Tracking Phenotypes with Automated Lensfree Microscopy

Soft Condensed Matter Group, Physics Faculty, LMU

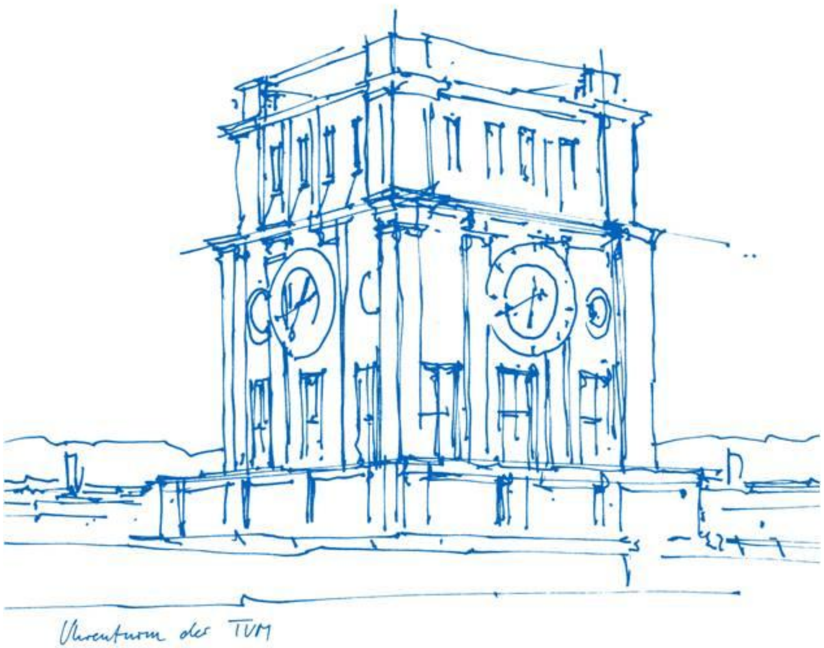
Dominik Schnaus, Maria Bordukova, Lion Gleiter, Bilal Al-Saeedi

Technical University of Munich

Department of Mathematics

TUM Data Innovation Lab

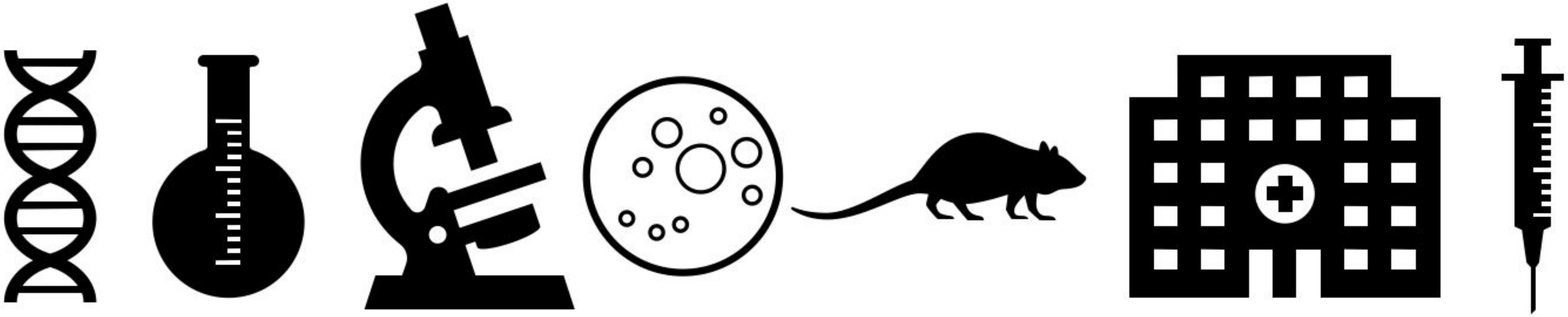
Munich, 24. February 2021



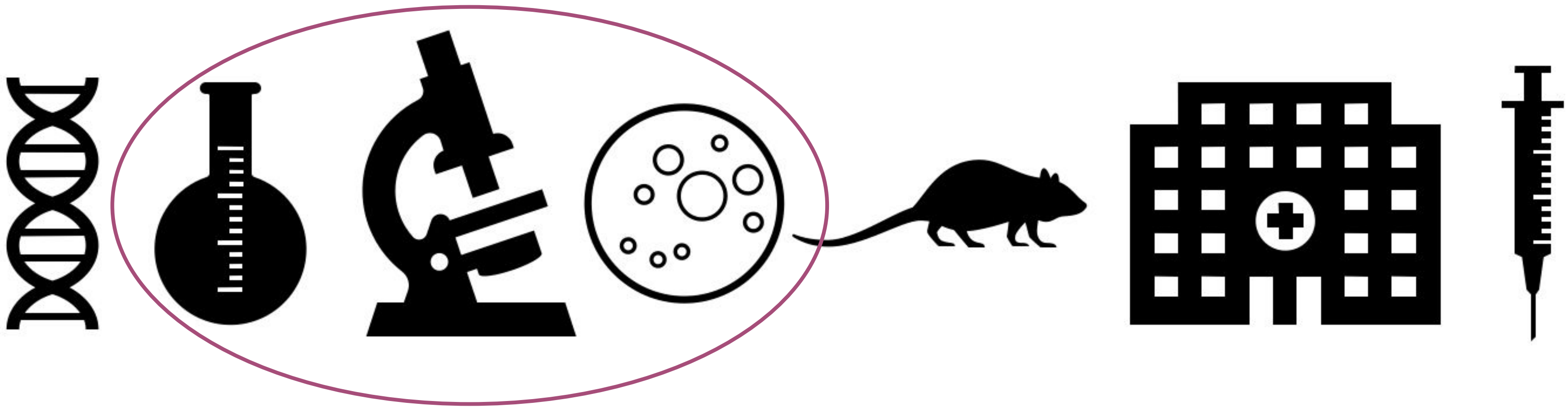
Agenda

- Motivation and project goals
- Data
- Cell counting and detection
- Confluency estimation
- Conclusion and outlook

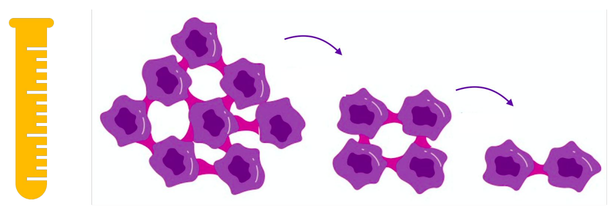
Introduction - Drug Development Pipeline



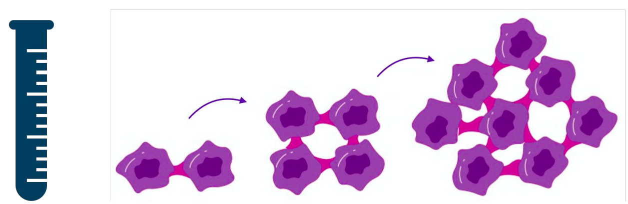
Introduction - Drug Development Pipeline



drug
A



drug
B

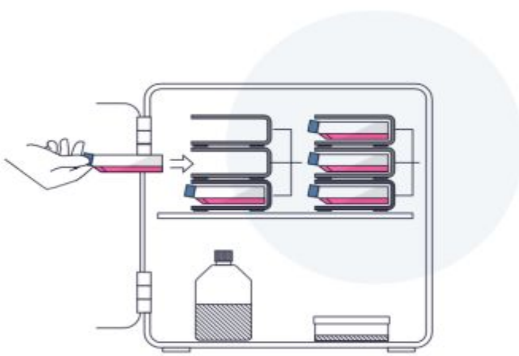


Introduction - Traditional Microscope

Introduction - Lens-free Microscope

1 Microscope

Easy to use & super compact microscope unit.

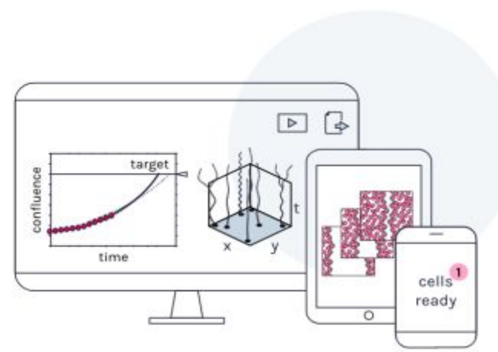


2 Analysis

Fully automatic cutting edge image & data-analysis.

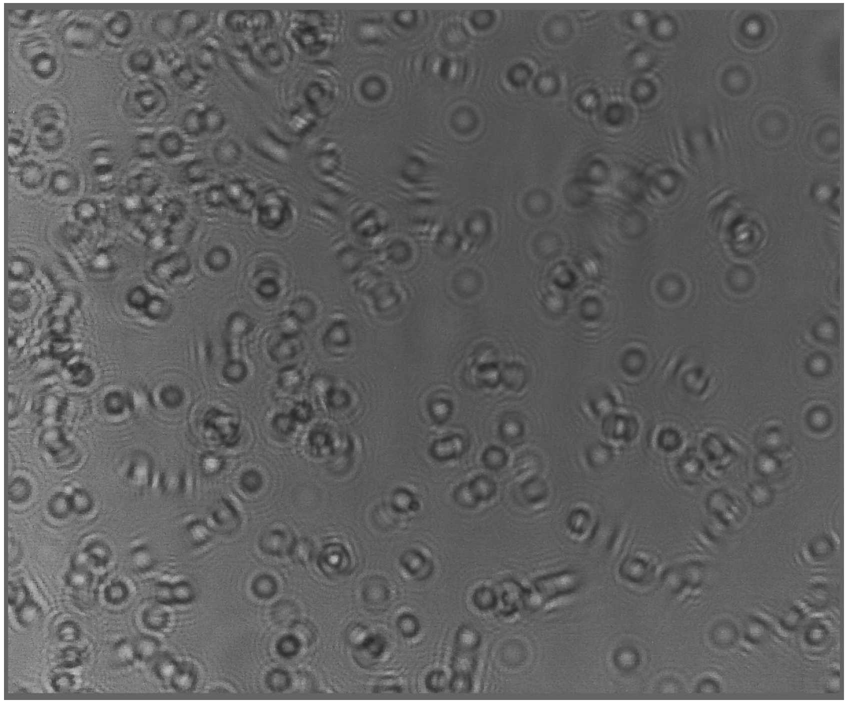
3 Applications

Quality management & cell-based assays.



Introduction - Project Goals

- Assess cell proliferation in lens-free microscopy images
- Two perspectives:
  - Cell counting and detection
  - Confluency estimation



- **289 cells**
- **19% of area covered by cells**

Introduction - Counting Chamber Dataset

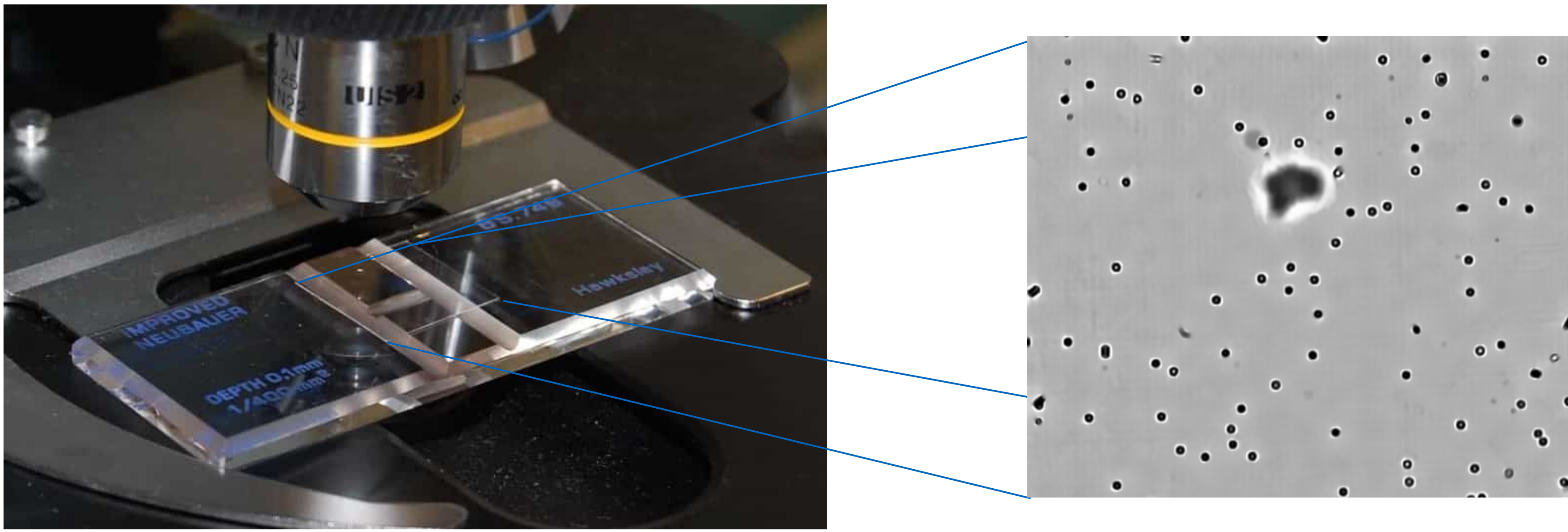
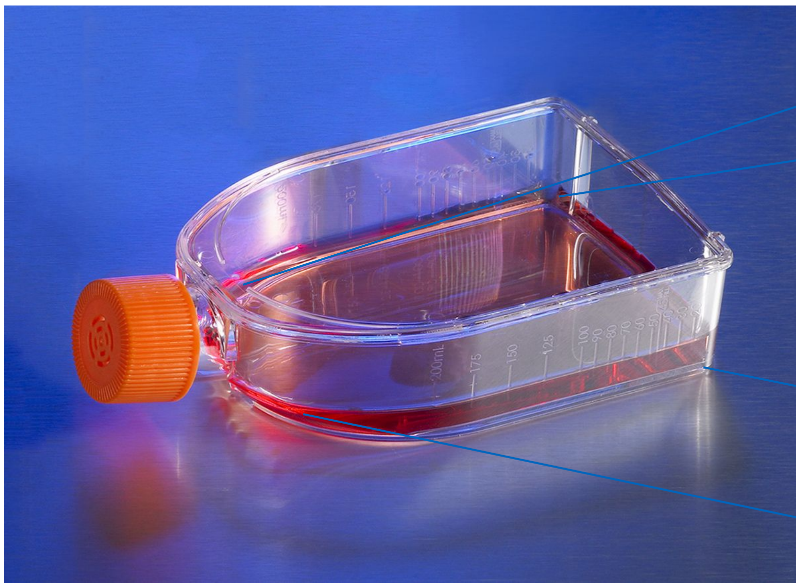
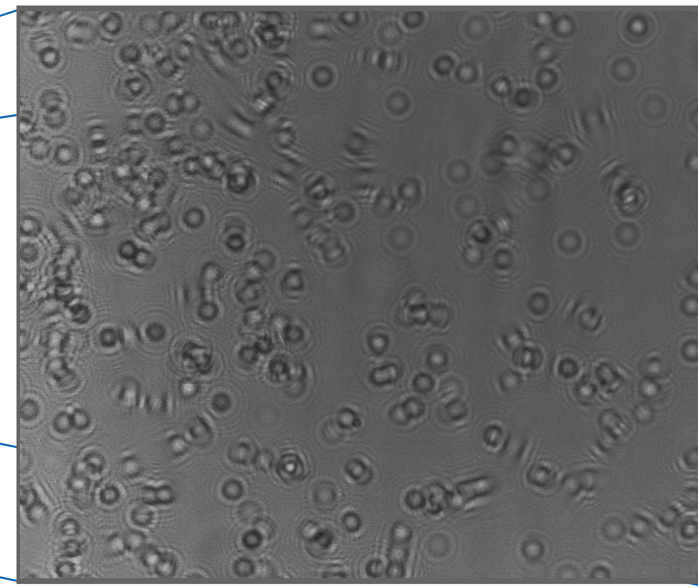


Image source:
https://www.hemocytometer.org/wp-content/uploads/2013/04/hemocytometer-cell-count-on-microscope.jpg

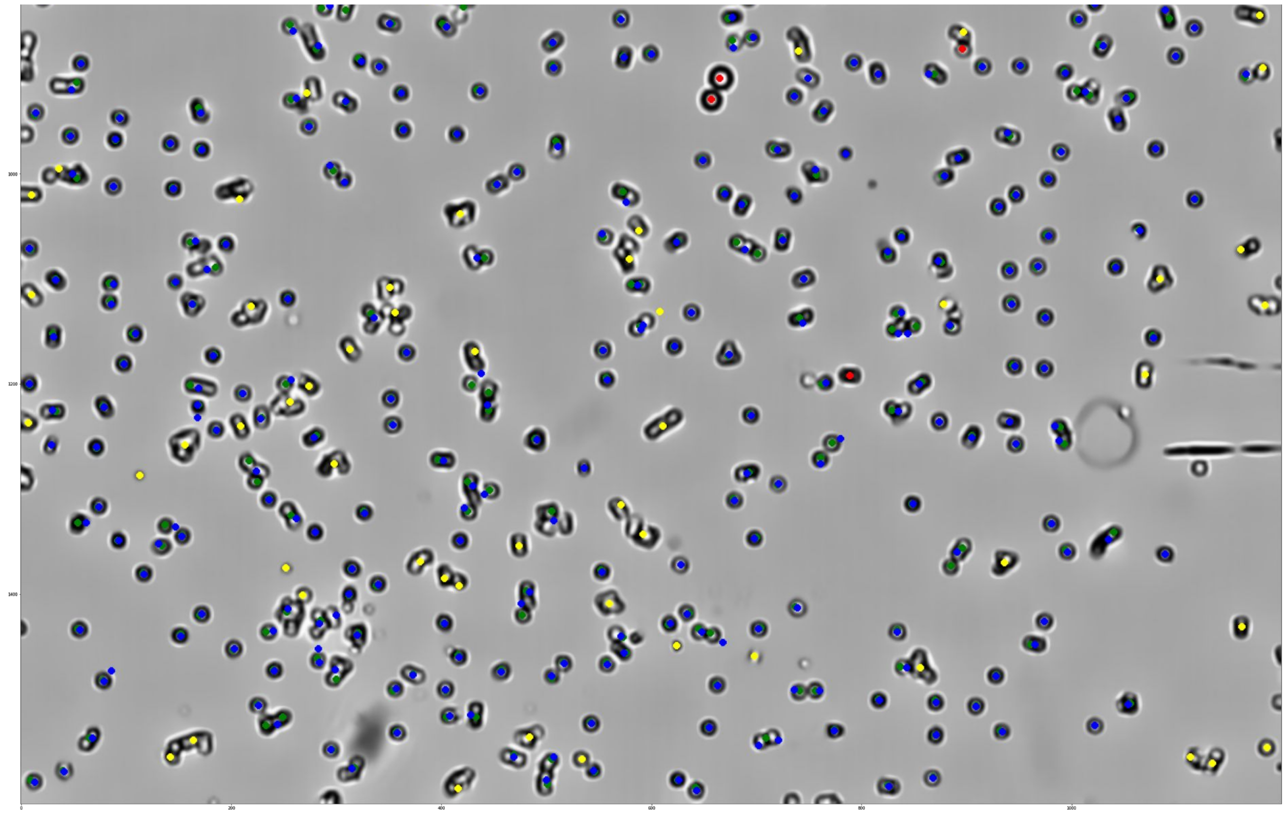
Introduction - Flask Dataset





Tasks

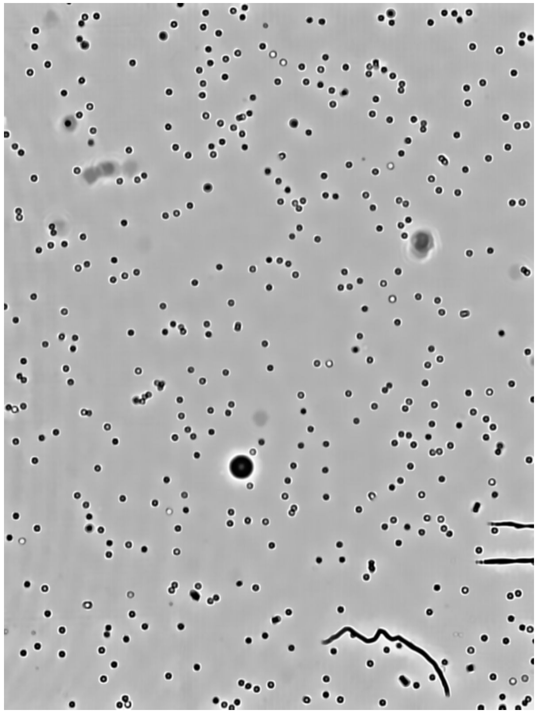
- **Cell counting and detection**
- Confluency estimation



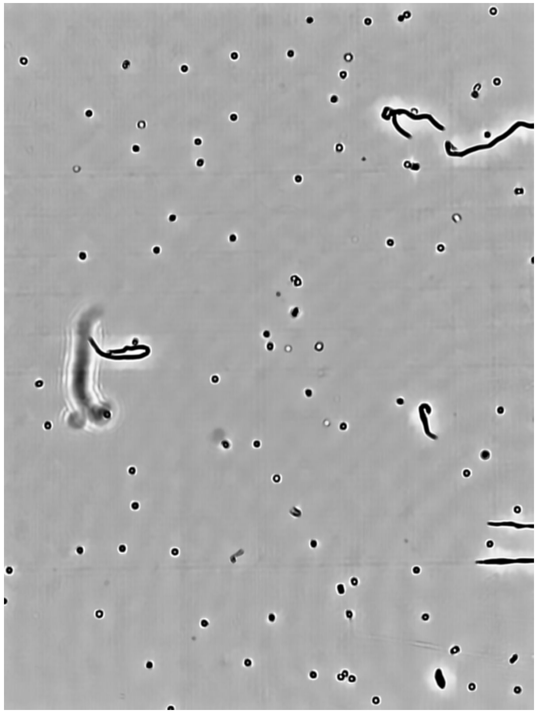
Data split

Cell Counting & Detection - Counting Chamber

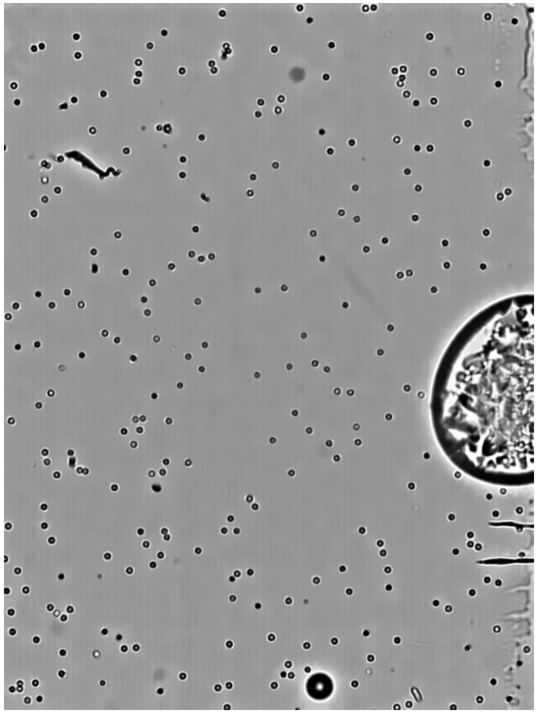
A549 (train)
126 images



HuH7 (val)
22 images



3T3 (test)
126 images

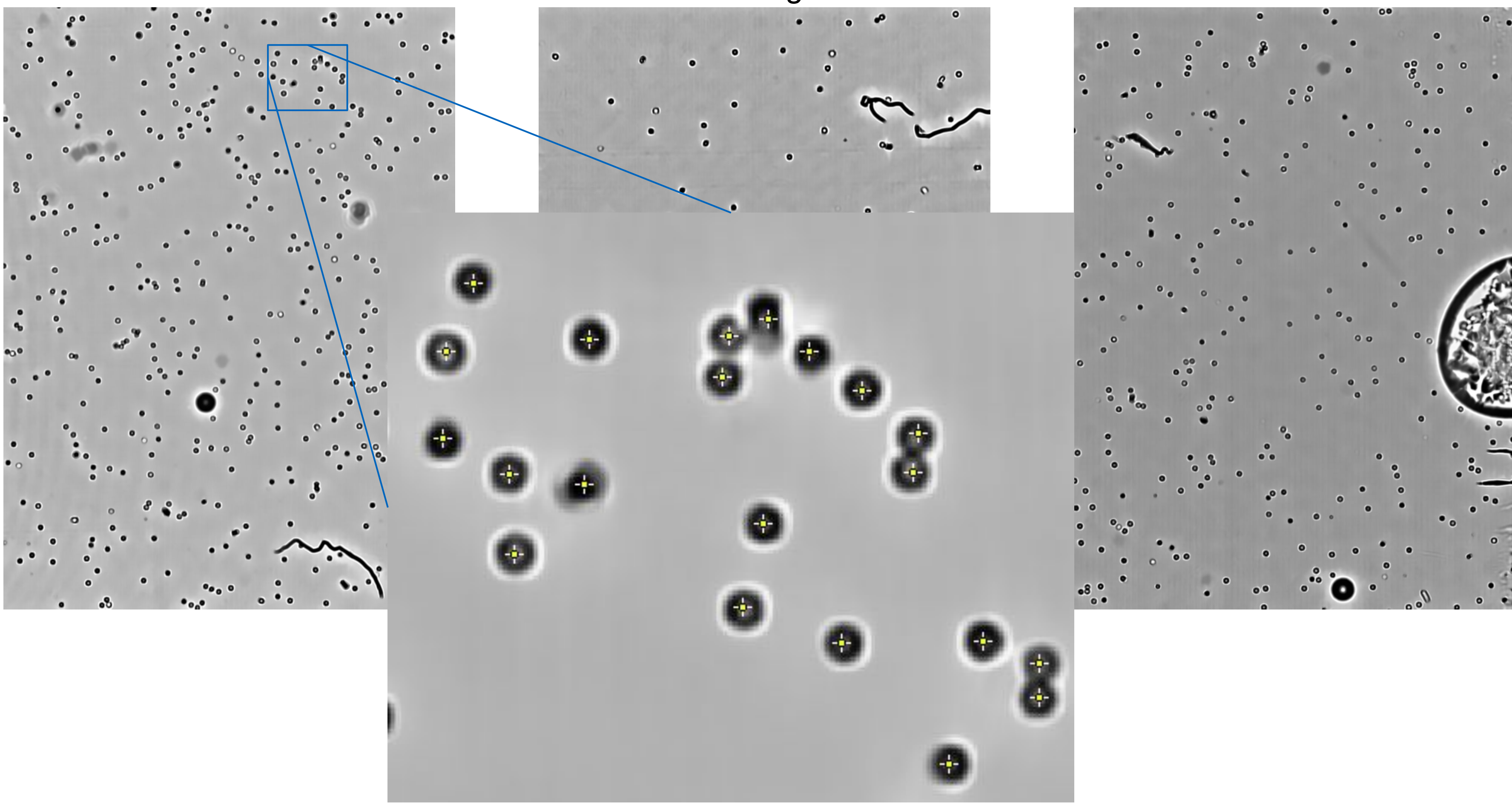


Cell Counting & Detection - Counting Chamber

A549 (train)
126 images

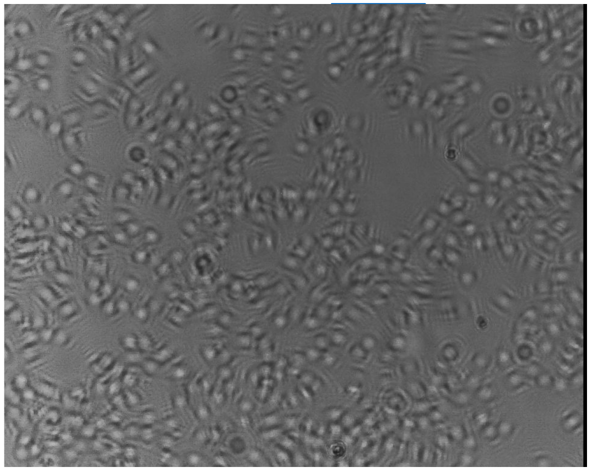
HuH7 (val)
22 images

3T3 (test)
126 images

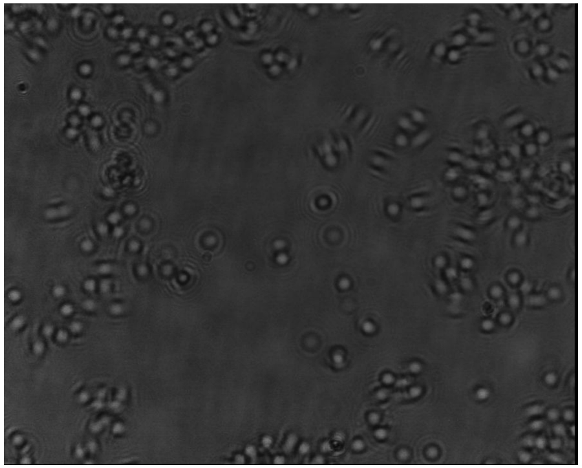


Cell Counting & Detection - Flask

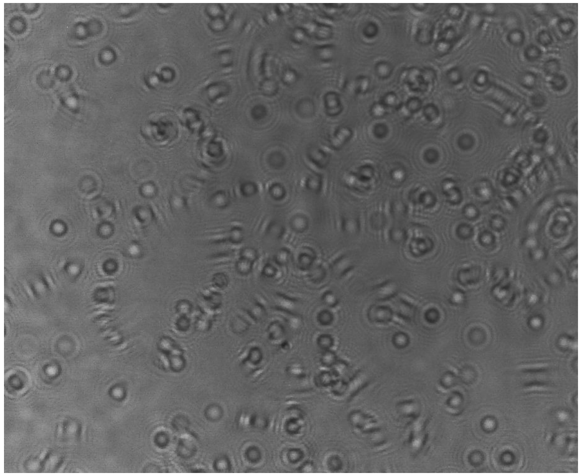
NRK (train)
1053 images / 6 experiments



A549 (val)
9 images / 4 experiments



3T3 (test)
829 images / 10 experiments



Methodology

Cell Counting & Detection - Methods

Method	**Count-Ception**	**SS-DCNet**	**CenterNet**	**ProximityNet**
Type	Counting		Detection	
State-of-the-art in	Cell counting	Crowd, plant, vehicle counting	Anchor-free object detection	Lens-free cell detection
Backbone CNN	Inception	VGG-16	Resnet-18	LinkNet

Cell Counting & Detection - Methods

Method	**Count-Ception**	**SS-DCNet**	**CenterNet**	**ProximityNet**
Type	Counting		Detection	
State-of-the-art in	Cell counting	Crowd, plant, vehicle counting	Anchor-free object detection	Lens-free cell detection
Backbone CNN	Inception	VGG-16	Resnet-18	LinkNet

Cell Counting & Detection - Counting

Method	**Count-Ception**	**SS-DCNet**
Redundant counting	True	False
Multiple stages	False	True
Smooth vs. binary count map	Binary	Smooth
Regression vs. classification	Regression	Classification

Cell Counting & Detection - Methods

Method	**Count-Ception**	**SS-DCNet**	**CenterNet**	**ProximityNet**
Type	Counting		Detection	
State-of-the-art in	Cell counting	Crowd, plant, vehicle counting	Anchor-free object detection	Lens-free cell detection
Backbone CNN	Inception	VGG-16	Resnet-18	LinkNet

Cell Counting & Detection - Detection

Method	**CenterNet**	**ProximityNet**
Exact localization by	Regression	Upsampling
Loss	Focal loss	Dice loss

Evaluation & Results

Cell Counting & Detection - Evaluation & Results

- Counting metrics:
  a. MAE
  b. MSE
- Detection metrics:
  a. Precision
  b. Recall
  c. F1 score

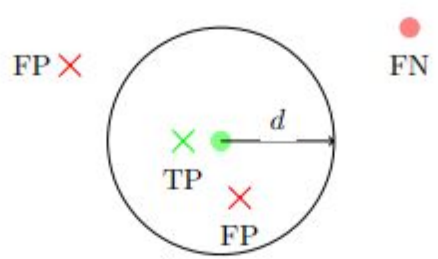
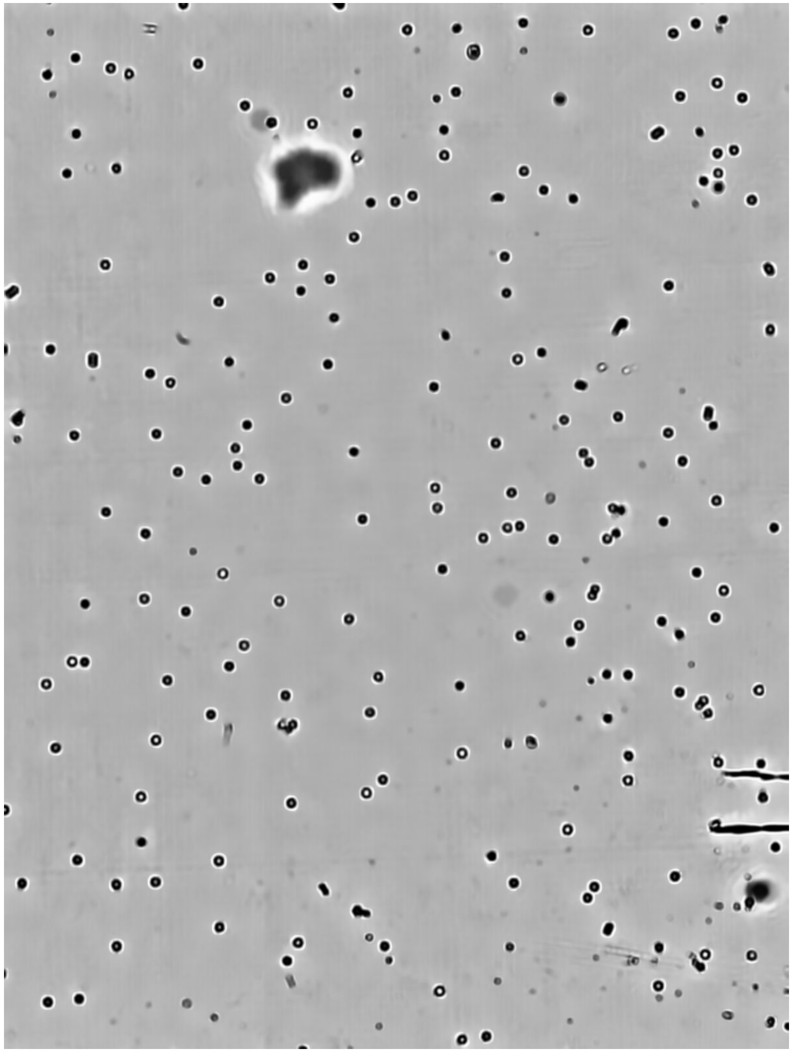


Figure 3: This figure shows how true positives (TP, green cross), false positives (FP, red cross) and false negatives are defined. The maximal distance d is chosen to be 20 pixels for our task. Adapted from [8].

Cell Counting & Detection - Counting Chamber



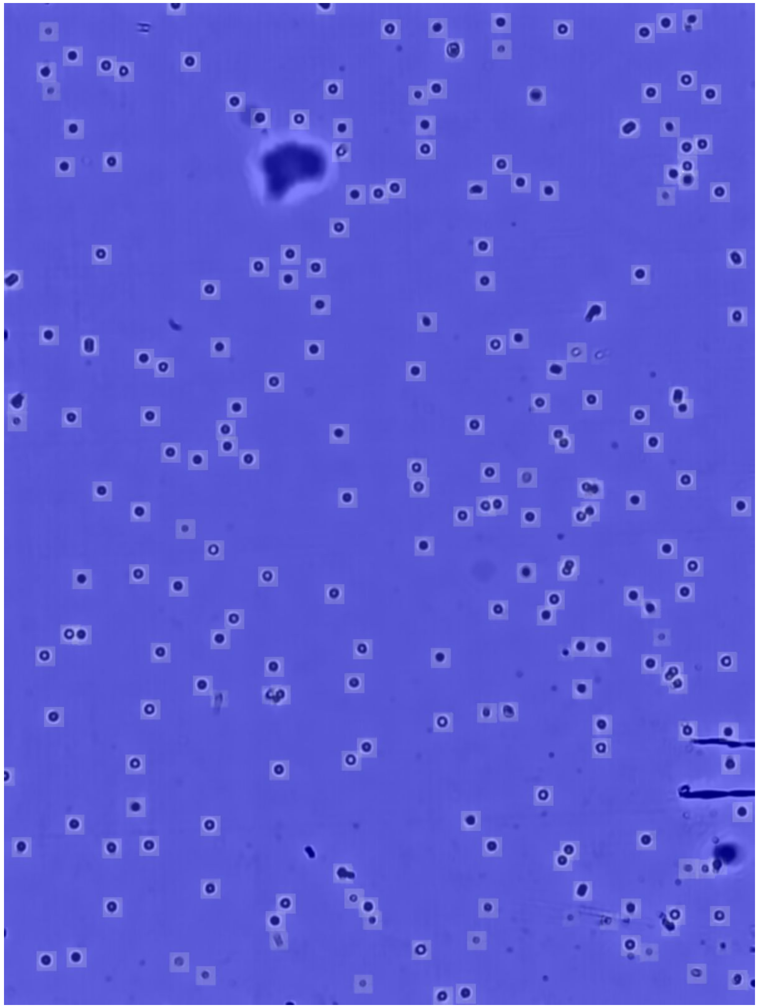
True count: 231

Cell Counting & Detection - Counting Chamber

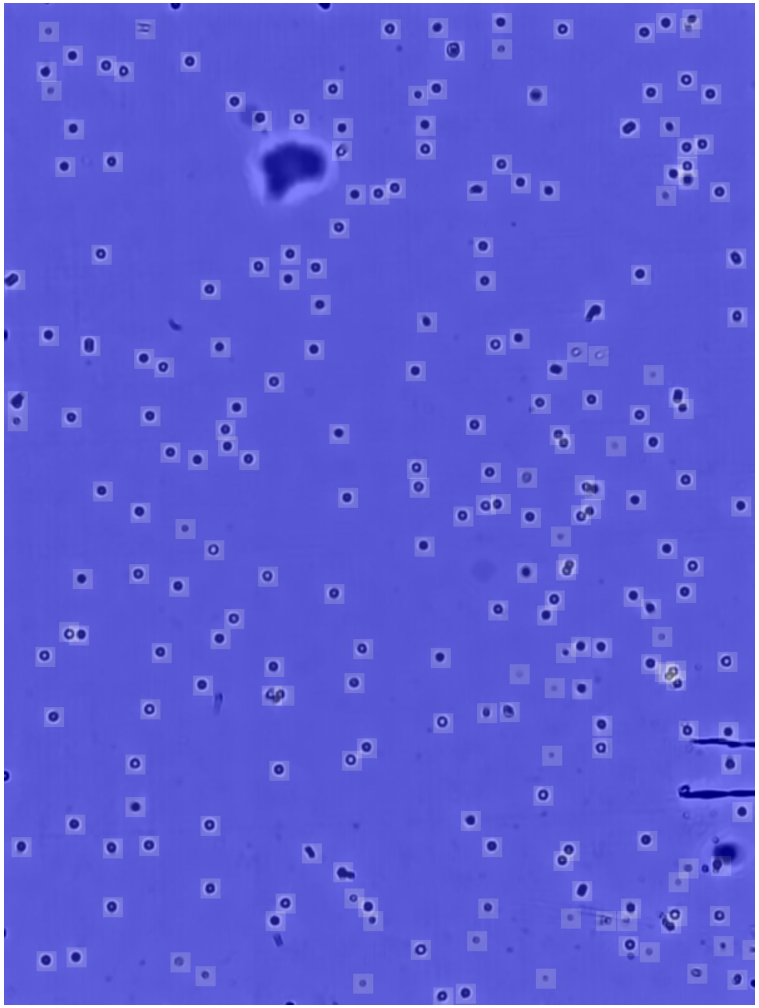
Method	MAE	MSE	Precision	Recall	F1 score
Count-Ception	13.67	321.43	-	-	-
SS-DCNet	10.92	460.89	-	-	-
CenterNet	6.40	84.24	**0.97**	0.96	0.96
ProximityNet	**5.81**	**76.48**	0.96	**0.97**	**0.97**

Best models out of 38 model configurations

Cell Counting - Count-Ception Results

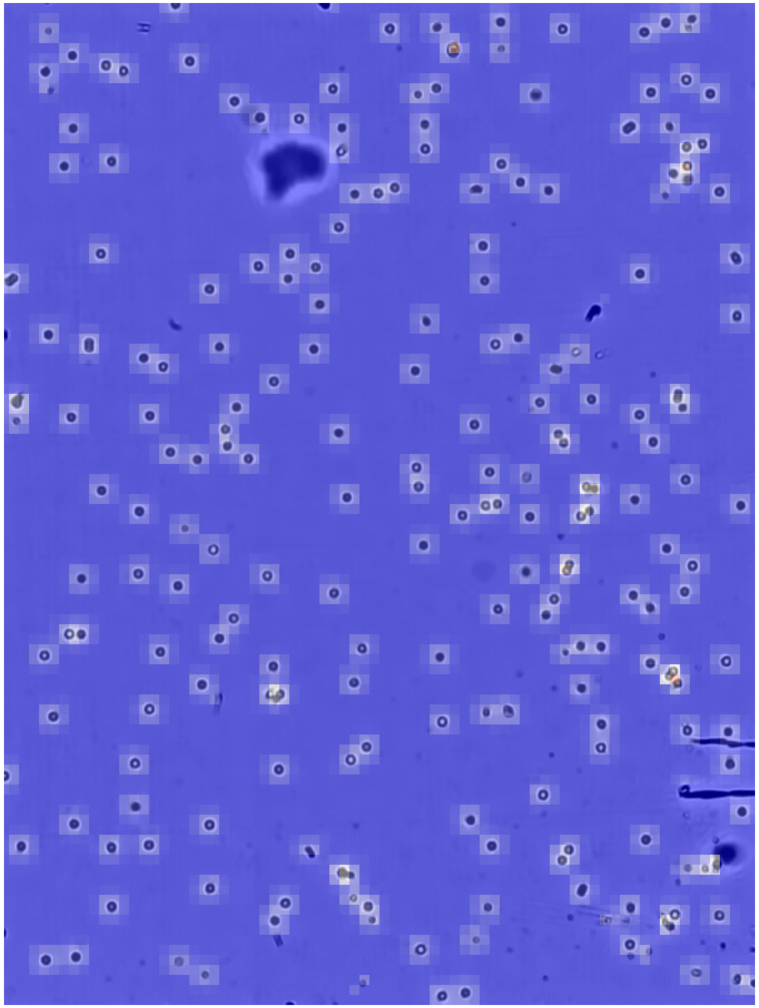


Predicted heatmap
Predicted count: 208

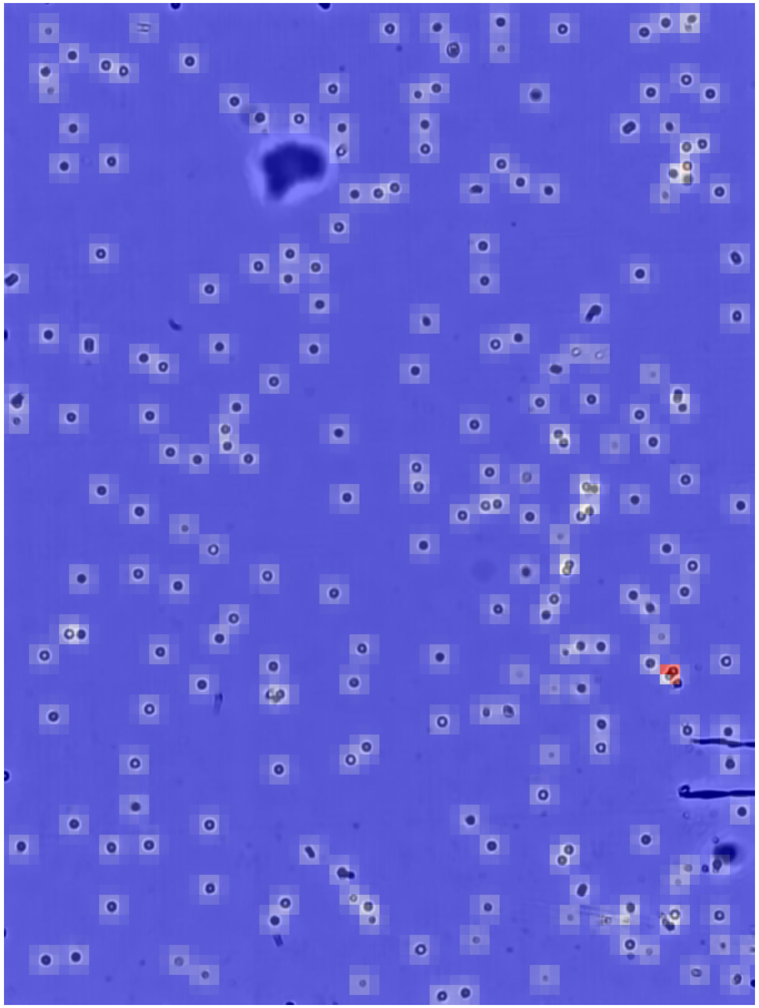


Target heatmap
True count: 231

Cell Counting - SS-DCNet Results

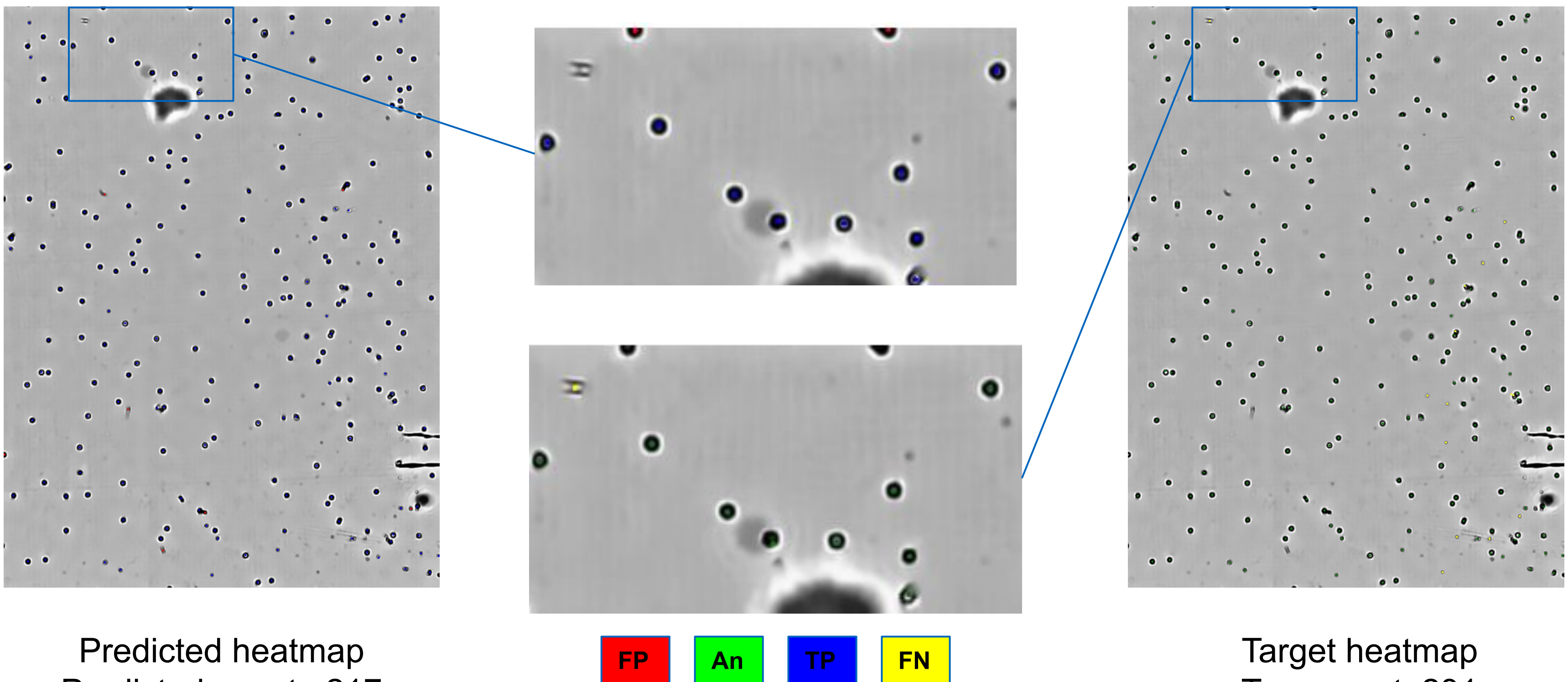


Predicted heatmap
Predicted count: 221



Target heatmap
True count: 231

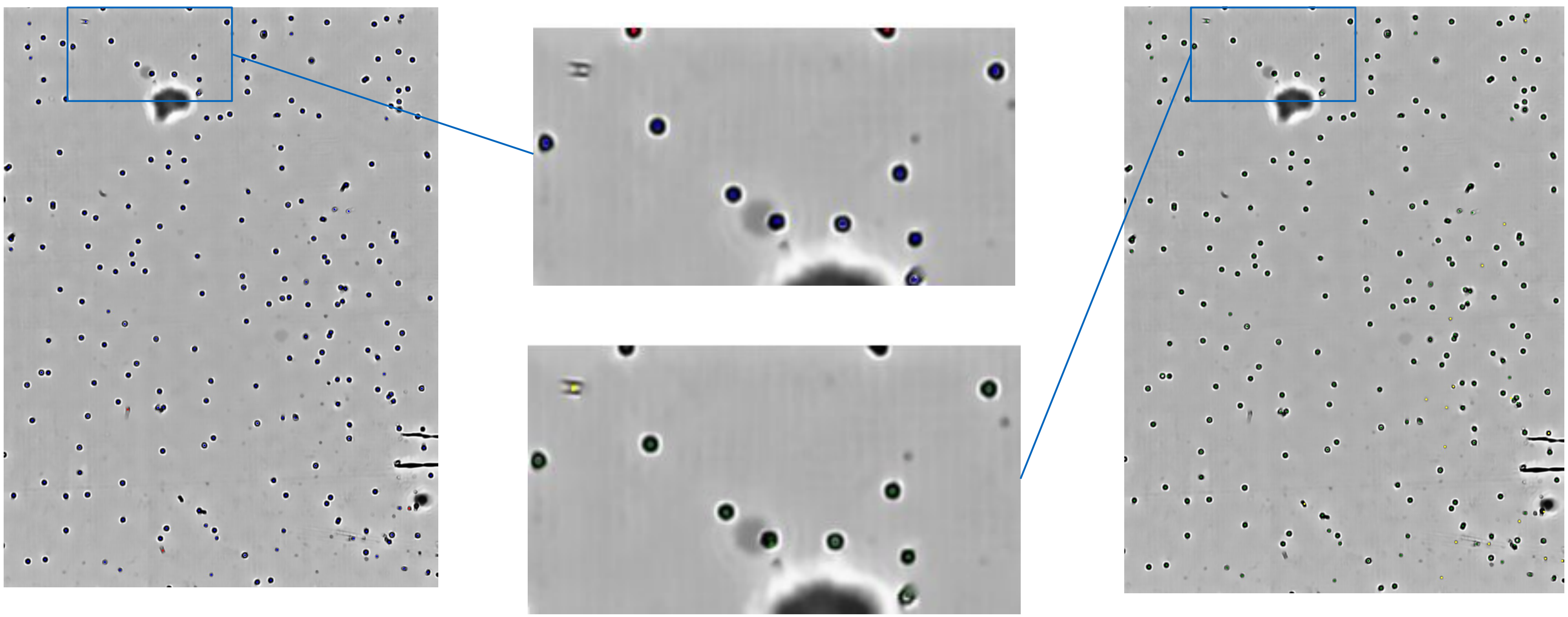
Cell Detection - CenterNet Results



Predicted heatmap
Predicted count: 217

Target heatmap
True count: 231

Cell Detection - ProximityNet Results

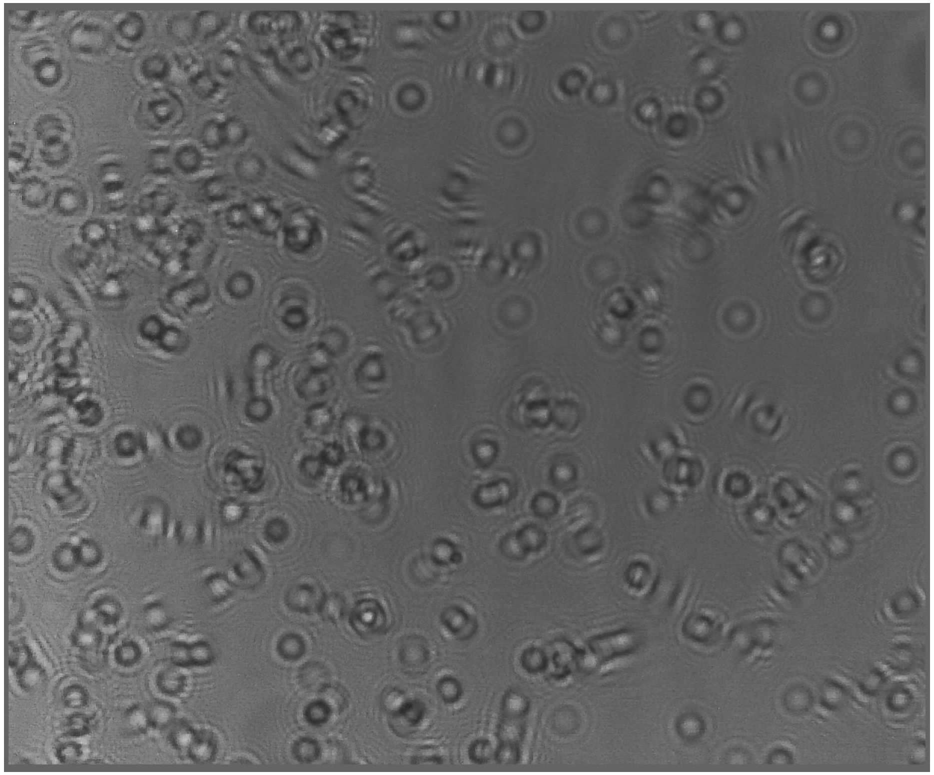


FP An TP FN

Predicted heatmap
Predicted count: 218

Target heatmap
True count: 231

Cell Counting & Detection - Flask



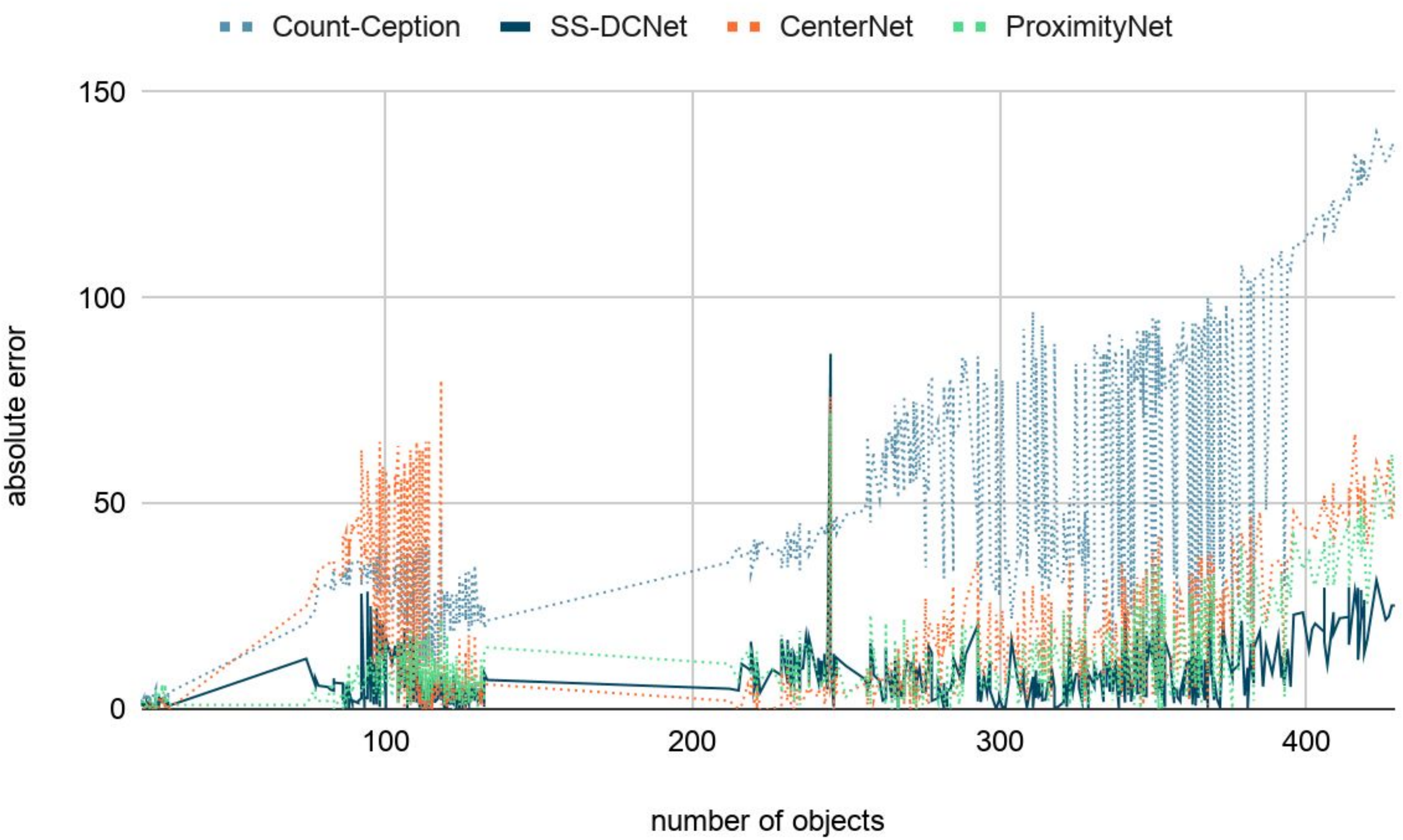
True count: 289

Cell Counting & Detection - Flask

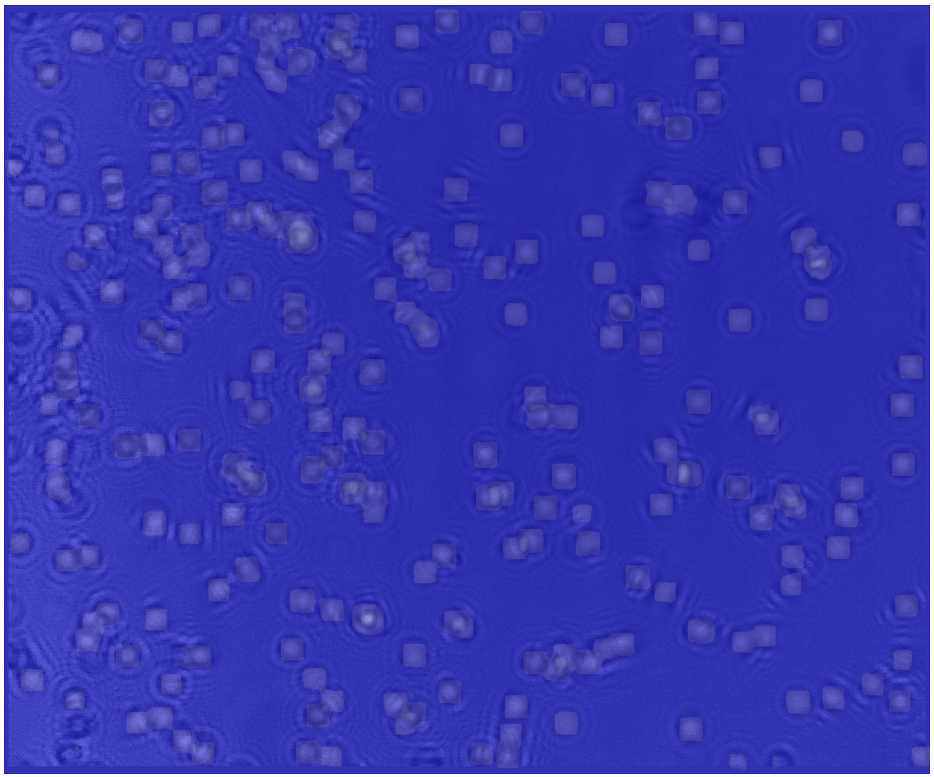
Method	MAE	MSE	Precision	Recall	F1 score
Count-Ception	43.07	2923.88	-	-	-
SS-DCNet	**7.10**	**97.80**	-	-	-
CenterNet	16.37	579.12	**0.94**	0.87	0.89
ProximityNet	10.33	202.96	0.93	**0.92**	**0.92**

Best models out of 44 model configurations

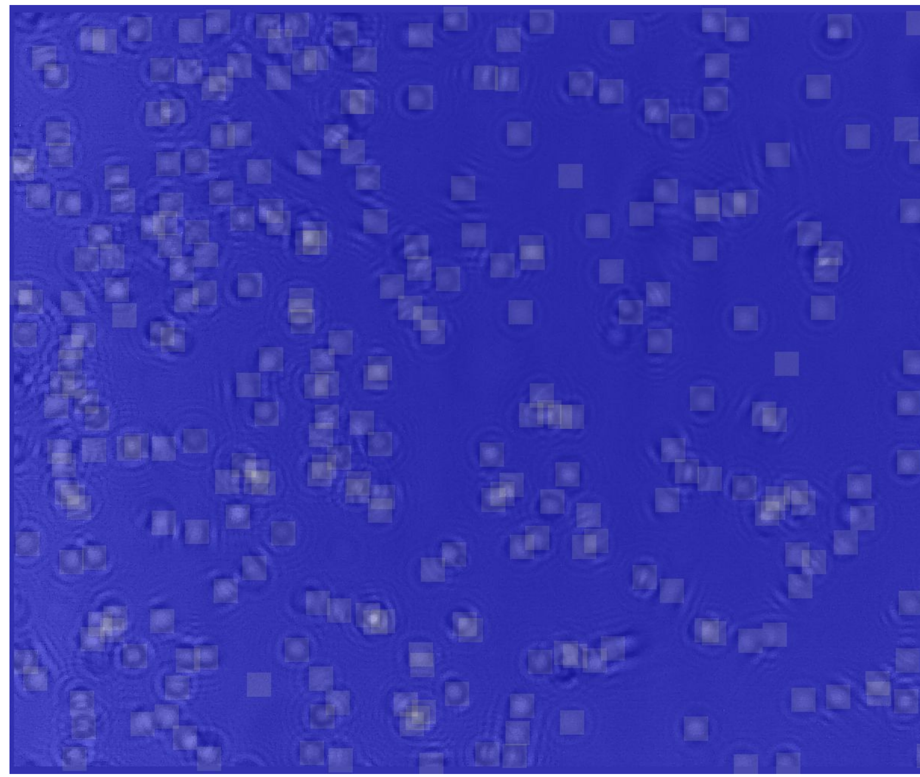
Cell Counting & Detection - Flask



Cell Counting - Count-Ception Results

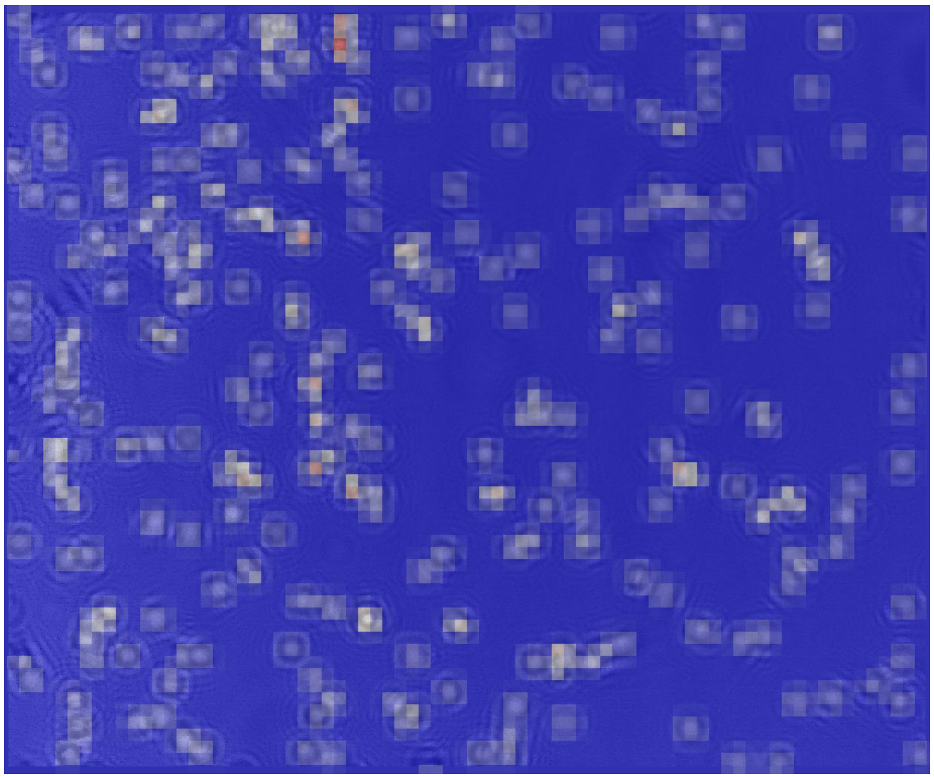


Predicted heatmap
Predicted count: 204

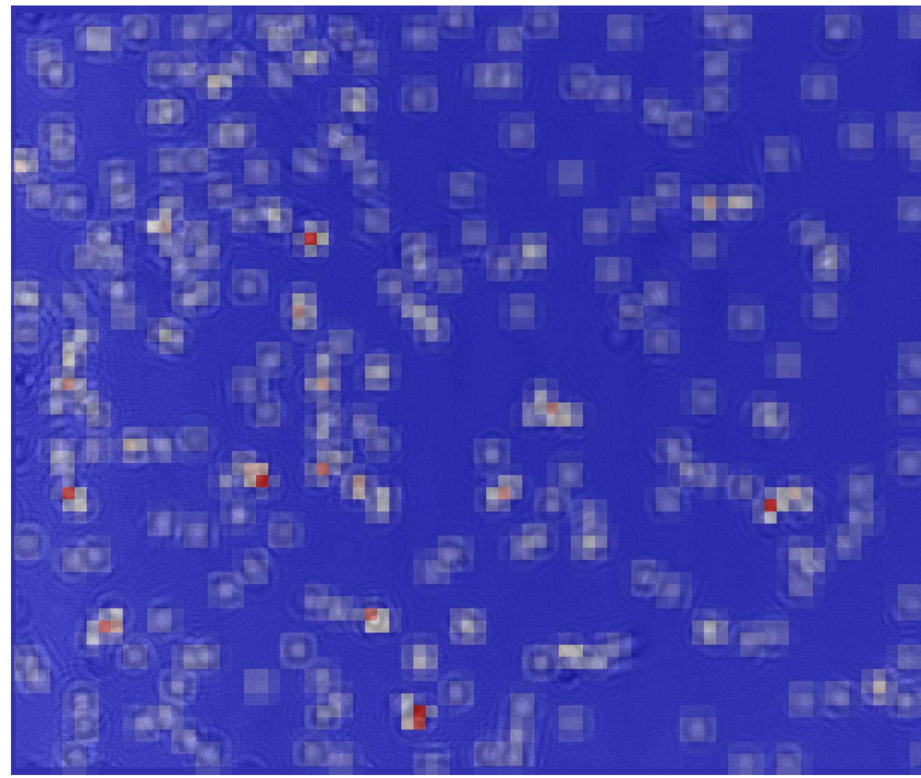


Target heatmap
True count: 289

Cell Counting - SS-DCNet Results

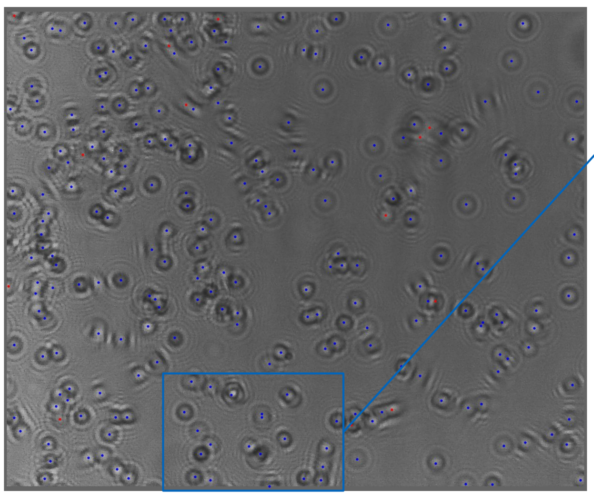


Predicted heatmap
Predicted count: 275

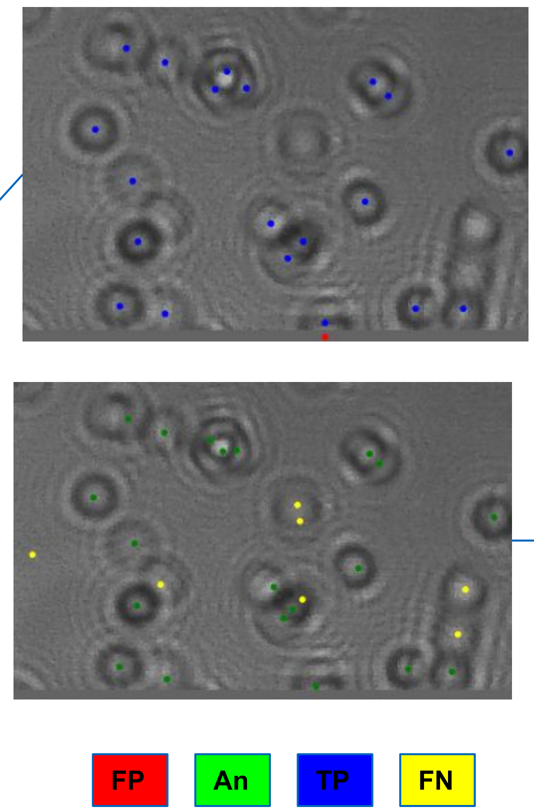


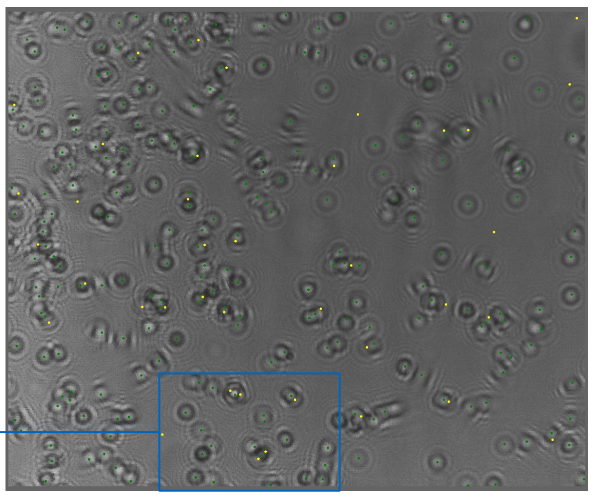
Target heatmap
True count: 289

Cell Detection - CenterNet Results



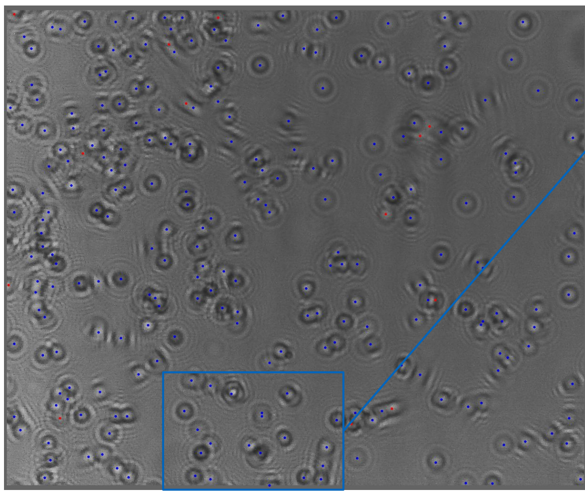
Predicted heatmap
Predicted count: 263



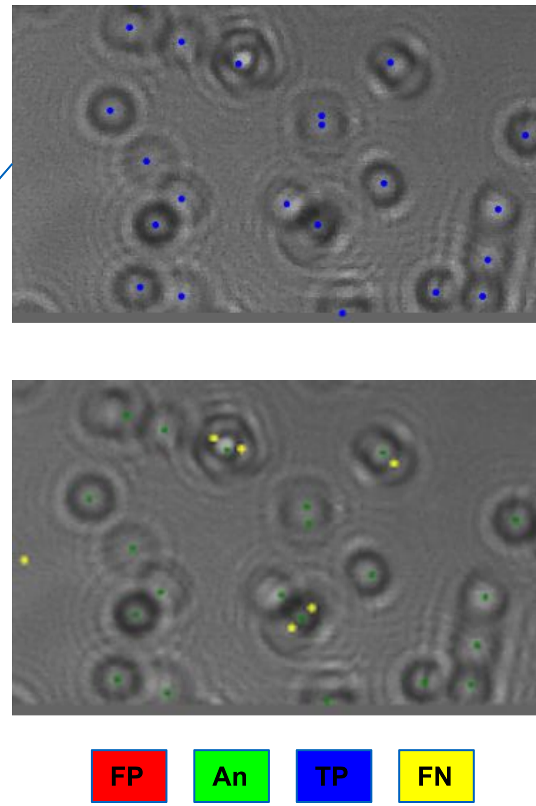


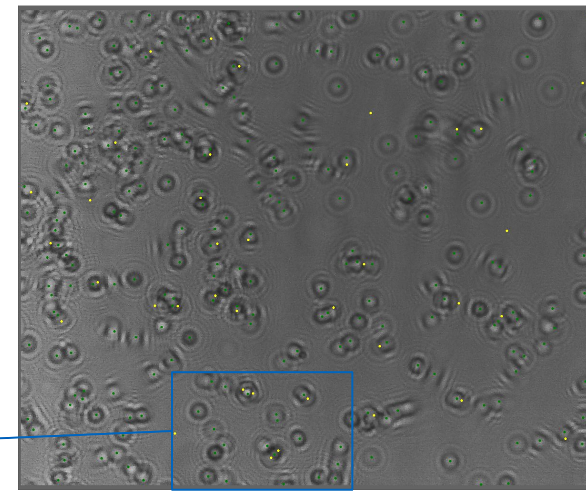
Target heatmap
True count: 289

Cell Detection - ProximityNet Results



Predicted heatmap
Predicted count: 270





Target heatmap
True count: 289

Tasks

- Cell counting and detection
- **Confluency estimation**

Cell Proliferation

Idea: number of cells per image and cell covered area probably correlate

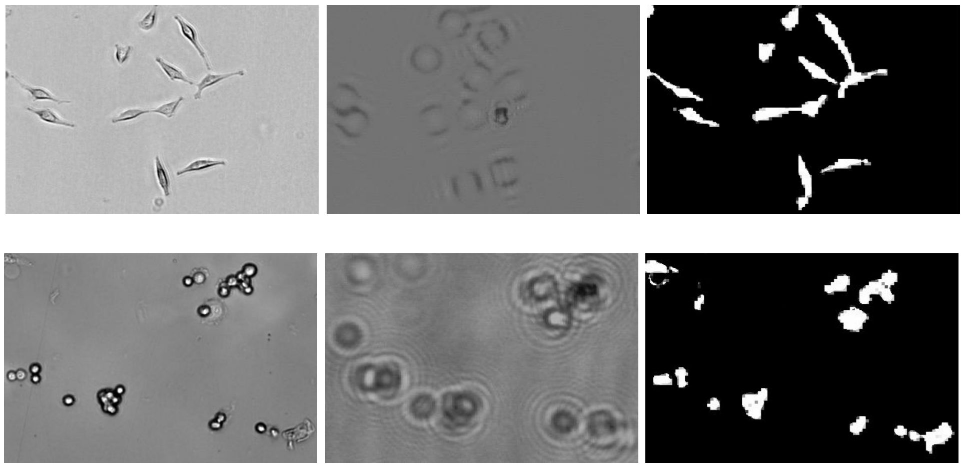
Baseline model: multiply the number of cells by a learned, average cell size

Possible disadvantages of the baseline model

bright-field | lens-free | mask

NRK

HeLa



Different cell shapes are not considered when counting cells

Count - Confluency relationship not linear for clustered, overlapping cells

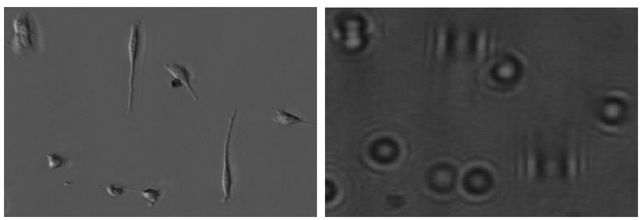
→ We think we can do better with a separate model

3 approaches: Count-ception, regression and classification

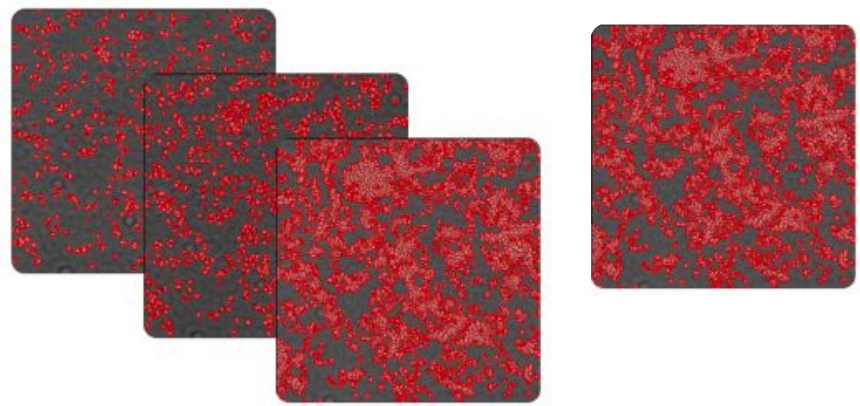
Data overview

- Only flask data, various cell types

Parallel bright-field and lens-free images



Time series vs. independent observations

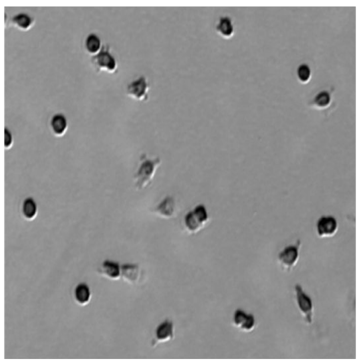


- 4224 observed images, but only 158 independent experiments

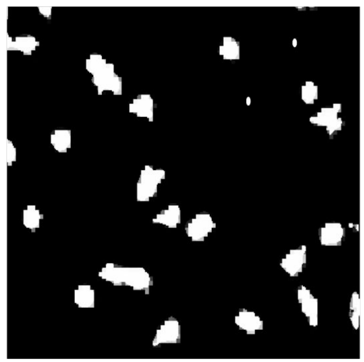
Ground truth generation

Model 1: Mask generation

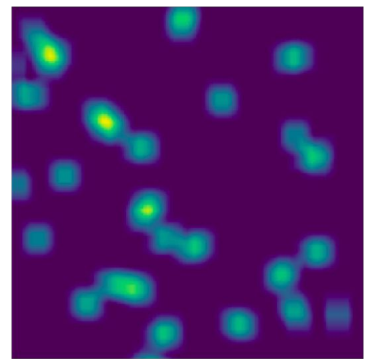
Target generation



bright-field image



mask



confluency map

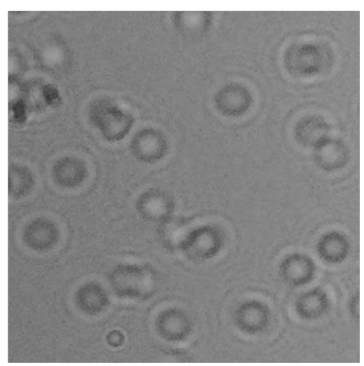
9%

confluency class

or

0.09

regression target



lens-free image

Model 2: Confluency estimation

Training/Validation/Test split

Cell type	Time series	Experiments (train/val/test)	Total observations
3T3	yes	13 (12/0/1)	1601
A549	yes	8 (8/0/0)	434
A549	no	33 (0/6/27)	33
HeLa	yes	7 (7/0/0)	1310
HeLa	no	18 (0/4/14)	18
Osteoblasts	no	41 (22/6/13)	41
BxPC-3	no	33 (0/0/33)	33
NRK	yes	5 (0/0/5)	754
Total		158 (49, 16, 93)	4224 (3280, 16, 928)

2 cell types only in the test set,

to evaluate generalization to unseen cell types

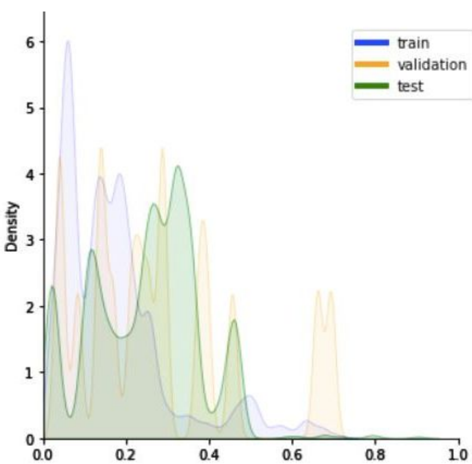
6 time series, to test consistency over time

Evaluation per cell type possible

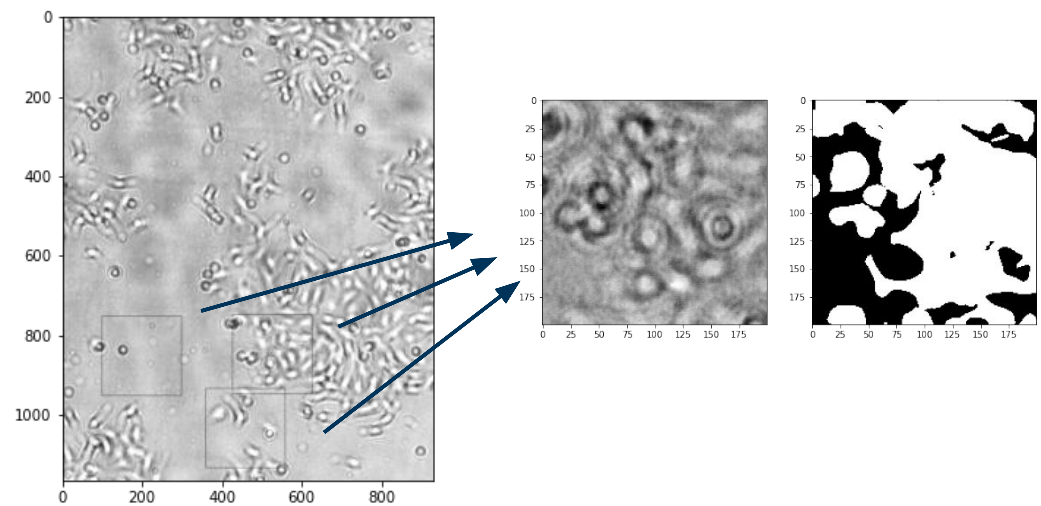
Methodology

Data augmentation

Only 49 training experiments → Rotation, flipping and Gamma-transformation



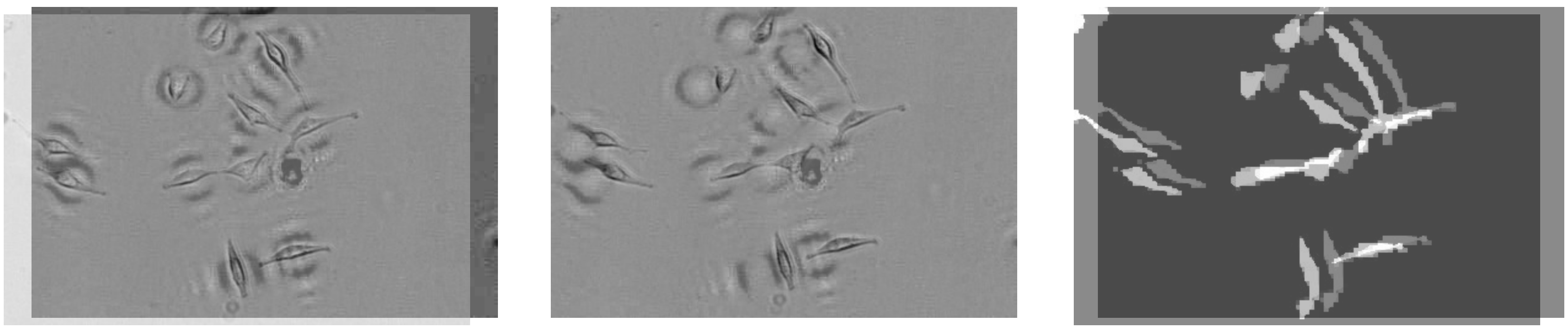
Skewed true confluency distribution
→ overlaying random patches



Methodology - Why not segmentation?

- original shape might be difficult to infer
- slight misalignments lead to inaccurate ground truth masks

perfect actual

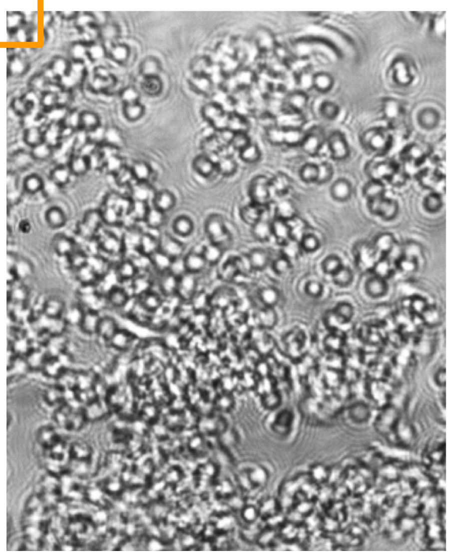


https://www.di-lab.tum.de/fileadmin/w00byz/www/LMU_lensless_microscopy_Final_Presentation_SS19.pdf

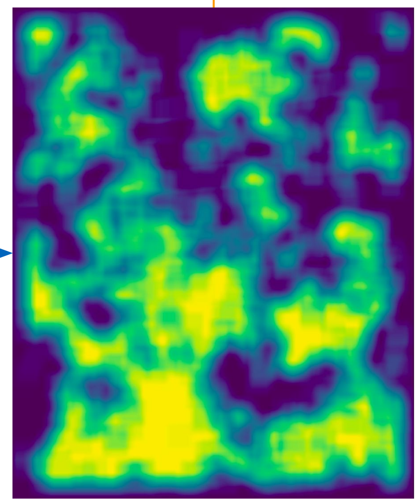
Methodology - Counting pixels with Count-ception

Pixel-wise L1 or L2 loss

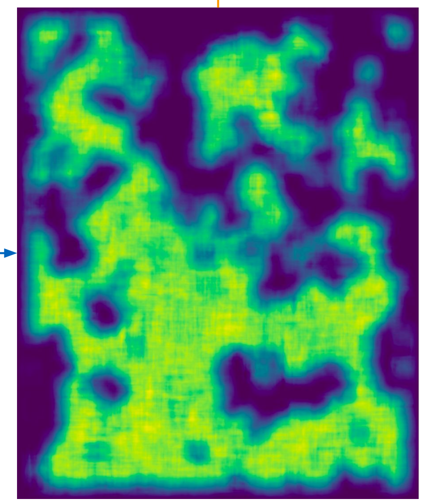
Receptive field (RF) 32



Input image



Target confluency map



Predicted confluency map M

$\in \mathbb{R}^{h \times w}$

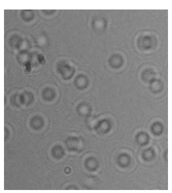
Count-ception Model

$$confluency = \frac{\left(\sum_{x,y} M[x,y]\right)/(h \times w)}{RF \times RF}$$

(redundant counts)

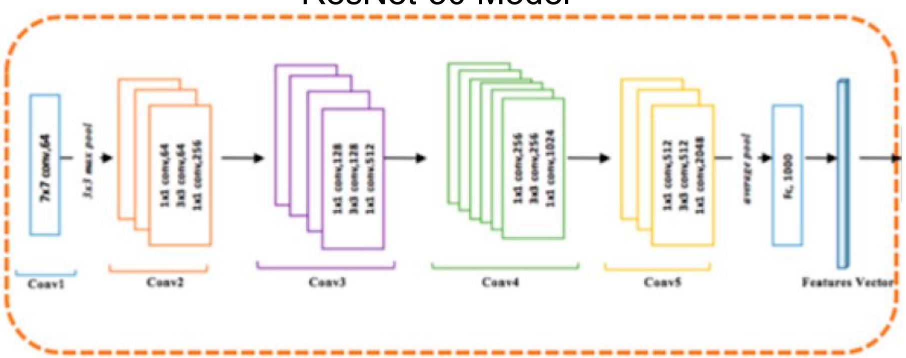
Methodology - Classification

input image
224 x 224



Feature Extraction

ResNet-50 Model



Classification

softmax →

class 1%
class 2%
…
class 99%
class 100%

(Cross-entropy loss)

Classification - Modifications

- Natural order and distance notion between confluency classes

Soft Labels (Train time)

- One-hot encoding

	11%	12%	13%	14%	15%	16%	17%	
...	0	0	0	**1**	0	0	0	...

- Soft labels

Convolution with Gaussian kernel, e.g., sigma = 0.5

	11%	12%	13%	14%	15%	16%	17%	
...	0	0.02	0.12	**0.72**	0.12	0.02	0	...

- Cross entropy loss

$$CE = -\sum_{x} p(x) log(q(x))$$

Classification: Modifications

- Natural order and distance notion between confluency classes

Soft Labels (Train time)

- One-hot encoding

	11%	12%	13%	14%	15%	16%	17%	
...	0	0	0	**1**	0	0	0	...

- Soft labels

Convolution with Gaussian kernel, e.g., sigma = 0.45

	11%	12%	13%	14%	15%	16%	17%	
...	0	0.02	0.12	**0.72**	0.12	0.02	0	...

- Cross entropy loss

$$CE = -\sum_{x} p(x) log(q(x))$$

Class aggregation (Inference time)

- Mode prediction

	11%	12%	13%	14%	15%	16%	17%	
...	0	**0.73**	0	0.12	0.03	0.09	0	...

- Weighted average of k best predictions

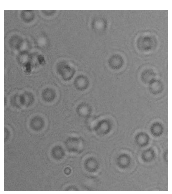
$$confluency = \sum_{k} w_k c_k, \quad \sum_{k} w_k = 1$$

For k = 3:

confluency = ⅓ (12 + 14 +16) = 14

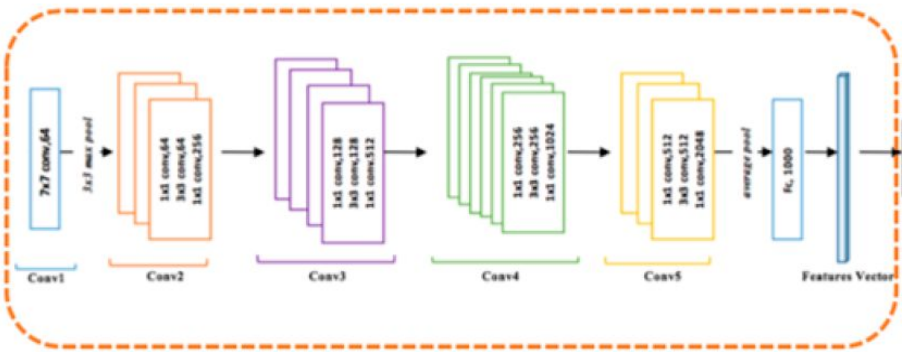
Methodology - Regression

input image
224 x 224



Feature Extraction

ResNet-50 Model



Classification

softmax → class 1%
class 2%
…
class 99%
class 100%

(Cross-entropy loss)

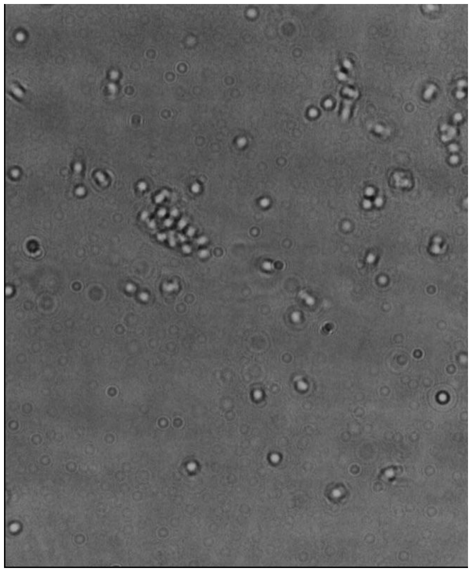
Regression

sigmoid → confluency

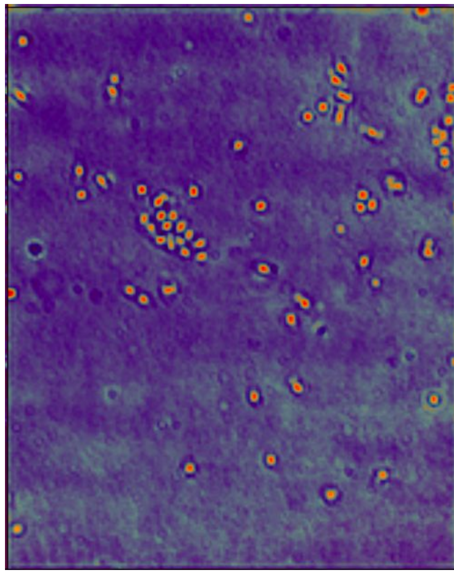
(L1 or L2 loss)

Interpretability

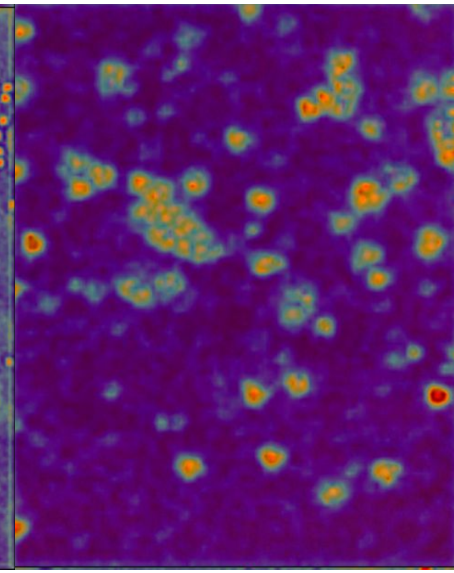
Classification and Regression: **Grad-CAM++**



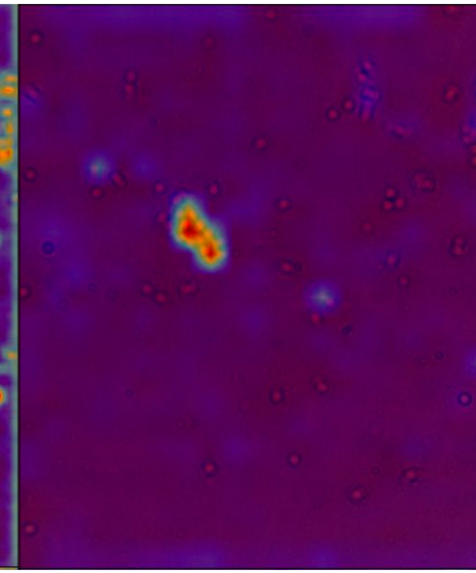
Input image (800 x 1000)



Stage 2 (200 x 250)



Stage 3 (100 x 125)



Stage 5 (25 x 32)

- **Grad-CAM++**: weighted sum of feature maps of a certain convolutional layer. The weights are calculated using backpropagation
- Low and middle level features instead of last convolutional layer feature maps
- The method also works for regression

Experiments & Results

Experiments and results

For a set $\mathcal{I}$ of images with true and predicted confluency values $y_i, \hat{y}_i$ we report

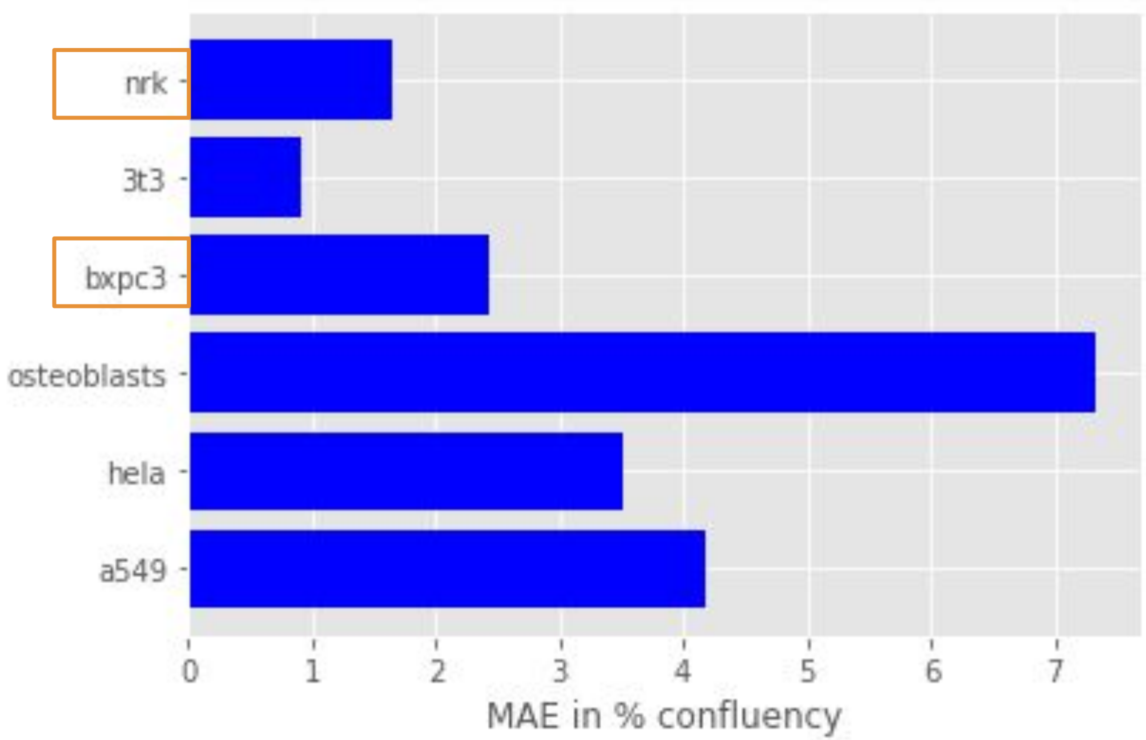
$$\mathrm{MAE} = \frac{100\,\%}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} |\hat{y}_i - y_i|, \quad \mathrm{relMAE} = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \frac{|\hat{y}_i - y_i|}{y_i}$$

Model	Baseline	Count-ception	Classification	Regression
MAE (relative MAE)	11.23 (54.13%)	1.92 (8.77%)	2.99 (11.14%)	1.87 (8.54%)

- Count-Ception: receptive field size 32, no downsampling, L1 loss
- Classification: no downsampling, soft labels with sigma = 0.45 and CE loss, weighted class aggregation for k = 5
- Regression: downsampling rate 0.5, L1 loss

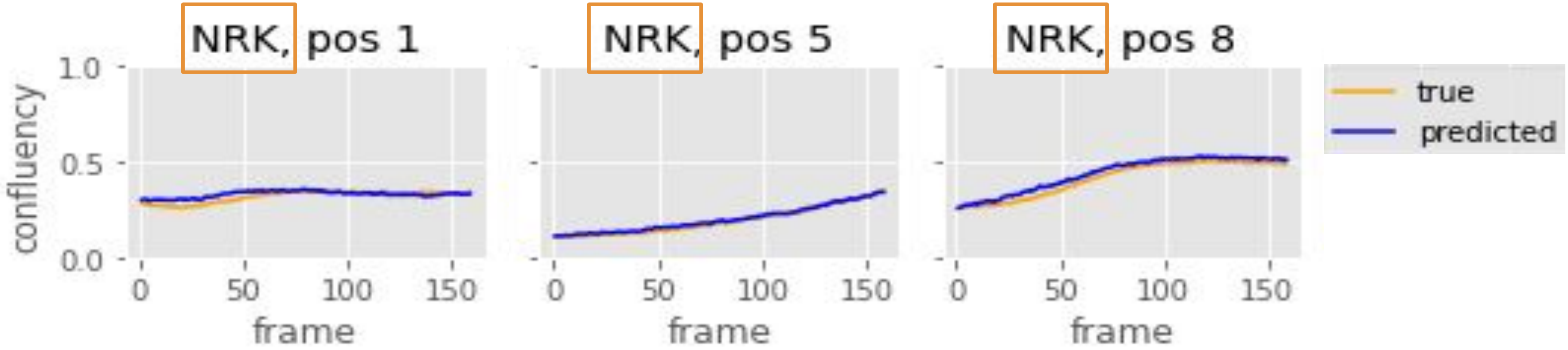
Results - Regression model I

- Generalization ability or robustness

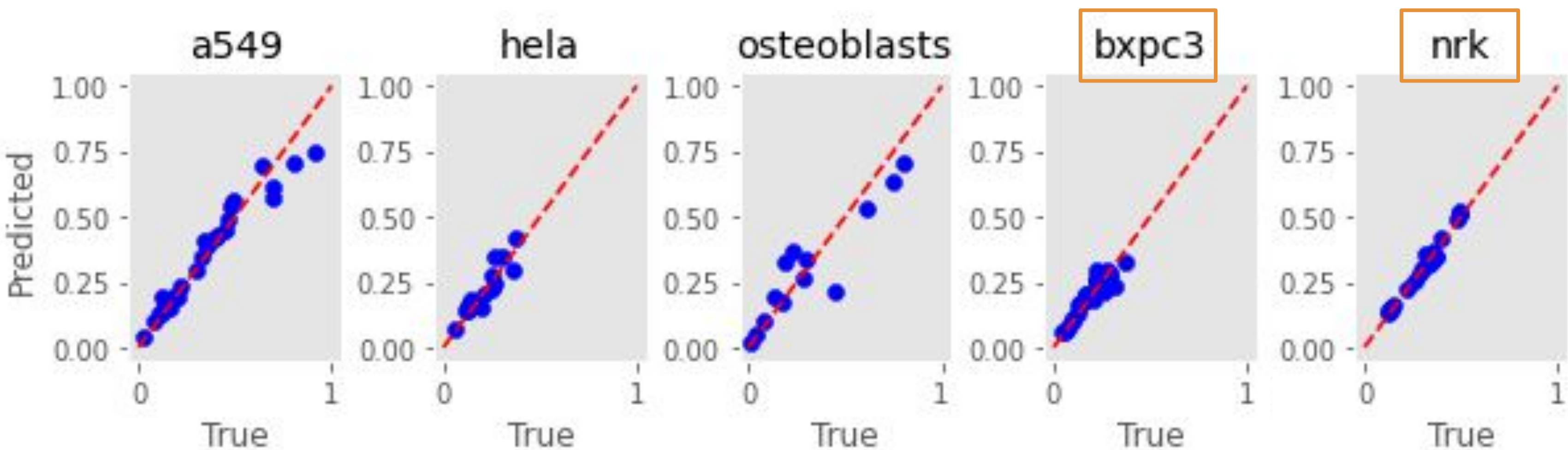


Results - Regression model II

- Time consistency

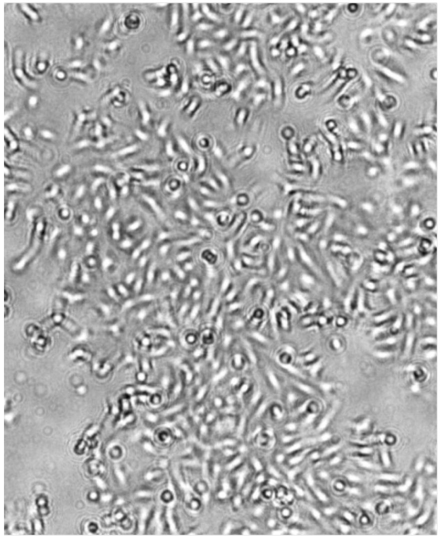


- Bias and error

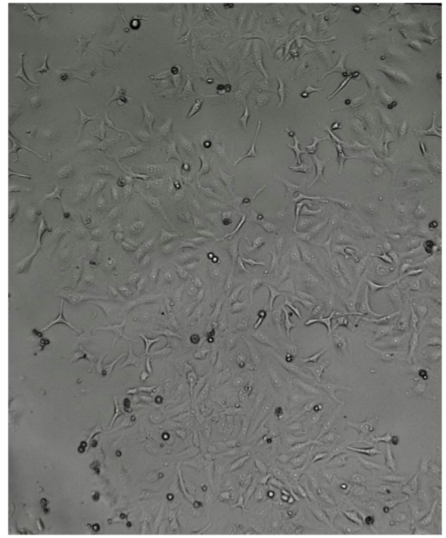


Qualitative evaluation

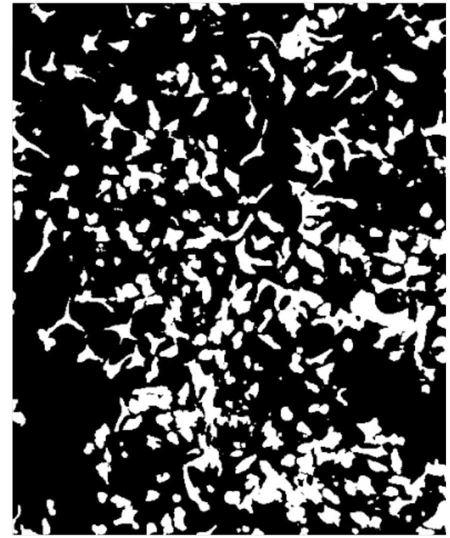
Actual confluency: 21.21 %



Lens-free image

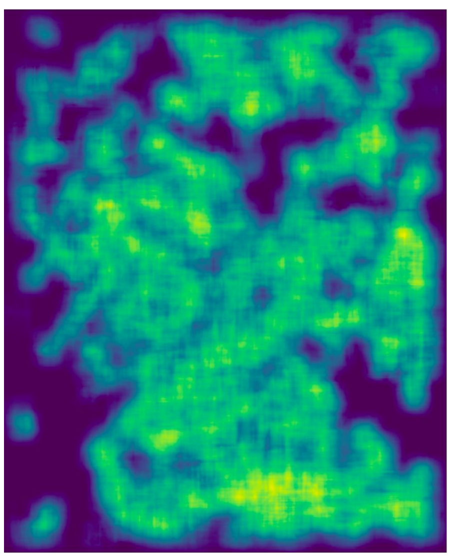


Bright-field image



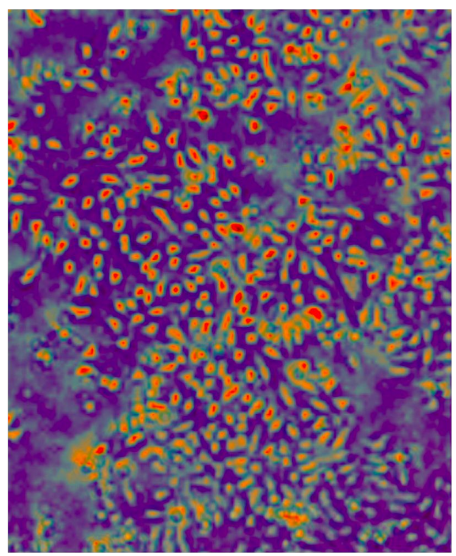
Segmentation mask

Predicted confluency: 41.95 %



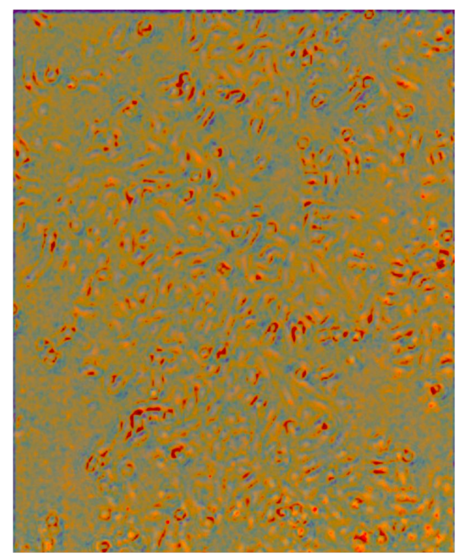
Predicted confluency map

Predicted confluency: 40.56 %



Grad-CAM heatmap for regression

Predicted confluency: 35.73 %



Grad-CAM heatmap for classification

Conclusion & Outlook

Conclusion and outlook

- Adapted state-of-the-art methods for object counting and detection on the lens-free microscopy images
- Innovative application of counting models for area estimation task
- Confluency estimation as a classification and regression problem
- Grad-CAM++ for explaining predictions

Further work

- Use detections for tracking
- Combine counting and detection models
- Evaluate Grad-CAM++ heatmaps for cell counting task
- Evaluate CenterNet and SS-DCNet and for confluency estimation task
- Consider low and middle level features for regression
- Implement attention mechanism to improve interpretability

Thank you for your attention!