

Robust and Efficient Representation Learning for High-Throughput Imaging Flow Cytometry Data

by

Andrey Kurenkov

Submitted to the Department of Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Data Science

at the

NAZARBAYEV UNIVERSITY

June 2026

© Nazarbayev University 2026. All rights reserved.

Author
Department of Computer Science
April 11, 2026

Certified by
Adnan Yazici
Professor of Computer Science
Thesis Supervisor

Certified by
Natalie Barteneva
Professor of Biology
Thesis Supervisor

Accepted by
Elizabeth Arkhangelsky
Dean, School of Engineering and Digital Sciences

Robust and Efficient Representation Learning for High-Throughput Imaging Flow Cytometry Data

by

Andrey Kurenkov

Submitted to the Department of Computer Science
on April 11, 2026, in partial fulfillment of the
requirements for the degree of
Master of Science in Data Science

Abstract

Imaging flow cytometry (IFC) produces millions of single-cell images per experiment, offering rich morphological and spatial information beyond conventional flow cytometry. However, the rapid growth of dataset size has created two significant bottlenecks: the lack of accessible, instrument-agnostic analysis software, and the fragility of deep learning models to domain shifts caused by instrumental variability or sample preparation.

To address these gaps, we present Eidocell, an open-source framework for interactive analysis and robust representation learning in IFC. Eidocell integrates self-supervised vision transformers (ViT-S/16) to extract biologically meaningful features without manual annotation. To enable consistent analysis across changing experimental conditions, we introduce an unsupervised Test-Time Adaptation (TTA) algorithm that realigns feature statistics to the target domain in real-time. Furthermore, we implement fuzzy clustering strategies to quantify continuous morphological transitions that are ill-defined by discrete logic.

We evaluated the framework across four diverse datasets: microalgae, immune synapse, red blood cells, and bovine sperm. Unsupervised clustering consistently recovered biologically meaningful structures, achieving accuracy scores up to 98% on biological benchmarks. Our TTA adaptation algorithm demonstrated improved performance over baseline and prominent inference performance.

Thesis Supervisor: Adnan Yazici
Title: Professor of Computer Science

Thesis Supervisor: Natalie Barteneva
Title: Professor of Biology

Acknowledgments

I would like to express my gratitude to my supervisors, Professor Adnan Yazici and Professor Natalie Barteneva, for their guidance and support throughout this work. I am thankful to the members of Professor Barteneva's laboratory for their help with data acquisition, biological expertise, and many productive discussions. I also thank the colleagues who contributed to the case studies and provided feedback on earlier versions of this manuscript.

Contents

1	Introduction	15
1.1	Single-Cell Analysis	15
1.2	Analysis Bottleneck	16
1.2.1	Workflow Gap	17
1.2.2	Robustness and Domain Shift	17
1.3	Research Objectives	18
1.4	Primary Contributions	19
1.5	Thesis Organization	21
2	Background and Related Work	23
2.1	Imaging Flow Cytometry	23
2.1.1	Data Characteristics and Spatial Features	25
2.2	Bio-Image Analysis Software	26
2.2.1	Vendor-Proprietary Software	26
2.2.2	Open-Source Platforms	27
2.2.3	Machine Learning-Centric Tools	28
2.3	Feature Engineering and Representation Learning	29
2.3.1	Classical Feature Engineering	29
2.3.2	Convolutional Neural Networks	30
2.3.3	Vision Transformers and Self-Supervised Learning	31
2.3.4	Self-Supervised Foundation Models for Cytometry	31
2.3.5	Transfer Learning and Fine-Tuning Strategies	32
2.3.6	The Generalization Gap	33

2.4	Robustness and Domain Adaptation	34
2.4.1	Domain Shift in Imaging	34
2.4.2	Test-Time Adaptation	35
2.4.3	Unsupervised Discovery and Fuzzy Clustering	40
2.4.4	Research Gap	41
3	Methodology: The Eidocell Framework and Algorithms	43
3.1	System Design and Architecture	43
3.2	The Computational Pipeline	45
3.2.1	Data Ingestion and Normalization	45
3.2.2	The Segmentation Module	46
3.2.3	Feature Extraction	47
3.3	Robust Feature Adaptation under Domain Shift	48
3.3.1	Problem Formulation	48
3.3.2	Statistical Feature Alignment via Normalization Adaptation .	49
3.3.3	Semantic Boundary Alignment via Entropy Minimization . . .	51
3.3.4	The Hybrid TTA Algorithm	53
3.4	The Analysis Engine	54
3.4.1	Dimensionality Reduction	54
3.4.2	Discovery and Fuzzy Clustering	55
3.4.3	Clustering Evaluation Metrics	57
3.4.4	Assisted Classification	59
4	Data Resources	61
4.1	Biological Case Studies	61
4.1.1	Case Study I: Bovine Sperm Morphology	61
4.1.2	Case Study II: Post-COVID-19 Red Blood Cells	62
4.1.3	Case Study III: Microalgae Database	63
4.1.4	Case Study IV: Immunological Synapse	63
4.2	Technical Benchmark Datasets	63

5	Experiments and Results	65
5.1	Phase I: Workflow Validation and Backbone Selection	65
5.1.1	Backbone Selection and Performance Analysis	65
5.1.2	Clustering Performance	66
5.1.3	AI Interpretability and Morphological Continuity	67
5.1.4	End-to-End Workflow: RBC Case Study	70
5.1.5	Baseline Performance on Corrupted Data	71
5.2	Phase II: Algorithms for Robust Representation Learning	73
5.2.1	Validation on Technical Benchmarks	73
5.2.2	Biological Domain Adaptation	73
5.2.3	Ablation Study and Computational Efficiency	74
6	Discussion	75
6.1	Reducing the Workflow Gap	75
6.2	Test-Time Adaptation Impact	76
6.3	Theoretical and Empirical Support for Hybrid Adaptation	77
6.4	Biological Implications of Fuzzy Analysis	78
6.5	Limitations	79
7	Conclusion	81
7.1	Summary	81
7.2	Future Work	82

List of Figures

3-1	Graphical user interface of Eidocell	44
3-2	The <i>Eidocell</i> computational pipeline architecture	46
4-1	Sample images and their corresponding masks from the biological case studies	62
5-1	Quantitative analysis of morphological ambiguity using Fuzzy Clustering on the Bovine Sperm dataset	68
5-2	Morphological Confusion Web	69
5-3	Practical workflow for the RBC case study in Eidocell	72

List of Tables

2.1	Advanced spatial features unique to IFC	26
2.2	Comparison of TTA methods	40
4.1	Summary of the four primary biological case studies used to validate the framework	63
4.2	Summary of the technical benchmark datasets	64
5.1	K-NN classification accuracy of various feature extractor backbones .	66
5.2	Comparative clustering performance between Soft K-means and X- means algorithms	67
5.3	Baseline Accuracy (%) of the static DINOv3 model under distribution shift.	72
5.4	Comparison of TTA strategies on technical benchmarks under severe distribution shift (Level 5 Corruption)	73
5.5	K-NN classification accuracy across biological case studies using Test- Time Adaptation methods	73
5.6	Ablation study of the Hybrid TTA components on the Sperm Cells dataset	74
6.1	Comparison of software platforms for IFC analysis	76

Chapter 1

Introduction

This chapter introduces the biological and technical context of the thesis. We review the transition from bulk cellular analysis to high-throughput single-cell imaging, focusing on the capabilities of Imaging Flow Cytometry (IFC). We identify the primary bottlenecks in current IFC data processing, and discuss limitations of existing software ecosystems and model degradation due to domain shift. Lastly, we define the primary contributions of this work in addressing these workflow and algorithmic challenges.

1.1 Single-Cell Analysis

Cellular heterogeneity is a fundamental characteristic of biological systems, significantly influencing the phenotype and function of organisms in both health and disease states [1]. Western blotting or bulk RNA sequencing are traditional methods of analyzing cellular characteristics that often fall short of capturing this heterogeneity due to their averaging approaches. By pooling genetic or proteomic material from thousands of cells, these bulk techniques distort the behavior of rare subpopulations, transition states, and stochastic variations between individual cells. To resolve this, modern biology has shifted toward methods capable of quantifying cellular parameters at single-cell resolution, approximating the set of observable characteristics of an organism or cell: its morphology, biochemical properties, and behavior [2, 3] .

Optical imaging is the default approach for capturing spatial information with high fidelity. Unlike molecular assays that require cell destruction (e.g., single-cell sequencing), imaging implicitly represents the morphological characteristics of the cell in its native or near-native state. However, there is a trade-off between statistical power and morphological resolution. Conventional flow cytometry (FCM) has high sample throughput of tens of thousands of cells per second, but provides limited morphological data in the form of a few integrated intensity values (scatter and fluorescence). Conversely, high-content microscopy provides detailed spatial resolution but is typically limited in throughput and speed, often requiring time-consuming sample preparation and scanning [4].

IFC is a hybrid technology that combines the statistical power of flow cytometry with the spatial resolution of microscopy, and allows researchers to record high-resolution brightfield, darkfield, and fluorescence images of cells traveling in a fluid stream at rates of up to 10,000 events per second [5]. Systems such as the Amnis ImageStream[®] X Mk II (Amnis-Cyte, USA) use Time-Delay Integration (TDI) CCD sensors to capture blur-free images of fast-moving cells, generating datasets that are high-dimensional, spatially correlated, and statistically robust.

The hardware breakthrough allowed studying of the samples where morphological context is necessary, e.g. cell-cell interactions (e.g., immune synapses) [6], characterizing nuclear shape changes in cancer, or monitoring phytoplankton diversity in environmental samples [7]. Unlike traditional cytometry, a single IFC file captures images across multiple spectral channels for every cell, and may contain hundreds of thousands of single-cell images. In the context of large experiments containing hundreds of files, data volume grows exponentially [5, 8].

1.2 Analysis Bottleneck

Modern research and analysis of IFC data presents two major challenges: a workflow gap caused by a limited pool of software tools, and an algorithmic gap caused by the limitations of conventional computational methods in large-scale biological data.

1.2.1 Workflow Gap

To analyze biological datasets, researchers rely on computational tools ranging from custom scripts to specialized software. However, existing bio-image analysis software is fragmented [4]. On one end of the spectrum are vendor-provided solutions, such as the IDEAS software for the Amnis ImageStream. Such platforms support feature extraction, gating, and bioassays, but are proprietary, closed-source, and often rely on classical image processing techniques (e.g., masking and feature engineering) that struggle to capture complex phenotypes. Furthermore, they create data silos, restricting analysis to specific hardware and preventing the integration of modern, open-source deep learning models or cross-device analysis.

Existing open-source solutions like ImageJ/Fiji or CellProfiler [9] are usually applied in microscopy. These tools were not originally designed for the flow cytometry paradigm (i.e., continuous streams of single objects requiring population gating), and adapting them to IFC often requires constructing multi-step pipelines and significant custom scripting [10].

Writing custom analysis scripts in Python, R, MATLAB and other languages is the most flexible approach, which offers limitless experimental capabilities and access to state-of-the-art methods and libraries like PyTorch or TensorFlow. However, this requires significant field knowledge and programming expertise that is not yet standard in the biological sciences. Consequently, high-dimensional data is often under-utilized as researchers revert to manual inspection or simplified gating strategies that ignore the rich morphological information contained in the images. There remains a need for an open-source tool specifically tailored to the IFC workflow: one that combines a user-friendly graphical interface with the power of modern, interactive machine learning.

1.2.2 Robustness and Domain Shift

Even when software tools are available, the underlying algorithms face data issues native to many scientific and industrial applications, like domain shift and model

deterioration. During the past decade Deep Learning (DL) and Convolutional Neural Networks (CNNs) have outperformed traditional feature engineering in classifying cell phenotypes [11]. However, supervised DL models are data-hungry and suffer from domain shift, requiring large sets of annotated examples to train. In biomedical imaging, ground-truth labels are expensive to obtain.

In IFC, domain shift can arise from biological variations (e.g., different types of cells such as fibroblasts, endothelial cells, spermatozooids, or comparing fresh vs. cryopreserved or fixed with formaldehyde samples) or technical variations (e.g., changes in instrument focus, illumination, or staining protocols). Our previous work on bovine sperm analysis [12] demonstrated that a model trained on fresh samples may suffer a significant drop in accuracy (10–15%) when applied to frozen sperm samples, despite the visual similarity of the cells. Similarly, optical artifacts such as defocus or rotations can worsen classification performance if the model has not been explicitly trained to handle them [13]. If an experimental condition changes or a device is recalibrated, model retraining and fine-tuning are often required to maintain the performance, which hinders analysis workflows in complex case studies. The issue exacerbates further with biology’s inherent complexity in morphological descriptors.

1.3 Research Objectives

The primary motivation of this thesis is to improve analytical potential of IFC by bridging the gap between high-throughput data acquisition and robust, interpretable machine learning. Because biological morphology exists on a continuous spectrum, and instrumental conditions inevitably drift during long-running experiments, traditional feature engineering and supervised learning become limited. The specific objectives are threefold:

1. **Development of an open-source analysis framework:** To overcome the limitations of proprietary software and the entry barrier of custom scripting, we aim to develop *Eidocell*, a standalone desktop application for IFC analysis. The objective is to provide a modular environment that integrates state-of-the-art

deep learning models (such as Vision Transformers and CNNs) with interactive visualization and clustering tools. The framework must be instrument-agnostic, capable of processing data from various platforms (e.g., ImageStream, Flow-Cam) to allow cross-platform and cross-domain analysis.

2. **Validation on case studies:** We aim to validate the proposed framework by applying it to distinct, real-world biological datasets exhibiting various data quality challenges native to IFC. This includes the morphological classification of bovine sperm (characterized by high class imbalance and subtle structural defects) [12], the analysis of red blood cells in post-COVID-19 patients (characterized by focus artifacts and cellular deformation) [13], the profiling of immune synapses (multi-object classification, visual obstruction), and the analysis of microalgae (high class count and imbalance) [7]. These studies serve to quantitatively evaluate proposed methods and conventional supervised learning in high-throughput analysis.
3. **Improving robustness via unsupervised adaptation:** Static deep learning models suffer from performance degradation in applications with frequently changing data distributions. Our primary technical objective is to develop algorithmic methods based on Test-Time Adaptation (TTA). We aim to implement and evaluate strategies that allow a pre-trained feature extractor to adapt its weights to incoming unlabeled data streams. By aligning feature representations between source and target domains without human supervision, we aim to improve classification performance over baseline and ensure consistent clustering across heterogeneous experiments.

1.4 Primary Contributions

In addressing the workflow and algorithmic bottlenecks of IFC this thesis proposes an open-source analysis platform augmented with unsupervised adaptation algorithms and probabilistic discovery tools, making the following primary contributions to the

fields of bio-image analysis and applied machine learning:

- **The *Eidocell* framework:** We designed and developed a novel, open-source desktop application that bridges the gap between high-throughput cytometric data and modern deep learning. The software is instrument-agnostic and provides a no-code, human-in-the-loop environment for interactive clustering, gating, and semi-supervised labeling of millions of single-cell images.
- **A Hybrid TTA algorithm:** We proposed and implemented an unsupervised, online adaptation strategy to mitigate instrumental and biological domain shifts. By dynamically combining Statistical Feature Alignment (updating normalization running statistics) with Semantic Boundary Alignment (entropy minimization), we demonstrated a near 10% recovery in classification accuracy on cryopreserved biological samples without requiring target-domain labels.
- **Integration of self-supervised vision transformers (ViTs):** We established that frozen, self-supervised ViT backbones (specifically DINOv3 [14]) consistently outperform traditional supervised CNNs in extracting robust morphological semantics from IFC data, particularly in the presence of random cellular rotations and positional shifts.
- **Quantification of morphological continuums via fuzzy logic:** Recognizing that cellular deformation and maturation are continuous processes, we moved beyond standard discrete classification by integrating Soft (Fuzzy) Clustering on top of deep neural embeddings. This probabilistic approach successfully captures, isolates, and quantifies transitional biological states (such as progressive cell damage or intermediate maturation stages) that traditional hard-clustering algorithms artificially obscure.
- **Empirical validation:** We curated and evaluated our framework across four diverse, real-world biological datasets (Bovine Sperm, Post-COVID RBCs, Freshwater Algae, and Immune Synapses) alongside standardized synthetic benchmarks (CIFAR-C, Tiny-ImageNet-C).

1.5 Thesis Organization

The remainder of this thesis is structured as follows:

- **Chapter 2 (Background and Related Work)** establishes the theoretical foundation of the thesis. It reviews the hardware mechanics of Imaging Flow Cytometry, evaluates existing bio-image analysis software, and traces the evolution of feature extraction. It also introduces the algorithmic concepts of domain shift, test-time adaptation, and clustering types.
- **Chapter 3 (Methodology)** details the technical design of the proposed solutions. It outlines the *Eidocell* software architecture and computational pipeline. We also provide the mathematical formulations for our Hybrid Test-Time Adaptation (TTA) algorithm and the integration of fuzzy clustering for interpretable unsupervised discovery.
- **Chapter 4 (Data Resources)** describes the empirical datasets used to validate the framework. It details the acquisition, biological context, and specific technical challenges of the four real-world biological case studies, alongside the synthetic benchmark datasets used for controlled algorithmic testing.
- **Chapter 5 (Experiments and Results)** presents evaluation results of the proposed methods. Phase I validates the static capabilities of *Eidocell* and explores the interpretability of fuzzy clustering. Phase II quantitatively evaluates the Hybrid TTA algorithm’s ability to recover accuracy under severe domain shifts.
- **Chapter 6 (Discussion)** interprets the experimental findings, explicitly contextualizing how *Eidocell* resolves the current software fragmentation in cytometric analysis. We also reflect on the biological implications of modeling morphological continuums and acknowledge the technical limitations of the current implementation.

- **Chapter 7 (Conclusion)** summarizes the main technical and biological contributions of the thesis, and outlines directions for future research and software development.

Chapter 2

Background and Related Work

In this chapter we overview relevant theoretical concepts and contextualize contributions within the broader field of bio-image analysis. We first review the hardware mechanics of IFC and evaluate the strengths and limitations of existing software solutions. We then trace the evolution of feature extraction from classical engineering to modern vision transformers. Lastly, we establish the algorithmic context for the challenges addressed in this work, specifically exploring the mechanics of domain shift, TTA strategies, and fuzzy clustering.

2.1 Imaging Flow Cytometry

IFC represents a technological convergence of FCM and fluorescence microscopy. Traditional FCM analyzes tens of thousands of cells per second (forward scatter, side scatter, and fluorescence intensity). Conversely, microscopy offers rich morphological detail but is limited by low throughput and limited statistics. IFC joins the two and offers the statistical robustness of conventional cytometry with the spatial resolution of the latter [5, 2].

The core innovation behind IFC is the decoupling of image acquisition from shutter speed. In standard microscopy, capturing a fast-moving object requires an extremely fast shutter to prevent motion blur, which significantly reduces the number of photons collected and thus the sensitivity of the instrument. IFC systems, such as the Am-

nis ImageStream[®] X Mk II, overcome this via TDI Charge-Coupled Device (CCD) sensors. In a TDI-CCD system, the accumulation of charge in the pixel wells is synchronized with the velocity of the cell flowing through the flow cell. As the cell moves across the laser interrogation point, the charge packet is shifted down the sensor array at the same rate. This allows the camera to integrate signal over a much longer effective exposure time without inducing motion blur, resulting in high-sensitivity fluorescence detection even at flow speeds exceeding thousands events/sec.

The data produced differs significantly from both FCM and microscopy: an individual event in IFC is recorded as a stack of up to 12 synchronized images, including Brightfield, Darkfield/SSC, and fluorescence channels. Because these channels are acquired from the same cell through a common optical path, they are inherently co-registered at the pixel level. This configuration allows quantification of the relative spatial distribution of markers—for example, the localization of a transcription factor within the nucleus versus the cytoplasm, or the co-localization of two fluorescent labels within a common subcellular compartment—directly from the image stack [10].

Although IFC instruments such as the Amnis ImageStream include on-board autofocus that continuously tracks the fluid core, individual cells still exhibit small, per-event Z-axis displacements within the hydrodynamic stream. Because these displacements occur at the level of the cell rather than the stream, they are not fully corrected by the autofocus system and produce residual defocusing artifacts. Such artifacts can structurally mimic biological phenotypes by blurring features like the central pallor of biconcave erythrocytes [13].

The random orientation of cells in suspension introduces rotational invariance as a significant confounder. While hydrodynamic forces tend to align elongated cells with the flow, spherical or irregular cells, such as leukocytes or cancer cells, adopt random rotations. This variability poses a challenge for CNNs unless specifically addressed through data augmentation or the implementation of rotation-invariant architectures.

2.1.1 Data Characteristics and Spatial Features

A distinguishing attribute of IFC is the complexity of the per-event data representation. IFC captures the full spatial distribution of fluorescence across every pixel of every channel. Such data requires spectral compensation: the correction for bleed-through between overlapping fluorochrome emission profiles, which must be executed at the individual pixel level rather than applied as a uniform scalar matrix [8]. Pixel-by-pixel deconvolution ensures that highly localized morphological signals, such as a punctate spot of fluorescence marking a DNA repair focus, are verifiable and not artifacts of spectral overlap.

Practically, this data is stored in multi-dimensional arrays. Conventional (non-imaging) cytometers serialize per-event scatter and fluorescence measurements into list-mode files following the Flow Cytometry Standard (FCS) format, where each row corresponds to a single event and each column to a scalar feature. IFC inherits this event-oriented list-mode structure but extends each entry from a vector of scalars to a full multi-channel image stack. Proprietary systems output Raw Image Files (`.rif`) containing uncalibrated pixel arrays, which are processed into Compensated Image Files (`.cif`) after spatial offset correction and spectral deconvolution. For deep learning pipelines, data is typically serialized into HDF5 datasets, which allow batched loading of multichannel image stacks (e.g., NumPy arrays of `uint16` integers) directly into GPU memory, bypassing the I/O bottleneck of processing individual TIFF files.

The data is initially analyzed through gradient-based focus metrics, such as root-mean-square (RMS) contrast across the brightfield channel. It quantifies the sharpness of cell boundaries and allows intuitive filtering of out-of-focus events. Additionally, area and aspect ratio measurements on the brightfield mask serve to filter debris (anomalously small area), doublets and clumps (disproportionately large area with irregular aspect ratios), and speed beads from the analytical population [5]. Beyond the standard morphometric features shared with microscopy (area, perimeter, circularity), IFC enables a suite of other advanced spatial fluorescence metrics that are unique to the modality [10]:

Table 2.1: Advanced spatial features unique to IFC that use the co-registered multi-channel image stack.

Feature Category	Description and Biological Application
Bright Detail Intensity	Measures intensity of punctate spots within a mask after local background subtraction. Used for enumerating internalized nanoparticles, viral inclusions, and autophagy.
Similarity / Cross-Corr.	Pearson correlation between pixel intensities of two fluorescent channels across the same spatial mask. Quantifies nuclear translocation (e.g., NF- κ B) and co-localization in immunological synapses.
Internalization Ratio	Ratio of fluorescence intensities within the cytoplasm versus the entire cell. Assesses phagocytosis, antibody-drug conjugate uptake, and etc.
Spot Distance	Physical distance (μm) between centroids of multiple distinct bright spots within a single cell. Essential for FISH spot counting.

These spatial metrics transform qualitative visual observations into quantitative data, allowing population-level statistical analysis of sub-cellular phenomena that are invisible to non-imaging flow cytometers. However, extracting hundreds of features across up to 12 spectral channels per cell creates a high-dimensional feature space where manual 2D gating and traditional feature engineering becomes a limiting factor.

2.2 Bio-Image Analysis Software

Collection of IFC databases has outpaced the development of accessible software tools. Existing software can be broadly categorized into vendor-provided solutions, open-source bio-image platforms, and specialized algorithmic toolkits.

2.2.1 Vendor-Proprietary Software

Imaging instruments are typically bundled with proprietary software, such as IDEAS[®] (Amnis/Luminex). These solutions support necessary data handling and multiple analysis features: visualizing recorded samples, gating, plotting extracted features,

and biological assays. They rely heavily on masking techniques, which define a region of interest based on a threshold and allow us to calculate features like *Intensity* or *Circularity*.

Despite manuals and documentations accompanying the software, these platforms act as black boxes. The algorithms used for feature extraction are often fixed, preventing researchers from implementing novel descriptors and computational methods. Furthermore, they operate as data silos; an analysis pipeline built in IDEAS cannot easily be applied to data from a FlowCam (Yokogawa Fluid Imaging Inc., USA) or a custom microfluidic device. By default, they lack native support for modern deep learning architectures or restrict their access behind paid versions. Existing versions include basic machine learning modules (e.g., the *Machine Learning Module* in IDEAS), where they usually rely on older paradigms like Linear Discriminant Analysis (LDA) and lack the flexibility of Python-based ecosystems.

2.2.2 Open-Source Platforms

The bio-image analysis community developed open-source solutions, primarily ImageJ/Fiji and CellProfiler [9].

ImageJ/Fiji is the standard for biological image analysis. Its primary feature is the support for public plugins that can be refactored to work for a wide range of tasks. However, ImageJ was designed for the *single image* or *stack* paradigm common in microscopy. Adapting it to the *list-mode* data structure of flow cytometry (where an experiment consists of a list of N discrete events) requires scripting and creates high memory overhead.

CellProfiler established the *Analysis Pipeline* concept [9], allowing users to sequence discrete modules, such as `IdentifyPrimaryObjects` followed by `MeasureObjectSize`, for efficient processing of large image datasets. This framework is highly effective for high-content screening (HCS) on plates, but encounters significant limitations within the context of IFC.

One constraint is the lack of interactivity in its design. CellProfiler is optimized for a workflow where users construct a pipeline, execute it, and evaluate the post-

processing results. Exploratory research, such as the identification of novel microalgal morphotypes or rare immune phenotypes, demands a human-in-the-loop approach. This requires the ability to interactively cluster and refine populations, which serves as a core design idea of the proposed Eidocell software.

Furthermore, significant input/output friction exists when integrating modern computational tools. Integrating deep learning models from popular frameworks into a CellProfiler pipeline requires specialized plugins like *CellProfiler Analyst* or custom scripting, complicating the adoption of advanced analytical methods [3].

2.2.3 Machine Learning-Centric Tools

Existing tools often rely on ML methods for their core features. *CellProfiler Analyst* allows users to train Random Forest classifiers on the features extracted by CellProfiler [15]. Ilastik utilizes the Random Forest algorithm for interactive pixel-level segmentation [16]. More recently, *EasyFlow* offers a cross-platform graphical interface requiring zero coding, allowing locally hosted cytometry analysis built on Matplotlib and Qt [17]. While EasyFlow lowers the entry barrier for basic analysis, it does not integrate deep learning feature extraction or online adaptation.

The listed tools use ML algorithms for analysis, but they rely primarily on shallow learning algorithms (Random Forests, SVMs) applied to hand-crafted features. As discussed in the following section, this approach is limited by the quality of the pre-defined features and fails to capture context and raw pixel information. Tools like DeepCytometry or Cytomine continue to integrate DL methods, but often require complex server-side setups or command-line interfaces, limiting their adoption in wet-lab environments.

2.3 Feature Engineering and Representation Learning

The analysis of biological images has historically relied on the extraction of manually engineered features. The features are mathematical descriptors explicitly defined by human experts to capture visual properties, i.e. object circularity, area, perimeter, etc. This approach is interpretable, but limited by the expert’s ability to define what is important a priori. DL has shifted this paradigm towards representation learning, where the optimal features for a given task are learned from the data itself.

2.3.1 Classical Feature Engineering

Standard workflows supported by IDEAS and CellProfiler begin with segmentation computing binary masks of the objects, followed by the calculation of predefined metrics. These metrics are generally categorized into three groups, the first being morphometric features. These are descriptors of shape and size, such as Area, Aspect Ratio, and Circularity. They provide robust measurements but might lack the sensitivity required to distinguish between cell types with similar gross morphologies. For instance, such metrics are frequently insufficient for differentiating T-cells from B-cells or distinguishing active from inactive sperm based solely on cellular outlines [18].

Beyond morphology, texture features and signal distribution metrics are used to quantify internal cellular characteristics. Haralick textures (entropy, contrast, and correlation) or Zernike moments, quantify the spatial arrangement of pixel intensities to capture intracellular interactions, such as the granularity observed in neutrophils. However, these metrics exhibit high sensitivity to image quality, noise, and focus artifacts [9]. Signal distribution metrics describe the localization of fluorescence and are used in combination with previous metrics.

These metrics are highly efficient for well-defined assays characterized by strong phenotypes and are interpretable. However, classical feature engineering might miss

non-linear patterns inherent in complex biological states, such as the early onset of apoptosis or stem cell differentiation. Furthermore, the process of feature selection is frequently labor-intensive and susceptible to researcher bias.

2.3.2 Convolutional Neural Networks

CNNs overcome the limitations of feature engineering by learning a hierarchy of features directly from raw pixel data. In a CNN, lower layers learn to detect edges and simple textures, while deeper layers aggregate these into complex, abstract representations of an object’s structure [19].

The application of CNNs to IFC data has demonstrated superior performance over classical methods. In a landmark study, Eulenberg et al. (2017) and Lippeveld et al. (2019) demonstrated that the **DeepFlow** architecture’s customized ResNet and VGG variants could accurately classify distinct human white blood cell lineages (eosinophils, neutrophils, and lymphocytes) using only stain-free brightfield and dark-field imagery [20][11]. Similarly, our own work on bovine sperm classification showed that models like **ConvNeXt** and **MobileNet** could distinguish between subtle morphological defects (e.g., *abnormal midpiece*, *distal cytoplasmic droplets*) with over 91% accuracy, significantly outperforming classical shape descriptors [12].

Beyond classification, deep learning has enabled **virtual fluorescent labeling**. The **DeepIFC** framework, built upon an Inception U-Net architecture, accepts label-free brightfield and darkfield images and predicts the corresponding fluorescent spatial distribution for specific biomolecular markers [21]. The network learns imperceptible morphological signatures associated with specific antigens, generating synthetic fluorescence images that can substitute physical staining. In the domain of clinical hematology, Doan et al. (2020) demonstrated CNN-based monitoring of acute myeloid leukemia via stain-free IFC [22] and objective assessment of red blood cell storage lesions using deep learning to track morphological degradation over time [23]. For environmental and public health applications, the **MCellNet** architecture achieved high-accuracy detection of waterborne pathogens (*Cryptosporidium* and *Giardia*) in IFC data [24]. Another direction of DL application is integration with the **Quant-**

tative Phase Imaging (QPI). The **RT-HAD** (Real-Time Holographic Aggregate Detection) framework combines holographic reconstruction with YOLOv8 detection to identify rare platelet aggregates at throughputs exceeding 10,000 events per second with sub-10 millisecond inference times [25, 26].

Collectively, these advances demonstrate that CNNs have matured from proof-of-concept classifiers to production-grade tools spanning classification, virtual staining, pathogen detection, and real-time diagnostics. However, all supervised architectures share a common limitation: they require labeled training data for each specific task and instrument configuration, motivating the shift toward self-supervised and foundation model approaches.

2.3.3 Vision Transformers and Self-Supervised Learning

Recent advances in computer vision have introduced the **ViT**. Unlike CNNs, which process images via local receptive fields (convolutions), Transformers utilize a self-attention mechanism that allows the model to weigh the importance of different parts of an image globally. Furthermore, the training approach has shifted towards **Self-Supervised Learning (SSL)**. Methods like **DINO** (Self-distillation with NO labels) allow models to learn rich feature representations from unlabeled data by solving pre-text tasks (e.g., predicting missing parts of an image or enforcing invariance between different augmented views of the same cell).

2.3.4 Self-Supervised Foundation Models for Cytometry

Training a robust CNN requires large quantities of expert-labeled single-cell images. In clinical cytometry, acquiring ground-truth labels is slow, expensive, and subjectively biased, as peer-to-peer agreement on morphological categories is often imperfect. Over the last two years, this bottleneck has been addressed by the aforementioned SSL methods and pretraining strategies.

SSL algorithms such as BYOL (Bootstrap Your Own Latent), SimCLR, and the highly influential **DINOv2** remove the requirement for human-annotated class labels

or significantly limit it to the scale of a specific task. Instead of learning to map an image to a predefined category, SSL trains a dual student-teacher network to minimize the distance between latent representations of augmented views of the same objects, while maximizing the distance from different objects. Through this self-distillation loss, the network learns rich, intrinsic embeddings that capture universal morphological structure rather than task-specific patterns.

SSL methods have been successfully applied to the field of biological imaging in the past years, microscopy in particular. **DinoBloom** is a ViT-based model family scaling up to 1.13 billion parameters, pre-trained via DINOv2 on a cohort of 13 diverse single-cell datasets from peripheral blood and bone marrow [27]. DinoBloom produces generalized 768-dimensional cellular embeddings that vastly outperform standard ImageNet-trained models. It excels at zero-shot and few-shot learning, linear probing for leukocyte subtyping, and weakly supervised acute myeloid leukemia (AML) classification without task-specific fine-tuning. **RedDino** represents an RBC-tailored adaptation of the DINOv2 framework, pre-trained on a meticulously curated dataset of 1.25 million diverse red blood cell images across varying hardware modalities [28]. RedDino achieves state-of-the-art performance in highly nuanced RBC shape classification tasks (e.g., spherocytosis, sickle cell disease) via simple k-nearest neighbor assessment on the learned embeddings, outperforming supervised baselines trained on the same data. The works demonstrate that researchers can extract generalized embeddings from pre-trained models and feed them into simple linear classifiers or clustering algorithms to achieve state-of-the-art accuracy with a minimized computational overhead. However, these models produce static embeddings—the features are frozen at inference time and do not adapt to distribution shifts in new experimental conditions.

2.3.5 Transfer Learning and Fine-Tuning Strategies

The standard paradigm for improving the model at a specific task and battling domain shift is fine-tuning: a model pre-trained on a large generic dataset (typically ImageNet) is fine-tuned on the target task using labeled data from the domain of

interest. Several popular strategies exist along a spectrum of adaptation level:

Linear probing freezes the entire backbone and trains only a new classification head on the downstream task. This preserves the pre-trained representations but limits the model’s ability to learn domain-specific features.

Head-only fine-tuning unfreezes the final few layers while keeping earlier feature extraction layers frozen, offering a compromise between adaptation capacity and computational cost.

Full fine-tuning updates all model parameters on the downstream task. This yields the best task-specific performance when sufficient labeled data is available, but introduces two significant risks. First, *catastrophic forgetting*: the model overwrites the general-purpose representations learned during pre-training, losing the ability to transfer to other tasks or domains. Second, *overfitting*: biological imaging datasets are often small (hundreds to low thousands of labeled cells per class), and fine-tuning millions of parameters on such limited data leads to poor generalization.

The fundamental insight motivating the present work is that SSL features are inherently general-purpose and robust, but they remain *static*—they do not adapt to the specific distribution characteristics of new instruments, sample preparations, or experimental conditions encountered at deployment time. We select a different adaptation strategy: instead of modifying the learned feature extractors through fine-tuning, we only tune the statistical parameters of the model at test time, preserving the SSL representations while correcting for domain shift.

2.3.6 The Generalization Gap

Domain shift remains a significant challenge in modern deep learning, as a model’s performance is tied to the specific data on which it was trained. In the context of biological imaging, the domain is defined not only by biological variables but also by the physical parameters of the instruments. For instance, slight variations in optical focus can alter texture features by blurring chromatin patterns, while fluctuations in illumination or gain, such as changes in LED intensity, shift the underlying pixel distributions. Sample preparation protocols introduce substantial variability; as ob-

served in sperm cells analysis study, cryopreservation alters the refractive index and cytoplasmic density of cells compared to fresh samples.

Standard supervised models frequently fail when applied to a new domain where the data distribution of the test set differs from the training set ($P_{train}(X) \neq P_{test}(X)$). This challenge represents a primary barrier to the deployment of artificial intelligence in clinical settings. Addressing this gap requires a shift in focus from training larger models toward developing algorithmic methods capable of adapting models to new domains at the time of inference.

2.4 Robustness and Domain Adaptation

While standard supervised learning assumes that training and testing data are drawn from the same probability distribution ($P_{train}(X, Y) = P_{test}(X, Y)$), this assumption does not always hold in real-world biological imaging due to the high variability of biological samples and instrument parameters, causing domain shift.

2.4.1 Domain Shift in Imaging

Domain shift can be mathematically categorized according to which component of the joint probability distribution undergoes change. The most prevalent scenario is covariate shift, which occurs when the distribution of input features changes, denoted as $P_{train}(X) \neq P_{test}(X)$, while the underlying relationship between features and labels, $P(Y|X)$, remains stable. We observe this during cryopreservation, which may alter the refractive index of a cell’s cytoplasm and thus modify the resulting image X , even though the biological classification Y of a *Normal* class cell remains constant.

Alternatively, label shift occurs when the distribution of class labels themselves changes, represented as $P_{train}(Y) \neq P_{test}(Y)$. This case is inherent to biological research and clinical diagnostics; for example, a healthy control sample will naturally exhibit a vastly different ratio of immune cell subtypes compared to a sample obtained from an infected subject.

Standard deep learning models, particularly those using Batch Normalization

(BN), encode statistics (mean μ and variance σ^2) specific to the training domain directly into the network weights. When $P_{test}(X)$ shifts, frozen statistics become mismatched, leading to feature corruption and erroneous predictions (e.g., due to a change in background illumination intensity).

2.4.2 Test-Time Adaptation

Test-Time Adaptation (TTA) addresses domain shift without the requirement of collecting new labeled data for every experiment. Compared to Unsupervised Domain Adaptation (UDA), which requires concurrent access to both labeled source and unlabeled target data during training, TTA adapts a pre-trained model solely using the unlabeled target data encountered during the inference phase. TTA requires zero original training data and assumes no modification to the source-training procedure [29]. This property is particularly valuable in biological imaging, where privacy constraints, proprietary data, and the impracticality of concurrent retraining make UDA infeasible.

A Taxonomy of TTA Approaches

Recent surveys categorize TTA methodologies based on the specific component adjusted during adaptation [30]:

1. **Model Adaptation:** Direct parameter updates using unsupervised objectives (entropy minimization, pseudo-labeling). Powerful but risks optimization instability and catastrophic forgetting.
2. **Inference Adaptation:** Adjusts the computational graph or prediction aggregation during the forward pass (dynamic routing, augmentation ensembling) without permanent weight changes.
3. **Normalization Adaptation:** Updates only the normalization layer statistics or affine parameters. Highly efficient but dependent on batch quality and size.

4. **Sample Adaptation:** Uses generative models to project target samples back onto the source manifold before classification. Eliminates risk of weight degradation.
5. **Prompt Adaptation:** For Vision-Language Models; tunes continuous text or visual prompt embeddings rather than the backbone parameters.

The following subsections outline the development history of TTA methods across these categories.

Normalization and Self-Supervised Foundations (2016–2020)

The theoretical foundation of modern TTA was laid by the observation that domain discrepancy presents mostly in the statistical distributions of intermediate feature maps.

Adaptive Batch Normalization (AdaBN) proposed a simple approach: during inference, discard the frozen source BN statistics $(\mu_{train}, \sigma_{train}^2)$ and replace them with the mean and variance computed directly from the incoming target batch [31]:

$$\hat{x} = \frac{x - \mu_{batch}}{\sqrt{\sigma_{batch}^2 + \epsilon}} \cdot \gamma + \beta \quad (2.1)$$

This is a parameter-free method and it introduces virtually zero computational overhead. However, its efficacy is directly linked to batch size: when test instances arrive individually or in highly correlated sequences, the estimated batch statistics collapse causing poor prediction quality.

Test-Time Training (TTT) introduced a complementary paradigm by using auxiliary self-supervised learning [32]. During offline training, the network is bifurcated into a shared feature extractor and two heads: a supervised classification head and a self-supervised head (e.g., rotation prediction). At test time, the self-supervised loss provides gradient signal from unlabeled data to update the shared feature extractor, implicitly learning domain-invariant representations. **TTT++** improved upon this by replacing rotation prediction with contrastive learning and

incorporating online feature alignment to stored source manifold statistics, preventing the self-supervised loss from distorting the semantic space [33]. However, both TTT variants require access to the source training procedure to jointly train the auxiliary head, limiting their applicability to pre-trained models.

Entropy Minimization and Stability (2021–2022)

The field shifted towards pure test-time strategies adapting unmodified, pre-trained architectures.

TENT (Test Entropy Minimization) relies on the principle that the Shannon entropy of a model’s softmax output serves as a proxy for alignment with the data distribution [34]. TENT minimizes this entropy via backpropagation, restricting gradient updates exclusively to the affine parameters (γ, β) of the normalization layers while freezing all feature extraction weights:

$$\mathcal{L}_{ent}(x) = - \sum_c p(y = c|x) \log p(y = c|x) \quad (2.2)$$

This constraint allows fast adaptation, establishing state-of-the-art results on CIFAR-10-C and ImageNet-C. However, TENT requires diverse, reasonably sized batches; under label shift or with small batches, it significantly degrades.

MEMO (Marginal Entropy Minimization) addressed the strict batch-size dependency by allowing instance-level adaptation (batch size = 1) [35]. For a single test image, MEMO generates multiple randomly augmented views, passes them through the network, and minimizes the entropy of the *marginal* output distribution across all augmentations. Unlike TENT, MEMO adapts all model parameters, using the consistency signal from augmented views for optimization.

EATA (Efficient Anti-Forgetting Test-Time Adaptation) injected regularization into the adaptation process to prevent continuous entropy minimization from causing catastrophic forgetting [36]. EATA has three main principles: (1) active sample filtration, discarding samples with extremely high entropy (too noisy) or high similarity to previously seen data (redundant); (2) Fisher Information Matrix-

based importance weighting, heavily penalizing changes to parameters critical for the source task; and (3) an explicit anti-forgetting regularizer that balances plasticity with stability.

Continual Test-Time Adaptation (2022–2024)

Standard TTA assumes a static domain shift, but real-world deployment involves continuously changing conditions. An IFC instrument may experience gradual drift in laser power, fluidic pressure, or sample degradation over the course of a long experiment.

CoTTA (Continual Test-Time Adaptation) addresses two problems of continual adaptation: error accumulation and catastrophic forgetting [37]. To mitigate error accumulation, CoTTA uses a weight-averaged “mean teacher” framework—an exponential moving average (EMA) of the student network’s historical weights that provides stable pseudo-labels. For each test sample, up to 32 augmented views are processed by the teacher, and the averaged predictions form a consensus label. CoTTA introduces *stochastic restoration* to mitigate catastrophic forgetting: during every iteration, $\sim 1\%$ of the student network’s neurons are forcibly reset to their original source-pretrained weights. This mechanism acts as a permanent anchor, enabling theoretically infinite-horizon adaptation without collapse.

EcoTTA addressed CoTTA’s prohibitive computational requirements by replacing full-parameter updates with lightweight meta-networks (adapters) injected into the architecture [38]. Only these adapters are updated during adaptation, reducing memory consumption by over 80% compared to CoTTA. Self-distilled regularization prevents the adapters from deviating significantly from the frozen original network’s behavior.

RoTTA (Robust Test-Time Adaptation) tackled the specific challenge of temporally correlated data streams by decoupling the incoming data from the optimization batch [39]. A category-balanced memory bank selectively caches streaming samples based on timeliness and uncertainty; during gradient updates, the optimization batch is sampled from this bank rather than the raw stream, artificially restoring

i.i.d.-like conditions.

Advanced Approaches: Causal Features, Generative, and Prompt-Based

Recent works demonstrated that entropy is a flawed proxy for accuracy under complex distribution shifts. **DeYO** showed that a network may exhibit low entropy not because it recognizes the causal structural features of an object, but because it has overfitted to spurious background textures [40]. DeYO introduced *Pseudo-Label Probability Difference (PLPD)*: the test image is subjected to an object-destructive augmentation (e.g., extreme patch shuffling), and the confidence difference between the original and destroyed image quantifies whether the prediction relies on shape (causal, domain-invariant) or texture (spurious). Only high-PLPD samples are used for adaptation.

Another approach uses **sample-space adaptation** instead of weight updates. **Diffusion-TTA** uses pre-trained denoising diffusion models to project corrupted test images back onto the clean source manifold before classification [41]. Catastrophic forgetting is minimized because the classifier’s weights remain completely frozen. This concept has been extended to 3D point clouds, where latent-space diffusion denoising achieves state-of-the-art robustness on corrupted ShapeNet and ModelNet40 benchmarks.

For big Vision-Language Models (VLMs), **prompt-based adaptation** is the dominant strategy. **TPT (Test-Time Prompt Tuning)** dynamically learns adaptive text prompt embeddings for each test sample by minimizing the marginal entropy across augmented views, while keeping the VLM encoders entirely frozen [42]. **DiffTPT** enhanced this by replacing simple geometric augmentations with Stable Diffusion-generated diverse variations [43]. Most recently, **SubTTA** works purely in the embedding geometry, aligning target visual features to the pre-trained vision-language space and projecting them onto a task-specific textual subspace that orthogonally filters task-irrelevant visual noise [44]. **DART** addresses the additional challenge of label distribution shifts by learning confusion matrices for class-wise correction [45].

Comparative Overview

Table 2.2 summarizes the key TTA methods discussed above.

Table 2.2: Comparison of key TTA methods. “Adaptation Target” indicates which model components are modified during test time.

Method	Year	Adaptation Target	Source Required	Batch Req.	Key Innovation
AdaBN	2016	BN running stats	No	Large	Stat replacement
TTT / TTT++	2021	Shared backbone	Train phase	Any	Self-supervised aux. task
TENT	2021	BN affine (γ, β)	No	Moderate	Entropy minimization
MEMO	2022	All parameters	No	1	Marginal entropy
EATA	2022	BN affine (γ, β)	No	Moderate	Fisher anti-forgetting
CoTTA	2022	All + EMA teacher	No	Any	Mean teacher + restoration
EcoTTA	2023	Adapter modules	No	Any	Meta-networks (80% mem. ↓)
RoTTA	2023	BN + teacher	No	Memory bank	Category-balanced sampling
DeYO	2024	BN affine (γ, β)	No	Moderate	PLPD causal filtering
Diffusion-TTA	2023	Input samples	No	Any	Diffusion purification
TPT / DiffTPT	2023	Text prompts	No	Augmented	Prompt tuning for VLMs
SubTTA	2026	Embedding geometry	No	Any	Subspace alignment
Hybrid (Ours)	2025	Norm. stats + affine	No	Moderate	Sequential stat + entropy

The Hybrid TTA algorithm proposed in this thesis combines elements from two categories: Normalization Adaptation (the AdaBN lineage of batch statistic replacement) [31] and Model Adaptation (the TENT lineage of entropy minimization on affine parameters) [34]. This combination is motivated by the requirements of interactive IFC analysis: (1) computational efficiency sufficient for real-time throughput of tens of thousands of images per second; (2) compatibility with Vision Transformer architectures via Layer Normalization adaptation; and (3) preservation of the self-supervised feature extractors without requiring access to the source training procedure. The sequential design, where statistical alignment precedes entropy minimization, is further explained in Chapter 3.

2.4.3 Unsupervised Discovery and Fuzzy Clustering

Beyond classification, a major goal of single-cell analysis is discovery and explainability. Traditional algorithms like *K-Means* force every cell into a hard, disjoint cluster, which can ignore biological nuance in some applications. Cellular differentiation can be a continuous process; a cell transitioning from the G1 to S phase of the cell cycle, or a stem cell differentiating into a neuron, exists in an intermediate state.

Hard clustering forces these transitional cells into arbitrary categories, obscuring the underlying biology.

Fuzzy (Soft) Clustering algorithms, such as Soft K-Means or density-based approaches like HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise), assign a probability of membership to multiple clusters.

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d_{ij}}{d_{ik}}\right)^{\frac{2}{m-1}}} \quad (2.3)$$

Where u_{ij} is the degree of membership of cell i to cluster j . By integrating fuzzy clustering into the *Eidocell* workflow, we can quantify ambiguity as a biological feature. A sperm cell with a 50% membership to *Normal* and 50% to *Irregular head shape* is not an error; it likely represents a subtle or transitional phenotype that warrants specific investigation.

2.4.4 Research Gap

We identify two major research points to be noted in the field of bioimage and IFC data analysis.

First, self-supervised foundation models such as DinoBloom [27] and RedDino [28] have demonstrated that SSL Vision Transformers can learn universal, high-quality cellular embeddings that generalize across instruments and datasets without task-specific fine-tuning. These models represent a fundamental advance in feature representation quality for single-cell imaging.

Second, the TTA literature has established a number of methods for mitigating domain shift at inference time. However, the majority of TTA research evaluates on synthetic corruption benchmarks (CIFAR-10-C, ImageNet-C) rather than the specific challenges of biological imaging, which include extreme class imbalance, subtle morphological gradients between phenotypes, and instrument-specific optical artifacts.

No existing IFC analysis tool integrates both of these advances. Proprietary platforms (IDEAS, VisualSpreadsheet) rely on classical feature engineering with no adaptation mechanisms. Open-source tools (CellProfiler, EasyFlow) offer greater algorithm-

mic flexibility but lack native deep learning pipelines. Recent ML-centric cytometry tools incorporate deep learning features but assume static model deployment. Furthermore, no existing software tool combines interactive human-in-the-loop analysis with online feature adaptation. This thesis aims to close this gap by implementing a Hybrid TTA algorithm within the *Eidocell* framework, creating an instrument-agnostic platform where self-supervised features dynamically adapt to each new experimental condition while preserving the interactive exploratory workflow that biologists require.

Chapter 3

Methodology: The Eidocell Framework and Algorithms

This chapter details the technical design and mathematical implementation of the proposed solutions. We first outline the software architecture of the *Eidocell* framework, designed to facilitate an interactive, instrument-agnostic analysis of high-throughput image streams. We include the formulation of our hybrid TTA algorithm for mitigating domain shift, and the integration of dimensionality reduction and soft clustering techniques to enable the unsupervised discovery of continuous biological phenotypes.

3.1 System Design and Architecture

The development of Eidocell is motivated to improve analysis of large IFC databases and increase the accessibility of advanced machine learning algorithms. Existing solutions, as discussed in Chapter 2, frequently force a trade-off between the usability found in proprietary solutions and the flexibility of custom Python scripts. We developed our open-source solution around three core design principles: instrument agnosticism, modularity, and human-in-the-loop interactivity.

The principle of instrument agnosticism implies that the system must decouple the analysis pipeline from the specific hardware used for acquisition. Regardless of the imaging platform, the internal representation of the single-cell event remains

consistent. Furthermore, we prioritize modularity to consider rapid advancements in computational research. Preprocessing scripts, feature extractors, and segmentation models can be swapped, updated, or fine-tuned without altering the core application logic. Finally, Eidocell emphasizes human-in-the-loop interactivity to improve the explainability and discovery process through quick access to high-level population statistics, such as scatter plots, clusters, and raw single-cell image data.

Eidocell is implemented as a desktop application using Python 3.9, using the PySide6 binding for Qt as the Graphical User Interface (GUI). The high-performance computational tasks are offloaded to specialized libraries: NumPy and Pandas for vectorial data manipulation, and PyTorch and ONNX Runtime for deep learning inference.

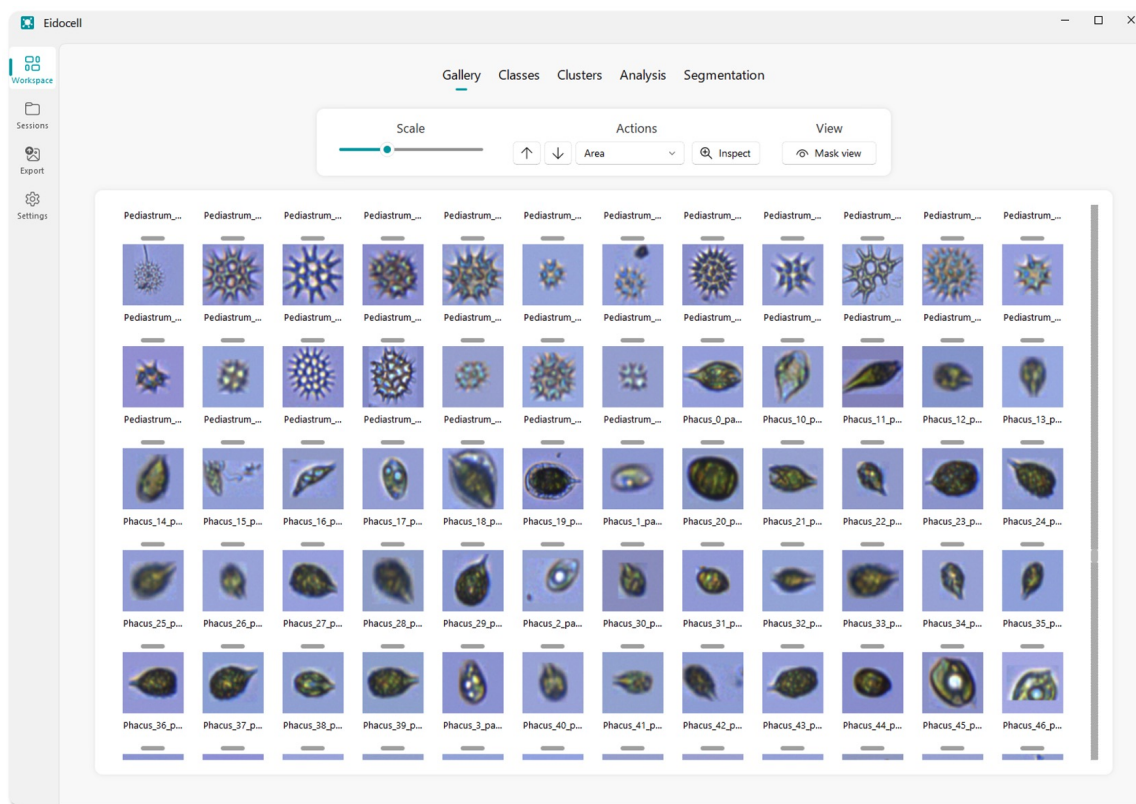


Figure 3-1: Graphical user interface of Eidocell. An algae dataset is shown in the gallery view, where single-cell images are displayed with adjustable gallery settings. The left panel allows navigation between workflow modules (segmentation, clustering, quantitative analysis), while the top panel provides contextual parameters and controls.

The software architecture follows the Model-View-Presenter (MVP) pattern, which

suggests a clean separation between data logic, user interface components, and mediation logic. This structural decoupling ensures that the underlying analytical tools remain independent of the graphical presentation.

The **Model** layer maintains application state and data integrity via a centralized DataManager class. This component serves as the single source of truth for the active session. To ensure ACID compliance and prevent data corruption during analysis, session configurations and gating hierarchies are stored in a lightweight SQLite database. Extracted feature vectors and embeddings are stored in memory-mapped NumPy arrays.

The **View** layer comprises the passive GUI components that display data and capture user input. These components contain no business logic and prioritize performance. For instance, the Gallery View utilizes a virtualized grid with lazy loading to render thousands of cell images without exhausting system RAM.

The **Presenter** layer acts as the functional bridge between the Model and the View. When a user interacts with the View, such as by defining a gate on a scatter plot, the Presenter intercepts the coordinates, queries the DataManager to identify the corresponding object indices, and updates the View to filter the displayed population. This architecture decouples the UI from the backend implementation, allowing for easier long-term testing and algorithmic development.

3.2 The Computational Pipeline

The main data pipeline consists of Data Ingestion, Preprocessing, Segmentation, and Feature Extraction. Each stage is designed to be modular, allowing individual algorithms to be integrated into the workflow.

3.2.1 Data Ingestion and Normalization

The ingestion module allows parallel loading of common image formats, such as TIFF, PNG, and JPG, and HDF5 containers used for raw IFC exports. Basic pre-processing is applied to the imported data for consistent analysis conditions. We start with bit-

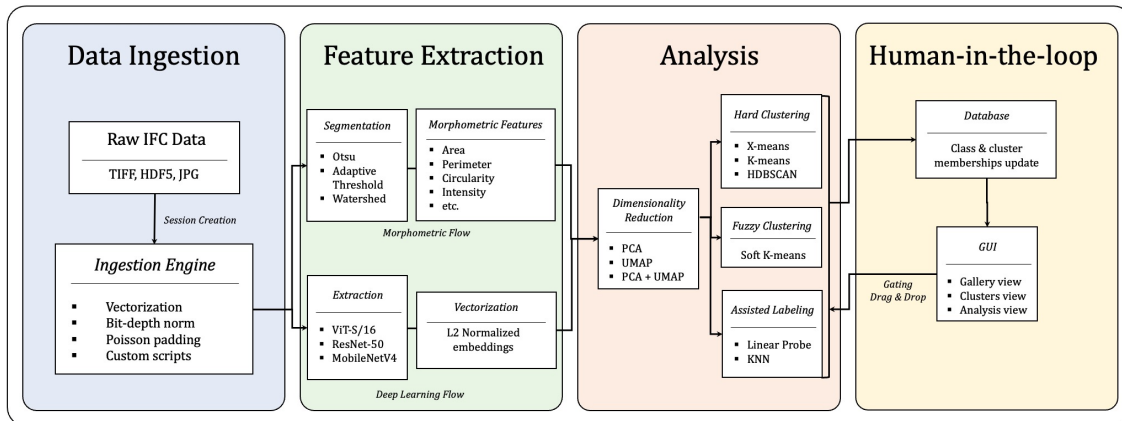


Figure 3-2: The *Eidocell* computational pipeline architecture. Raw multi-channel images are normalized and segmented before entering a feature extraction step. Users can extract interpretable classical morphometric features or use a pre-trained backbone for high-dimensional latent vectorization. These features are subsequently passed to the analysis step comprising dimensionality reduction, clustering, and subsequent human-in-the-loop refinement.

depth conversion, where images (typically ranging from 8-bit to 32-bit floats) are normalized to a standard range (0–255 for visualization or 0–1 for inference) while preserving the dynamic range for quantitative fluorescence analysis. The module performs automated channel mapping to decompose multi-channel stacks into discrete components. Based on metadata or user input, the system identifies the primary *Brightfield* or *Morphology* channel to serve as the base for subsequent analysis tasks.

3.2.2 The Segmentation Module

A region-of-interest can span one or multiple objects within the same image. Simpler analyses generally require whole-cell segmentation, while the analysis of multi-cellular objects or internal structures could benefit from more advanced segmentation methods.

Classical Segmentation

For whole-cell segmentation in high-contrast samples with simple backgrounds (e.g., cell cultures in clear media), we use classical computer vision techniques:

- **Otsu’s Method:** An automatic clustering-based thresholding algorithm that minimizes intra-class variance. This is computationally inexpensive and effective for identifying cells where the signal-to-background ratio is high.
- **Adaptive Thresholding:** For images with uneven illumination, local adaptive thresholding is used. This method calculates thresholds for small regions of the image, robustly identifying objects across illumination gradients.
- **Watershed Algorithm:** To address the problem of cell aggregation (e.g., doublets or clusters), the distance transform of the binary mask is computed, followed by the Watershed algorithm. This separates touching objects based on their topological centers.

Deep Learning-Based Segmentation

For complex biological samples (e.g., lysed blood, debris-rich environmental water), classical methods might fail or limit the possibilities of analysis. To address this, we integrate a lightweight segmentation head on top of a ViT-S/16 backbone, allowing for mask generation based on user prompts (e.g., a mouse click).

3.2.3 Feature Extraction

Once cells are segmented, the system calculates a standard suite of morphometric features using the `scikit-image` library. These include geometric properties (Area, Perimeter, Circularity, Eccentricity, Solidity) and intensity-based properties (Mean Intensity, Standard Deviation, Skewness). These features provide a baseline for interpretable, rule-based gating (e.g., `Select all cells with Area > 100  $\mu m^2$`).

For more advanced feature engineering, we rely on representation learning through pre-trained backbone models. The primary extractor used in this framework is the DINOv3 [14] (Self-distilled Vision Transformer) ViT-S/16 architecture, alongside support for other established models (e.g., MobileNetV3, ResNet-50) and the ability to import custom backbones via `.ONNX`. The feature extraction process operates as follows:

1. **Preprocessing:** Single-cell crops are resized to the model’s native input resolution (e.g., 224×224) using bicubic interpolation. To prevent aspect ratio distortion, we implement a *pad-to-square* strategy rather than artificially stretching the image.
2. **Forward Pass:** The images are passed through the neural network. Instead of taking the output of a final classification layer (which would restrict the system to pre-defined classes), we extract the activation vectors from the penultimate layer, representing the latent space of the image.
3. **Vector Normalization:** The raw activation vectors are L_2 -normalized. This ensures that the distance between vectors in the high-dimensional space corresponds to cosine similarity, a metric that has been shown to correlate strongly with semantic visual similarity.

3.3 Robust Feature Adaptation under Domain Shift

The baseline Eidocell framework serves as a high-level tool to extract high-dimensional features. As we discussed, the reliability of integrated static deep learning models is constrained by the assumption of i.i.d. data. This section formalizes the domain shift problem in the context of IFC and presents our adaptation strategy.

3.3.1 Problem Formulation

Let $\mathcal{D}_S = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ represent the source domain (training data), where $x^s \in \mathcal{X}$ are images and $y^s \in \mathcal{Y}$ are the associated morphological labels. A neural network f_θ is trained to minimize the empirical risk:

$$\theta^* = \arg \min_{\theta} \frac{1}{N_s} \sum_{i=1}^{N_s} \ell(f_\theta(x_i^s), y_i^s) \quad (3.1)$$

During deployment, the model encounters a target domain $\mathcal{D}_T = \{x_i^t\}_{i=1}^{N_t}$ (e.g., cryopreserved cells), where labels y^t are unavailable. In IFC, the physical variability

of sample preparation and instrument calibration causes *covariate shift*: the marginal distributions of inputs differ ($P_S(X) \neq P_T(X)$), while the conditional label distribution remains stable ($P_S(Y|X) \approx P_T(Y|X)$).

We can characterize IFC-specific domain shift through the physical parameters of the imaging system. Let $\phi = (z, I, n, s)$ denote the vector of physical parameters, where z is the focal depth, I is the illumination intensity, n is the refractive index of the medium, and s encodes the sample preparation protocol (e.g., fresh vs. cryopreserved). The target image is then modeled as:

$$x^t = g(x^s; \Delta\phi) \quad (3.2)$$

where g is the image formation function perturbed by the parameter deviation $\Delta\phi = \phi_T - \phi_S$. Empirically, this perturbation might manifest as alterations in background intensity, focus blur, or cytoplasmic texture, which are changes that do not alter the semantic identity of the cell but drastically degrade the performance of f_θ .

The objective of TTA is to find adapted parameters θ_T^* that minimize the expected risk on the target domain using only unlabeled target data:

$$\theta_T^* = \arg \min_{\theta} \mathbb{E}_{x \sim P_T(X)} [\ell(f_\theta(x), y)] \quad (3.3)$$

subject to the constraint that no labels from $\mathcal{D}_T$ and no data from $\mathcal{D}_S$ are accessible. To address this, we embed an unsupervised Hybrid TTA strategy directly into the Eidocell inference module, dynamically aligning both the feature statistics and the decision boundaries of the model to the incoming data stream.

3.3.2 Statistical Feature Alignment via Normalization Adaptation

Deep neural networks rely heavily on BN layers to stabilize feature distributions. A BN layer normalizes an input feature map x using the mean μ and variance σ^2 :

$$\hat{x} = \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} \cdot \gamma + \beta \quad (3.4)$$

During inference, the model uses fixed running statistics $(\mu_{train}, \sigma_{train})$ computed during training. Under domain shift these fixed statistics no longer match the activation statistics of the target data. This statistical mismatch propagates through the network, corrupting the feature representation.

To mitigate this, we add Test-Time Normalization (TTN). When a new session is loaded into Eidocell, we re-estimate the statistics μ_{test} and σ_{test} using the incoming batch of randomly sampled unlabeled test images B_T . The adaptation rule for the normalization statistics at layer l is defined as:

$$\mu_l \leftarrow (1 - \alpha)\mu_l + \alpha \cdot \frac{1}{|B_T|} \sum_{x \in B_T} f_\theta^l(x) \quad (3.5)$$

$$\sigma_l^2 \leftarrow (1 - \alpha)\sigma_l^2 + \alpha \cdot \text{Var}(f_\theta^l(B_T)) \quad (3.6)$$

Where $\alpha \in [0, 1]$ is a momentum coefficient. By updating these statistics, we tune the model to possible texture, light, or noise variations in images. This step is computationally light requiring only a single forward pass. For ViT architectures, which utilize Layer Normalization (LN) rather than Batch Normalization, the adaptation is performed by unfreezing the affine parameters (γ, β) of the LN layers, allowing the model to rescale the features of the target domain to match the expected dynamic range of the self-attention heads.

Information-Theoretic Justification. The basis of normalization adaptation is distribution divergence. Each normalization layer l encodes source-specific sufficient statistics $(\mu_S^l, \sigma_S^{l,2})$ that implicitly define a Gaussian prior over the expected activation distribution. Under covariate shift, the true activation distribution $P_T(z_l)$ at layer l diverges from this prior. Assuming approximately Gaussian activations[46] the KL

divergence between the target activations and the source prior grows as:

$$D_{KL}(P_T(z_l) \parallel \mathcal{N}(\mu_S^l, \sigma_S^{l^2})) = \frac{(\mu_T^l - \mu_S^l)^2}{2\sigma_S^{l^2}} + \frac{\sigma_T^{l^2}}{2\sigma_S^{l^2}} - \frac{1}{2} \log \frac{\sigma_T^{l^2}}{\sigma_S^{l^2}} - \frac{1}{2} \quad (3.7)$$

Replacing the source statistics with the target statistics $(\mu_T^l, \sigma_T^{l^2})$ eliminates this divergence entirely for Gaussian-distributed activations, achieving optimal alignment at every layer. The momentum-based update (Equations 5–6) represents a smoothed variant of the direct replacement proposed in AdaBN [31], exchanging optimality for temporal stability.

3.3.3 Semantic Boundary Alignment via Entropy Minimization

Statistical alignment corrects for global shifts in pixel intensity and texture, but does not guarantee that the decision boundaries of the classifier remain optimal. A model may have corrected feature statistics but produce low-confidence predictions for samples lying near the decision boundary (e.g., a probability distribution like $[0.4, 0.4, 0.2]$).

To sharpen the decision boundaries without access to labels, we use Entropy Minimization. The hypothesis is that a classifier adapted to the target domain should make confident predictions (low entropy), regardless of whether those predictions are correct. By forcing the model to be confident, we push samples away from the decision boundaries and toward the class centroids.

For a given input x , the model outputs a probability distribution $\hat{y} = f_\theta(x)$. The Shannon entropy $H(\hat{y})$ is defined as:

$$\mathcal{L}_{ent}(x) = - \sum_{c=1}^C \hat{y}_c \log \hat{y}_c \quad (3.8)$$

where C is the number of classes. During the adaptation phase, we minimize this entropy loss on the target batch B_T . However, optimizing all model parameters θ can lead to *model collapse* or catastrophic forgetting. We restrict the optimization

to the affine transformation parameters (γ, β) of the normalization layers, keeping the convolutional weights and attention heads frozen. This approach in combination with the TENT methodology [34] allows the network to adapt its feature scaling while preserving the learned semantic filters.

Information-Theoretic Justification. Shannon entropy provides an unsupervised proxy for prediction quality through several theoretical connections. First, entropy is an upper bound on the Bayes error rate: $P_e \leq \frac{1}{2}H(\hat{Y})$ for binary classification, with analogous bounds for the multiclass case [30]. Minimizing entropy therefore pushes the expected error rate downward. Second, under the assumption of a uniform marginal label distribution, minimizing the conditional entropy $H(\hat{Y}|X)$ is equivalent to maximizing the mutual information $I(X; \hat{Y})$ between inputs and predictions:

$$I(X; \hat{Y}) = H(\hat{Y}) - H(\hat{Y}|X) \quad (3.9)$$

This maximization encourages the model to preserve the maximum amount of discriminative information from the input in its predictions. Third, entropy minimization connects to the *cluster assumption* in semi-supervised learning: decision boundaries should pass through low-density regions of the data distribution. By pushing predictions toward the vertices of the probability simplex (one-hot vectors), entropy minimization drives samples away from uncertain boundary regions and toward high-density class centroids.

However, entropy minimization implemented without constraints might assign all probability mass to a single class and globally achieve zero entropy, causing class collapse. The restriction of gradient updates to only the affine parameters (γ, β) of normalization layers is safer, as they can rescale and shift feature distributions but cannot alter the learned class-discriminative filters, limiting the optimization’s capacity to reach degenerate solutions.

3.3.4 The Hybrid TTA Algorithm

The Hybrid TTA algorithm integrates statistical feature alignment and semantic boundary alignment into a unified, unsupervised adaptation loop. The algorithm executes them sequentially on each incoming batch to maximize both adaptation stability and computational throughput.

First, the statistical alignment acts as a zero-gradient pre-conditioning step. By recalculating the normalization statistics directly from the unlabeled target batch, we neutralize global covariate shifts (fluctuations in illumination, focus blur, or changes in the sample’s refractive index). This step projects the shifted target features back into the normalized distribution bounds for the pre-trained network.

Second, we perform semantic boundary alignment via entropy minimization. Because the features have already been globally aligned, the entropy loss function operates on a smoother landscape, potentially improving the convergence. Statistical alignment performs a first-order correction (mean and variance alignment) that is convex and admits a closed-form solution. By projecting the shifted target features back into the normalized distribution bounds of the pre-trained network, the statistical alignment step reduces the magnitude of the entropy loss gradient $\|\nabla_{\gamma,\beta}\mathcal{L}_{ent}\|$, since the features are already approximately centered and scaled. This means the subsequent entropy minimization requires fewer gradient update steps to converge, explaining the empirical observation in our ablation study (Section 5): the hybrid method achieves a throughput of ~ 250 images/second, compared to only ~ 145 images/second for semantic alignment alone.

There is risk of catastrophic forgetting and model collapse when TTA approaches are applied. Unconstrained self-training or entropy minimization can cause a model to overfit to an anomalous batch of data, or cause prediction of a single majority class to artificially minimize uncertainty. To mitigate this, we add previously mentioned stability mechanisms, such as a momentum-based moving average (controlled by the coefficient α) for the running statistics. By interpolating between the historical parameters and the current batch estimates, the model retains generalized

representations learned from the source domain while remaining dynamically responsive to the target stream. Furthermore, by strictly confining all gradient updates to the lightweight affine transformation parameters (γ, β) of the normalization layers we structurally prevent the network from collapsing into low-entropy solutions. The complete procedure is detailed in Algorithm 1.

Algorithm 1 Hybrid Test-Time Adaptation

- 1: **Input:** Pre-trained model f_θ with normalization parameters $\Phi = \{\mu, \sigma, \gamma, \beta\}$.
 - 2: **Input:** Target data stream D_T (unlabeled).
 - 3: **Hyperparameters:** Learning rate η , Momentum α .
 - 4: Initialize target model $f_{\theta^*} \leftarrow f_\theta$.
 - 5: **for** each batch $x_b \in D_T$ **do**
 - 6: *// Step 1: Statistical Alignment*
 - 7: Compute batch statistics: $\mu_b, \sigma_b^2 \leftarrow \text{Stats}(x_b)$
 - 8: Update running stats: $\mu \leftarrow (1 - \alpha)\mu + \alpha\mu_b$
 - 9: *// Step 2: Semantic Alignment*
 - 10: Forward pass: $\hat{y}_b = f_{\theta^*}(x_b)$
 - 11: Compute Entropy Loss: $L = \frac{1}{|B|} \sum \mathcal{L}_{ent}(\hat{y}_b)$
 - 12: Compute Gradients: $\nabla_{\gamma, \beta} L$
 - 13: Update Affine Params: $\gamma \leftarrow \gamma - \eta \nabla_\gamma L, \quad \beta \leftarrow \beta - \eta \nabla_\beta L$
 - 14: **Return:** Adapted model f_{θ^*} .
-

3.4 The Analysis Engine

The final step is the interpretation of the extracted feature vectors. Because deep representation learning results in high-dimensional embeddings (e.g., 384 or 768 dimensions for a Vision Transformer), they cannot be analyzed using standard 2D gating. We perform dimensionality reduction, unsupervised clustering, and semi-supervised label propagation.

3.4.1 Dimensionality Reduction

To use visualization plots and improve the feature interpretability of the high-dimensional latent space, we implement multiple non-linear dimensionality reduction (DR) methods. Alongside Principal Component Analysis (PCA) for linear variance estimation,

we often use Uniform Manifold Approximation and Projection (UMAP) [47]. UMAP was selected over t-SNE due to its superior preservation of global data structure and lower computational overhead. The system allows users to tune key hyperparameters:

- **n_neighbors:** Controls how UMAP balances local versus global structure. Low values focus on fine-grained differences between cells, while higher values emphasize broader biological groupings.
- **min_dist:** Sets the minimum distance between points in the 2D plane, preventing over-packing of points and facilitating easier manual gating.
- **Metric Selection:** Users can choose between Euclidean, Cosine, or Manhattan distances (e.g., Cosine is typically preferred for L_2 -normalized deep learning embeddings).

3.4.2 Discovery and Fuzzy Clustering

We include a set of clustering algorithms for iterative discovery. The default baseline is X-Means—an extension of K-Means that uses the Bayesian Information Criterion (BIC) to automatically estimate the optimal number of clusters (K) [48]. X-Means identifies statistically distinct morphological groups without the researcher needing to define a taxonomy beforehand. For more complex, density-variable data (such as identifying rare cell doublets), we include HDBSCAN, which can identify noise points and outliers, filtering debris. However, traditional algorithms like X-Means and HDBSCAN assign hard labels to data points, meaning every event must belong to exactly one cluster. This discretization can be biologically inaccurate. Hard clustering forces these transitional cells into arbitrary categories, obscuring the underlying biology. To address this, we implement Soft (Fuzzy) Clustering. Instead of a binary assignment, we compute a membership vector $\mathbf{u}_i = [u_{i1}, u_{i2}, \dots, u_{iK}]$ for each cell x_i , where u_{ij} represents the degree of membership to cluster j . We use a Soft K-Means approach, minimizing the objective function:

$$J_m = \sum_{i=1}^N \sum_{j=1}^K (u_{ij})^m \|x_i - c_j\|^2 \quad (3.10)$$

where $m > 1$ is the fuzzifier exponent (controlling the softness of the boundaries) and c_j is the centroid of cluster j . The membership degree is calculated as:

$$u_{ij} = \frac{1}{\sum_{k=1}^K \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad (3.11)$$

Defining the input and output parameters, the full algorithm could be represented as:

Algorithm 2 Soft K-Means (Fuzzy Clustering) Algorithm

```

1: Input: Extracted feature set  $X = \{x_1, x_2, \dots, x_N\}$ .
2: Input: Number of clusters  $K$ , Fuzzifier  $m > 1$ , Error tolerance  $\epsilon$ .
3: Output: Cluster centroids  $C = \{c_1, \dots, c_K\}$ , Membership matrix  $U$ .
4: Initialize membership matrix  $U^{(0)}$  randomly, such that  $\sum_{j=1}^K u_{ij} = 1$  for all  $i$ .
5: Set iteration counter  $t \leftarrow 0$ .
6: repeat
7:   // Step 1: Update Cluster Centroids
8:   for  $j = 1$  to  $K$  do
9:      $c_j^{(t+1)} = \frac{\sum_{i=1}^N (u_{ij}^{(t)})^m x_i}{\sum_{i=1}^N (u_{ij}^{(t)})^m}$ 
10:  // Step 2: Update Membership Degrees
11:  for  $i = 1$  to  $N$  do
12:    for  $j = 1$  to  $K$  do
13:       $u_{ij}^{(t+1)} = \left[ \sum_{k=1}^K \left( \frac{\|x_i - c_j^{(t+1)}\|}{\|x_i - c_k^{(t+1)}\|} \right)^{\frac{2}{m-1}} \right]^{-1}$ 
14:     $t \leftarrow t + 1$ 
15: until  $\|U^{(t)} - U^{(t-1)}\| < \epsilon$  ▷ Convergence
16: return  $C, U^{(t)}$ 

```

We can now quantify phenotypic ambiguity as a biological feature through probabilistic output. This allows researchers to distinctively filter for archetypal cells (high certainty) versus transitional cells (high ambiguity). If a cluster is found to be heterogeneous, the user can manually split or merge groups using the GUI drag-and-drop interface, a process that updates the class labels in the underlying SQLite database in real-time.

3.4.3 Clustering Evaluation Metrics

To quantitatively assess the performance of our unsupervised representation learning and clustering algorithms (e.g., X-Means and HDBSCAN), we use metrics that compare the algorithmic cluster assignments against expert-annotated ground truth labels. Because unsupervised clustering does not assign semantic labels (i.e., the algorithm outputs *Cluster 1* instead of *Biconcave RBC*), standard supervised metrics like Accuracy or F1-Score cannot be directly applied without solving the bipartite matching problem. Instead, we use two standard, permutation-invariant clustering metrics: Normalized Mutual Information (NMI) and the Adjusted Rand Index (ARI). Both metrics evaluate the structural agreement between two partitions of the data, independent of the label values.

Normalized Mutual Information (NMI)

NMI is an information-theoretic measure that quantifies the amount of information shared between the ground truth labels $U = \{u_1, u_2, \dots, u_R\}$ and the predicted clusters $V = \{v_1, v_2, \dots, v_C\}$.

The foundation of NMI is the Mutual Information (MI), defined as:

$$I(U, V) = \sum_{i=1}^R \sum_{j=1}^C P(u_i \cap v_j) \log \frac{P(u_i \cap v_j)}{P(u_i)P(v_j)} \quad (3.12)$$

where $P(u_i)$ is the probability that a randomly selected cell belongs to class u_i , $P(v_j)$ is the probability it belongs to cluster v_j , and $P(u_i \cap v_j)$ is the joint probability of the cell belonging to both.

Because standard MI is not upper-bounded and tends to increase as the number of clusters increases, it is normalized using the Shannon entropy H of the partitions:

$$\text{NMI}(U, V) = \frac{2 \cdot I(U, V)}{H(U) + H(V)} \quad (3.13)$$

where $H(U) = -\sum_i P(u_i) \log P(u_i)$.

The resulting NMI score ranges from 0 to 1. A score of 0 indicates that the pre-

dicted clusters share no mutual information with the true biological classes (random assignment), while a score of 1 indicates perfect agreement.

Adjusted Rand Index (ARI)

While NMI measures shared information, the Rand Index (RI) approaches the problem from a combinatorial perspective, counting the number of pairs of cells that are assigned consistently (either grouped together or separated) in both the ground truth U and the prediction V .

Let n be the total number of cells. The contingency table elements n_{ij} represent the number of cells that are in both class u_i and cluster v_j . Let a_i be the number of cells in class u_i , and b_j be the number of cells in cluster v_j .

The unadjusted Rand Index is strictly the ratio of agreeing pairs to the total possible pairs $\binom{n}{2}$. However, in high-dimensional biological data with high class imbalance, random clustering can yield a deceptively high RI. Therefore, we use the ARI, which corrects for chance grouping by establishing a baseline of expected similarity:

$$\text{ARI} = \frac{\text{Index} - \text{Expected Index}}{\text{Max Index} - \text{Expected Index}} \quad (3.14)$$

Mathematically, this is computed as:

$$\text{ARI} = \frac{\sum_{i,j} \binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2} \right] / \binom{n}{2}}{\frac{1}{2} \left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2} \right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2} \right] / \binom{n}{2}} \quad (3.15)$$

The ARI score ranges from -1 to 1. A score of 0 signifies that the clustering performs no better than random assignments drawn from the same marginal distributions, negative values indicate anti-correlation, and a score of 1 indicates identical partitioning.

3.4.4 Assisted Classification

To maximize sample efficiency, we include a semi-supervised labeling module. The program allows the user to label a small subset of representative cells (e.g. through unsupervised clustering or by manually selecting events) and then automates the labeling of the remaining session images. This is implemented via a k-Nearest Neighbors (k-NN) propagation logic:

1. For every unlabeled cell i , the system identifies its k nearest neighbors in the high-dimensional latent space.
2. The cell is assigned a predicted label based on the majority vote of its neighbors, weighted by their distance:

$$\hat{y}_i = \operatorname{argmax}_c \sum_{j \in \mathcal{N}_i} w_{ij} \mathbb{I}(y_j = c) \quad (3.16)$$

where w_{ij} is the inverse of the distance between cell i and neighbor j .

3. A confidence score is calculated for each prediction. Cells with scores below a user-defined threshold are flagged for manual review, ensuring that the final classification remains firmly under expert control.

Chapter 4

Data Resources

To evaluate the performance of the suggested framework and algorithms, we obtained a set of databases that contain challenges frequently occurring in IFC: class imbalance, optical aberrations, multi-cellular interactions, and domain shift. We outline the details of the acquisition, preprocessing, and characteristics of the four primary biological case studies and the technical benchmarks utilized in this thesis.

4.1 Biological Case Studies

We utilized four distinct biological datasets acquired via IFC. Each dataset represents a case study designed to test a unique capability of the *Eidocell* framework and the proposed adaptation algorithms.

4.1.1 Case Study I: Bovine Sperm Morphology

The first dataset is derived from a study investigating the impact of cryopreservation on bull sperm morphology [12], and was acquired using an ImageStream[®] X Mk II at 40x magnification. The samples represent three distinct bull breeds—Kazakh Whitehead, Auliekol, and Simmental. The collection contains 436,374 single-cell brightfield images. From this total, a subset of 12,000 images was manually annotated across eight morphological categories: *Normal*, *Irregular head shape*, *Twisted or elongated*

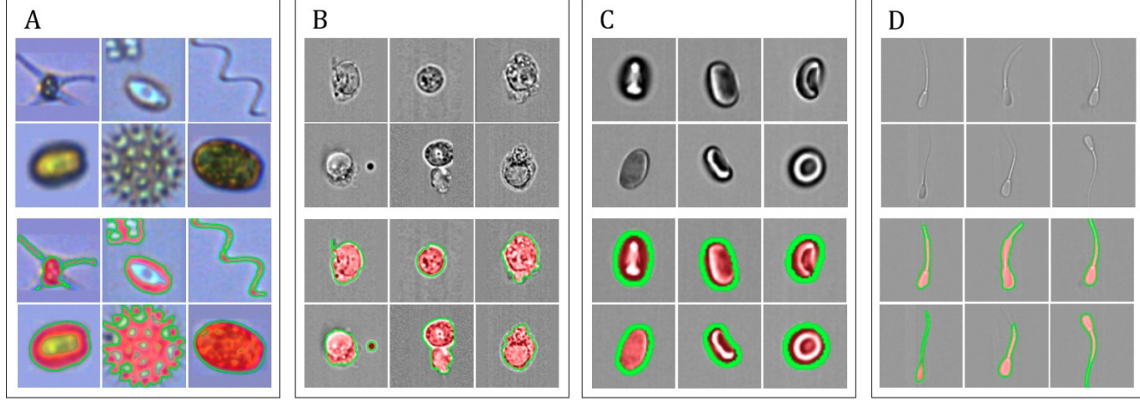


Figure 4-1: Sample images and their corresponding masks from the biological case studies. (A) Microalgae dataset. Intensity-based Otsu’s thresholding is used for segmentation. (B) Immune synapse dataset. Edge-based Otsu’s thresholding is applied (Otsu + edge detection + closed contours). (C) RBC dataset. Adaptive thresholding is applied for segmentation. (D) Sperm cells dataset. Same segmentation settings as in case B.

head, Abnormal midpiece, Abnormal tail, Proximal Cytoplasmic Droplet (Proximal CD), Distal Cytoplasmic Droplet (Distal CD), and Coiled Tail/Midpiece (CTM).

The main challenge is the two data acquisition states: frozen and fresh. Cryopreservation induces subtle alterations in cytoplasmic density and the structural and biophysical parameters of the cells.

4.1.2 Case Study II: Post-COVID-19 Red Blood Cells

The data originates from a study conducted at the National Research Center for Cardiac Surgery investigating rheological changes in erythrocytes in patients with post-COVID-19 syndrome [13]. Using the ImageStream X Mk II, approximately 35,000 Brightfield images were captured and categorized into three morphological states: healthy Biconcave cells, Side-View orientations, and pathological Elongated cells.

The primary challenge within this dataset stems from the physics of hydrodynamic focusing, which often causes red blood cells to rotate or drift out of the focal plane. Consequently, the collection contains a high proportion of out-of-focus events. These instances serve as a proxy for validating the robustness of feature extractors against

instrumental noise and blur, ensuring that the software can maintain classification accuracy despite physical variations in the flow stream.

4.1.3 Case Study III: Microalgae Database

This dataset was collected from field studies as well as mesocosm research using the FlowCam VS4 (Yokogawa Fluid Technologies, USA) [49, 50, 51], and characterized by high-class imbalance and morphological diversity. From this study, a filtered collection of 27,000 events comprising 29 distinct microalgal groups, was used for validation..

4.1.4 Case Study IV: Immunological Synapse

We utilized data from the recent immunological synapse profiling study conducted by Boushehri and co-authors (2023). This work introduced scifAI, a machine learning framework specifically designed to analyze multi-channel IFC data. The original dataset has images of primary human T and B cells forming immunological synapses. These images were captured under various experimental conditions, including different immune stimulations and therapeutic antibody treatments. The images were manually classified into nine biological categories, which distinguished between single cells, doublets, multicell interactions, and functional versus non-functional synapses. In total, 5221 images were labeled.

Table 4.1: Summary of the four primary biological case studies used to validate the framework.

Case Study	Total Events	Classes	Primary Challenge
1. Bovine Sperm	436,374	8	Domain Shift (Fresh $\leftrightarrow$ Frozen)
2. RBCs	35,000	3	Optical Defocus & Blur
3. Algae	27,000	29	Taxonomic Diversity & Class Imbalance
4. Immune Synapse	5,221	9	Multi-Modal Feature Fusion

4.2 Technical Benchmark Datasets

The biological datasets described above provide the necessary domain context but contain uncontrolled variables (e.g., simultaneous changes in size, texture, and ro-

tation) that make it difficult to isolate the specific effects of distribution shift algorithms. To robustly validate the proposed TTA approach, we utilize the standardized -*C* (Corrupted) dataset family introduced by Hendrycks and Dietterich [52]. These benchmarks apply 15 distinct algorithmic corruptions across five severity levels to test images, serving as controlled proxies for physical artifacts encountered in high-throughput microscopy. Noise corruptions (such as Gaussian, shot, and impulse noise) simulate sensor noise and low-light grain; blur corruptions (defocus, motion, and zoom) parallel hydrodynamic focus drift and motion blur; and digital corruptions (contrast, brightness, and JPEG compression) mimic variations in illumination and large-scale storage artifacts.

This approach is tested across three datasets of different complexity. The first dataset is CIFAR-10-C, a 10-class dataset comprising 50,000 source domain training images and 750,000 target domain evaluation images. This serves as a base benchmark to validate statistical alignment capabilities. The semantic difficulty is increased using CIFAR-100-C, which contains 100 fine-grained classes. The third dataset, Tiny-ImageNet-C, featuring 200 classes with images resized to 64×64 pixels.

Table 4.2: Summary of the technical benchmark datasets. Each dataset applies 15 distinct synthetic corruptions across 5 severity levels to simulate physical microscopy artifacts (e.g., optical defocus, sensor noise, illumination shifts).

Dataset	Classes	Resolution	Source Images	Target Images	Purpose
CIFAR-10-C	10	32×32	50,000	750,000	Baseline validation of statistical alignment on low-resolution, broad semantic classes.
CIFAR-100-C	100	32×32	50,000	750,000	Assessing adaptation stability across highly imbalanced, fine-grained taxonomies.
Tiny-ImageNet-C	200	64×64	100,000	750,000	Scalability testing on higher-dimensional feature spaces and complex visual variances.

Chapter 5

Experiments and Results

In this chapter, we evaluate the performance of the proposed workflows and algorithms. The evaluation is conducted in two phases. First, we validate the *Eidocell* system as a static analysis tool, selecting an optimal backbone architecture and assessing its capabilities in unsupervised biological discovery. Second, we evaluate the proposed TTA strategies against state-of-the-art baselines under real-world domain shifts.

5.1 Phase I: Workflow Validation and Backbone Selection

The downstream performance of the Eidocell workflow relies entirely on the quality of the feature representations extracted from the single-cell images. We prioritized SSL ViTs over traditional supervised CNNs to ensure generalizability without task-specific retraining.

5.1.1 Backbone Selection and Performance Analysis

We evaluated several pre-trained backbones on our primary biological datasets using a k-NN classifier as a proxy for feature separability. As shown in Table 5.1, the **DINOv3 (ViT-Small/16)** backbone demonstrated superior performance in class

separability.

Table 5.1: K-NN classification accuracy of various feature extractor backbones. Results are reported as mean $\pm$ standard deviation. DINOv3 consistently outperforms CNN baselines.

Backbone	Algae	Imm. Synapse (BF)	Imm. Synapse (MC)	RBC	Sperm Cells
ViT-S/16 (DINOv3)	86.12 $\pm$ 0.84	54.07 $\pm$ 2.59	78.00 $\pm$ 2.26	98.28 $\pm$ 0.46	74.54 $\pm$ 0.85
EfficientNetV2	79.61 $\pm$ 0.83	51.73 $\pm$ 1.97	69.63 $\pm$ 1.63	97.40 $\pm$ 0.75	69.16 $\pm$ 1.13
MobileNetV3 (large)	82.77 $\pm$ 0.78	53.73 $\pm$ 2.14	68.97 $\pm$ 1.87	98.09 $\pm$ 0.66	69.34 $\pm$ 1.12
ResNet-50	80.81 $\pm$ 0.63	50.87 $\pm$ 3.12	71.71 $\pm$ 1.98	97.55 $\pm$ 0.72	67.08 $\pm$ 0.85
ViT-S/16 (ImageNet-21k)	82.19 $\pm$ 1.50	53.67 $\pm$ 4.54	70.18 $\pm$ 2.03	98.77 $\pm$ 0.26	67.29 $\pm$ 1.05

The performance gap between ViT-S/16 and traditional CNNs (like ResNet-50) might be explained by an architectural and pretraining strategy progress. Objects in IFC are randomly rotated and vary in their position within the cropped frame. CNNs rely on local receptive fields, making them sensitive to spatial translations. In contrast, the self-attention mechanism of vision transformers computes global contextual relationships across the entire image patch grid, rendering the extracted features inherently more robust to the random rotations and positional shifts native to hydrodynamic focusing.

5.1.2 Clustering Performance

To validate the human-in-the-loop discovery workflow, the pipeline was applied to the biological datasets to evaluate clustering performance against ground-truth annotations.

Using X-Means (hard clustering) on UMAP-reduced DINOv3 features, we extracted biologically meaningful clusters. As shown in Table 5.2, X-means demonstrated a significant performance advantage over standard soft clustering. For instance, in the Immune Synapse dataset (multi-channel), it achieved an ARI of 53.22 ± 5.90 , spontaneously separating single T-cells from conjugates with high precision.

While Soft K-means consistently yielded lower ARI and NMI scores compared to X-means, it provides probabilistic membership vectors that capture transitional phenotypes, as demonstrated in the following section.

Table 5.2: Comparative clustering performance between Soft K-means and X-means algorithms. Results are reported as mean ARI and mean NMI ($\pm$ SD) over five runs. UMAP was configured with $n_neighbors=15$, $min_distance=0.1$

Dataset	Modality	ARI	NMI
<i>Soft K-means</i>			
Algae	BF	34.14 ± 7.16	41.21 ± 9.20
Immune Synapse	BF	13.04 ± 6.47	18.12 ± 5.52
Immune Synapse	Multi-channel	38.20 ± 5.15	42.05 ± 7.58
RBC	BF	54.26 ± 5.46	61.60 ± 7.31
Sperm Cells	BF	16.43 ± 7.81	23.17 ± 9.77
<i>X-means (Hard Clustering)</i>			
Algae	BF	55.45 ± 7.47	73.68 ± 10.16
Immune Synapse	BF	25.17 ± 9.94	32.18 ± 9.88
Immune Synapse	Multi-channel	53.22 ± 5.90	56.05 ± 6.03
RBC	BF	85.20 ± 8.95	80.21 ± 7.15
Sperm Cells	BF	29.85 ± 8.50	40.05 ± 8.77

5.1.3 AI Interpretability and Morphological Continuity

Standard hard-clustering forces every cell into a single, discrete category, ignoring the continuous nature of biological deformation. By applying fuzzy clustering to the neural network’s latent space, we can quantify how cells transition between morphologies. Figure 5-1 illustrates this ambiguity.

Figure 5-1A measures the overall similarity between the average fuzzy profiles of each class. The network effectively captures biological reality: classes with massive structural deformities, such as the Coiled Tail/Midpiece (CTM), exhibit low similarity (red) to all other classes, indicating the network easily isolates this gross defect. Conversely, Normal cells and Distal CD cells show extremely high similarity (0.97). This aligns with biological intuition, as a distal cytoplasmic droplet is a minor, localized perturbation on a flagellum that is otherwise structurally identical to a normal cell.

To understand directional uncertainty, Figure 5-1B maps the flow of fuzzy membership. By assigning each mathematical cluster a dominant biological label, we can observe how probability mass from a true source class (y-axis) spills over into target clusters (x-axis). Crucially, this relationship is asymmetric, revealing how the network

prioritizes specific features. For instance, a Normal cell frequently registers partial membership in the Proximal CD category (0.49), likely because minor optical noise or debris near the cell neck can mimic a droplet. However, the reverse is not true: an actual Proximal CD cell has a highly distinct physical feature, meaning the network rarely confuses it for a pristine Normal cell (0.23). This asymmetry demonstrates that the fuzzy clustering algorithm is not generating random noise, but rather capturing a logical, hierarchical continuum of morphological features.

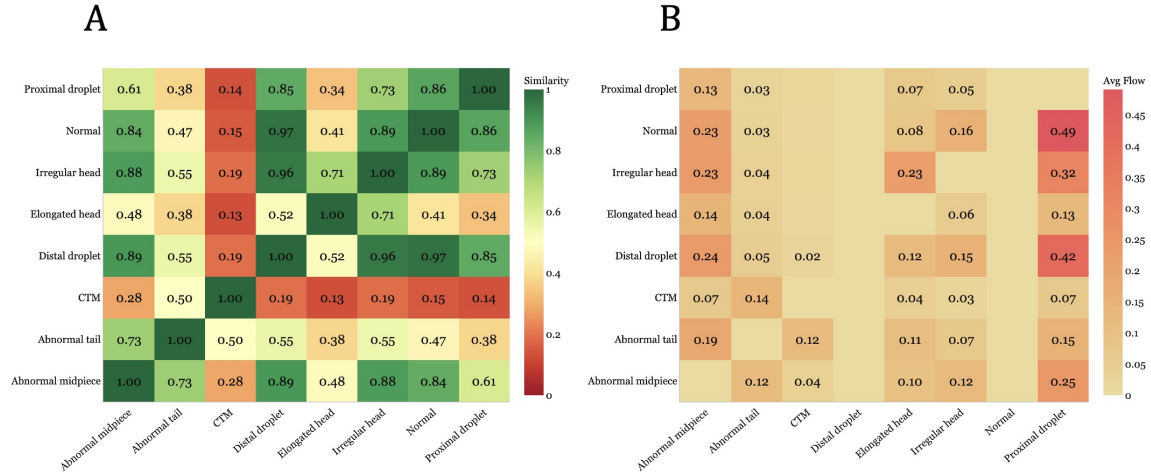


Figure 5-1: Quantitative analysis of morphological ambiguity using Fuzzy Clustering on the Bovine Sperm dataset. (A) Class Membership Similarity: A symmetric heatmap showing the overall morphological overlap between ground-truth classes in the model's latent space. Green values indicate classes that the model views as structurally highly similar (e.g., Normal and Distal CD, 0.97). Red values indicate gross deformities that the model easily isolates from other classes (e.g., CTM). (B) Cross-Class Membership Flow: An asymmetric heatmap showing directional confusion or morphological transition between classes. The y-axis represents the true class of the cells, and the x-axis represents the dominant clusters where their fuzzy probabilities spill over. The asymmetry highlights hierarchical feature detection: e.g., simple Normal cells frequently exhibit partial resemblance to Proximal CD clusters (0.49) due to subtle image noise, whereas actual Proximal CD cells possess distinct features and are less frequently confused for Normal cells (0.23). The diagonal is masked to zero to highlight only cross-class interactions.

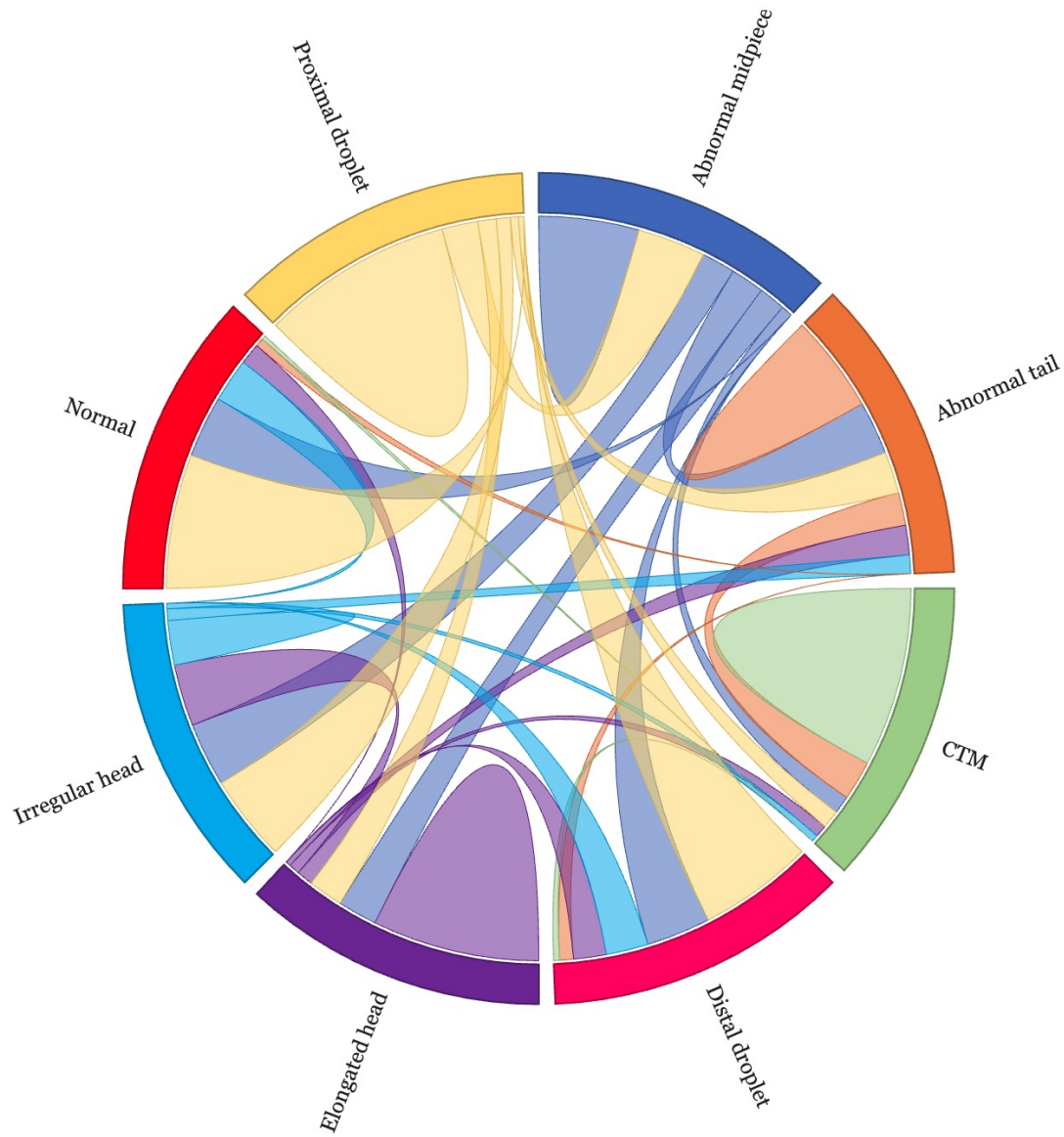


Figure 5-2: Morphological Confusion Web. A chord diagram visualizing the cross-class fuzzy membership flow (from Figure 5-1B) across the Bovine Sperm dataset. The outer arc segments represent the discrete biological classes, while the connecting ribbons represent the volume of shared fuzzy probability mass between them. Ribbon width at the base correlates with the degree of morphological overlap. This visualization highlights biological continuums: for instance, the massive ribbon connecting Twisted or elongated head and Irregular head shape demonstrates that the network perceives these not as strictly distinct categories, but as overlapping phenotypes on a continuous spectrum of head deformation. Conversely, CTM remains highly isolated, sharing minimal probability mass with head abnormalities.

Cross-Class Flow and The Morphological Confusion Web

To visualize directional ambiguity, we create a morphological confusion web (Figure 5-2). Wider ribbons indicate stronger fuzzy overlap between two classes. The significance of the overlap can represent both biological transition states and model’s confusion at inference time.

The most prominent feature is the massive flow connecting Normal, Proximal CD, and Distal CD. Biologically, this represents a maturation cycle. As a sperm cell matures, a cytoplasmic droplet physically migrates from the base of the head (Proximal) down the annulus (Distal), which connects the mid-piece with a principal piece before shedding completely (Normal). A cell captured while the droplet is halfway down the tail represents an intermediate state.

Practical Application via Entropy Gating

This translates into a software feature based on entropy gating. A researcher establishing a clinical baseline can filter for archetypal cells by setting an aggressive entropy threshold (e.g., $u_{normal} > 0.90$), removing slightly deformed cells. Conversely, to identify serious structural deformation we can invert this gate ($0.4 < u_c < 0.6$) to isolate transitional phenotypes caught precisely in the middle of a continuous process.

5.1.4 End-to-End Workflow: RBC Case Study

To validate the workflow of the *Eidocell* framework, we used the RBC dataset to demonstrate an end-to-end workflow from raw instrument output to labeled, exportable populations (referencing the workflow illustrated in Chapter 3, Figure 5-3).

The workflow executes in six distinct phases:

1. **Data Ingestion (Figure 5-3A):** Raw `.cif` or `.hdf5` list-mode files generated by the ImageStream X Mk II are imported, normalizing the multi-channel optical data into memory.
2. **Segmentation (Figure 5-3B):** Because RBCs frequently exhibit uneven illumination (particularly the central pallor of biconcave cells), users apply local

adaptive thresholding to generate highly accurate morphological masks.

3. **Classical Gating (Figure 5-3C):** Before applying computationally heavy deep learning models, basic morphometric features (Area and Circularity) are extracted. The user draws a manual 2D gate to exclude obvious debris, doublets, and platelets, ensuring only single RBCs pass to the next stage.
4. **Latent Clustering (Figure 5-3D):** The filtered cells are passed through the frozen DINOv3 backbone. The resulting 384-dimensional embeddings are reduced via UMAP and automatically grouped using X-Means clustering.
5. **Interactive Refinement (Figure 5-3E, H):** X-Means typically over-segments biological data (e.g., splitting the *Elongated* cells into two slightly different clusters based on rotation). The researcher visually inspects the randomly sampled events from these clusters in the GUI and uses the drag-and-drop interface to merge them into a single, biologically ground-truth *Elongated* class.
6. **Export (Figure 5-3F, G):** The finalized class labels, bounding boxes, and extracted feature vectors are exported into standard `.csv` formats for downstream statistical analysis or directly as cropped images to train external supervised models.

This seamless transition from deterministic classical gating to interactive deep-learning discovery eliminates the need to switch between multiple fragmented software environments, directly addressing the workflow bottleneck inherent to IFC.

5.1.5 Baseline Performance on Corrupted Data

While the static backbone has plausible performance, its performance drops under distribution shifts. Evaluated on the corrupted technical datasets (Table 5.3), accuracy drops by up to 27% in the presence of noise and blur. This significant degradation motivates the need for dynamic representation adaptation.

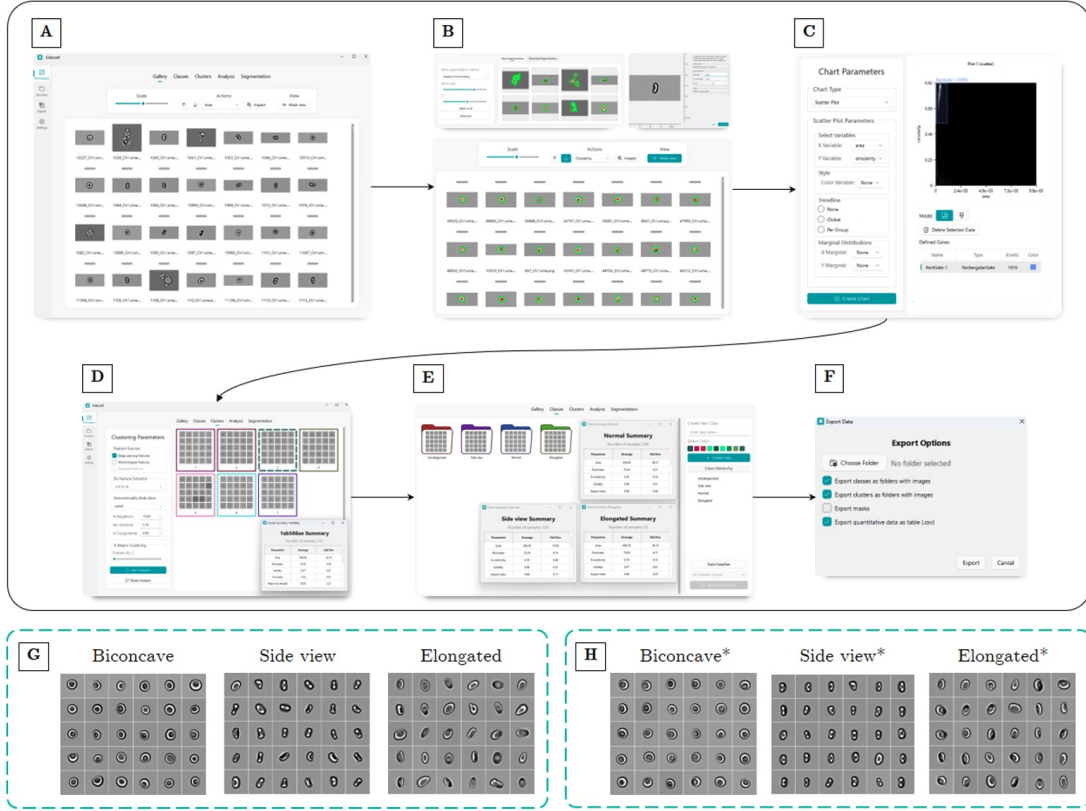


Figure 5-3: Practical workflow for the RBC case study in Eidocell. (A) Import of ImageStream X Mk II data. (B) Segmentation with adaptive thresholding. (C) Gating on area-circularity plot to exclude platelets. (D) Clustering with ViT-S/16 followed by interactive refinement. (E) Creation of target classes from refined clusters. (F) Export of labeled images. (G) Representative ground-truth images. (H) Randomly sampled events from clusters before refinement.

Table 5.3: Baseline Accuracy (%) of the static DINOv3 model under distribution shift.

Dataset	Source (Clean)	Target (Shifted)	Degradation
CIFAR-10	94.2%	71.5% (Avg. Corruption)	-22.7%
CIFAR-100	78.5%	51.2% (Avg. Corruption)	-27.3%
Tiny-ImageNet-C	83.4%	59.3% (Avg. Corruption)	-24.1%

5.2 Phase II: Algorithms for Robust Representation Learning

Even within a continuous IFC experiment, data is subject to micro-shifts: subtle variations in hydrodynamic focus, fluidic pressure, LED illumination, and sample degradation. We evaluate whether our proposed adaptation algorithms can mitigate this.

5.2.1 Validation on Technical Benchmarks

Table 5.4 shows a hierarchy of robustness under severe corruption evaluated in an unsupervised, online setting ($B = 64$). Our Hybrid approach achieves the highest accuracy recovery.

Table 5.4: Comparison of TTA strategies on technical benchmarks under severe distribution shift (Level 5 Corruption). Results are reported as mean classification accuracy (%).

Method	Mechanism	CIFAR-10-C	CIFAR-100-C	Tiny-ImageNet-C
Source Only	Static Weights	71.5	51.2	59.3
IST [53]	Pseudo-Labeling	84.1	61.3	68.7
TEA [54]	Energy Alignment	86.5	63.8	70.4
Hybrid TTA (Ours)	Stat. + Entropy	88.2	66.1	73.1

5.2.2 Biological Domain Adaptation

Applied to our four primary biological case studies (Table 5.5), unsupervised adaptation consistently yields higher classification accuracy compared to the frozen baseline.

Table 5.5: K-NN classification accuracy across biological case studies using Test-Time Adaptation methods. Results report the accuracy of the dynamically adapted embeddings.

Method	Microalgae	Imm. Synapse (BF)	Imm. Synapse (MC)	RBC	Sperm Cells
Baseline	86.12	54.07	78.00	98.28	74.54
IST	87.45	55.60	79.15	98.40	81.20
TEA	88.01	56.12	80.40	98.55	83.50
Hybrid TTA (Ours)	88.93	57.85	81.90	98.81	85.90

The most significant improvement is observed in the Sperm Cells dataset (+11.36%). This dataset inherently contains a massive covariate shift (Fresh vs. Frozen samples). While the static model misclassified subtle *Irregular head shape* phenotypes due to cryo-induced blurred membranes, the Hybrid TTA successfully recalibrated the model’s internal statistics. Even for datasets without specific preparation shifts (Microalgae and Immune Synapse datasets), the Hybrid TTA still provided a +2.8% to +3.9% increase.

5.2.3 Ablation Study and Computational Efficiency

Table 5.6 isolates the components of the Hybrid TTA on the sperm cells dataset.

Table 5.6: Ablation study of the Hybrid TTA components on the Sperm Cells dataset. Combining both alignments yields the highest accuracy while maintaining a practical throughput for interactive analysis.

Config.	Active Components	Accuracy (%)	Inference Speed
A	Source Only (Static Baseline)	74.5	>1000
B	Statistical Alignment Only	80.8	≈ 950
C	Semantic Alignment Only	82.3	≈ 145
D	Hybrid (Stat + Sem)	85.9	≈ 250

Statistical alignment (config B) recovers a substantial portion of accuracy by correcting global illumination shifts with almost no impact on processing throughput (950 imgs/s). Conversely, semantic alignment (config C) is computationally heavy due to entropy backpropagation, dropping throughput to 145 imgs/s.

By combining the two (config D), the hybrid method achieves the highest accuracy. Crucially, executing the statistical alignment first initializes the entropy minimization in a loss landscape closer to a local minimum, requiring fewer gradient update steps per batch. This synergy results in a significantly higher processing throughput (250 imgs/s) compared to pure semantic alignment.

Chapter 6

Discussion

This chapter interprets the experimental findings in the context of our original research objectives. We discuss how the *Eidocell* framework bridges the software workflow gap in single-cell analysis by combining the interactivity of proprietary tools with modern deep learning representations. We evaluate the practical implications of TTA for long-running biological studies and reflect on the biological insights produced by fuzzy clustering. Lastly, we address the current technical limitations of our methodology and software implementation.

6.1 Reducing the Workflow Gap

Eidocell addresses two persistent gaps in IFC analysis: (i) scalable, platform-agnostic exploration of large image collections and (ii) low-overhead generation of biologically meaningful groupings that keep the domain expert “in the loop.” Across the four case studies, self-supervised features coupled with interactive refinement consistently recovered structure aligned with expert annotations, without requiring predefined taxonomies. This workflow is particularly consequential in exploratory scenarios such as bloom dynamics or rare subpopulation discovery in clinical events, where ontologies are incomplete and shift across experiments. Practically, Eidocell provides early, coarse partitions that can be promoted to stable labels only after expert review. This reduces both biases from premature hard labeling and the time spent on manual

selection.

To explicitly contextualize how *Eidocell* resolves the fragmented landscape of bio-image analysis, we summarize its capabilities against current standard platforms in Table 6.1.

Table 6.1: Comparison of software platforms for IFC analysis. *Eidocell* specifically targets the intersection of high-throughput list-mode data and modern, representation-learning methodologies.

Platform	Open Source	IFC Targeted	DL Features	TTA	Interactive Clustering	Fuzzy Logic
Amnis IDEAS®	No	Yes	Partial (Paid)	No	No	No
ImageJ/Fiji	Yes	No	Partial (Plugins)	No	No	No
CellProfiler	Yes	No	Partial (Plugins)	No	No	No
EasyFlow [17]	Yes	Yes	No	No	No	No
Custom Python Scripts	Yes	Yes	Yes	Possible	No (Requires GUI)	Yes
Eidocell (Ours)	Yes	Yes	Yes	Yes	Yes	Yes

Traditional microscopy tools like ImageJ and CellProfiler fundamentally lack native support for the *list-mode* data structures required by flow cytometry (processing a continuous stream of millions of isolated single-cell crops). Conversely, proprietary tools like Amnis IDEAS handle list-mode data efficiently but lock researchers out of the modern deep learning ecosystem, relying instead on rigid, classical feature engineering. While custom Python scripting offers wide algorithmic flexibility, it hinders the interactive, visual human-in-the-loop gating that biologists rely on.

As demonstrated in Table 6.1, *Eidocell* successfully merges the visual interactivity of proprietary cytometric software with the algorithmic flexibility of the open-source Python ecosystem.

6.2 Test-Time Adaptation Impact

A central finding of this thesis is that static deep learning models, even those based on robust self-supervised backbones, are vulnerable to distribution shifts inherent in longitudinal or multi-center biomedical studies. As demonstrated in the Bovine Sperm case study, the transition from fresh to frozen samples introduces global changes in image statistics that degrade the performance of classifiers.

Deploying such models in clinical or veterinary settings without adaptation mecha-

nisms poses a significant risk. We showed that TTA, specifically through the lightweight realignment of BN statistics, offers a practical solution. By dynamically adjusting to the *optical fingerprint* of the current sample stream, the model allows for purely algorithmic mitigation of instrumental drift, reducing the need for constant re-calibration or collection of new labeled datasets for every new experimental condition.

6.3 Theoretical and Empirical Support for Hybrid Adaptation

The design choices underlying the Hybrid TTA algorithm are supported by converging evidence from both the foundation model and TTA literature.

On the feature representation side, recent self-supervised foundation models for hematology have demonstrated that SSL Vision Transformer features are inherently high-quality and transferable. DinoBloom [27] achieved strong zero-shot and few-shot performance across 13 diverse hematological datasets, proving that SSL embeddings capture universal morphological semantics without any task-specific training. RedDino [28] further demonstrated that simple k-nearest neighbor classification on SSL embeddings outperforms supervised baselines specifically trained for RBC morphology assessment. These results validate our choice of a self-supervised ViT backbone (DINOv3) as the feature extractor: the pre-trained embeddings provide a strong baseline that does not require labeled data to achieve competitive performance.

On the adaptation side, the TTA literature provides extensive evidence that entropy minimization and normalization adaptation are effective, complementary strategies. TENT [34] established that updating only the affine parameters of normalization layers via entropy minimization recovers significant accuracy on standard corruption benchmarks (CIFAR-10-C, ImageNet-C) with minimal computational overhead. CoTTA [37] extended this to continual, non-stationary environments, demonstrating that TTA can maintain performance over extended deployment periods—a requirement for long-running IFC experiments. EATA [36] introduced principled mechanisms

for preventing catastrophic forgetting during continuous adaptation, validating the feasibility of sustained online adaptation.

Our Hybrid TTA algorithm joins two approaches: it applies the normalization adaptation and entropy minimization specifically to SSL features, preserving their general-purpose quality while correcting for domain-specific distribution shifts. We hypothesize that TTA-adapted features will achieve the best trade-off between target accuracy and label efficiency, achieving competitive or superior accuracy to supervised fine-tuning without requiring any target labels.

6.4 Biological Implications of Fuzzy Analysis

In the analysis of both algal morphospecies and sperm abnormalities, we observed that many cells fall into transitional states that do not fit strictly into *Type A* or *Type B* categories.

This continuity is explicitly visualized in the Morphological Confusion Web (Figure 5-2). The chord diagram recontextualizes probabilistic confusions as shared probability mass. The thick ribbons connecting categories (e.g. between Twisted or elongated head and Irregular head shape, or Normal cells and Distal CD) highlight populations of cells that visually reside at the boundary between two defined morphologies.

These transitional states are further explored by our quantitative fuzzy membership heatmaps. For instance, the high cosine similarity between Normal sperm cells and CD-carrying cells (Proximal CD, Distal CD) reflects the fact that a cytoplasmic droplet is a highly localized perturbation on an otherwise structurally well-defined cell; the self-supervised network correctly encodes this global structural identity. Moreover, the cross-class membership flow reveals distinct asymmetrical relationships. For example, Normal cells frequently distribute partial membership mass into the Abnormal midpiece cluster, but the reverse is much weaker. This mathematical asymmetry might mirror a hierarchical biological degradation pathway, capturing the continuous onset of cryo-damage where the midpiece acts as a broad, high-variance indicator of

early structural damage.

The application of fuzzy clustering and soft-labeling approaches in Eidocell reveals that these transitional states might represent biologically meaningful information (e.g. gradual morphological degradation during osmotic stress or the intermediate stages of binary fission in cyanobacteria). By preserving these continuous relationships in the feature space rather than forcing hard decisions, we provide experts with a more accurate reflection of the underlying physiological heterogeneity.

6.5 Limitations

Despite these advances, the limitations of our current work remain explicit:

- (i) **Performance and handling of large databases:** The software experiences UI stutters with experiments containing hundreds of thousands of events due to threading and memory complications. In-memory processing of large databases and session caching remain complex software challenges.
- (ii) **Features and data processing native to IFC analysis:** IFC experiments often involve joint analysis of brightfield, darkfield, and fluorescence channels. While supported, the current feature extraction pipeline is optimized for morphology (brightfield/darkfield) and handles fluorescence naively, limiting the implementation of complex co-localization assays.
- (iii) **Data complexity:** Multi-channel data and various platforms create an inherently difficult data domain. Feature projection in combination with X-means is a pragmatic approach, but it is sensitive to parameter tuning. We essentially resort to conventional gating for initial filtering (doublets/debris detection) to achieve meaningful downstream clustering.
- (iv) **TTA hyperparameter sensitivity:** The Hybrid TTA algorithm introduces two key hyperparameters: the momentum coefficient α for statistical alignment and the learning rate η for entropy minimization. Optimal values depend on the

magnitude and nature of the domain shift, and no single configuration is universally optimal across all datasets. Under extreme distribution shifts that exceed the representational capacity of affine-only adaptation (e.g., drastic changes in imaging modality), the method may provide insufficient correction. Future work should explore adaptive hyperparameter schedules and shift-magnitude estimation to automate this tuning.

Chapter 7

Conclusion

This chapter concludes the thesis by synthesizing the core research efforts and summarizing the major technical and biological contributions. We review how the integration of self-supervised representation learning, test-time adaptation, and interactive fuzzy clustering addresses the current bottlenecks in Imaging Flow Cytometry. Finally, we outline promising directions for future research and the continued development of the *Eidocell* platform.

7.1 Summary

We presented Eidocell, an open-source platform designed for IFC data analysis. It combines a user-friendly GUI with self-supervised feature extraction and interactive clustering, lowering technical barriers while keeping experts in control. The software provides an end-to-end workflow that spans from segmentation to exploratory clustering and quantitative visualization, within a single, instrument-agnostic environment.

Through evaluation on four diverse datasets covering immunology, veterinary science, and environmental science, our workflow recovered biologically meaningful structures while demonstrating competitive performance in clustering and classification. Our key technical contributions include:

- The integration of a general-purpose ViT-S/16 (DINOv3) that generalizes well across IFC domains without fine-tuning.

- The implementation of a Hybrid TTA algorithm that mitigates instrument drift and recovers up to 5-10% classification accuracy on biological datasets.
- The application of Fuzzy Clustering and *Entropy Gating*, which successfully quantifies and visualizes continuous biological transitions that are traditionally obscured by hard-clustering algorithms.

7.2 Future Work

Eidocell is currently released as a functional prototype. Future development will focus on scaling performance for very large datasets (>1 million events) by implementing optimized data loaders and caching mechanisms. We also aim to expand support for multi-channel IFC data, enabling complex assays such as internalization and nuclear localization scores.

Furthermore, we plan to explore weakly supervised segmentation approaches, where the system learns to segment novel cell types from coarse user scribbles or point annotations, further reducing the manual burden on researchers. Ultimately, we aim to provide a transparent, extensible platform that serves as a broadly useful tool for single-cell imaging studies in basic research, ecology, and clinical diagnostics.

Bibliography

- [1] Natasha S Barteneva, Elizaveta Fasler-Kan, and Ivan A Vorobjev. Imaging flow cytometry: coping with heterogeneity in biological systems. *Journal of Histochemistry & Cytochemistry*, 60(10):723–733, 2012.
- [2] Natasha S Barteneva and Ivan A Vorobjev, editors. *Imaging Flow Cytometry: Methods and Protocols*, volume 1389 of *Methods in Molecular Biology*. Springer, 2016.
- [3] Kevin Smith, Filippo Piccinini, Tamas Balassa, Krisztian Koos, Tivadar Danko, Hossein Azizpour, and Peter Horvath. Phenotypic image analysis software tools for exploring cellular biology and drug discovery. *Cell systems*, 6(6):636–653, 2018.
- [4] Robert Haase, Elnaz Fazeli, David Legland, Michael Doube, Si’an Culley, Ilya Belevich, Eija Jokitalo, Martin Schorb, Anna Klemm, and Christian Tischer. A hitchhiker’s guide through the bio-image analysis software universe. *FEBS letters*, 596(19):2472–2485, 2022.
- [5] David A Basiji, William E Ortyn, L Liang, V Venkatachalam, and P Morrissey. Cellular image analysis and imaging by flow cytometry. *Clinics in laboratory medicine*, 27(3):653–670, 2007.
- [6] Sayedali Shetab Boushehri, Katharina Essig, Nikolaos-Kosmas Chlis, Sylvia Herter, Marina Bacac, Fabian J Theis, Elke Glasmacher, Carsten Marr, and Fabian Schmich. Explainable machine learning for profiling the immunological

synapse and functional characterization of therapeutic antibodies. *Nature communications*, 14(1):7888, 2023.

- [7] Veronika Dashkova, Dmitry Malashenkov, Nicole Poulton, Ivan Vorobjev, and Natasha S Barteneva. Imaging flow cytometry for phytoplankton analysis. *Methods*, 112:188–200, 2017.
- [8] Paul Rees, Huw D Summers, Andrew Filby, Anne E Carpenter, and Minh Doan. Imaging flow cytometry. *Nature Reviews Methods Primers*, 2(1):86, 2022.
- [9] 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 biology*, 7(10):1–11, 2006.
- [10] Holger Hennig, Paul Rees, Thomas Blasi, Lee Kametsky, Jane Hung, David Dao, Anne E Carpenter, and Andrew Filby. An open-source solution for advanced imaging flow cytometry data analysis using machine learning. *Methods*, 112:201–210, 2017.
- [11] Philipp Eulenberg, Niklas Köhler, Thomas Blasi, Andrew Filby, Anne E Carpenter, Paul Rees, Fabian J Theis, and F Alexander Wolf. Reconstructing cell cycle and disease progression using deep learning. *Nature Communications*, 8(1):463, 2017.
- [12] Anel Umirbaeva, Andrey Kurenkov, Bolat Seisenov, Kanat Zhumanov, Kadyrbay Tajiyev, Mirzhan Mustafin, Ivan A Vorobjev, and Natasha S Barteneva. Deep learning-enabled sperm morphology analysis of bovine sperm for label-free imaging flow cytometry. *bioRxiv*, 2025. Preprint.
- [13] Andrey Kurenkov, Aigul Kussanova, and Natasha S Barteneva. Advancing precision single-cell analysis of red blood cells through semi-supervised deep learning

using database of patients with post-COVID-19 syndrome. In *Imaging, Manipulation, and Analysis of Biomolecules, Cells, and Tissues XXII*, volume 12846, page 1284602. SPIE, 2024.

- [14] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Rammamondjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025.
- [15] Thouis R Jones, In Han Kang, Douglas B Wheeler, Robert A Lindquist, Adam Papallo, David M Sabatini, Polina Golland, and Anne E Carpenter. Cellprofiler analyst: data exploration and analysis software for complex image-based screens. *BMC bioinformatics*, 9:1–16, 2008.
- [16] Stuart Berg, Dominik Kutra, Thorben Kroeger, Christoph N Straehle, Bernhard X Kausler, Carsten Haubold, Martin Schiegg, Janez Ales, Thorsten Beier, Markus Rudy, et al. ilastik: interactive machine learning for (bio) image analysis. *Nature methods*, 16(12):1226–1232, 2019.
- [17] Ye Ma, Inbal Eizenberg-Magar, and Yaron Antebi. EasyFlow: An open-source, user-friendly cytometry analyzer with graphic user interface (GUI). *PLOS ONE*, 19(11):e0308873, 2024.
- [18] Thomas Blasi, Holger Hennig, Huw D Summers, Fabian J Theis, Joana Cerveira, James O Patterson, Derek Davies, Andrew Filby, Anne E Carpenter, and Paul Rees. Label-free cell cycle analysis for high-throughput imaging flow cytometry. *Nature Communications*, 7:10256, 2016.
- [19] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

- [20] Maxim Lippeveld, Carly Knill, Emma Ladlow, Andrew Fuller, Louise J Michaelis, Yvan Saeys, Andrew Filby, and Daniel Peralta. Classification of human white blood cells using machine learning for stain-free imaging flow cytometry. *Cytometry Part A*, 97(3):308–319, 2020.
- [21] Veera A Timonen, Erja Kerkelä, Ulla Impola, Leena Penna, Jukka Partanen, Outi Kilpivaara, Mikko Arvas, and Esa Pitkänen. DeepIFC: Virtual fluorescent labeling of blood cells in imaging flow cytometry data with deep learning. *Cytometry Part A*, 103(10):807–817, 2023.
- [22] Minh Doan, Marian Case, Dino Masic, Holger Hennig, Claire McQuin, Juan Caicedo, Shantanu Singh, Allen Goodman, Olaf Wolkenhauer, Huw D Summers, David Jamieson, Frederik W Van Delft, Andrew Filby, Anne E Carpenter, Paul Rees, and Julie Irving. Label-free leukemia monitoring by computer vision. *Cytometry Part A*, 97(4):407–414, 2020.
- [23] Caicedo JC et al. Doan M, Sebastian JA. Objective assessment of stored blood quality by deep learning. *Proceedings of the National Academy of Sciences*, 117(35):21381–21390, 2020.
- [24] Shaobo Luo, Khanh T Nguyen, Bich T T Nguyen, Shaobin Feng, Yinning Shi, Ahmed Elsayed, Yanlong Zhang, Xinnan Zhou, Bo Wen, Giovanni Chierchia, Hugues Talbot, Tarik Bourouina, Xudong Jiang, and Ai Qun Liu. Deep learning-enabled imaging flow cytometry for high-speed *Cryptosporidium* and *Giardia* detection. *Cytometry Part A*, 99(11):1123–1133, 2021.
- [25] Kerem Delikoyun, Qianyu Chen, Liu Wei, Si Ko Myo, Johannes Krell, Martin Schlegel, Win Sen Kuan, John Tshon Yit Soong, Gerhard Schneider, et al. Real-time deep learning phase imaging flow cytometer reveals blood cell aggregate biomarkers for haematology diagnostics. *arXiv preprint*, 2024.
- [26] Juyeon Park, Bijie Bai, DongHun Ryu, Tairan Liu, Chungha Lee, Yi Luo, Mahn Jae Lee, Luzhe Huang, Jeongwon Shin, Yijie Zhang, Dongmin Ryu, Yuzhu

- Li, Geon Kim, Hyun-Seok Min, Aydogan Ozcan, and YongKeun Park. Artificial intelligence-enabled quantitative phase imaging methods for life sciences. *Nature Methods*, 20(11):1645–1660, 2023.
- [27] Valentin Koch, Sophia J Wagner, Salome Kazeminia, Ece Sancar, Matthias Hehr, Julia A Schnabel, Tingying Peng, and Carsten Marr. DinoBloom: A foundation model for generalizable cell embeddings in hematology. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2024*, pages 520–530. Springer, 2024.
- [28] Luca Zedda, Andrea Loddo, Cecilia Di Ruberto, and Carsten Marr. Reddino: A foundation model for red blood cell analysis. In *Lecture Notes in Computer Science*, volume 15963, pages 445–455. Springer, 2026.
- [29] Zehao Xiao and Cees G M Snoek. Beyond model adaptation at test time: A survey. *arXiv preprint arXiv:2411.03687*, 2024.
- [30] Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, 133(1):31–64, 2025.
- [31] Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. *arXiv preprint arXiv:1603.04779*, 2016.
- [32] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning*, pages 9229–9248. PMLR, 2020.
- [33] Yuejiang Liu, Parth Kothari, Bastien van Delft, Baptiste Bellot-Gurlet, Taylor Mordan, and Alexandre Alahi. Ttt++: When does self-supervised test-time training fail or thrive? In *Advances in Neural Information Processing Systems*, volume 34, pages 21808–21820, 2021.

- [34] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.
- [35] Marvin Zhang, Sergey Levine, and Chelsea Finn. Test time robustness via adaptation and augmentation. In *Advances in Neural Information Processing Systems*, volume 35, pages 14651–14664, 2022.
- [36] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International Conference on Machine Learning*, pages 16888–16905. PMLR, 2022.
- [37] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7201–7211, 2022.
- [38] Junha Song, Jungsoo Lee, In So Kweon, and Sungha Choi. Ecotta: Memory-efficient continual test-time adaptation via self-distilled regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11920–11929, 2023.
- [39] Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15922–15932, 2023.
- [40] Jonghyun Lee, Dahuin Jung, Saehyung Lee, Junsung Park, Juhyeon Shin, Uiwon Hwang, and Sungroh Yoon. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In *International Conference on Learning Representations (ICLR)*, 2024.
- [41] Mihir Prabhudesai, Tsung-Wei Ke, Alexander C Li, Deepak Pathak, and Katerina Fragkiadaki. Test-time adaptation of discriminative models via diffusion

- generative feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023.
- [42] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 14274–14289, 2022.
- [43] Chun-Mei Feng, Kai Yu, Yilu Liu, Salman Khan, and Wangmeng Zuo. Diverse data augmentation with diffusions for effective test-time prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2704–2714, 2023.
- [44] Zhihao Zeng, Wuyang Bao, Xingxing Lin, Rongpeng Qiu, Tianfu Wei, Xin Ning, Yixuan Yan, Chunyan Luo, M. X. Cheng, Jiyang He, and Hanghang Tong. Subspace alignment for vision-language model test-time adaptation. *arXiv preprint arXiv:2601.08139*, 2026.
- [45] Matteo Farina, Gianni Franchi, Giovanni Iacca, Massimiliano Mancini, and Elisa Ricci. Frustratingly easy test-time adaptation of vision-language models. *arXiv preprint arXiv:2405.18330*, 2024.
- [46] Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. In *Advances in Neural Information Processing Systems*, volume 33, pages 11539–11551, 2020.
- [47] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.
- [48] Dan Pelleg, Andrew W Moore, et al. X-means: Extending k-means with efficient estimation of the number of clusters. In *ICML*, volume 1, pages 727–734, 2000.

- [49] Adina Zhumakhanova, Yersultan Mirasbekov, Ayagoz Meirkhanova, Dmitry V Malashenkov, Thomas A Davidson, Eti Ester Levi, Erik Jeppesen, and Natasha S Barteneva. From colonial clusters to colonial sheaths: Imaging flow cytometry analysis of *Microcystis* morphospecies dynamics in mesocosm and links to CyanoHABs management. *Ecological Indicators*, 163:112100, 2024.
- [50] Ayagoz Meirkhanova, Adina Zhumakhanova, Polina Len, Christian Schoenbach, Eti Ester Levi, Erik Jeppesen, Thomas A Davidson, and Natasha S Barteneva. Heatwave-induced thermal stratification shaping microbial-algal communities under different climate scenarios as revealed by long-read sequencing and imaging flow cytometry. *Toxins*, 17(8):370, 2025.
- [51] Dmitry V Malashenkov, Lajos Vörös, Aigerim Duisen, Veronika Dashkova, Aidana Abilkas, Ivan A Vorobjev, and Natasha S Barteneva. Assessing Aral Sea residual lake system: impact of fluctuating salinity on phytoplankton communities. *bioRxiv*, 2024. Preprint.
- [52] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.
- [53] Jing Ma. Improved self-training for test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23701–23710, 2024.
- [54] Yige Yuan, Bingbing Xu, Liang Hou, Fei Sun, Huawei Shen, and Xueqi Cheng. TEA: Test-time energy adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23901–23911, 2024.