minimize the leave-one-out error. This adapts the feature space so distances match the target regression outputs, pulling together points with similar values while pushing apart dissimilar points. The transformed space focuses on distances that improve generalization, benefiting kernel regression along with other techniques relying on pairwise distances [**?**].

Local Fisher Discriminant Analysis for Supervised Dimensionality Reduction (LFDA) is a supervised version of linear discriminant analysis. It projects data into a lower-dimensional space that maximizes between-class separability while minimizing within-class scatter to improve class discrimination [**?**].

If there is not enough information in classical image features to sort classes into clusters of biological relevance, we aimed to proceed to extract complex features from the images, using neural networks leading to clustering based on the latter.

This was targeted to be achieved in two steps. Initially, we utilize pre-trained models in their respective default configurations to evaluate their efficiency in clustering. Next, we identify the top-performing model among those and proceed to fine-tune it by training the same on our data.

3 Materials & Methods

3.1 Data

There are three datasets viz. *Salmonella* dataset, Phytoplankton strains dataset and Cell-Cycle dataset. The *Salmonella* dataset contains the images of macrophages infested with *Salmonella*. Cells are classified on the basis of *Salmonella*(e) present in the cells - One, two and many **??**. The classified images in the *Salmonella* dataset were 655. In phytoplankton dataset, the labelled images were 4304 classified based on the strains. In cell cycle dataset, the annotated images were 17774 based on different mitotic cell cycle stages **??**. Imaging data from the FACSVulcan (ICS) comes in two forms - as numerical values which are included in the exported flow cytometry standard (FCS) file and the actual cell images which are exported as a Z stack of tag image file format (TIFF). Each imaging parameter has 8 numerical values associated with it which correspond to 8 individual TIFFs in the Z stack.

The images are of dimensions - 8 channels (Z-stack), 104 pixels in width, and have variable height ranging from 40 - 90 pixels.

3.2 Data Format for Classical Image features for Data Analysis, Processing

For the analysis, processing of classical image features, the data output file .csv, was fit into an AnnData object. AnnData.X contained the features as an observations x variations data matrix, the names of the features in AnnData.var as one-direction variables annotation, and the metadata in AnnData.obs as one-direction observations annotations.

Anndata is a data format and a python package for handling annotated data matrices, positioned between pandas and xarray [**?**]. Multidimensional annotations are stored in obsm and varm. Unstructured data which doesn't fit the model, but should stay associated with the dataset are stored in uns. Principal components and the transformed dimensionality-reduced data matrix obtained through PCA can be stored as multi-dimensional annotations of variables and observations respectively (in obsm and varm) [**?**].

The data was processed and analyzed using the tools and methods provided by the ScanPy package, an open-source Python library for single-cell RNA sequencing (scRNA-seq) data analysis [**?**]. ScanPy offers a comprehensive set of functionalities for quality control, preprocessing, visualization, and downstream analysis of scRNA-seq data. It implements various algorithms and approaches for tasks such as dimensionality reduction, DineshPalliMasterThesis/Figures/twosalmonellaesampleimg.png

(a) Sample image showing cell with two *Salmonellae*

DineshPalliMasterThesis/Figures/moresalmonellaesampleimg.png

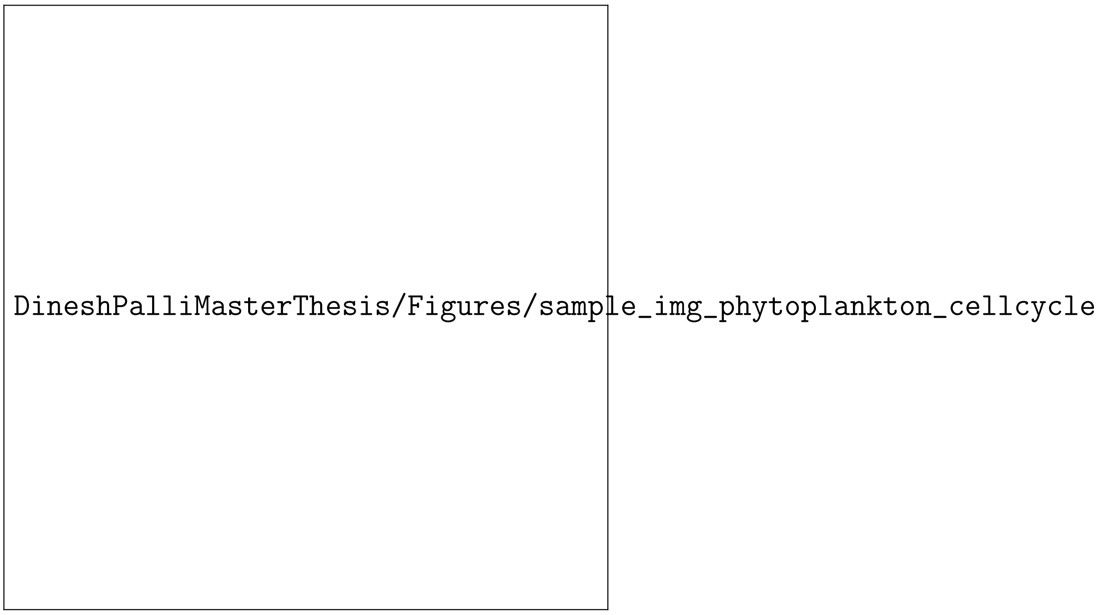
DineshPalliMasterThesis/Figures/sample_img_phytoplankton_cellcycle

Figure 6: Random images from the Plankton and Cell Cycle datasets belonging to different classes, respectively (Top to Bottom), are displayed in this figure. The image width is fixed at 104 pixels, while the height varies across the images.

clustering, trajectory inference, and differential expression analysis [**?**].

3.3 Scaling

The objective of feature scaling is to establish an equal contribution from all features to prevent the undue influence of features with larger values. This becomes particularly crucial when dealing with datasets that encompass features with dissimilar ranges, units of measurement, or magnitudes. The implementation of feature scaling allows for the transformation of the dataset's features onto a more uniform scale. By ensuring a balanced comparison between features, scaling not only improves model convergence but also mitigates the risk of certain features dominating others based purely on their magnitude.

3.4 Distance Metric Learning

Distance metric learning is a machine learning technique that aims to learn a suitable distance metric (or similarity measure) for a given dataset. The goal is to find a distance function that can capture the underlying structure and relationships between data points, allowing for more effective and meaningful comparisons between them.

In many machine learning algorithms and applications, distance metrics play a crucial role. They are used to measure the similarity or dissimilarity between data points, which is essential for tasks such as clustering, classification, retrieval, and anomaly detection.

DineshPalliMasterThesis/Figures/anndata_schema.png

Figure 7: **AnnData schema**, a data structure used in single-cell genomics, combines a data matrix X with annotations of observations (obs), variables (var), and unstructured annotations (uns). The obs component contains information about individual cells or samples, while the var component stores details about variables, typically genes. The uns component provides a flexible container for any additional annotations or metadata. Together, these components within the AnnData format provide a comprehensive and standardized framework for organizing and analyzing single-cell data.

Some of the popular distance metric learning algorithms include, NCA, LMNN, LFDA and MLKR.

NCA (Neighborhood Component Analysis) is a distance metric learning algorithm that aims to learn a linear transformation of the input data such that the K-Nearest Neighbors (K-NN) classification accuracy is maximized.

Formula (Objective Function):

$$\max_W \sum_{i=1}^{N} P(y_i|x_i;W)$$

where W is the transformation matrix, N is the number of samples, x_i is the input data, y_i is the true label of x_i, and $P(y_i|x_i;W)$ is the probability of x_i being correctly classified by its K-NN using the transformed data.

LMNN (Large Margin Nearest Neighbor) is a distance metric learning algorithm that aims to learn a Mahalanobis distance metric such that the K-NN classification accuracy is maximized, while simultaneously pushing each data point closer to its K-NN neighbors with the same class label and away from data points with different class labels.

The objective function for LMNN is as follows:

$$\min_M \sum_{i,j} L_{ij} + \lambda \sum_{i,j,k} \max(0, 1 + d(x_i, x_k)^2 - d(x_i, x_j)^2)$$

where M is the Mahalanobis metric matrix to be learned, L_{ij} is the loss term defined as:

$$L_{ij} = \{\ d\ (x_i, x_j)^2 if y_i = y_j and x_j is one of the K - NN of x_i 0 otherwise$$

and $d(x_i, x_j)$ is the Euclidean distance between data points x_i and x_j.

The additional term $\lambda \sum_{i,j,k} \max(0, 1+d(x_i, x_k)^2 - d(x_i, x_j)^2)$ in the objective function is the large margin term, which encourages the distances between data points with different labels to be larger than a margin. This helps to create a larger separation margin between different classes in the transformed space, leading to better discriminative power and improved performance for the nearest neighbor classification task.

LFDA (Local Fisher Discriminant Analysis) is a distance metric learning algorithm that aims to learn a transformation of the input data such that the local structure of the data is preserved, and the Fisher criterion is maximized for better class separation.

Formula (Objective Function):

$$\max_W tr(W^T S_B W)/tr(W^T S_W W)$$

where W is the transformation matrix, $tr(\cdot)$ denotes the trace of a matrix, S_B is the between-class scatter matrix, and S_W is the within-class scatter matrix.

MLKR (Metric Learning for Kernel Regression) is a distance metric learning algorithm designed specifically for kernel regression tasks. It aims to learn a transformation of the input data such that the kernel regression performance is optimized.

Formula (Objective Function):

$$\min_M \sum_{i=1}^{N} \frac{(y_i - \hat{y}_i)^2}{\sum_{j=1}^{N} k(x_i, x_j)}$$

where M is the metric matrix, N is the number of samples, y_i is the true label of x_i, $\hat{y}_i$ is the predicted label of x_i using kernel regression, and $k(x_i, x_j)$ is the kernel function between data points x_i and x_j.

3.5 Pre-trained Models

We selected the neural network models ResNET50, ScDINO, TransPath out-of-the box to analyse how the models perform in forming meaningful clusters. The best performing network is selected for fine-tuning on the available datasets.

To fit our images to the models, to extract features, the images were padded with zeros to fit the dimensions of the images. The images were processed in a channel-wise manner, with similar calculations applied to each channel.

3.5.1 ResNet50

ResNET50, part of Residual Network (ResNet) family, is a deep convolutional neural network architecture having 50 layers which is used for image recognition and classification tasks [**?**]. ResNet50 was trained on ImageNet dataset - over 14 million images that are organized into more than 20,000 categories. The dimensionality of the feature vector of the ResNet50 model is 2048. This means that the feature vector is a 2048-dimensional vector that represents the image that is input to the model. The feature vector is extracted from the last pooling layer of the model. The network was trained on images of dimensions 224x224x3, RGB [**?**]. The ResNet50 model has 50 layers, and the number of parameters in the model is approximately 25 million. The images were pre-processed before they were used to train the model. This preprocessing included resizing the images to 224x224 pixels, subtracting the mean RGB values from the images, and normalizing the images. ResNet50 is commonly employed as a pre-trained model for transfer learning, where the model is pre-trained on a large dataset, ImageNet, and then fine-tuned for a specific classification task with a smaller dataset [?]. By leveraging the learned features from the pre-trained model, transfer learning often leads to better performance compared to training a model from scratch, particularly when the available training data is limited. The loss function of ResNet50 depends on the specific task it is being used for [?]. However, in the case of image classification, the typical loss function used with ResNet50 is the softmax cross-entropy loss. Given an input image, the ResNet50 network predicts the probabilities of each class using the softmax activation function in the last layer. The softmax function normalizes the output logits into probabilities, representing the network's confidence for each class. The softmax cross-entropy loss measures the discrepancy between the predicted class probabilities and the true labels. It encourages the network to assign higher probabilities to the correct class and lower probabilities to the incorrect classes. The loss function is computed as the average of the cross-entropy losses over all the training samples. Mathematically, the softmax cross-entropy loss for ResNet50 can be defined as:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(p_{ij})$$

Where:

- $\mathcal{L}$ is the loss function,
- N is the number of training samples,
- C is the number of classes,
- y_{ij} is the true label (1 if the sample belongs to class j, 0 otherwise),
- p_{ij} is the predicted probability for class j for the i-th sample [?].

Working of ResNet50: ResNet's primary innovation is the use of residual connections, also known as skip connections or shortcut connections, which enable the network to overcome the vanishing gradient problem and facilitate the training of much deeper networks [?]. The vanishing gradient problem arises when gradients become too small during the backpropagation process, making it difficult for the network to learn and update weights effectively [?].Residual connections work by allowing the output of one layer to be added to the output of a layer further down the network. This means that the network essentially learns the residual function (i.e., the difference between the input and output of a series of layers), rather than the direct mapping between input and output [?]. This approach helps maintain gradient flow through the network, resulting in improved training performance and better convergence [?].

3.5.2 scDINO

Single Cell Distillation of knowledge with No labels is the application of DINO model for automated microscopy-derived fluorescent imaging datasets of single cells [?]. DINO is a self-supervised vision transformer capable of learning representations from unlabelled data using a contrastive loss function [?]. The model consists of a student ViT (Vision Transformer) that learns to predict global features in an image from local patches, supervised by the cross-entropy loss from a momentum teacher ViT's embeddings while doing centering and sharpening to prevent mode collapse [?]. The dimensionality of the DINO vision transformer 16s is 13.3 million. The dimensionality of the feature vector is 1024. The dimensionality of the DINO Vision Transformer 16s is relatively small compared to other vision transformers, such as the ViT-B/32 model, which has 120 million parameters. This is because the DINO Vision Transformer 16s uses a smaller patch size (16x16 pixels) and a smaller number of layers (12 layers). The DINO model is able to automatically learn class-specific features leading to accurate unsupervised object segmentation [?]. The loss function used in the DINO model is a contrastive loss function that is applied to all views at once, and the goal of backpropagation is to minimize the cross-entropy loss between the student and teacher models. The loss function of the DINO model is a cross-entropy loss function applied to all views at once [?].

$$L(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{v=1}^{V}\log\frac{e^{\frac{z_{i,v}^{s}\cdot z_{i,v}^{t}}{\tau}}}{\sum_{j=1}^{N} e^{\frac{z_{i,v}^{s}\cdot z_{j,v}^{t}}{\tau}}}$$

Where:

- $L(\theta)$: Loss function
- N: Number of samples
- V: Number of views
- $z_{i,v}^{s}$: Student embedding for the i-th sample and v-th view
- $z_{i,v}^{t}$: Teacher embedding for the i-th sample and v-th view
- τ: Temperature scaling factor

The goal of the DINO model is to minimize this loss function, which encourages the student model to learn representations that are similar to the teacher model. The temperature scaling factor τ controls the sharpness of the probability distribution over the embeddings [?].

DineshPalliMasterThesis/Figures/scDINO_working.png

Figure 8: **Workflow for single-cell phenotyping using DINO**-based self-supervised vision transformers (ss-ViTs). The upper part illustrates the DINO-based self-supervised learning approach, which includes a student and teacher ViT. The lower part showcases the ViT-based image embedding and feature representations, with emphasis on the classification (CLS) token (is a special token that is used in the Transformer architecture. CLS token is added to the beginning of the input sequence, and it is used to represent the overall meaning of the sequence.), utilized for single-cell clustering [**?**].

scDINO was adapted to accept variable numbers of input channels and averaged the RGB-specific embedding weights, giving each channel equal weight [?]. scDINO was adjusted to give out 384 feature vectors. scDINO was trained on 89,564 cells in the immune single-cell training dataset, while using 224x224x5 as input data. All the pixel intensities were rescaled to range between 0 and 1 for each channel [?].

Training of scDINO To enable training of the scDINO vision transformer on our image datasets, the input images were standardized by padding the height with zeros to create uniform dimensions across all images. The full dataset was utilized including non-annotated images, taking advantage of scDINO's self-supervised learning capability. Each image was treated as a single local crop with no splitting The model was trained with a dataset split fraction of 0.25, seed set to 40, a learning rate of 0.0005, and the AdamW optimizer, set to run for 200 epochs, checkpointing every 2 epochs.

3.5.3 TransPath

TransPath is a transformer based unsupervised contrastive learning model for histopathological image classification. TransPath is a hybrid model that combines a convolutional neural network (CNN) with a swin-transformer architecture [?]. The CNN is used to extract local features from the images, while the transformer is used to learn long-range dependencies between the features [?]. This allows TransPath to learn both local and global features from the images, which is important for histopathological image classification. TransPath was trained on a dataset of histopathological images that were labeled with the type of cancer. Contrastive learning involves learning to distinguish between similar and dissimilar images [?]. This allows TransPath to learn to identify subtle differences in histopathological images, which is important for cancer classification. The model was trained on histopathological images of dimensions 224 x 224 x 3 and the feature vector is 768. TransPath has been shown to achieve state-of-the-art performance on the task of histopathological image classification. It has been shown to be able to classify different types of cancer with high accuracy [?]. The loss function is the similar to the DINO - contrastive loss with cross-entropy:

$$L = -\sum_{i=1}^{N} \log \left(\frac{e^{\frac{q_i \cdot k_i}{\tau}}}{\sum_{j=1}^{N} e^{\frac{q_i \cdot k_j}{\tau}}} \right) - \sum_{i=1}^{N} \log \left(\frac{e^{\frac{k_i \cdot q_i}{\tau}}}{\sum_{j=1}^{N} e^{\frac{k_i \cdot q_j}{\tau}}} \right) + \lambda \cdot \mathrm{CELoss}(q, y)$$

In this formula:

- N represents the total number of instances or samples.
- q_i and k_i are the query and key embeddings, respectively, for instance i.

- τ denotes the temperature hyperparameter that controls the similarity scaling. $CELoss(q, y)$ refers to the cross-entropy loss between the predicted logits q and the true labels y.
- λ represents the weighting factor for the cross-entropy loss.

This loss function encourages the query and key embeddings to form positive pairs (similar instances) while discouraging negative pairs (dissimilar instances) based on their similarities measured using the dot product. Additionally, the cross-entropy loss term ensures the model's predictions align with the true labels.

3.6 Cluster Purity Evaluates the Performance of Biologically Meaningful Cluster Formation

Cluster purity was developed to quantitatively measure the cluster separation. Purity provides a simple and transparent evaluation of clustering quality. It is computed by assigning each cluster to the class that is most frequent within the cluster and then measuring the accuracy of this assignment by counting the number of correctly assigned documents [**?**]. Purity values range between 0 and 1, where higher values indicate better clustering quality [**?**]. In our case, to calculate Purity, the following steps are followed. Firstly, a matrix is created, which involves counting the number of documents classified as each class within each cluster. To compute purity , each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned documents and dividing by N (Total number of data points). This is represented with the formula:

$$purity(\Omega, C) = \frac{1}{N} \sum_{k} \max_{j} |\omega_k \cap c_j|$$

where $\Omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$ is the set of clusters and $C = \{c_1, c_2, \ldots, c_J\}$ is the set of classes. We interpret ω_k as the set of documents in ω_k and c_j as the set of documents in c_j in the above equation.

4 Results

4.1 Classical image features without machine learning are inadequate for forming biologically meaningful clusters

After the preprocessing of data - *Salmonella*, Phytoplankton and cell cycle, by normalization and scaling, PCA (principle component analysis, [?]) and UMAP (Uniform Manifold Approximation and Projection [?]) were performed using ScanPy functions **??**, [?]. The classical images features were extracted and plotted **??**.

It can be inferred from the plots that the classical image features without the application of machine learning, do not clearly form clusters of biological relevance.

4.2 Scaling provides modest amplification but insufficient separation of biologically meaningful clusters

Scaling of data is the process of adjusting the values of the features in a dataset so that they are all on the same scale. Scaling was performed to reduce the affect of magnitude while clustering. Scaling of data is important for machine learning because it helps to improve the accuracy and efficiency of the machine learning algorithm. When the features are not scaled, the algorithm may give more weight to features with larger values, even if those features are not as important as features with smaller values.

However, upon the scaling and transformation of the feature space, a clear demarcation between the strains becomes evident, as illustrated in **??**. The transformation process involves the scaling of features after subsetting the dataset, a step that amplifies the subtle differences in features across various classes. The basis for these lies in the weighted Euclidean distances between data points in the feature space (**??**). This amplification, facilitates the separation of the clusters. By magnifying the differences, we are able to discern clear boundaries between clusters that might otherwise appear to overlap or blend into each other. The clusters, 269, 1164, 1333, 3015, 3541 are used to illustrate the effect of transformation of feature space and scaling on cluster separation.

Despite the transformation of feature space, some clusters could not be clearly distinguished from each other as shown in **??** - depicts the clusters of one salmonella and two salmonellae not being able to be separated from each other, yet after scaling and the transformation of the feature space. Moreover, similar trend is observed when trying to separate the phytoplankton strains - 1512, 3003 and 4281.

DineshPalliMasterThesis/Figures/salmonella_clusters_before_DML.png

(a) Leiden clustering of *Salmonella* data. This visualization represents the distinct clusters identified in the Salmonella dataset following the pre-processing stage.

DineshPalliMasterThesis/Figures/plankton_clusters_before_DML.png

DineshPalliMasterThesis/Figures/phytoplanktonclusterseparationaft

Figure 10: The image illustrates the distinct separation of phytoplankton strain clusters 269, 1164, 1333, 3015, 3541, a result of the applied scaling procedure. This clear demarcation contrasts with the initial state where these clusters were not clearly separated, as depicted in **??**. The scaling procedure has thus significantly enhanced the clarity and differentiation of these phytoplankton strain clusters.

DineshPalliMasterThesis/Figures/maxintensitymembranestain_fitc_scaledandunscaled_plan

(a) Histogram comparing the distribution of the maximum intensity of membrane stain FITC in two different subsets of data.

DineshPalliMasterThesis/Figures/fitc-h_sc

(b) Histogram that compares the intensity of the stain FITC-H in two different data.

Figure 11: **Each figure depicts how the differences across the clusters are amplified upon scaling, in comparison to the unscaled.** Each figure represents a classical feature selected - MaxIntensity_membrane_stain_FITC and FITC-H among the selected plankton strains - 4281, 3003, 1512, referred in the images as strains of interest.

DineshPalliMasterThesis/Figures/clustersmixedafterscalingsalmonella

(a) Depicts the *Salmonella* clusters of cells containing one and two *Salmonellae*, demonstrating their lack of clear separation despite undergoing scaling and transformation.

DineshPalliMasterThesis/Figures/clustersmixedafterscalingplankton.

(b) Illustrates the plankton clusters of strains 1512, 3003, and 4281, emphasizing their continued overlap even after scaling and transformation.

Figure 12: **Illustration of the two datasets' clusters post scaling and feature space transformation, highlighting the lack of clear separation between them.**

4.3 Insufficient Separation of Biologically Meaningful Clusters Using Distance Metric Learning on Classical Image Features

To ascertain the feature space that facilitates the formation (and separation) of biologically relevant clusters, we applied distance metric learning. Distance metric learning was performed on the classical image features. The distance metric learning algorithms used include: Neighborhood Component Analysis (NCA), Large Margin Nearest Neighbors (LMNN), Metric Learning for Kernel Regression (MLKR) and Local Fischer Discriminant Analysis for Supervised Dimensionality Reduction (LFDA). Distance metric learning was performed after the dimensionality reduction was executed on the dataset using PCA [**?**].

Out of the aforementioned distance metric learning algorithms, NCA was the best-performing algorithm based on the results from cluster purity calculation - Table **??**.

Table 1: Performance of different distance metric learning algorithms across multiple datasets. The performance was measured using the cluster purity metric mentioned in **??**.

Distance Metric	**Salmonella**	**Phytoplankton**	**Cell Cycle**
No Distance Metric Learning	0.568433	0.673014	0.85258
NCA	**0.72579**	**0.76682**	**0.99339**
LMNN	0.63115	0.67778	0.99325
MLKR	0.69221	0.76703	0.99182
LFDA	0.56843	0.72847	0.87293

The cluster purity increased noticably. Some clusters couldn't be separated despite the application of distance metric learning as observed in **??**.

To assess the performance of metric learning on a merged dataset, we combined three individual datasets and conducted evaluations using cluster purity. This was done to check if there is a feature space that suits all three datasets and aids in relevant clustering. However, the cluster purity decreased when compared to the application of Distance Metric Learning (DML) on each dataset separately. This observation is clearly depicted in **??** and **??**.

Table 2: Performance of distance metric learning algorithm NCA across dataset formed by merging the three individual datasets. The performance was measured using the cluster purity metric mentioned in **??**.

Distance Metric	**Salmonella**	**Phytoplankton**	**Cell Cycle**
NCA	0.65196	0.79570	0.93944

Due to the same, it can be noticed that the information in / from the classical image

DineshPalliMasterThesis/Figures/salmonellaclustering_afternca.png

(a) Leiden clustering of *Salmonella* data after the application of distance metric learning algorithm NCA.

DineshPalliMasterThesis/Figures/planktonclustering_afternca.png

DineshPalliMasterThesis/Figures/mergeddataset_salmonella.png

(a) Leiden clustering of *Salmonella* data from the merged dataset after the application of distance metric learning algorithm NCA.

DineshPalliMasterThesis/Figures/mergeddataset_plankton_newlabel.png features is not enough to form clusters of biological relevance.

4.4 Pre-trained Deep Learning Models show inferior separation compared to classical image features

As distance metric learning on classical image features was not able to clearly form the clusters of biological relevance, we proceeded to extract features from the images, using deep neural networks.

The previously mentioned deep-learning models out-of-the-box were used to extract the features from the images. The performance of the models is mentioned in Table **??** and Figure **??**.

The results show three different models performing best in the three datasets. scDINO showed a significant improvement before and after NCA relative to other two networks. Due to the same we proceeded to train the scDINO network, to evaluate the increment in the performance of the model, if any.

Table 3: Performance of different pre-trained deep learning networks used to extract features from the images. The performance is measured using the cluster purity metric mentioned in **??**.

Deep Learning Network	***Salmonella***		**Phytoplankton**		**Cell Cycle**	
	Before NCA	**After NCA**	**Before NCA**	**After NCA**	**Before NCA**	**After NCA**
ResNet50	0.47567	0.47321	0.70098	**0.70348**	0.57469	0.61025
ScDINO	0.52953	**0.57076**	0.59414	0.69794	0.54569	0.54789
TransPath	0.53083	0.53133	0.60485	0.63519	0.65229	**0.71893**
scDINO (trained)	0.46939	**0.62408**	0.61468	**0.67523**	**0.55161**	0.54701

4.5 Performance of scDINO Marginally Improved After Training

The inferior performance of pre-trained deep learning models compared to classical features implies these generic networks have not learned feature representations tailored to our clustering task. To address this, we trained the model, scDINO which showed improvement out-of-the-box, on the full corpus of labeled and unlabelled images across the phytoplankton, *Salmonella*, and cell cycle datasets. The training loss decreased over epochs Figure: **??**, indicating the model was learning from the diverse image data. After training, model weights were extracted and used to extract features from the images.

DineshPalliMasterThesis/Figures/outoftheboxscDINOpostNCA_salmonella.png

(a) Leiden clustering of *Salmonella* data after the extraction of complex features from images using pre-trained network - scDINO and post NCA.

DineshPalliMasterThesis/Figures/outoftheboxscDINOpostNCA_plankton_nooverlap.png

However, clustering performance based on Neighborhood Components Analysis of the scDINO embeddings was still inferior compared to classical features, as shown in **??** and **??**. In the case of the cell cycle dataset, it dropped marginally. This shows that the model learned biologically irrelevant features along with the relevant ones.

Table 4: Table showing the performance of scDINO deep learning network after training, used to extract features from the images. The performance is measured using the cluster purity metric mentioned in **??**.

Deep Learning Network	**Salmonella Dataset**	**Salmonella Dataset**	**Plankton Dataset**	**Plankton Dataset**	**Cell Cycle Dataset**	**Cell Cycle Dataset**
	Before NCA	**After NCA**	**Before NCA**	**After NCA**	**Before NCA**	**After NCA**
scDINO	0.46939	**0.62408**	0.61468	**0.67523**	**0.55161**	0.54701

DineshPalliMasterThesis/Figures/scDINOlossduringtrainingsalmonella

(a) Training loss plotted against the epoch during scDINO training on *Salmonella* data.

DineshPalliMasterThesis/Figures/scDINOlossduringtrainingplankton.p

(b) Loss vs epochs during scDINO training on Plankton data.

DineshPalliMasterThesis/Figures/scDINOlossduringtrainingcellcycle.

DineshPalliMasterThesis/Figures/scDINOaftertrainingafterNCA_salmonella.png

(a) Leiden clustering of *Salmonella* data after the extraction of complex features from images after training scDINO on the *Salmonella* dataset and post NCA.

DineshPalliMasterThesis/Figures/scDINOaftertrainingafterNCA_plankton.png

5 Discussion

Advances in microscopy and imaging technologies have enabled the collection of high-dimensional data across diverse biological contexts. In this regard, ICS produces high-dimensional data that requires an embedding on which biologically relevant clustering can be done. As image volumes grow, manual analysis becomes infeasible, requiring more automated computational solutions to identify relevant features and patterns. In this study, we aimed to develop an embedding that can disentangle complex image features relevant to cell-type and state variation from high-dimensional ICS data. Our goal was to enable biologically meaningful clustering of single cells, which could be applied across diverse datasets from multiple experiments. We evaluated whether classical image features contain enough information for effective clustering, or if more complex embeddings learned via deep neural networks on images are required. We used DML and neural networks for the same. We tested dimensionality reduction and clustering methods on classical features, applied distance metric learning to transform the feature space, and analyzed pre-trained deep learning models for their embedding performance. We also trained the deep learning networks on our data, to evaluate their performance.

The results presented indicate that relying solely on classical image features is insufficient for forming biologically meaningful clusters across the diverse datasets - *Salmonella*, phytoplankton, and cell cycle images. While these features provide a preliminary separation between some states, they lack the complexity and discriminative power to finely differentiate between subtle phenotypic differences that denote variations in cell type or condition **??**. Some reasons why this occurred might be capacity of hand-engineered features to not capture the intricate textural, shape, and morphological characteristics that distinguish cell phenotypes of the cells. Moreover, the classical image features are designed generically and not specific to the individual datasets. But the cluster purity of the cell cycle data was higher than other datasets. Light scattering profiles have been closely correlated to the nuclear morphology and changes during cell cycle [**?** **?**]. Nuances in the staining patterns, spatial protein distributions, and textural motifs within cells, which are specific to dataset (cells and their morphology) are unlikely to be encoded by classical features.

The histograms comparing feature distributions before and after scaling (**??** and **??**) demonstrate that scaling can help amplify the differences between clusters to some extent. This scaling has proved that the clusters can be separated, in the case of some phytoplankton strains. However, this amplification is modest, and several strains still exhibit significant overlap even after scaling. For instance, in the phytoplankton data, strains 1512, 3003 and 4281 remain blended together post-scaling. Similarly, the *Salmonella* clusters denoting one vs two bacteria per cell show little separation following scaling. But for inseparable clusters, the differences may be more subtle and not well captured by the classical features even after scaling.

In relation to the plankton biology, the composition of the culture medium is another critical factor to consider when isolating phytoplankton strains. Despite the potential effects that different media may have on the success of phytoplankton isolation and growth, few studies have systematically evaluated the impacts of various media formulations. Harrison and Berges (2005) quoted McLachlan (1973): "Numerous enriched and synthetic media have been formulated, which together with generally trivial modifications, almost equal the number of investigators" [**?**]. Cells infected with *Salmonella* have one, two or more *Salmonella* inside. Cells with more *Salmonella* show high intensity than the cells with one or two *Salmonella* inside them. When the *Salmonella* is multiplying inside the cell, there is change in the pixel area over time, which is useful for the clustering, but the size of *Salmonella* is relatively small in comparison to the cell and the ICS captures the features of the entire cell [**?**]. As the *Salmonella* multiply the intensity of the fluorescence increase in the cell, and this can also be noticed as the cluster with more *Salmonella* is relatively more isolated from clusters with one and two *Salmonellae* inside them **??**. In the case of cells with one or two *Salmonellae*, during the bacterial reproduction, the fluorescent signals might be close to each other during separation. This can result in an elongated spot of fluorescence, rather than two different spots resulted from completion of multiplication of the bacterium. Capturing more details by the ICS, in the case of *Salmonella* data, including the non-biologically relevant data, from the cells encapsulating the bacteria, which makes it difficult to infer the biologically relevant features. This can also be noticed from the complex features extracted from the images.

This implies that while scaling transforms the range of feature values to highlight differences, the features themselves do not fully encapsulate the intrinsic characteristics that demarcate distinct cell types in this data. More complex features are needed to delineate boundaries between biologically meaningful groups.

Applying dimensionality reduction techniques like PCA and UMAP visualization directly on the classical features provides some preliminary clusters, but many are indistinguishable (Figure 1). For example, in the *Salmonella* data, clusters representing one bacterium per cell versus two bacteria per cell are not differentiated clearly. This indicates the classical features do not adequately capture morphological differences between these phenotypes.

Using distance metric learning algorithms like NCA on the features improves purity (cluster accuracy) to some degree but still does not lead to a complete separation of relevant biological groups (Table 1, Figure 5). Some cell type clusters remain mixed, like those seen earlier. There are two reasons that might have caused this, one, the cell cycle consists of fixed progression of stages that cells pass through sequentially. Each stage has distinct phenotype that is reflected in the cells. But the transitions between adjacent phases are gradual. Cells entering a new phase retain remnants of the prior one. This means proximate cell cycle stages are intrinsically more similar than distant stages. Distance Metric Learning (DML) can take advantage of this inherent structure by pulling closer points near each other in the progression while pushing apart distant phases. Essentially, DML is able to embed information about the sequential ordering of the cycle into the transformed feature space. DML techniques like NCA learn linear transformations of the feature space [**?**]. The phenotypic differences across cell cycle phases follow observable patterns in size, shape etc. These morphological changes can likely be captured through linear feature transformations. But for the *Salmonella* and phytoplankton data, discriminative patterns are likely more complex and non-linear. In case of *Salmonella*, the bacteria might be small and the differences subtle to capture biologically relevant information. The physiology of the plankton changes from strain to strain and some strains could possibly be similar to each other with subtle feature differences. Simple linear transformations are insufficient to disentangle these complex feature relationships. Datasets with gradual, ambiguous differences require learning highly non-linear mappings from features to phenotypes. These datasets (salmonella and plankton), because of their variations demand complex non-linear feature dependencies that linear DML cannot represent. In contrast, the more structured cell cycle changes are amenable to linear separations, explaining DML's success. This orderly alignment and natural similarities between adjoining phases might likely explains why DML was effective for cell cycle separation. NCA performed better than other DML algorithms, because as a probabilistic model, it is less prone to overfitting compared to margin-based techniques like LMNN, especially with limited training data. NCA is designed to optimize leave-one-out classification accuracy directly, whereas other methods optimize different proxy objectives like relative distances or margins between classes. This may have made NCA better suited for our goal.

To check if there is feature space that suits all the three datasets properly and aids in clustering, we merged the three datasets to apply DML on the same. The dataset formed by merging the three datasets, did not perform any better and it is justified as there are many features to learn from and some of the features from different datasets are overlapping and might conflict with each other which hinders the formation of biologically relevant clusters **??** and **??**.

The inferior performance of pre-trained deep learning models compared to classical features (**??**) implies these generic networks have not learned feature representations tailored to this specific clustering task when used out-of-the-box. Their internal representations may not fully disentangle the properties critical for phenotype-based clustering. Models like ResNet50, scDINO and TransPath were pre-trained on natural image datasets like ImageNet or histopathology data. They have not seen enough relevant cell images to learn features optimized for differentiating cell types and states based on their morphology. scDINO was trained on single cell data, which is similar to our datasets **??**. ScDINO learns both local patterns within image patches and global context from the full image via the vision transformer architecture. This multi-scale representation may have aided clustering. Also, all the three networks used performed inferiorly on the *Salmonella* data. This can also be due to the small size of the labelled data, which is 655 images from which the out-of-the-box network was used to extract features, typically considered small for a neural network.

Taken together, these results strongly indicate that while classical features and scaling provide a partial separation between some states, they lack the complexity and dimensionality needed to finely discriminate between subtle biological variations that correspond to changes in cell type or condition. Pre-trained deep learning models also do not readily learn feature spaces conducive for this goal without explicit training on relevant data. This strongly motivates the need for learning specialized embeddings tailored to the specific task of clustering. The generic embeddings derived from pre-trained models are unlikely to effectively disentangle the intricate combination of morphological features that denotes variations in cell type and state without dedicated training on such data.

A deep neural network trained in an end-to-end fashion on diverse datasets spanning different cell types, states, and conditions may potentially learn a rich, high-dimensional embedding space capable of separating clusters belonging to distinct phenotypes. For the same, we trained the scDINO network on our data to evaluate the performance of the network after training. For the same, we first trained the network on datasets separately to evaluate the individual increment in performance. The trained models performed relatively better when compared with pre-trained network and performed inadequately compared with DML. This could possibly be due to the capturing all the features of the images both biologically relevant and non-relevant. To address the same, the model could be adjusted with weighted loss function - a loss function that assigns higher weights to biologically relevant features or classes. Hard negative mining to select more informative negative examples during training could also be used. This focuses the model on its weaknesses. This can be especially true in relation to *Salmonella* dataset as the details that are biologically relevant are the ones in bacteria, which is actually a minor part of the entire image of the cell. But the network tends to learn all the features of the cell too. But with plankton, the performance improved on average, relatively as the strains look different from each other morphologically. Some strains which tend to look similar are not clearly separated when the clusters are formed as can be seen in **??**.

Cluster purity was utilized as metric to evaluate the quality of clusters generated by the clustering algorithms in our work. Cluster purity provides a measure of how well clusters map to single classes, by calculating the percentage of samples assigned to their dominant class in each cluster. Additionally, the fixed Leiden resolution used for clustering can impact purity measurements. We did not evaluate the robustness of purity across different resolutions, which could provide more insights into the clustering performance. Leiden starts from a random initial partitioning. Running Leiden multiple times with different seeds may produce variation in purity across runs.

While cluster purity gives an intuitive sense of cluster accuracy, it is seldom perfect. Purity can be biased by imbalanced dataset sizes, and does not penalize having redundant or excessive clusters. Further, purity only considers the dominant class in each cluster, ignoring potential mixing of other classes. For the same reason, we recommend future studies involving intrinsic metrics (do not require ground truth labels) viz., Silhouette Coefficient [**?**].

Silhouette Coefficient

$$s = \frac{b - a}{\max(a, b)} \tag{1}$$

Where a is the mean intra-cluster distance and b is the nearest cluster distance for each sample. The score is averaged over all samples [**?**].

Further improvements to the disentangling of complex features extracted from the images can be done, with slightly changing the loss function. Combining the contrastive loss with cluster separation loss would improve the separation of clusters and improve the accuracy. Using a contrastive loss like Normalized Temperature-scaled Cross Entropy Loss (NT-Xent) over cross entropy could better capture similarities between images of the same phenotype [**?**]. NT-Xent is more robust to noise and is a metric learning approach. It operates on both pairs or triplets of samples which suits to Salmonella dataset. Additionally, it can be combined with cluster separation loss, since it penalises the cluster centroids to push entire clusters away from each other [**?**]. This combination helps the network learn fine similarities in local samples (with contrastive loss) and higher level inter-cluster relations (with cluster separation loss). We have not tried DINO2, the latest update to the DINO, proven to perform better than DINOv1 [**?**], with improved attention mechanism, which could also be trained and tried on our data [**?**].

$$\mathcal{L}hybrid = \mathcal{L}NT - Xent + \lambda\mathcal{L}_{separation} \tag{2}$$

Where:

$$\mathcal{L}NT - Xent = -\frac{1}{N}\sum i = 1^N \log \frac{\exp(x_i \cdot x_j/\tau)}{\sum_{k\neq i}\exp(x_i \cdot x_k/\tau)} \tag{3}$$

is the NT-Xent loss between positive pairs (i, j),
and the cluster separation loss is defined as:

$$\mathcal{L}separation = \sum c_1 \neq c_2 \max(0, m - D(c_1, c_2)) \tag{4}$$

which applies a margin m between the centroids c_1 and c_2 of different clusters, with D measuring the distance.

The hyperparameter λ balances the two loss components. This hybrid formulation allows jointly modeling sample similarity with NT-Xent and cluster separation.

In addition to adjusting the loss function, training regime can be fine-tuned for optimal performance including, oversampling labeled images during training to focus learning on phenotypic differences. Unlabelled images can be used for pre-training then fine-tuned on labeled data. Curriculum learning approach can be tried, by starting with easy separations then progressively make clustering harder as training improves [**?**]. Two-stage training could be attempted, where the model is first trained on all data, then fine-tuned on labeled subsets for phenotypically relevant features. Providing class/phenotype labels during training to directly optimizes the embedding for clustering rather than only reconstruction.

In this study, we showed that while classical features and generic deep learning embeddings provided preliminary cluster separations, neither fully captured the nuanced morphological patterns distinguishing cell types and states. DML on classical image features was the best performer in forming biologically relevant clustering. Further optimizations to the loss function, training procedure, and network architecture could better focus learning on biologically-meaningful traits. Overall, this study highlighted both the challenges and prospects of clustering high-dimensional data using distance metric learning and automated phenotypic characterization from images using deep neural networks. With problem-specific tuning, we are optimistic deep learning can reveal new phenotypic relationships within complex cellular populations in an unbiased manner.

References

[1] Gianluca Pegoraro and Tom Misteli. High-throughput imaging for the discovery of cellular mechanisms of disease. *Trends Genet.*, 33(9):604–615, September 2017.

[2] Albert J. Mach and Dino Di Carlo. Continuous scalable blood filtration device using inertial microfluidics. *Biotechnology and Bioengineering*, 107(2):302–311, 2010.

[3] Katy Vandereyken, Alejandro Sifrim, Bernard Thienpont, and Thierry Voet. Methods and applications for single-cell and spatial multi-omics. *Nature Reviews Genetics*, 2023.

[4] Xiaofeng Liao, Melissa Makris, and Xin M. Luo. Fluorescence-activated cell sorting for purification of plasmacytoid dendritic cells from the mouse bone marrow. *Journal of Visualized Experiments*, (117), 2016.

[5] Katherine M. Mckinnon. Flow cytometry: An overview. *Current Protocols in Immunology*, 120(1), 2018.

[6] Fundamentals of Flow Cytometry — AAT Bioquest — aatbio.com. `https://www.aatbio.com/resources/assaywise/2019-8-1/fundamentals-of-flow-cytometry`. [Accessed 20-Jul-2023].

[7] Michael J. Sanderson, Ian Smith, Ian Parker, and Martin D. Bootman. Fluorescence microscopy. *Cold Spring Harbor Protocols*, 2014(10):pdb.top071795, 2014.

[8] Andrea Cossarizza, Hyun-Dong Chang, Andreas Radbruch, Andreas Acs, Dieter Adam, Sabine Adam-Klages, William W. Agace, Nima Aghaeepour, Mübeccel Akdis, Matthieu Allez, and et al. Guidelines for the use of flow cytometry and cell sorting in immunological studies (second edition). *European Journal of Immunology*, 49(10):1457–1973, 2019.

[9] Virginia Espina, Julia D Wulfkuhle, Valerie S Calvert, Amy VanMeter, Weidong Zhou, George Coukos, David H Geho, Emanuel F Petricoin, 3rd, and Lance A Liotta. Laser-capture microdissection. *Nat. Protoc.*, 1(2):586–603, 2006.

[10] Daniel Schraivogel, Terra M. Kuhn, Benedikt Rauscher, Marta Rodríguez-Martínez, Malte Paulsen, Keegan Owsley, Aaron Middlebrook, Christian Tischer, Beáta Ramasz, Diana Ordoñez-Rueda, Martina Dees, Sara Cuylen-Haering, Eric Diebold, and Lars M. Steinmetz. High-speed fluorescence image&#x2013;enabled cell sorting. *Science*, 375(6578):315–320, 2022.

[11] Zicheng Hu, Sanchita Bhattacharya, and Atul J. Butte. Application of machine learning for cytometry data. *Frontiers in Immunology*, 12, 2022.

[12] Isaac Virshup, Sergei Rybakov, Fabian J. Theis, Philipp Angerer, and F. Alexander Wolf. anndata: Annotated data. 2021.

[13] F. Alexander Wolf, Philipp Angerer, and Fabian J. Theis. Scanpy: large-scale single-cell gene expression data analysis. *Genome Biology*, 19(1), 2018.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[15] Ramon Pfaendler, Jacob Hanimann, Sohyon Lee, and Berend Snijder. Self-supervised vision transformers accurately decode cellular state heterogeneity. *bioRxiv*, 2023.

[16] Xiyue Wang, Sen Yang, Jun Zhang, Minghui Wang, Jing Zhang, Wei Yang, Junzhou Huang, and Xiao Han. Transformer-based unsupervised contrastive learning for histopathological image classification. *Medical Image Analysis*, 81:102559, 2022.

[17] Prabhakar Raghavan Christopher D. Manning and Hinrich Schütze. *Introduction to Information Retrieval.* Cambridge University Press, Cambridge University Press Assessment Shaftesbury Road Cambridge CB2 8EA, 2008.

[18] Andrzej Maćkiewicz and Waldemar Ratajczak. Principal components analysis (pca). *Computers Geosciences*, 19(3):303–342, 1993.

[19] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020.

[20] A BRUNSTING and PF MULLANEY. Differential light-scattering from spherical mammalian-cells. *BIOPHYSICAL JOURNAL*, 14(6):439–453, 1974.

[21] Mitchell C. Benson, Denise C. Mcdougal, and Donald S. Coffey. The application of perpendicular and forward light scatter to assess nuclear and cellular morphology. *Cytometry*, 5(5):515–522, 1984.

[22] Paul Harrison. Harrison, p. j. and j.a. berges. 2005. marine culture media. in: Algal culturing techniques. r. andersen (ed.) academic press, ny, pp. 21-33. 02 2005.

[23] Anna Fàbrega and Jordi Vila. Salmonella enterica serovar typhimurium skills to succeed in the host: Virulence and regulation. *Clinical Microbiology Reviews*, 26(2):308–341, 2013.

[24] John Chiotellis William de Vazelhes. Neighbourhood Components Analysis. `https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.NeighborhoodComponentsAnalysis.html`.

[25] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.

[26] M Meila and A Jain. Normalized information distance as a measure of cluster similarity. *Journal of Machine Learning Research*, 11:2219–2237, 2003.

[27] P J Rousseeuw et al. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20(1):53–65, 1987.

[28] Gregory Koch, Richard S Zemel, and Ruslan R Salakhutdinov. Siamese neural networks for one-shot learning. *arXiv preprint arXiv:1503.02531*, 2015.

[29] Yufei Xu, Zhuang Liu, and Dong Zhou. Variational bayesian clustering with cluster separation loss. In *Proceedings of the 35th International Conference on Machine Learning*, pages 5586–5595. PMLR, 2018.

[30] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021.

[31] Zihang Chen, Xingyu Wang, Han Zhang, Han Zhang, Guiguang Cui, Hao Sun, Yihuan Yang, Xiangyu Zhang, and Yang Lu. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.01021*, 2023.

[32] Yoshua Bengio, J Louradour, R Collobert, and J Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 113–120, 2009.

6 Acknowledgement

First and foremost, I would like to express my deepest gratitude to my principal investigator, Prof. Dr Fabian Theis, and my internal supervisor, PD Dr Michael Menden, for their guidance and for examining my thesis. Their expertise and insights have been invaluable to my academic journey.

I am particularly indebted to my supervisor, Louis Kümmerlee, who has provided constant support throughout this process. His patience, knowledge, and being a Swiss Army Knife in answering all my queries and doubts has been instrumental in the completion of this thesis. He was always there for me when needed, and for that, I am truly grateful.

I would also like to extend my thanks to my lab members, especially Felix Fischer and Giovanni Palla, among others. Their support and camaraderie have made this journey a happy, rewarding and enriching experience.

On a personal note, I wish to express my heartfelt gratitude to my family: my parents, Madhava Reddy Palli and Samrajya Lakshmi Kalluru, and my brother, Sai Swaroop Palli. They have been my pillars of strength, providing unwavering support and encouragement throughout my academic journey and through life.

I am profoundly indebted to my friends, Atharv Arora, Shrey Parikh, Krishna Sai, Divya and Bhavna Menon. During the times when I found myself in the deepest valleys of my journey, they were there, ready to jump-start my faltering spirit with their unwavering belief in me. Their friendship has been more than just a source of camaraderie; it has been a beacon of hope in my darkest hours. The emotional support they provided has been my anchor, grounding me when the storms of doubt threatened to carry me away. Their presence in my life is a gift I will forever cherish.

In the grand tradition of alphabetical order, I extend my heartfelt thanks to Aarushi Davesar, Sabeel Un Naeem, and Siddhi Pawar. You have been my personal cheerleading squad, my stress relief hotline, and my partners in laughter. Thank you for being the 'safety valves' for my occasional steam-letting sessions and for standing by me when life decided to play limbo. Your unwavering positivity and constant encouragement have been the lighthouse guiding me through the foggy nights of this academic journey. For that, I owe you a debt of gratitude... and perhaps a round of ice cream!

In conclusion, I recognize that this achievement would not have been possible without the support, guidance, and encouragement of each individual mentioned above, and for this, I am eternally grateful.

Last page of the master's thesis (including statement of originality)

This form must be filled out and is to be included as the last page in the printed copy of your master's thesis for the Examination Committee.

Dieses Formblatt ist ausgefüllt als letzte Seite in das beim Prüfungsausschuss abzugebende Exemplar der Masterarbeit einzubinden.

Last name, first name: PALLI, DINESH REDDY

Signature of the (internal) supervisor: PD Dr. MICHAEL MENDEN

Starting date of the master's thesis: 13.02.2023

Submission date of the master's thesis: 31.07.2023

Master's thesis statement of originality

I hereby confirm that I have written the accompanying thesis myself, without contributions from any sources other than those cited in the text and acknowledgements. This applies also to all graphics, drawings, maps and images included in the thesis.

Erklärung zur Masterarbeit

Hiermit versichere ich, dass die vorliegende Arbeit von mir selbstständig verfasst wurde und dass keine anderen als die angegebenen Quellen und Hilfsmittel benutzt wurden. Diese Erklärung erstreckt sich auch auf in der Arbeit enthaltene Graphiken, Zeichnungen, Kartenskizzen und bildliche Darstellungen.

München, 31.07.2023

Place, date
Ort, Datum

Signature
Unterschrift