Towards a Large-Scale Unbiased Machine Learning Benchmark for Cell Instance Segmentation: Final Report for CPSC 537 Intro to Database Systems

Chen Liu
Yale University
chen.liu.cl2482@yale.edu

Danqi Liao
Yale University
danqi.liao@yale.edu

Shuangge Wang
Yale University
shuangge.wang@yale.edu

Abstract

Our objective is to develop a comprehensive quantitative benchmark designed to impartially assess deep learning techniques using open cell segmentation datasets. Our goal is to establish a standard similar to "CIFAR" or "ImageNet" in the realms of histology and cellular biology. So far, we have examined seven datasets, with a range of 30 to 7,000 images and encompassing between 7,000 to 1.2 million cells. Two of the largest datasets have been integrated into our benchmark. We have evaluated ten deep learning methods, selecting two for their ease of use in inference processes. We plan to further refine and expand this project and will ultimately launch a website to facilitate widespread access and community involvement.

1 Introduction

There is a significant lack of publicly available datasets and common tasks in cell image segmentation to benchmark existing or new methods, unlike fields such as natural image classification (consider MNIST [1], CIFAR [2], or ImageNet [3]). As a result, it is difficult for researchers to run extensive comparisons and/or to build upon the work of others, ultimately impeding the progress in this field.

To encourage open collaboration and ensure transparency and reproducibility, we propose an online system that brings together a list of publicly available datasets to form a large-scale quantitative evaluation benchmark. Our system allows researchers or developers to register and submit their models for various tasks. It also hosts a leaderboard that showcases the performance of different models on these open datasets. Users will be able to browse and filter images for visualization and can also compare different models based on tasks, metric, and performances. Our system also encourages user contribution by uploading new methods that users develop for evaluation and benchmarking.

We envision that our benchmark can be valuable to people from various backgrounds.

1. **For cellular biologists and similar scientists**, our project aims to provide a unified platform to help them find the best cell segmentation models for their research projects, according to the properties of their data, and to help them easily install and deploy the models.
2. **For researchers who want to follow the latest developments**, we provide an extensive and fair comparison of the methods on a variety of datasets.
3. **For researchers developing cell segmentation models**, we provide a platform to compare their models fairly with existing ones and promote their work. In the long run, we hope

Preprint. Under review.

that all new research on cell segmentation will use our platform for a transparent and comprehensive evaluation of their model.

4. **For beginners interested in this field**, we provide a clear and friendly introduction, showcasing the state-of-the-art models and pointing them to background and tutorials.

For this course project, we further limit the scope and focus on the following scenarios.

1. We focus on **instance segmentation** as we believe it is more important and influential than *semantic segmentation* for the context of cell images.
2. We exclusively evaluate the **generalization ability** of the models. To that end, we gather models whose *pretrained weights are available* on public repositories and evaluate them on the same set of *previously unseen* dataset that we collect.
3. We manually run the evaluation without hosting an automated service for that purpose.

2 Architecture

2.1 Database

In the following section, we describe the tables, relationships, and constraints of our database.

2.1.1 Tables

Our current design incorporates 7 tables in the database, as listed below.

The following visualization techniques are used for better illustration.

1. The keys and corresponding data types are specified.
2. Tables with names colored gray are weak entities.
3. For clarity, the referenced foreign keys are color-coded. Their corresponding primary keys are indicated with matching colors.

cell_image

key	dtype
ID	string
cell_type	string
imaging_modality	string
instance_seg	bool
task_ID	string
image_URL	string
label_URL	string

task

key	dtype
ID	string
name	string
num_images	int
description	string

model

key	dtype
ID	string
developer_ID	string
name	string
backbone	string
learning_method	string
num_params	int
paper_URL	string
code_URL	string
upload_time	date

prediction

key	dtype
model_ID	string
task_ID	string
cell_image_ID	string
prediction_URL	string

performance

key	dtype
model_ID	string
task_ID	string
metric_name	string
score	float

developer

key	dtype
ID	string
email	string
name	string
pass_hash	string

metric

key	dtype
name	string
description	string

2.1.2 Relationships

1. ***cell_image*** vs. ***task*** is a many-to-many relationship, with '*task_ID*' being the foreign key in ***cell_image***.

2. ***task*** vs. ***performance*** is a one-to-many relationship, with '*task_ID*' being the foreign key in ***performance***.
3. ***model*** vs. ***performance*** is a one-to-many relationship, with '*model_ID*' being the foreign key in ***performance***.
4. ***metric*** vs. ***performance*** is a one-to-many relationship, with '*metric_name*' being the foregin key in ***performance***.
5. ***model*** vs. ***developer*** is a many-to-one relationship, with '*developer_ID*' being the foreign key in ***model***.

2.1.3 Constraints

1. Every '*ID*' in each table should be unique.
2. '*name*' in the ***developer*** table cannot be null.
3. '*pass_hash*' in the ***developer*** table should have a length constraint (more than a certain number) to ensure a hashed security password.
4. '*num_images*' in the ***task*** table should be checked to make sure it is larger than some threshold number.

Finally, the ER digram is shown in Figure 1. It specifies the entity sets, the relationships, whether the entities are strong or weak entities, the participation, etc.

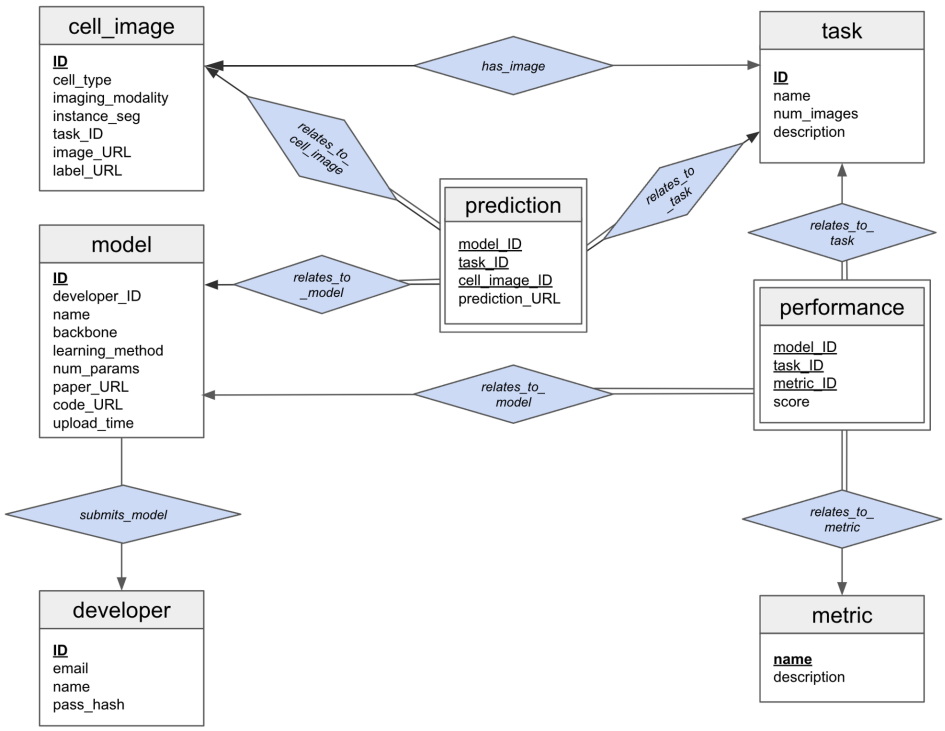

Figure 1: **The ER diagram.** Underlined keys are primary keys. Double-boxes indicate weak entities. Single-ended arrows (→) indicate one-to-many relationships, pointing from the "many" side to the "one" side. Double-ended arrows (↔) indicate many-to-many relationships. Double-line arrows (⇒) indicate total participation.

2.2 Tech Stack

The architecture of our project is as follows.

For the **backend**, we used Python. Python is a good choice because of our team's experience and its many libraries for machine learning and data processing. We used a micro-web framework, Flask, to listen on frontend calls before serving data from backend to frontend. Flask is a lightweight, easy-to-use Python plugin, especially if we are using a SQL-based database. We also used PostgreSQL as our DBMS for storing cell images, task and metrics information.

For the **frontend**, we used HTML, CSS and JavaScript, which could integrate with the React framework. We believe this is a good choice given how well documented the framework is and our team's experience with react development. We will require an SQL database for managing user info, cell images, tasks, models, and other information needed.

We use Git for version control and team collaboration. We used localhost to host our website. We have not yet deployed the website through any third-party hosting service.

3 Key Features

3.1 Leaderboard Page

This is a page where users compare how each model performs in each metric. Data are transcribed from numeric to visual information to make the comparison easier. We also normalized the score for each metric to make the visualization more appealing. Users can sort the dataset of models based on specific metric name (Figure 2).

By far, on the dataset side, we have included two datasets with instance segmentation ground truth labels, both of which are among the datasets with the most number of images. The datasets are described in detail in Section 4.1.

On the model side, we have incorporated two machine learning methods with three pre-trained checkpoints in total. The methods are described in detail in Section 4.2.

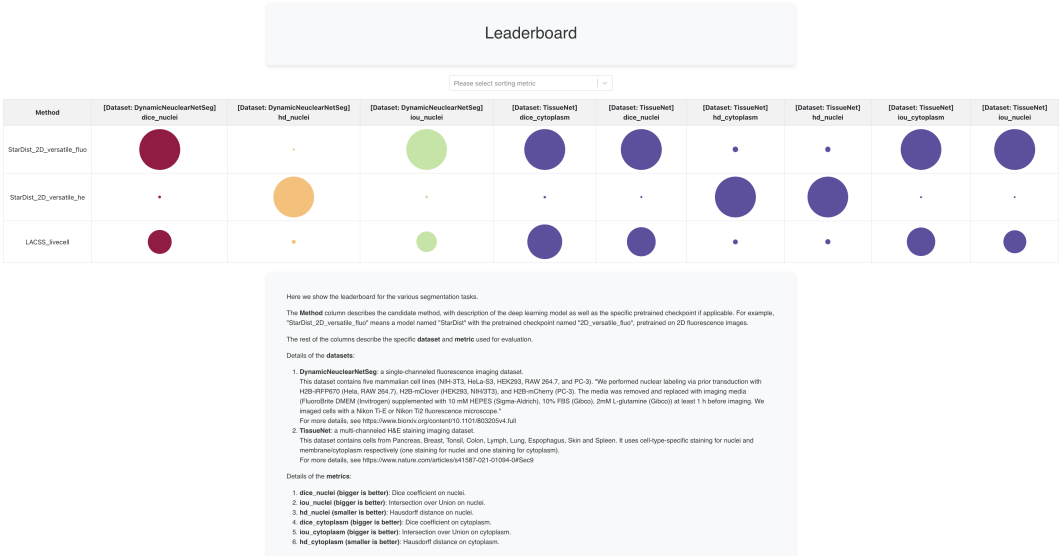

Figure 2: **The leaderboard Page** shows which methods are performing well on which tasks in a visually intuitive manner. Developers can showcase their methods by competing and outperforming others. Adopters can find their go-to methods to try on their tasks. Researchers can study model generalization as well as other intriguing topics.

3.2 Cell Page

In the database, each cell image is tagged by the corresponding cell type. Some examples are: lung cells, pancreas cells, lymph node cells, HeLa cells, among many others. Users can query which cell

type they want to display on the page. Users can also slide a counter to adjust the number of images displayed on the web page (Figure 3).

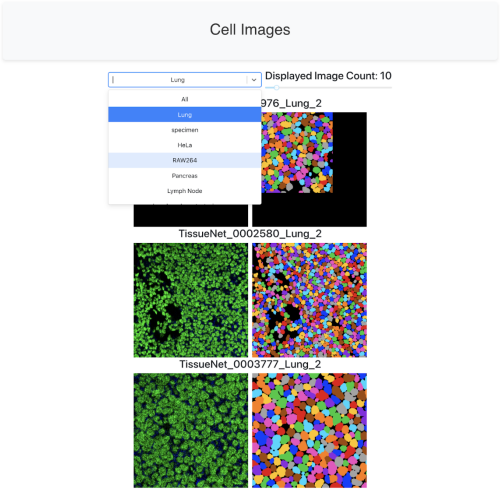

Figure 3: **The Cell Page** helps the users better understand the data, grouped by different cell types (shown in the drop-down menu).

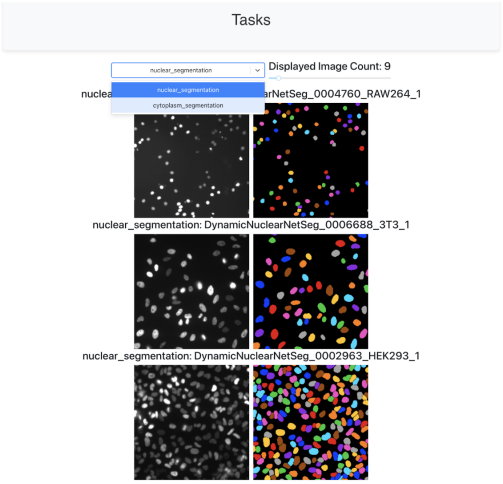

Figure 4: **The Task Page** helps the users better understand the data, grouped by different tasks (shown in the drop-down menu).

3.3 Task Page

Similarly to the cell page, on the task page users can query which task they want to display. Users can also slide a counter to adjust the number of images displayed on the web page (Figure 4). By far, we have included two tasks: cell (instance) segmentation and nuclei (instance) segmentation. The latter only segments the nucleus of each cell, whereas the former segments the cytoplasm, i.e., the border of the cell membrane.

Users can select a set of models and ground truth. The database, by performing a 'join' operation on cell images and tasks, generates a table of different models that infer the same image. This functionality gives the user a direct comparison of how different models perform on the same image.

3.4 Interactive Comparison

Users can choose multiple methods for side-by-side comparison with ground truth segmentation maps. Visualization results are shown in Figure 5. The side-by-side comparison is a helpful visualization for the users to qualitatively examine the selected methods on selected images.

As mentioned previously, the datasets and models included are further described in Section 4.1 and Section 4.2, respectively.

3.5 Participation Page

Currently, we envision the following manners for users to participate:

1. They may download the models and/or datasets and try them out on their own.
2. They may upload their own datasets to the database.
3. They may upload their method to the database.

As a starting point, we would like to host the website in a Kaggle-like manner. For researchers who want to set up their method to compete against existing ones, they shall download the datasets (through a link on our website redirecting them to the official data providers) and upload their inference results to us. We will then evaluate the performance and update the pages, including but not limited to the leaderboard.

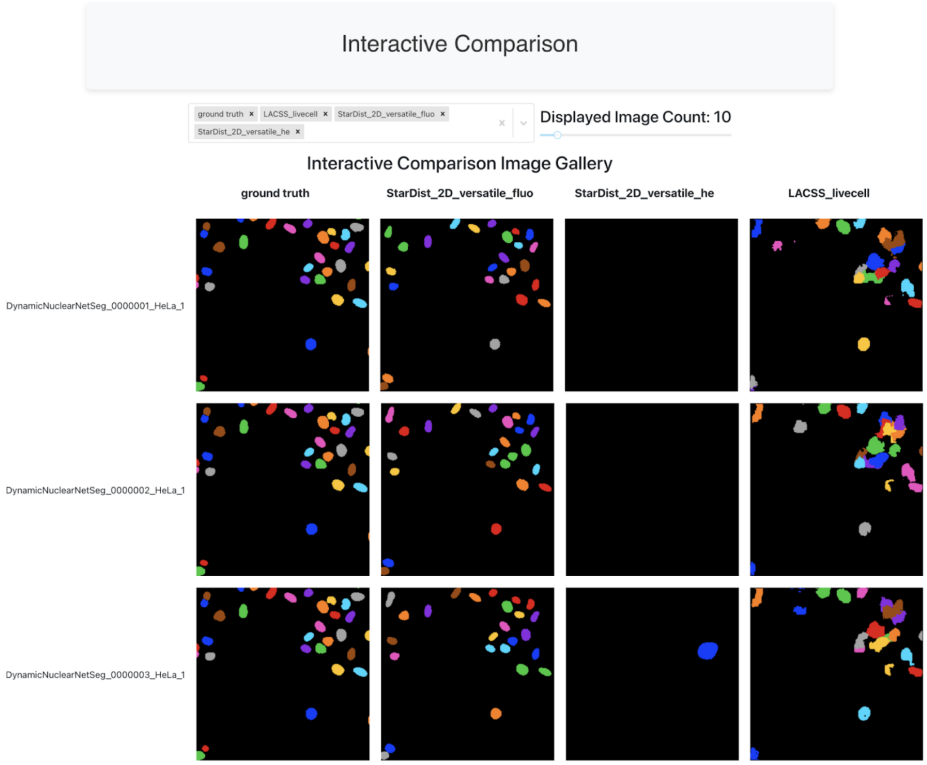

Figure 5: **The Interactive Comparison Page** provides a venue for qualitative comparison of selected methods on selected cell images.

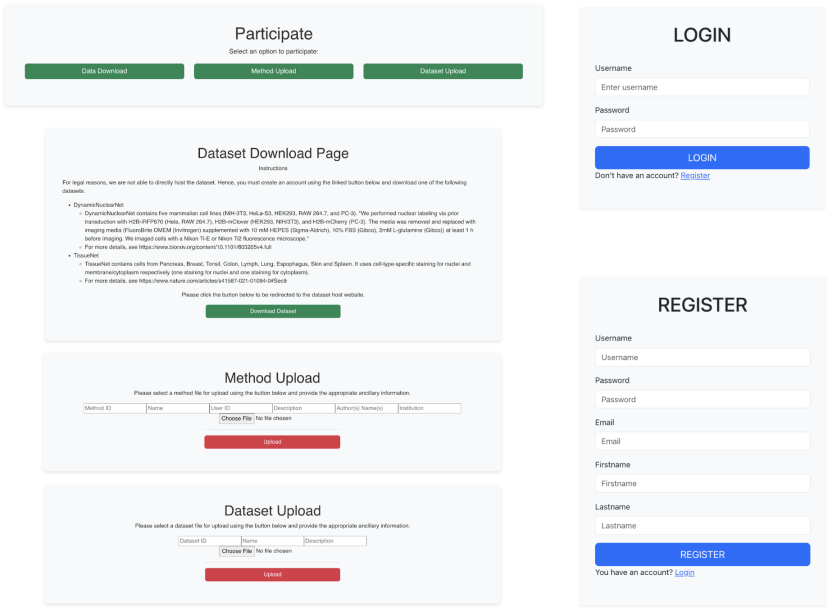

Figure 6: T**he Participation Page** allows the users to register and further utilize the models and datasets hosted.

The registration and login process on our website are shown in Figure 6.

4 Description of Data

4.1 Datasets benchmarked

We searched for publicly available datasets for cell segmentation and further narrowed it down to two datasets with the most number of images. The complete list of candidates can be found in Table 1.

DynamicNuclearNet contains five mammalian cell lines (NIH-3T3, HeLa-S3, HEK293, RAW 264.7, and PC-3). "We performed nuclear labeling via prior transduction with H2B-iRFP670 (Hela, RAW 264.7), H2B-mClover (HEK293, NIH/3T3), and H2B-mCherry (PC-3). The medium was removed and replaced with imaging medium (FluoroBrite DMEM (Invitrogen) supplemented with 10 mM HEPES (Sigma-Aldrich), 10% FBS (Gibco), 2 mM L-glutamine (Gibco)) at least 1 hour before imaging. We imaged cells with a Nikon Ti-E or Nikon Ti2 fluorescence microscope."

TissueNet contains cells from Pancreas, Breast, Tonsil, Colon, Lymph, Lung, Espophagus, Skin and Spleen. It uses cell-type-specific staining for nuclei and membrane/cytoplasm respectively (one staining for nuclei and one staining for cytoplasm).

Table 1: Datasets we explored for the benchmark. The ones included in this course project are **bolded**.

Dataset Name	# images	Image size	# cells	Staining	Pub. year	Pub. venue	Link
CryoNuSeg [4]	30	256×256	7,596	H&E	2023	Computers in Biology and Medicine	https://www.sciencedirect.com/science/article/pii/S0010482521001438
DigestPath [5]	682	2000×2000	14,859	H&E	2022	Medical Image Analysis	https://www.sciencedirect.com/science/article/pii/S1361841522001323
DynamicNuclearNet Segmentation [6]	**7,084**	**512×512**	**606,455**	**Fluorescence**	**2023**	**arXiv**	https://www.biorxiv.org/content/10.1101/803205v4.full
EVICAN [7]	4600	1024×1024	26,000	CellMask/DAPI	2020	Bioinformatics	https://academic.oup.com/bioinformatics/article/36/12/3863/5814923
LIVECell [8]	5,239	1408×1040	1.6 M	stain-free	2021	Nature Methods	https://www.nature.com/articles/s41592-021-01249-6
NuInsSeg [9]	665	256×256	30,698	H&E	2023	arXiv	https://arxiv.org/abs/2308.01760
TissueNet [10]	**7,022**	**512×512**	**1.2 M**	**H&E**	**2021**	**Nature Biotechnology**	https://www.nature.com/articles/s41587-021-01094-0

4.2 Models evaluated

We searched for publicly available models for cell segmentation and further narrowed it down to two models that provided pretrained weights. The complete list of candidates can be found in Table 2.

The models we included are listed below.

1. **LACSS** with its publicly accessible weights pretrained on the 'LiveCell' dataset.
2. **StarDist** with 2 sets of pretrained weights, trained on '2D fluorescent images' and '2D H&E images' respectively.

For this project, we incorporated a total of **three candidate models** if we count models with different weights as distinct candidate models.

4.3 Evaluation metrics

We used the following quantitative metrics to evaluate the performance of the models.

Dice Coefficient measures the similarity of two binary maps. It ranges from 0 to 1, with 0 being the worst and 1 being the best.

$$Dice(X, Y) = \frac{2|X \cap Y|}{|X| + |Y|}$$

Table 2: Models we explored for the benchmark. The ones included in this course project are **bolded**.

Model Name	Task	Pub. year	Pub. venue	Weights?	Pub. Link	Code Link
2DCellSeg [11]	instance seg.	2018	BMC Bioinformatics	×	https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-018-2375-z	Not Found
3DCellSeg [12]	instance seg.	2022	Scientific Reports	✓	https://www.nature.com/articles/s41598-021-04048-3	https://github.com/AntonotnaWang/3DCellSeg
CellPose [13]	instance seg.	2020	Nature Methods	✓	https://www.nature.com/articles/s41592-020-01018-x	https://github.com/mouseland/cellpose
CellSeg [14]	semantic seg.	2022	BMC Bioinformatics	✓	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8767664/	https://michaellee1.github.io/CellSegSite/index.html
CUTS [15]	semantic seg.	2024	MICCAI	×	https://arxiv.org/abs/2209.11359	https://github.com/ChenLiu-1996/CUTS
DeepCeNS [16]	instance seg.	2021	IJCNN	×	https://ieeexplore.ieee.org/document/9533624	Not Found
DiffKillR [17]	instance seg.	2024	arXiv	×	https://arxiv.org/abs/2410.03058	https://github.com/KrishnaswamyLab/DiffKillR
Edge Enhancement [18]	semantic seg.	2022	IEEE EMBC	×	https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9871026	https://github.com/SAIL-GuoLab/Cell_Segmentation_and_Tracking
LACSS [19]	**instance seg.**	**2023**	**Communications Biology**	✓	https://www.nature.com/articles/s42003-023-04608-5	https://github.com/jiyuuchc/lacss
StarDist [20]	**instance seg.**	**2018**	**MICCAI**	✓	https://arxiv.org/abs/1806.03535	https://github.com/stardist/stardist
WSISPDR [21]	instance seg.	2019	MICCAI	✓	https://arxiv.org/pdf/1911.13077.pdf	https://github.com/naivete5656/WSISPDR

Intersection over Union (IoU) is defined very similarly to the Dice coefficient, with the same dynamic range.

$$IoU(X, Y) = \frac{|X \cap Y|}{|X \cup Y|}$$

Hausdorff Distance (HD) measures the distance between two binary maps. It ranges from 0 to infinity — though technically it is capped by the size of the image. Unlike the other two metrics, for HD lower is better.

$$HD(X, Y) = \max\left\{\sup_{x \in X} d(x, Y), \sup_{y \in Y} d(X, y)\right\},$$

where d is any distance measure, typically the L2 distance.

Since these metrics are designed for binary segmentation, we **made the following adaptations** to meaningfully evaluate multi-cell instance segmentation.

For each image, we iterate over all ground-truth cell masks, and for each cell mask, we compute these metrics with each individual cell mask from the prediction (generated by a candidate model). Among all (x, y) pairs, we choose the best score and take the average across all masks. Formally, we define **mean Dice coefficient**, **mean IoU** and **mean HD** as follows.

$$mDice(X, Y) = \frac{1}{\sum_{y \in Y} 1} \sum_{y \in Y} \left\{\max_{x \in X} Dice(x, y)\right\}$$

$$mIoU(X, Y) = \frac{1}{\sum_{y \in Y} 1} \sum_{y \in Y} \left\{\max_{x \in X} IoU(x, y)\right\}$$

$$mHD(X, Y) = \frac{1}{\sum_{y \in Y} 1} \sum_{y \in Y} \left\{\min_{x \in X} HD(x, y)\right\}$$

5 Technical Challenges

For interactive comparison, we were initially having trouble maintaining a consistently appealing visualization because the range of the metrics are so different cross metrics. We then rescaled Hausdorff Distance within each metric so they are visually comparable.

Some of the database operations are too complicated to operate with a single-shot query. Therefore, for some queries, e.g., the interactive comparison page, we decompose the large query to few subqueries to leverage the flexibility of Python. We also delegated some trivial operations like sorting to the frontend to reduce communication and query overhead.

We did not sanitize user inputs so there might be vulnerability for SQL injection.

We initially did not include prediction table which later was added because we wanted to interactively compare prediction results by different models.

6 Future Work

Inclusion of more datasets and models The first step is obviously to extend the project to more datasets and models. For models, we aim to include all publicly available models with pretrained weights. For datasets, we plan to start by including all publicly available datasets of H&E staining, and later expand to other imaging modalities and staining types. Eventually, it will be great if we can build the most comprehensive benchmark in this field.

Upgrade of evaluation script The current inference code is very slow to execute. Running 7,000 images on a model takes more than a day. This may be related to the double for-loop in the computation of mDice, mIoU, and mHD, where we may want to find a more efficient implementation.

More metrics and visualization In the future we may explore additional metrics and visualizations of cell segmentation results, including but not limited to entropy and mutual information measures [22].

Hosting automatic evaluation One potential mega-upgrade is to set up a service on the cloud that automatically runs inference and evaluation when participants submit their own methods. This would be a nice feature to have when most other aspects are mature enough.

References

[1] LeCun, Y. & Cortes, C. MNIST handwritten digit database (2010). URL `http://yann.lecun.com/exdb/mnist/`.

[2] Krizhevsky, A., Hinton, G. *et al.* Learning multiple layers of features from tiny images (2009).

[3] Deng, J. *et al.* Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255 (Ieee, 2009).

[4] Mahbod, A. *et al.* Cryonuseg: A dataset for nuclei instance segmentation of cryosectioned h&e-stained histological images. *Computers in biology and medicine* **132**, 104349 (2021).

[5] Da, Q. *et al.* Digestpath: A benchmark dataset with challenge review for the pathological detection and segmentation of digestive-system. *Medical Image Analysis* **80**, 102485 (2022).

[6] Moen, E. *et al.* Accurate cell tracking and lineage construction in live-cell imaging experiments with deep learning. *Biorxiv* 803205 (2019).

[7] Schwendy, M., Unger, R. E. & Parekh, S. H. Evican—a balanced dataset for algorithm development in cell and nucleus segmentation. *Bioinformatics* **36**, 3863–3870 (2020).

[8] Edlund, C. *et al.* Livecell—a large-scale dataset for label-free live cell segmentation. *Nature methods* **18**, 1038–1045 (2021).

[9] Mahbod, A. *et al.* Nuinsseg: A fully annotated dataset for nuclei instance segmentation in h&e-stained histological images. *arXiv preprint arXiv:2308.01760* (2023).

[10] Greenwald, N. F. *et al.* Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. *Nature biotechnology* **40**, 555–565 (2022).

[11] Al-Kofahi, Y., Zaltsman, A., Graves, R., Marshall, W. & Rusu, M. A deep learning-based algorithm for 2-d cell segmentation in microscopy images. *BMC bioinformatics* **19**, 1–11 (2018).

[12] Wang, A. *et al.* A novel deep learning-based 3d cell segmentation framework for future image-based disease detection. *Scientific reports* **12**, 342 (2022).

[13] Stringer, C., Wang, T., Michaelos, M. & Pachitariu, M. Cellpose: a generalist algorithm for cellular segmentation. *Nature methods* **18**, 100–106 (2021).

[14] Lee, M. Y. *et al.* Cellseg: a robust, pre-trained nucleus segmentation and pixel quantification software for highly multiplexed fluorescence images. *BMC bioinformatics* **23**, 46 (2022).

[15] Liu, C. *et al.* Cuts: A deep learning and topological framework for multigranular unsupervised medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 155–165 (Springer, 2024).

[16] Khalid, N. *et al.* Deepcens: An end-to-end pipeline for cell and nucleus segmentation in microscopic images. In *2021 International Joint Conference on Neural Networks (IJCNN)*, 1–8 (IEEE, 2021).

[17] Liu, C. *et al.* Diffkillr: Killing and recreating diffeomorphisms for cell annotation in dense microscopy images. *arXiv preprint arXiv:2410.03058* (2024).

[18] Zhu, N. *et al.* Segmentation with residual attention u-net and an edge-enhancement approach preserves cell shape features. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2115–2118 (IEEE, 2022).

[19] Shrestha, P., Kuang, N. & Yu, J. Efficient end-to-end learning for cell segmentation with machine generated weak annotations. *Communications Biology* **6**, 232 (2023).

[20] Schmidt, U., Weigert, M., Broaddus, C. & Myers, G. Cell detection with star-convex polygons. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II 11*, 265–273 (Springer, 2018).

[21] Nishimura, K., Ker, D. F. E. & Bise, R. Weakly supervised cell instance segmentation by propagating from detection response. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part I 22*, 649–657 (Springer, 2019).

[22] Liao, D. *et al.* Assessing neural network representations during training using noise-resilient diffusion spectral entropy. In *2024 58th Annual Conference on Information Sciences and Systems (CISS)*, 1–6 (IEEE, 2024).