UNIVERSITY OF TARTU

Institute of Computer Science

Software Engineering Curriculum

Danver Hans Värv

Mother-bud detection and classification in yeast cells

Bachelor's Thesis (9ECTS)

Supervisors: Mohammed A. S. Ali, MSc

Karla Juárez Núñez, MSc

Tartu 2022

Mother-bud detection and classification in yeast cells

Abstract:

In different organisms, asymmetric cell division is a conserved process in which asymmetric inheritance of cellular components gives rise to two cells with different characteristics. In budding yeast, *Saccharomyces cerevisiae*, after the asymmetric cell division two cells are generated: mother and daughter. Fluorescence microscopy has been widely used to study such microorganisms and hence study the process of cell division. Advances in microscopy have increased output data volumes, making the manual processing of such images expensive. Therefore, developing a pipeline to analyze microscopy images coming from screening experiments would have a high practical impact. In this work, we built a deep-learning-based pipeline for detecting budding (dividing) yeast cells and distinguishing bud from mother cells during yeast division. The final goal is to study if older proteins are being retained on the mother side and whether newly synthesized proteins are inherited towards the bud. The results show that the pipeline was able to detect the cells that are about to divide with an accuracy of 70.42%. Furthermore, 87.72% of the mothers and buds were accurately classified.

Keywords:

deep learning, neural networks, fluorescence microscopy, yeast cells segmentation, mother-bud detection

CERCS:

T111 - Imaging, image processing; P176 - Artificial intelligence; B110 - Bioinformatics, medical informatics, biomathematics biometrics

Ema-tütarrakkude tuvastamine ja klassifitseerimine pärmirakkudes

Lühikokkuvõte:

Erinevates organismides toimub asümmeetriline rakkude paljunemise protsess, mille käigus tekivad kaks erinevate omadustega rakku. Paljunevas pärmis, *Saccharomyces cerevisiae*, tekib pärast asümmeetrilist rakkude jagunemist kaks rakku: ema- ja tütarrakk. Fluoretsentsmikroskoopia on laialdaselt levinud mikroorganismide uurimiseks, kaasa arvatud pärmirakkude vaatlemiseks. Mikroskoopia areng on suurendanud töödeldavate andmete mahtu, muutes mikroorganismide käsitsi uurimise kulukaks. Lõputöö raames luuakse automaatrakendus, mis suudab süvaõpet kasutades tuvastada paljunevaid pärmirakke ning klassifitseerida need ema- või tütarrakuna. Lõppeesmärk on uurida, kas raku paljunemisega jäävad vanad valgud emarakule või päranduvad tütarrakule. Tulemused näitavad, et meie töötatud automaatsüsteem suutis tuvastada 70.42% täpsusega paljunevaid rakke. Lisaks suudab rakendus klassifitseerida ema- ja tütarrakku 87.72% täpsusega.

Võtmesõnad:

süvaõpe, närvivõrgud, fluorestsentsmikroskoopia, pärmirakkude segmenteerimine, pungumise tuvastamine

CERCS:

T111 – Pilditehnika; P176 – Tehisintellekt; B110 - Bioinformaatika, meditsiiniinformaatika, biomatemaatika, biomeetrika

Table of Contents

Introduction

In cell biology, image analysis is a valuable tool for collecting quantitative measurements in time and space with accuracy, speed, and sensitivity (Caicedo et al., 2019). Microscopy images have helped researchers understand biological processes like the effect that environmental conditions have on the phenotype of cells.

Among different imaging techniques, fluorescence microscopy has proven to have advantages over other approaches. While chemical dyes to stain cells can be toxic, using a fluorescent protein to tag a protein of interest has been largely chosen in recent years.

Asymmetric cell division is a widespread process in all organisms, ranging from bacteria to mammals (Venkei & Yamashita, 2018). This is a process whereby asymmetric inheritance of cellular components gives rise to two cells with different characteristics and fates (Shahriyari & Komarova, 2013). After each asymmetrical division in budding yeast, *Saccharomyces cerevisiae*, two new cells are born: mother and daughter. While the mother cells have a larger size (Hartwell & Unger, 1977), they can undergo a lesser number of replicative cell cycles (Steinkraus et al., 2008). The daughter cells are age-free (Higuchi-Sanabria et al., 2014), and their cell cycle is longer than their mothers' previous cell cycle (Hartwell & Unger, 1977). This has made the budding yeast a model organism to study cellular aging.

Fluorescence microscopy data is often recorded and saved as images utilizing digital imaging devices attached to fluorescence microscopes (Riffle & Davis, 2010). An active research topic is developing automated tools to examine fluorescent images (S.-C. Chen et al., 2007). Pipeline analysis of microscopy images could be utilized as a high-throughput tool to identify factors involved in different biological processes.

Deep learning methods are state of the art in most computer vision tasks like image classification (Krizhevsky et al., 2012), object detection (Redmon et al., 2016) and segmentation (L.-C. Chen et al., 2018). Deep learning has transformed medical imaging over the last decade because of its capacity to automate repetitive activities and integrate large amounts of data to make accurate predictions (LeCun et al., 2015).

High-throughput screening is a new approach used to look through large biological data sets to understand a given biological question. Therefore, developing a pipeline to analyze microscopy images from screening experiments is needed to extract as much detailed information as possible. For the biological question of this project, the detection and classification of the mother and bud compartments within a dividing pair is vital. In order to analyze if older proteins are being retained at the mother side and whether newly synthesized

proteins are inherited towards the bud, we need to be able to quantify fluorescence within each segmented compartment.

Contributions

- Classification of two different populations of cells per microscopy image. This classification was performed using blue fluorescent proteins of different intensities to distinguish between two cell populations: negative control and sample cells.
- Identification of subpopulations within the sample cells group. The images analyzed correspond to images from budding yeast during the exponential phase when they are dividing. Nevertheless, as the yeast cultures were asynchronous, in the images we can find the majority of cells dividing but there are also some single cells present. This is why we needed to identify only dividing cells within the sample cell population.
- Classification of compartments within dividing pairs. Within the dividing pairs of cells, we needed to identify which cellular compartment corresponded to the mother cells (bigger) and which one was the daughter cell/bud (smaller).

Outline

Background chapter introduces the fundamental concepts required to understand the thesis. It provides an overview of budding, fluorescence microscopy, image processing, deep learning, evaluation metrics and related work.
Dataset description gives an overview of what kind of data is used for the thesis.
Methodology introduces the approaches investigated to solve the budding yeast problem in fluorescent microscopy images.
Results and experiments discusses the results and experiments of detecting budding yeast.
Conclusion summarizes the thesis and discusses future work.
References include the literature used for the thesis.
Appendix includes the license.

Background

This chapter introduces the key concepts required to understand the thesis. It provides an overview of budding yeast cells, fluorescence microscopy, deep learning, image segmentation, evaluation metrics and related work.

Budding in yeast cells

Yeasts are single-celled fungi that reproduce asexually by budding, in which a small bud emerges from the mother cell (Wang et al., 2017). The bud grows in size until becoming an independent cell, which is called the daughter cell. The division occurs at the neck, where the mother and bud are connected (Figure 1). The budding yeast cells are around 5 to 10 μm in size; thus, microscopy is widely used to study biological processes in yeast (Zakhartsev & Reuss, 2018).

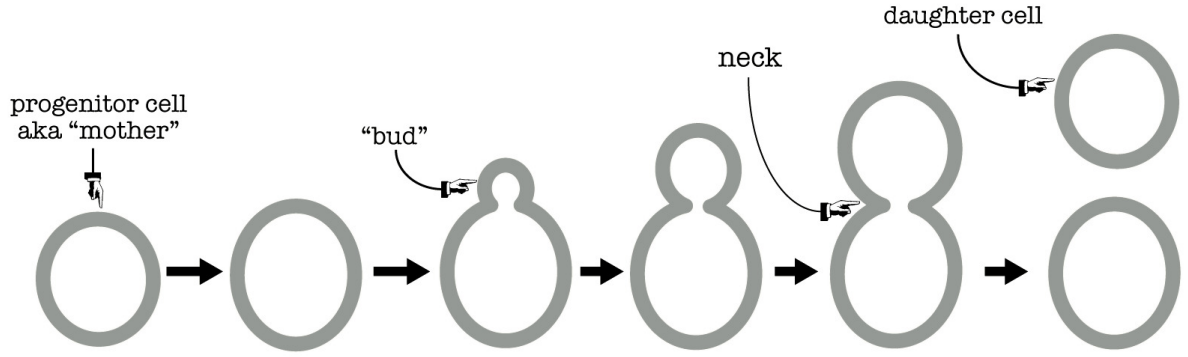

Figure 1. Yeast cell reproduction (*Collins Lab Blog*, 2010)

Fluorescence microscopy and its use in biology

In fluorescence microscopy, fluorophores emit light by absorbing a particular wavelength, making cells easily visible to the human eye aided by microscopes (Yuste, 2005). Fluorescence microscopy is a widely used imaging modality among molecular biologists, allowing the observation of a wide range of biological processes in a cell (Haider et al., 2016). Fluorescent labeling using genetically encoded fluorescent proteins located next to the protein of interest has made it possible to study the localization of proteins under different cellular conditions (Baggett et al., 2003). This has allowed us to deepen the study of protein functionality in all organisms. Depending on the goal of the experiment and on how many proteins of interest one aims to study at once, different colors of fluorescent proteins can be

used simultaneously (Jensen, 2012). Recently there was a dramatic increase in the volumes of microscopy data which makes the processing of such images very expensive (Ali et al., 2022).

Deep learning

Deep learning refers to algorithms that evaluate data using a logic structure comparable to how a person would draw conclusions. Deep learning applications use a layered structure of algorithms known as an artificial neural network (ANN) (Du et al., 2016). The architecture of such an ANN (Figure 2) is based on the biological neural network of the human brain, resulting in a significantly more competent learning process than typical machine learning models (Wolfewicz, 2022).

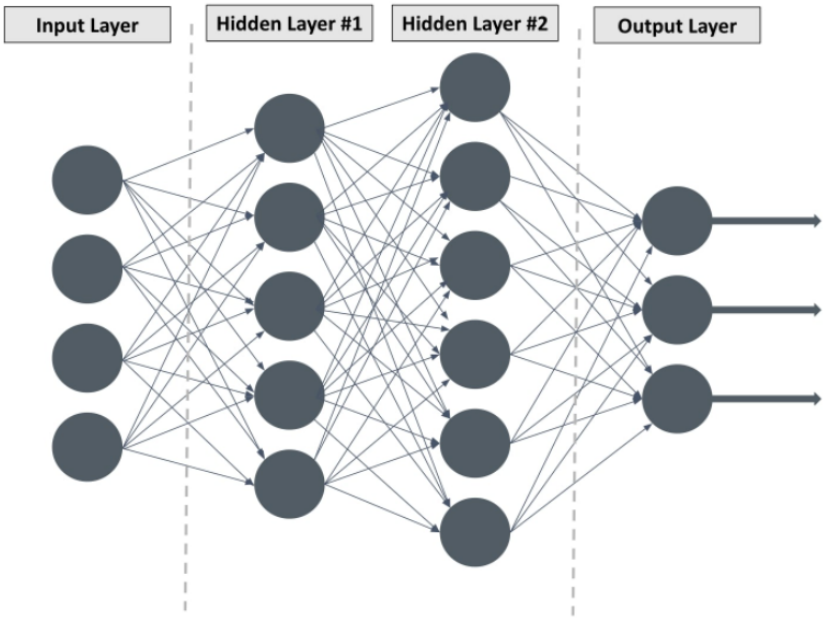

Figure 2. A simple artificial neural network (Wolfewicz, 2022).

Deep learning allows computational models with several processing layers to learn multiple levels of abstraction for data representations (LeCun et al., 2015). Deep neural networks consist of multiple layers of interconnected nodes, each building upon the previous layer to refine and optimize the prediction or categorization (Wolfewicz, 2022). A deep neural network's input and output layers are called visible layers. The input layer is where the deep learning model ingests the data for processing, and the output layer is where the final prediction or classification is made. A hidden layer is located between the input and output of the algorithm, in which the function applies weights to the inputs and directs them through an

activation function as the output (Wolfewicz, 2022). In short, the hidden layers perform nonlinear transformations of the inputs entered into the network. In general, any ANN with two or more hidden layers is referred to as a deep neural network.

Convolutional neural network

One of the most popular deep neural networks is the Convolutional Neural Network, also known as CNN. This name comes from the linear mathematical operation between matrices called convolution. Convolutional Neural Network has outperformed performance in the applications that deal with image data (Albawi et al., 2017). CNN has three main layers: convolutional, pooling and fully-connected layer (Figure 3).

The convolutional layer is the first layer in CNN. At each layer, the complexity of CNN increases. The earliest layers focus on simple features - colors, lines and edges. The convolutional layer requires input data, a filter and an output array (IBM Cloud Education, 2020). The filter is a two-dimensional (2D) array of weights representing a part of the image. The filter is applied to input data, shifting horizontally and vertically by a stride (Figure 3). The process is repeated until calculation has been done between input pixels and the filer. The final output from the series is also known as the activation map or feature map.

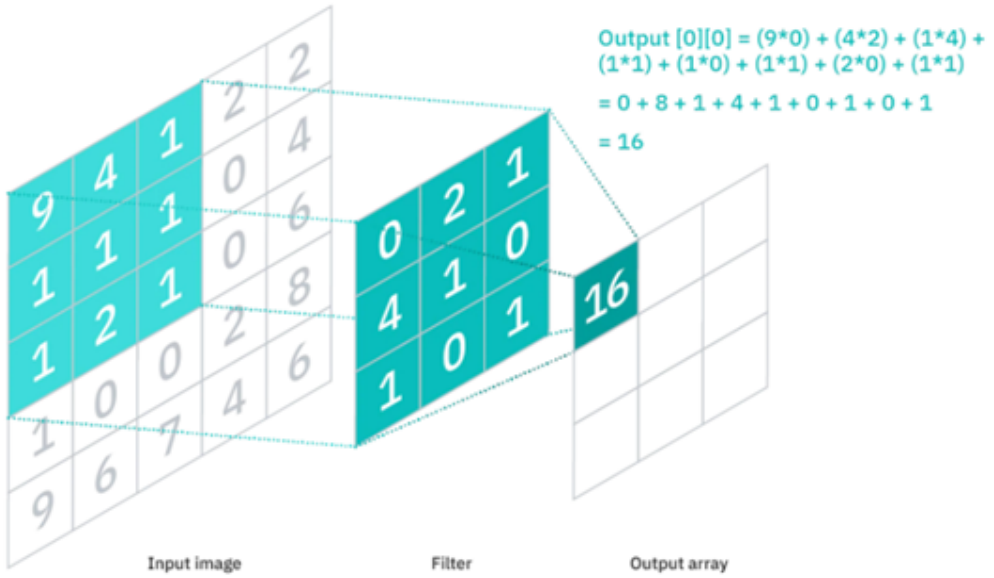

Figure 3. Workflow of Convolutional Neural Network

The pooling layer is also known as downsampling, reducing the number of input parameters. Like the convolutional layer, the pooling operation passes through the filter across the input, but this filter has no weights. There are two main types of pooling: max and average pooling. While the filter moves across the input, max-pooling calculates pixels with maximum value and average pooling with average values (IBM Cloud Education, 2020). Pooling helps to reduce complexity, improve efficiency and thus limit the probability of overfitting.

In the fully-connected layer, each node in the output layer connects directly to a node in the previous layer (IBM Cloud Education, 2020). This layer performs classification tasks based on the features extracted beforehand. Fully-connected layers use activation functions to classify the inputs and create a probability of 0 to 1.

Image segmentation

Since manually processing cells is impractical, automated delineation of cells is needed to standardize the analysis of big imaging datasets (Moen et al., 2019). In image analysis, the first step is to separate cell regions from the background, known as image segmentation. Image segmentation is a process in which an image is clustered into regions based on pixels that help to separate objects in the image (Matcha, 2021). There are two types of segmentation techniques – semantic and instance segmentation. Semantic segmentation separates the foreground from the background and returns a pixel-wise mask for each object on the image. Instance segmentation differs from semantic segmentation because it gives a unique label to every instance of a particular object in the image.

Related work

Several image-based yeast cell morphology detection algorithms have been published earlier. Yu *et al.* developed an image processing algorithm that is not dependent on fluorescent staining and can distinguish between non-budding, small bud and large bud yeast cells (Yu et al., 2011). The algorithm uses a ratio between the major and the minor axes to split dividing cells into mother and buds. Koschwanez *et al.* reported an imaging system using a fibre-optic imaging bundle to classify budding and non-budding yeast cells (Koschwanez et al., 2004). This system cannot absorb information on the mother or bud level.

Evaluation metrics

This section introduces the main performance metrics used for evaluating the outcome of the experiments.

Main performance metrics

Classifying objects or pixels can have four outcomes:

- True positive (TP) - object or pixel was correctly predicted as positive;
- False positive (FP) - object or pixel was incorrectly predicted as positive;
- False negative (FN) - object or pixel was incorrectly predicted as negative;
- True negative(TN) - object or pixel was correctly predicted as negative.

It is important to keep in mind that true negative (TN) does not apply in the context of object detection since there are infinite numbers of bounding boxes that should not be detected (Padilla et al., 2020).

Object detection methods are primarily evaluated by using precision and recall. Precision measures only relevant objects, calculating the percentage of correct positive predictions (TP/TP+FP). Recall calculates the percentage of correct positive predictions out of all the available ground truths (TP/TP+FN). In addition, F1 score (2*TP/(2*TP + FP + FN)) is used to express harmonic mean of precision and recall (Lipton et al., 2014).

Intersection over Union (IoU)

Intersection over Union is used for evaluating correct and incorrect identifications. The IoU measures the area of overlap between segmentation and the ground truth divided by the area of union between segmentation and the ground truth (Figure 4).

$$IOU = \frac{\text{area of overlap}}{\text{area of union}} =$$

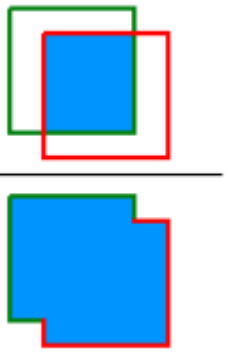

Figure 4. Illustration of Intersection over Union (Padilla et al., 2020)

We can classify a detection as correct or incorrect by comparing the IoU to a given threshold t. If IoU >= t, the detection is considered correct, if IoU < t, the detection is considered incorrect (Padilla et al., 2020).

Dataset description

To determine patterns of protein inheritance based on protein age in budding yeast (*Saccharomyces cerevisiae*), we used a dataset of fluorescence microscopy images. This dataset was generated through a high-throughput process in which all ORFs (proteins) in the yeast genome (around 6000 proteins) were tagged with a tandem-fluorescent protein timer (tFT). The tFT is a fusion of a red and a green-fluorescent protein (red-green FPs). Those FPs have different maturation times, green being a fast-maturing protein and red being a slower maturing protein. The ratio of red/green intensity is used to estimate protein age.

We used a batch of images from the genome-wide dataset available. In this dataset, the fluorescence microscopy images of yeast cells were acquired using four different channels: DAPI channel, green channel, red channel and brightfield channel. Due to the aims of the present work, only the DAPI channel was used. As seen in Figure 5, we have two cell populations per image: negative control and sample cells. The negative control cells express an intense blue fluorescent protein in the cytoplasm, while the sample cells express a dimmer blue fluorescent protein in the same location. The sample cells were also tagged with an intense blue fluorescent protein at the bud neck to identify cells that are dividing pairs from cells that are just close to each other. In each imaged well, a different protein was tagged with a fusion of red-green fluorescent proteins, so only sample cells will have any signal under these respective channels. The control cells will be used to have an average of the autofluorescence levels of the yeast cells when under the different fluorescent channels.

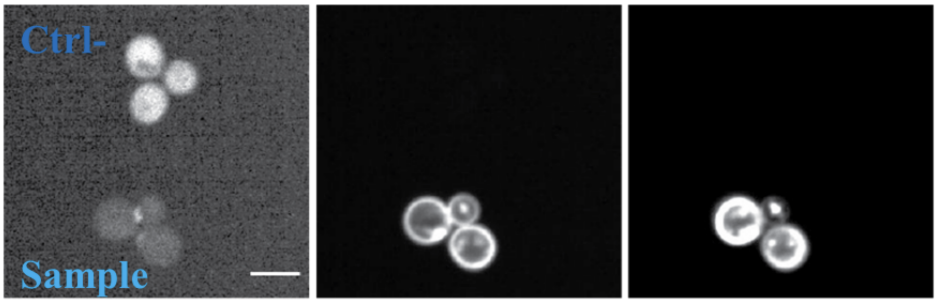

Figure 5. Final blue cytosolic markers for cell segmentation. From left to right, the DAPI channel used in this work shows negative control and sample cells; green and red channels indicate sample cells that have been tagged with red-green protein. Only sample cells have signals under respective green and red channels.

For cell segmentation, the DAPI channel, used to detect the blue fluorescence, was utilized to recognize cell shape and to classify cell populations based on the different blue intensities (Figure 6). We worked with 20 annotated fluorescence images of yeast. Twelve images were used for training, two for validation and six for testing. All of the images and corresponding masks had a size of 2160×2160 pixels. This dataset is referred to as the "mother-bud yeast" dataset.

We also used a dataset of yeast fluorescent microscopy images for pre-training with the name "yeast cell segmentation" dataset. In this dataset, the fluorescent microscopy images of yeast cells were acquired using four different channels: DAPI channel, green channel, red channel, and brightfield channel. Like the previous dataset, only the DAPI channel was used. Unlike the previous data, we have only one cell population per image. We worked with 10 annotated fluorescent images of yeast cells of size 2160×2160 pixels for pretraining.

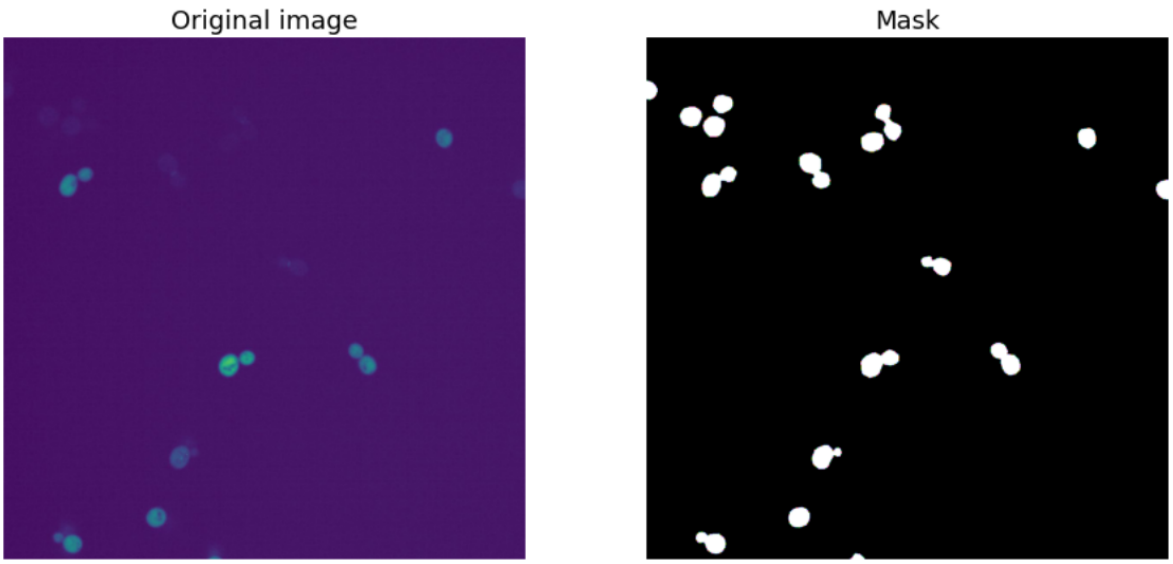

Figure 6. Example of a fluorescence image (left) with its corresponding ground truth mask (right).

In addition, we used the seven cell lines dataset from PerkinElmer. The dataset has seven types of cells including human cells from breast cancer (MCF7), fibrosarcoma (HT1080), cervical cancer (HeLa), hepatocellular carcinoma (HepG2), alveolar basal epithelial (A549), dog cells from kidney tissue (MDCK), and mouse embryonic fibroblast cells (NIH3T3). The dataset has, in total, 3024 images of size 1080×1080 pixels. A detailed description of this dataset can be found here (Fishman et al., 2021).

Methodology

This chapter introduces the approaches investigated to solve the budding yeast problem in fluorescent microscopy images.

Pre-processing

To train the deep learning models, we started by preprocessing the input data (Figure 7). We noticed that the illumination is not evenly distributed across the field of view. Illuminations are corrected using flat-field correction. We calculate the median image from the training set and divide each image with the median image to renormalize intensities across an image. All fluorescent image pixels are then log-normalized to bring high-intensity values of negative control cells closer to sample cells' low-intensity values (Figure 7). Eventually, all pixel values are normalized between 0 and 255.

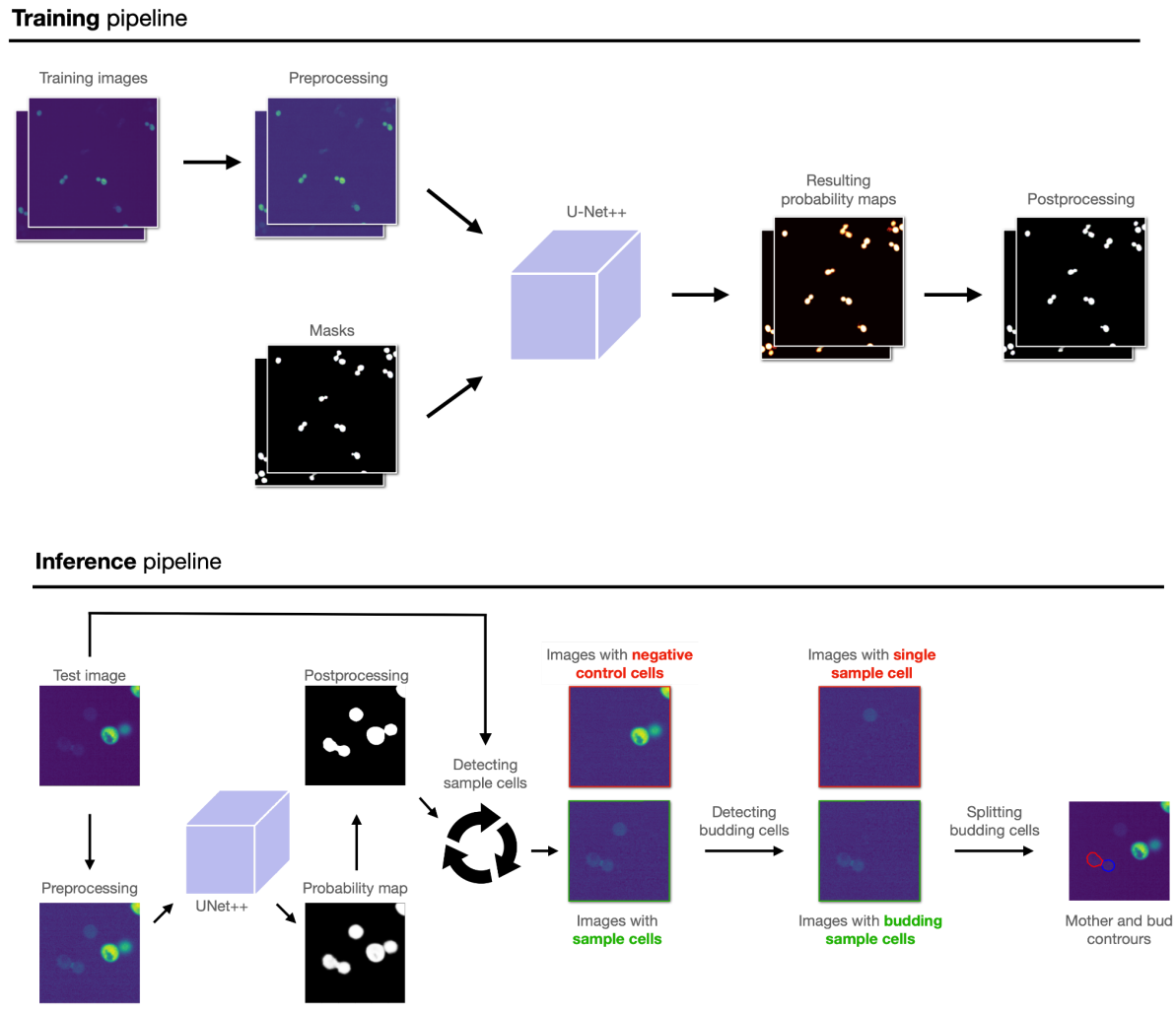

Figure 7. Pipeline of mother-bud classification and detection in yeast cells. During training (top), preprocessed images with corresponding binary masks are used to train the U-Net++ segmentation model. During the inference (bottom), the trained U-Net++ is used to segment yeast cells from the images. After probability maps are post-processed, sample cells and single cells are filtered out; the mother is separated from the budding compartment of dividing sample cells.

Network Architectures

UNet++

The encoder-decoder architecture has become more common in feature extraction due to its great flexibility and superiority (Lei et al., 2021). UNet++ (Figure 8) has a series of nested dense convolutional blocks that connect an encoder (contraction path) and a decoder (expansion path) (Zhou et al., 2018). There are four blocks in both the contraction and expansion paths. Each block consists of two convolution layers, batch normalization (Ioffe & Szegedy, 2015), 2 × 2 max-pooling in the downsampling path, and transposed convolution in the upsampling path (Ali et al., 2021). The bottleneck is formed with a similar block but without max pooling. U-Net++ has, in total, approximately 9 million trainable parameters.

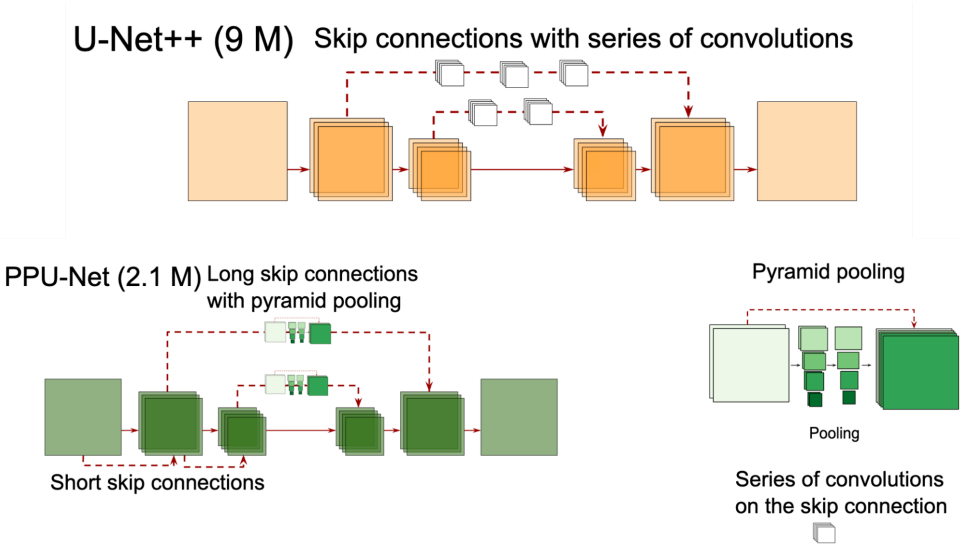

Figure 8. Neural network architectures used in the work. Top) a series of convolutions in the skip connection layer is introduced by U-Net++. Bottom) PPU-Net introduces the pyramid pooling module (bottom-right) between downsampling and upsampling paths and uses skip connections between downsampling blocks (Ali et al., 2021). Next to architecture names in the brackets are the number of trainable parameters.

PPU-Net

PPU-Net (Figure 8) architecture has a contraction path, an expansion path, a bottleneck between them, and a skip pathway connection (Ali et al., 2021). The pyramid pooling module in the skip pathway processes the contraction path block output and concatenates it to the input of the corresponding block in the expansion path. PPU-Net has, in total, approximately 2.1 million trainable parameters.

Model training

The models used in this work were pre-trained on a “seven cell lines” dataset that contains seven types of mammalian cells (Ali et al., 2022). We fine-tuned those pre-trained models with 12 manually annotated fluorescent microscopy images of yeast. Both architectures were trained for up to 500 epochs. The initial learning rate was 3e-4, and once the validation loss had not improved for ten consecutive epochs, it was lowered by a factor of ten. If validation loss did not improve for 20 consecutive epochs, training was stopped entirely. The input size for both models was 288 x 288 pixels with batch sizes of 8. The Adam optimizer (Kingma & Ba, 2014) and cross-entropy loss were used to train both models. All experiments were carried out on a Tesla V100-PCIE-16GB GPU.

Post-processing

The model's image probability maps were post-processed in order to evaluate the models. The results are based on binarized pixel-level outputs at a 0.7 cutoff. Objects under observation are identified with the help of *skimage.measure.label (Van der Walt et al., 2014)*. The label function from the skimage package connects regions from pixels that have the same values as their neighbor pixels. Holes less than 25 pixels are filled for more convenient results. Objects smaller than 800 pixels can be removed because budding cells are larger than the given threshold.

Detecting sample cells

We are only interested in the budding cells from the sample cell population and not in the negative control. Therefore, we separated the two cell populations after image probability maps were post-processed. We used the Otsu method to get a single image threshold for classifying each object in the picture. Otsu threshold is calculated by maximizing the

between-class variance and dividing pixels into two different classes (Liu & Yu, 2009). Sample cells are differentiated from negative control cells by filtering out objects that have an average pixel value less than Otsu's threshold, leaving us with only sample cells.

Detecting budding cells

After filtering out negative control cells, we detected the budding sample cells. Detecting budding cells requires filtering out cells that do not have a bud. Hence, the shape descriptors were used to look for peanut-like cells, indicating the cells that are about to divide. We, therefore, used solidity from *skimage.measure.regionprops* function from the skimage package, which finds the ratio between pixels in the area to the pixels of the convex hull image. If this ratio (solidity) is lower than 0.95, it indicates that the cell is less circular, dividing it (has a bud).

We noticed that cells could sometimes be quite near to one another. This results in large clusters of cells that the segmentation model fails to separate. Therefore, we decided to leave out cell clumps that are more than 3000 pixels by area.

Splitting buds from mothers

After negative control cells, big clumps, and single cells have been removed, only dividing sample cells are left. Now we need to separate the mother from the bud compartment. Two algorithms are discussed in this section for splitting the dividing cell.

Splitting from neck marker location

When there is a dividing cell pair, a neck marker is visible in between the mother and the bud compartments. The place where bud emerges from the mother cell has high-intensity value pixels (Methodology), making it look bright in the image (Figure 9a).

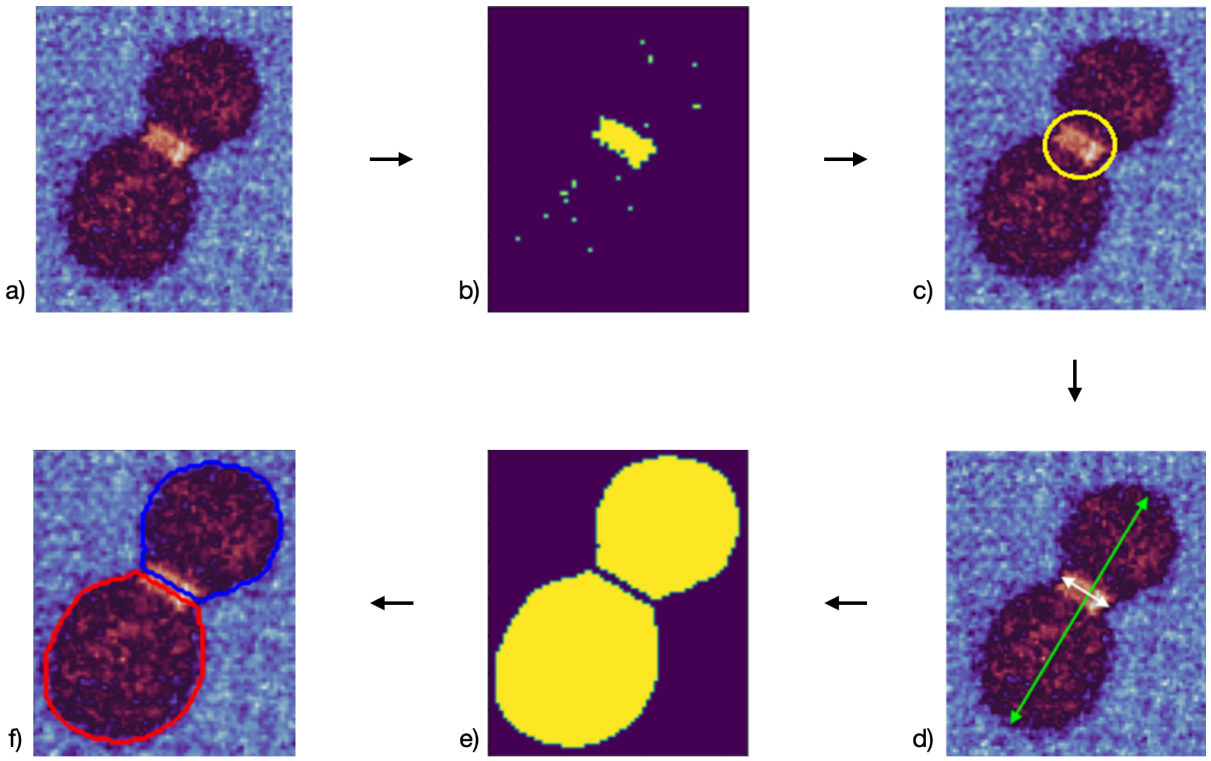

Figure 9. Pipeline of splitting budding cell to mother and bud in microscopy fluorescent images. **A**, one dividing cell pair. **B**, binary objects are created after filtering out 5% of the highest pixel values. **C**, the neck marker centroid is detected by finding the largest component centroid from the binary image. **D**, finding the major axis (green) and the perpendicular line to the major axis (white) which is passing through the neck marker centroid. **E**, splitting the dividing cell into two objects by setting perpendicular line (white) pixel values to 0. **F**, detecting the mother (red contour) and daughter (blue contour).

Using the coordinates of the segmented cells, we first patch each cell in the image separately. The location of the dividing cell neck marker is determined by filtering out 5% of the highest pixel values from the image patch (Figure 9b) and then determining the largest connected component centroid (Figure 9c). If the most significant connected component is smaller than 30 pixels, the processing is stopped. Second, the longest axis of the dividing cell is computed using the *skimage.measure.regionprops* from the skimage package. Third, a perpendicular line to the major axis is drawn, passing through the previously calculated neck marker location (Figure 9d). The pixels covered by the newly drawn perpendicular line are all set to 0 on a binary image, resulting in the creation of two objects from the budding cell (Figure 9e). Finally, a larger cell is allocated to the mother and a smaller cell to the bud (FIgure 9f).

Splitting using watershed

Watershed segmentation is one of the most common algorithms used to process medical images (Kornilov & Safonov, 2018). The algorithm requires the user to define markers from which basins are flooded. The method floods basins from the markers until basins from various markers intersect on watershed lines.

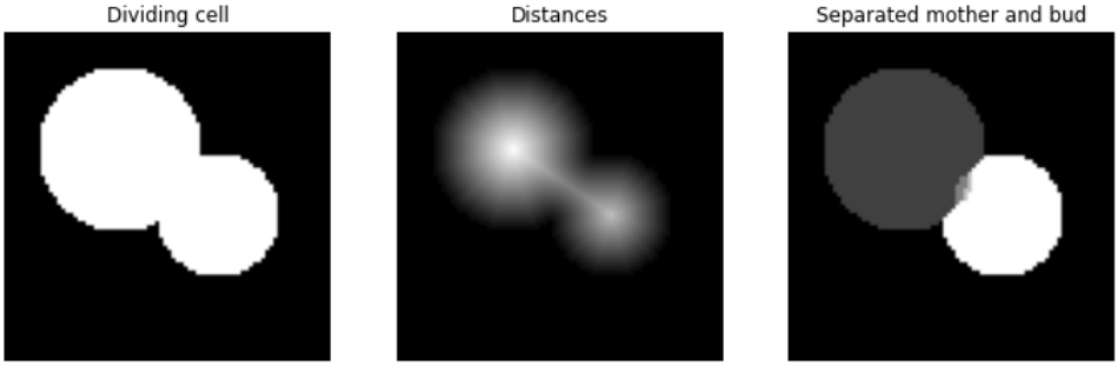

Figure 10. Example of dividing cell mask (left), distance to the background and separated dividing cell (right).

Two seeds representing mother and bud need to be given as input to the water-shedding algorithm. We calculated the distance between each cell's pixel and the nearest background pixel to get those seeds. Only two of the highest values of those groups were kept and used as seeds for the watershed algorithm. (Figure 10, middle image). In case there is only one seed detected, meaning that there is actually no peanut-like shape, then the splitting process is stopped. Flooding of basins separates the two cells along a watershed line (Figure 10, rightmost image).

Results and experiments

To detect and split mothers from buds in yeast cells, we tested two state-of-the-art deep learning segmentation models; we used two datasets and applied a series of image analysis techniques.

Cell segmentation

A common foundational step in our pipeline is to delineate cells. For that, two deep learning models were used, UNet++ and PPU-Net. To test domain shift, both models were pre-trained on two different datasets (source datasets). The UNet++ was pre-trained on the “yeast segmentation” dataset, while the PPU-Net was pre-trained on the “seven cell lines” dataset to segment mammalian cells' nuclei. We then fine-tuned both models using the “mother-bud yeast” dataset, leaving us four models to experiment with.

	Object-wise f1	Object-wise IoU	Pixel-wise precision	Pixel-wise recall	Pixel-wise f1	Pixel-wise IoU
UNet++	0.70021	0.62171	0.84071	0.91054	0.87422	0.77655
UNet++ *	0.80229	0.70063	0.93084	0.85707	0.89243	0.80576
PPU-Net	0.41024	0.29215	0.78499	0.50371	0.61365	0.44264
PPU-Net *	0.79213	0.65732	0.91459	0.85162	0.88198	0.78888

Table 1. Quantitative performance comparison of four segmentation models on yeast microscopy images. Models using transfer learning are marked with an asterisk (*). The best results of each metric are highlighted in green.

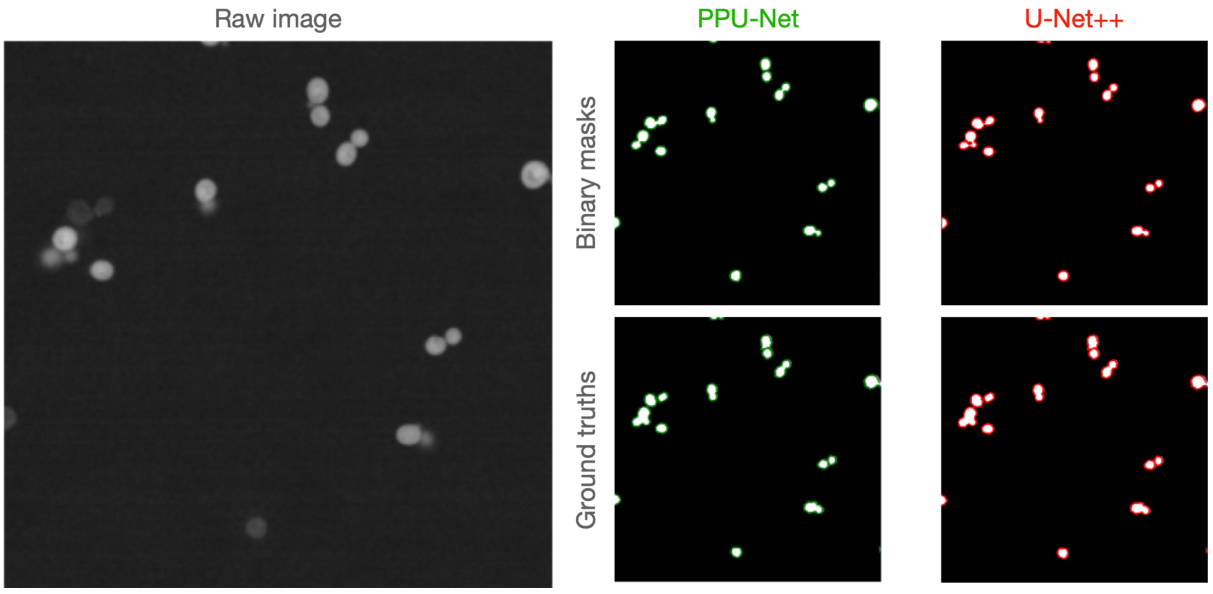

Figure 11. Example of fluorescent raw image matched with corresponding ground truths (bottom row) and binary masks binarized at 0.7 threshold from PPU-Net and U-Net++ (top).

Interestingly, the UNet++ pre-trained on the "yeast segmentation" dataset has the best segmentation performance (Table 1), with a pixel-wise f1 score of 0.89243. These results suggest that when the domain shift between the source and target data is small, more knowledge is transferred to the target.

We also noticed that having the U-Net++ trained on the "yeast segmentation" dataset with no fine-tuning performs better than the PPU-Net pre-trained on the "seven cell lines" without fine-tuning, U-Net++ pixel-wise f1 score is 0.87422 compared to PPU-Net 0.61365. Most of the time, an inspection of the results qualitatively, fine-tuned UNet++ and PPU-Net model predictions align with the ground truth (Figure 11).

Classifying sample and negative control cells

The next step is to filter out the negative control cells. We, therefore, classified the negative control and the sample cells using Otsu's method for thresholding. For the evaluation, we used the test split.

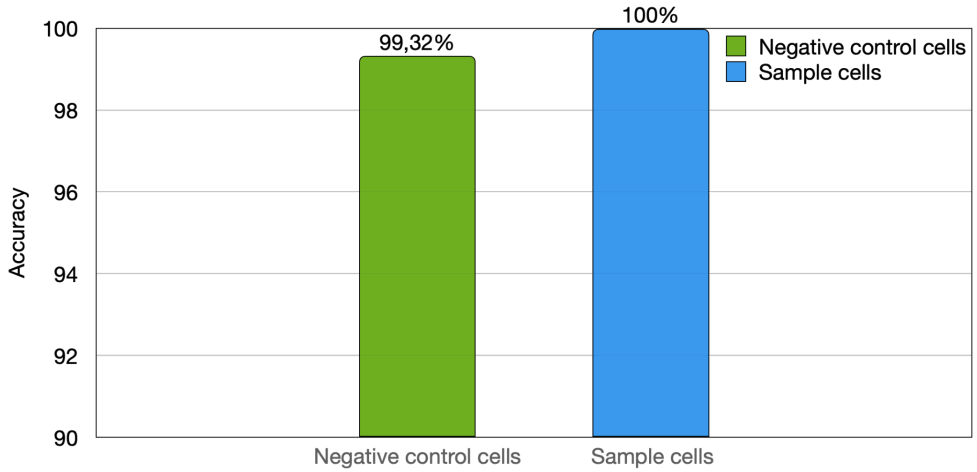

Figure 12. Sample cells and negative control cells classification performance.

We found that sample cells can be well separated from negative control cells. We found that 100% (165 out of 165) of the sample cells and 99.32% (147 out of 148) of the negative were classified correctly (Figure 12). The high performance of distinguishing the two cell populations is attributed to the average pixel values between the sample, and negative control cells are far apart. Therefore it was easy for Otsu's method to choose the best performing threshold.

Identifying budded (dividing) cells

We are only interested in dividing cells (budded cells) within the previously identified population of sample cells. Therefore, we identified within this population two subpopulations: dividing and single cells.

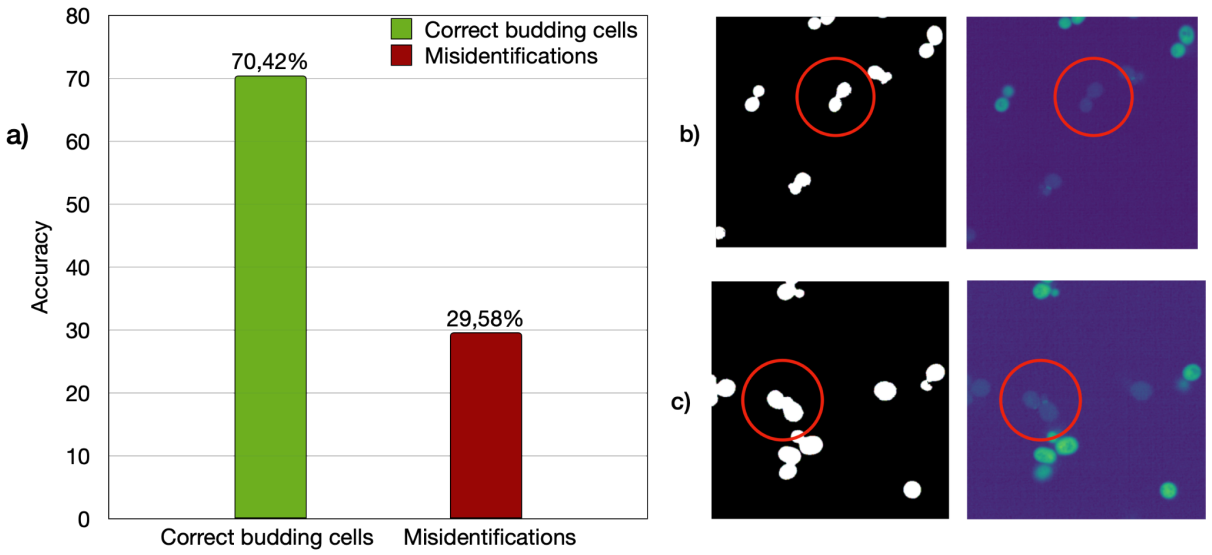

Figure 13. **A**, the result of classifying dividing/budding cells. Correct budding cells have a bud neck marker in between mother and bud. **B**, misidentification of two single cells as one budding cel. **C**, segmentation failure where the dividing sample cell is very close to another single cell.

The accuracy of identifying budding cells based on their size, shape, and intensity was 70.42% (50 out of 71). We experienced 21 misidentifications (Figure 13a). The main reason for not detecting some of the budding cells was the formation of cell clusters where cells were so close to each other that identification was impossible. In 13 cases, the algorithm confused two adjacent single cells with one budding cell (Figure 13b). In this case, segmentation failed to detect budding cells. In addition, we considered 5 cases as misidentifications where there was a budding cell very close to another cell and therefore forming clumps (Figure 13c). In addition, three budding cells were not detected since they were enormous and did not fit the size threshold. Not including 13 adjacent single cells detected as budding cells, the algorithm detected 86.2% of actual budding cells on the image (50 out of 58).

Mother-bud split and classification

Finally, we need to identify buds and mothers in each dividing/budding cell pair. For this, we used two algorithms, namely a neck-marker-based algorithm and a water-shedding-based. From the previous experiment, we are left only with the sample cells that are dividing.

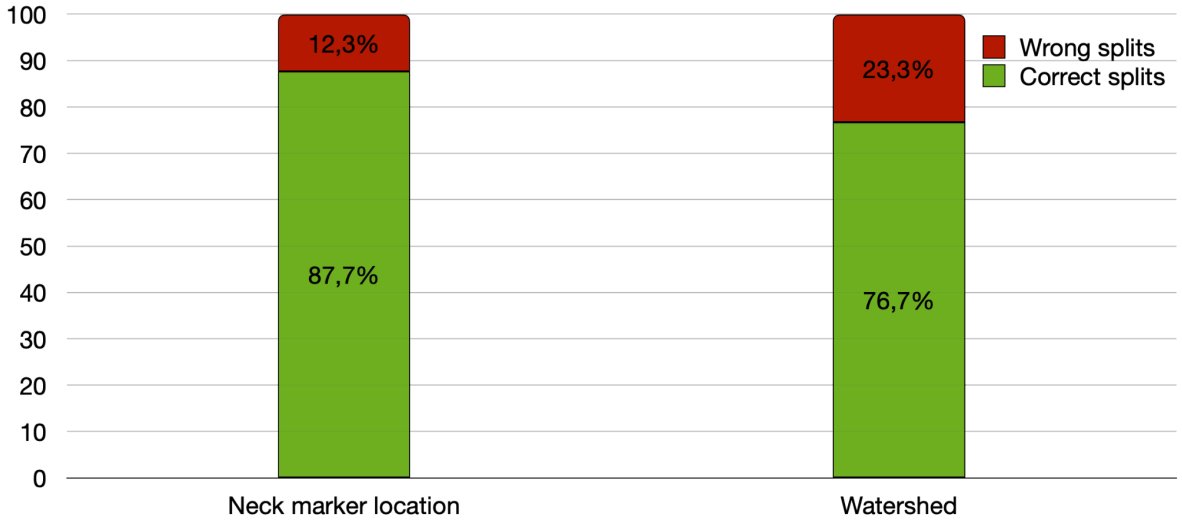

Figure 14. Comparison of performance for mother-bud identification using a neck marker location and the watershed method.

From the previous step, we have 68 dividing sample cells, but 23.53% of them should have been filtered out due to segmentation errors (Figures 13b and 13c). Therefore each algorithm tried to filter out such cases. From the segmentation errors, the watershed algorithm filtered out 50% (8 out of 16) and the neck-marker-based algorithm 68.75 % (11 out of 16). Splitting from neck marker location splits correctly 87.72% (50 out of 57) of the mother-bud pairs (Figure 14). The remaining errors are caused by segmentation failures, which means there is a visible dividing cell with a marker, but due to another cell being very close, both cells are segmented as one cell (Figure 13c). Splitting the mother from the bud using the watershed method resulted in 76.67% accuracy (46 out of 60). Compared to the neck-marker-based algorithm, the watershed method does not look for the neck marker, resulting in more misidentified cells. The watershed algorithm filters out samples with no possibility of using two seeds, meaning that objects are tiny and do not have a peanut-like shape.

Conclusion

The goal of the thesis was to build a pipeline for mother-bud detection and classification among sample cells. For this, we conducted four main experiments.

In the first experiment, we evaluated segmentation performance using UNet++ and PPU-Net network architectures. The UNet++ pre-trained on the “yeast segmentation” dataset has the best segmentation performance. These results suggest that when the domain shift between the source and target data is small, more knowledge is transferred to the target.

In the second experiment, we classified negative control and sample cells using Otsu’s method for thresholding. The results show that we were able to classify almost every cell correctly.

In the third experiment, we detected the dividing sample cells per image. If the segmentation did not fail with adjacent cells, the use of solidity and area showed high performance. One idea is to improve the segmentation by using data augmentation to provide more data for training.

We compared our proposed neck-marker-based algorithm with the watershed algorithm in the last experiment. As a result, our proposed algorithm split more dividing cells correctly and produced fewer wrong splits. Our algorithm filtered out more segmentation failures than a watershed approach.

As a result of four different experiments, we automated the mother-bud detection and classification of dividing sample cells. As mentioned in experiments three and four, the segmentation model failed in segmenting cells that were very close to each other. Thus, the segmentation model needs to be improved in the future. Furthermore, we need to establish a classification between different bud sizes to automatically evaluate how advanced each dividing pair is in the cell cycle. Finally, quantifying protein abundance based on the green fluorescence detected in the green channel images and cell-to-cell variability in protein turnover (red/green fluorescence) need to be identified in the computational pipeline to evaluate protein age in yeast cells.

References

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. *2017 International Conference on Engineering and Technology (ICET)*, 1–6.

Ali, M. A. S., Hollo, K., Laasfeld, T., Torp, J., Tahk, M.-J., Rinken, A., Palo, K., Parts, L., & Fishman, D. (2022). ArtSeg: Rapid Artifact Segmentation and Removal in Brightfield Cell Microscopy Images. In *bioRxiv* (p. 2022.01.24.477467). https://doi.org/10.1101/2022.01.24.477467

Ali, M. A. S., Misko, O., Salumaa, S.-O., Papkov, M., Palo, K., Fishman, D., & Parts, L. (2021). Evaluating Very Deep Convolutional Neural Networks for Nucleus Segmentation from Brightfield Cell Microscopy Images. *SLAS Discovery : Advancing Life Sciences R & D*, *26*(9), 1125–1137.

Baggett, J. J., Shaw, J. D., Sciambi, C. J., Watson, H. A., & Wendland, B. (2003). Fluorescent labeling of yeast. *Current Protocols in Cell Biology / Editorial Board, Juan S. Bonifacino ... [et Al.]*, *Chapter 4*, Unit 4.13.

Caicedo, J. C., Roth, J., Goodman, A., Becker, T., Karhohs, K. W., Broisin, M., Molnar, C., McQuin, C., Singh, S., Theis, F. J., & Carpenter, A. E. (2019). Evaluation of Deep Learning Strategies for Nucleus Segmentation in Fluorescence Images. *Cytometry. Part A: The Journal of the International Society for Analytical Cytology*, *95*(9), 952–965.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(4), 834–848.

Chen, S.-C., Zhao, T., Gordon, G. J., & Murphy, R. F. (2007). Automated image analysis of protein localization in budding yeast. *Bioinformatics* , *23*(13), i66–i71.

Collins Lab Blog. (2010, February 18). Retrieved May 8, 2022, from: https://blogs.cornell.edu/collinslab/page/97/

Du, X., Cai, Y., Wang, S., & Zhang, L. (2016). Overview of deep learning. *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, 159–164.

Fishman, D., Salumaa, S., Majoral, D., Laasfeld, T., Peel, S., Wildenhain, J., Schreiner, A., Palo, K.,

& Parts, L. (2021). Practical segmentation of nuclei in brightfield cell images with neural networks trained on fluorescently labelled samples. In *Journal of Microscopy* (Vol. 284, Issue 1, pp. 12–24). https://doi.org/10.1111/jmi.13038

Haider, S. A., Cameron, A., Siva, P., Lui, D., Shafiee, M. J., Boroomand, A., Haider, N., & Wong, A. (2016). Fluorescence microscopy image noise reduction using a stochastically-connected random field model. *Scientific Reports*, *6*, 20640.

Hartwell, L. H., & Unger, M. W. (1977). Unequal division in Saccharomyces cerevisiae and its implications for the control of cell division. *The Journal of Cell Biology*, *75*(2 Pt 1), 422–435.

Higuchi-Sanabria, R., Pernice, W. M. A., Vevea, J. D., Alessi Wolken, D. M., Boldogh, I. R., & Pon, L. A. (2014). Role of asymmetric cell division in lifespan control in Saccharomyces cerevisiae. *FEMS Yeast Research*, *14*(8), 1133–1146.

IBM Cloud Education. (2020, October 20). *What are Convolutional Neural Networks?* Retrieved May 8, 2022, from: https://www.ibm.com/cloud/learn/convolutional-neural-networks

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In F. Bach & D. Blei (Eds.), *Proceedings of the 32nd International Conference on Machine Learning* (Vol. 37, pp. 448–456). PMLR.

Jensen, E. C. (2012). Use of fluorescent probes: their effect on cell biology and limitations. *Anatomical Record* , *295*(12), 2031–2036.

Kingma, D. P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. In *arXiv [cs.LG]*. arXiv. http://arxiv.org/abs/1412.6980

Kornilov, A. S., & Safonov, I. V. (2018). An Overview of Watershed Algorithm Implementations in Open Source Libraries. *Journal of Imaging*, *4*(10), 123.

Koschwanez, J., Holl, M., Marquardt, B., Dragavon, J., Burgess, L., & Meldrum, D. (2004). Identification of budding yeast using a fiber-optic imaging bundle. *The Review of Scientific Instruments*, *75*(5), 1363–1365.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, *25*. https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.

html

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Lei, M., Li, J., Li, M., Zou, L., & Yu, H. (2021). An Improved UNet Model for Congestive Heart Failure Diagnosis Using Short-Term RR Intervals. In *Diagnostics* (Vol. 11, Issue 3, p. 534). https://doi.org/10.3390/diagnostics11030534

Lipton, Z. C., Elkan, C., & Narayanaswamy, B. (2014). Thresholding Classifiers to Maximize F1 Score. In *arXiv [stat.ML]*. arXiv. http://arxiv.org/abs/1402.1892

Liu, D., & Yu, J. (2009). Otsu Method and K-means. *2009 Ninth International Conference on Hybrid Intelligent Systems*, *1*, 344–349.

Matcha, A. C. N. (2021, May 19). *A 2021 guide to semantic segmentation*. AI & Machine Learning Blog. Retrieved May 8, 2022, from: https://nanonets.com/blog/semantic-image-segmentation-2020/

Moen, E., Bannon, D., Kudo, T., Graf, W., Covert, M., & Van Valen, D. (2019). Deep learning for cellular image analysis. *Nature Methods*, *16*(12), 1233–1246.

Padilla, R., Netto, S. L., & da Silva, E. A. B. (2020). A Survey on Performance Metrics for Object-Detection Algorithms. *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, 237–242.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 779–788.

Riffle, M., & Davis, T. N. (2010). The Yeast Resource Center Public Image Repository: A large database of fluorescence microscopy images. In *BMC Bioinformatics* (Vol. 11, Issue 1). https://doi.org/10.1186/1471-2105-11-263

Shahriyari, L., & Komarova, N. L. (2013). Symmetric vs. asymmetric stem cell divisions: an adaptation against cancer? *PloS One*, *8*(10), e76195.

Steinkraus, K. A., Kaeberlein, M., & Kennedy, B. K. (2008). Replicative aging in yeast: the means to the end. *Annual Review of Cell and Developmental Biology*, *24*, 29–54.

Van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., & Yu, T. (2014). *scikit-image: image processing in Python, PeerJ, 2, e453*.

Venkei, Z. G., & Yamashita, Y. M. (2018). Emerging mechanisms of asymmetric stem cell division. *The Journal of Cell Biology*, *217*(11), 3785–3795.

Wang, Y., Lo, W.-C., & Chou, C.-S. (2017). A modeling study of budding yeast colony formation and its relationship to budding pattern and aging. *PLoS Computational Biology*, *13*(11), e1005843.

Wolfewicz, A. (2022, April 21). *Deep learning vs. Machine learning – what's the difference?* https://levity.ai/blog/difference-machine-learning-deep-learning

Yu, B. Y., Elbuken, C., Ren, C. L., & Huissoon, J. P. (2011). Image processing and classification algorithm for yeast cell morphology in a microfluidic chip. *Journal of Biomedical Optics*, *16*(6), 066008.

Yuste, R. (2005). Fluorescence microscopy today. *Nature Methods*, *2*(12), 902–904.

Zakhartsev, M., & Reuss, M. (2018). Cell size and morphological properties of yeast Saccharomyces cerevisiae in relation to growth temperature. *FEMS Yeast Research*, *18*(6). https://doi.org/10.1093/femsyr/foy052

Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2018). UNet++: A Nested U-Net Architecture for Medical Image Segmentation. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support : 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, S...*, *11045*, 3–11.

License

Non-exclusive license to reproduce thesis and make thesis public

I, **Danver Hans Värv**,

1. herewith grant the University of Tartu a free permit (non-exclusive license) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

Mother-bud detection and classification in yeast cells,

supervised by Mohammed A. S. Ali and Karla Juárez Núñez.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons license CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright..

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive license does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Danver Hans Värv
10 / 05 / 2022