Technische Universität München

Prediction of organoid growth with artificial neural networks

Dissertation by

Fabian Englbrecht

Physik-Department

TUM School of Natural Sciences

Prediction of organoid growth with artificial neural networks

Fabian J. Englbrecht

Vollständiger Abdruck der von der TUM School of Natural Sciences der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitz: Prof. Dr. Martin Zacharias

Prüfer*innen der Dissertation:
1. Prof. Dr. Andreas Bausch
2. Prof. Dr. Matthias Rief

Die Dissertation wurde am 27.03.2023 bei der Technischen Universität München eingereicht und durch die TUM School of Natural Sciences am 22.06.2023 angenommen.

Abstract

The structure formation and development of organoids is a highly promising area of research with the potential to change the way we think about stem cell differentiation or tissue regeneration. Organoid growth is driven by complex underlying structure formation processes. The detailed analysis of these processes was hindered for many years, due to limitations of conventional data analysis algorithms. The development of more sophisticated machine learning methods such as deep learning allows to obtain new insights in the development of these complex organoid morphologies. This information can be used by a deep artificial neural network to make predictions of the future morphology of individual organoids. Machine learning algorithms for future frame prediction such as recurrent neural networks (RNNs) or generative adversarial networks (GANs) have proven to provide reliable results in predicting future images. The analysis of cell biology images however requires even more sophisticated tools due to limitations in the cultivation and microscopy image acquisition of organoids.

In this thesis, machine learning - especially deep learning - is used to predict the future morphology of organoids. The machine learning method is able to analyze the overall growth process of an organoid image dataset and use this information to predict the future morphology of individual organoids. The model was trained on biological and simulated image sequence data of different structural complexity. The results indicate that this technique is able to reliably predict and generate realistic images of future organoids. Short-term as well as long-term future predictions demonstrate an accurate representation of the structure formation in organoids.

This thesis demonstrates the potential of future frame prediction in organoid research and enables a variety of applications to analyze complex physiological mechanisms in organoids that can eventually help to improve the individualized treatment of diseases.

In summary, this thesis sets a framework for further analysis of structure formation processes in live-cell imaging applications and organoid research. It highlights the significance of using deep learning methods for analysis of complex growth processes, where conventional algorithms are limited and sufficient amount of image data is provided.

Publications

Parts of this thesis have appeared previously in the following publication:

Englbrecht, F., Ruider, I. E., & Bausch, A. R. (2021). Automatic image annotation for fluorescent cell nuclei segmentation. PloS one, 16(4), e0250093.

Acknowledgments

An dieser Stelle möchte ich mich bei allen Personen bedanken, die zur Entstehung dieser Arbeit beigetragen haben.

Besonders bedanken möchte ich mich zuerst bei meinem Doktorvater Prof. Andreas Bausch. Für die Möglichkeit an diesem spannenden Projekt zu arbeiten und die Freiheit viele eigene Ideen umsetzen zu können. Für die Betreuung, die Unterstützung und die Möglichkeiten in der Forschung, die wir am CPA haben. Vielen Dank dafür!

Großer Dank gilt auch Samuel Randriamanantsoa, für die Unterstützung bei meinen Experimenten mit Organoiden, aber auch für die zahlreichen fachlichen Diskussionen zum Projekt. Weiterer Dank gilt Anna Pestucha, Marion Raich und Iris Ruider für die Unterstützung am Mikroskop und in der Zellkultur, sowie Franz Hutterer und Alfredo Sciortino für die Hilfe in der IT.

Bedanken möchte ich mich auch bei Sandra Andrusca für die Unterstützung während ihrer Bachelorarbeit und als Werkstudentin, sowie Richard Piekara, Matthias Sagerer und Sándor Battaglini-Fischer.

Vielen Dank insbesondere meinen Kolleginnen und Kollegen im „organoid office“, Samuel, Franz, Marion, Sophie und Arsenii, für die angenehme Arbeitsatmosphäre. Spezieller Dank geht an Marion Raich und Samuel Randriamanantsoa für das Korrekturlesen dieser Arbeit.

Bedanken möchte ich mich auch bei Gabi Chmel, Karin Vogt, Monika Rusp und Thi-Hieu Ho, für die Arbeit in der Zellkultur.

Ebenfalls bedanke ich mich bei dem gesamten Lehrstuhl für Zellbiophysik (E27) und Lehrstuhl für Biophysik (E22) für das tolle Arbeitsumfeld. Insbesondere meinen Kolleginnen und Kollegen Caro, Chiao, Daphné und Timon, sowie unserer Sekretärin Katerina Girgensohn.

Abschließend möchte ich mich noch besonders bei meinen Eltern, meinen Geschwistern, sowie allen Freunden bedanken. Vielen Dank für eure fortwährende Unterstützung und den Rückhalt während meiner Promotion.

Contents

1. Introduction

Machine learning has become one of the most promising and cutting-edge technologies in recent years. With applications such as search engines, the lately published chatbot ChatGPT[51], or the prediction of protein structures with AlphaFold[38], machine learning is an integral part of today's modern life. In recent years, the use of machine learning - especially deep learning - has become more widespread, driving the development of a wide variety of research and real-world applications. In situations where a high amount of data can be generated or is already available, deep learning methods provide a reliable solution to find patterns within these data and derive new insights from this information. Complex scientific questions such as the folding of protein structures seemed to be impossible to solve for many years. Machine learning methods however have proven to provide predictions with high accuracy for those challenges[38].

This technical improvement opens up the possibility to tackle further fundamental questions in research. Especially in the area of recently established organoids which is one of the most promising research fields in cell biology. Organoids are miniaturized cellular structures that can be derived from embryonic or adult stem cells, but also cancer cells. By studying organoids, scientists can increase their understanding of organ and tissue development, the rise and progress of diseases, and which approaches and drugs may be effective in treating these diseases. Organoid research can lead to the development of personalized treatments based on the patient's own cells, an improved understanding of many diseases, as well as a fundamental understanding of tissue development.

Organoids are grown from cells which are cultured in a three-dimensional extracellular matrix with optimized growth conditions. After several days, these cells and cell clusters eventually form organoids. The morphological structure of these organoids is highly individual. Figure 1.1 illustrates samples of microscopy images of pancreatic ductal adenocarcinoma (PDAC) organoids. Even though the duration of cultivation after seeding the PDAC cells is identical, the organoids show highly individual morphologies. These individual structures are based on an underlying structure formation process. The structure formation however is dependent on multiple factors. These can be either from biological nature or environmental driven. To understand how organoids are growing and

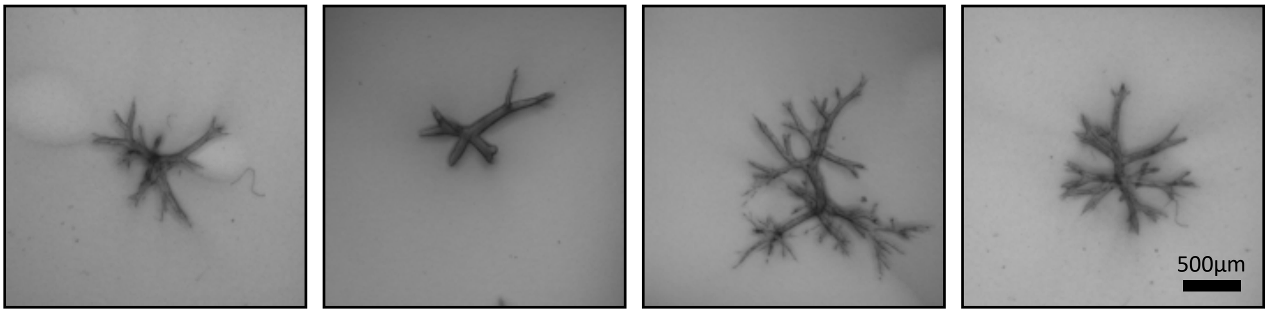


Figure 1.1: Microscopy images of pancreatic ductal adenocarcinoma organoids. All organoids are imaged five days after seeding and culturing the tumor cells, while developing highly individual structures. The underlying growth process of organoids is complex and depends on multiple biological and environmental factors.

which mechanisms lead to their complex structural morphology is an integral part for future organoid research.

In this thesis, machine learning is used to analyze the growth patterns of organoids that promotes the formation of its morphological complexity and to use this information to predict the future structure of an organoid at a given time point. This approach will open the way for many applications in future organoid research.
The prediction can for instance be used as a digital control for experiments with organoids. If a drug for example affects the growth behavior of an organoid, the prediction will show the future organoid without the intervention of the drug treatment.

In live-cell imaging applications where a lot of time series data is generated, the use of deep learning models for future frame prediction is highly promising. Due to the technical improvement of microscopes, the capabilities of image resolution and image throughput massively increased. Compared to typical real world applications, such as autonomous driving, microscopy images in cell biology are limited in some aspects. The limitations are both on the technical side and the biological side, such as restrictions in the frame rate, colorization of images, duration of imaging or the cultivation of organoids itself. For real world scenarios, a variety of future frame prediction algorithms was successfully tested already[52]. The objective of this thesis is to investigate the capabilities and limitations of future frame prediction with deep learning algorithms to predict the future morphology of individual organoids.

The first experimental part of this thesis focuses on a self-supervised learning algorithm for automatic segmentation of cell nuclei. As manual labelling is often a very time-consuming and tedious task, a self-supervised approach for automatic cell / cell-nuclei labelling can provide a solution for this challenge. To understand growth patterns of cells, it is important to register and track individual cells. This requires precise cell or cell nuclei segmentation. While the challenge of cell / cell nuclei segmentation in 2D is widely solved, it still remains some hurdles dealing with different input / training data. The training data is individual to the experiment and can for example differ in magnification due to the microscope objective, or the type of imaging such as bright-field or fluorescence imaging. As many labeled training datasets of cells / cell nuclei are publicly available, it still does not cover the entire variance represented by cell culture experiments. The experiment shows the possibilities, challenges and limitation of a self-supervised approach for automated labelling of cell nuclei. Parts of this experiment have been published previously[25].

In the second part of this thesis, the segmentation problem of complex cellular structures such as organoids is addressed. Organoids cultured in an extracellular matrix such as collagen form highly individual and complex structures. To analyze and parametrize organoids, a segmentation mask is highly beneficial. The challenge of the experiment was to automatically segment organoids in microscopy images to make them applicable for further analysis with deep neural networks or conventional analysis algorithms. A data processing pipeline for accurate organoid segmentation of 3D image projections was implemented.

The third part of this thesis focuses on the prediction of the morphology of organoids with deep learning algorithms. The goal of this experiment is to predict the morphological structure of an organoid from a specific time point to a later developmental period, such as 24 hours or 48 hours in the future. The results of the first two mentioned experiments enable the implementation of a future frame prediction network to do such tasks. The prediction of organoids not only helps to better understand the complex underlying structure formation processes in organoids, but also provides a basis for future applications in organoid research. The prediction can for example be used as a digital control when a drug is tested on organoids. The effect of the drug on the morphology of an organoid is hard to quantify. With a prediction of the future state of the organoid, it is possible to analyze the effect of the drug on the morphology of the organoid.

2. Theoretical background

2.1. Machine learning

Machine learning methods are widely used in a variety of data analysis methods. Since the *ImageNet* Competition in 2012, machine learning methods - especially deep neural networks - have shown high performance in the challenge of image classification[41]. Deep learning methods for image analysis gained massive improvements since then.

2.1.1. Machine learning methods

Machine learning algorithms are traditionally categorized into three different types of machine learning methods. Supervised learning, unsupervised learning and reinforcement learning. Each of these methods is used in a different context.

Supervised learning: A characteristic of supervised learning is the availability of annotated training data. The goal of the machine learning model is to identify and learn a relationship between unlabeled input data and labeled output data. Based on training of the annotated dataset, unseen test data can be classified[21].

Unsupervised learning: Unsupervised machine learning models are used in cases where the model is not provided with annotated data. In this case the machine learning model is trained on a dataset in order to find patterns and structure within the data[29]. Examples of unsupervised learning include methods such as k-means clustering, principal component analysis (PCA) or the use of auto-encoder neural networks[80].

Reinforcement learning: Reinforcement learning is a type of machine learning model which focuses on learnings in sequential decision making problems where feedback is limited[81]. The model uses a feedback-loop to update its understanding of the environment and to improve future decisions. Reinforcement learning is used in applications such as robotics or natural language processing.

Figure 2.1 illustrates the different types of machine learning algorithms. Besides this three-fold categorization, there are other types of machine learning methods used for specific applications. Commonly used methods are semi-supervised and self-supervised learning.

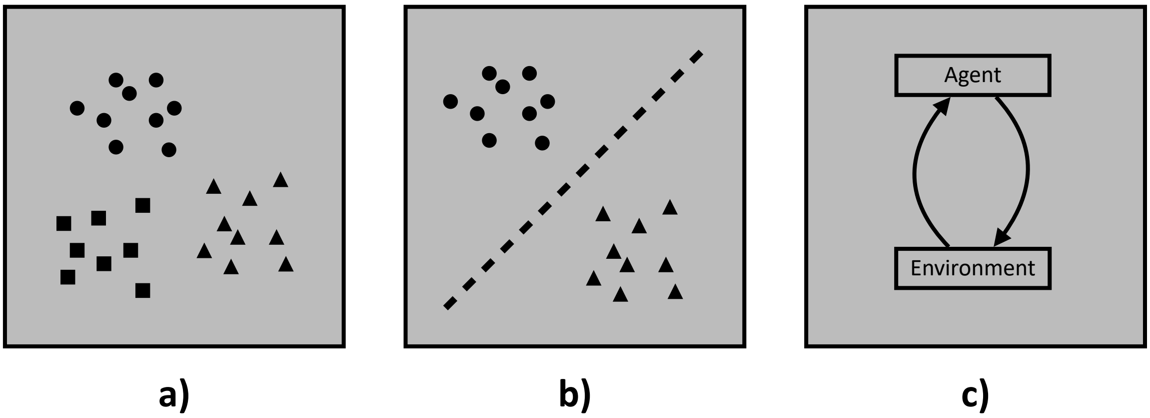


Fig. 2.1: Types of machine learning algorithms[2,3]. Traditional classification of machine learning algorithms with a) supervised learning: the model is trained on a labeled dataset. For each output, a correct input is provided. b) unsupervised learning: the model is trained on unlabeled data. It finds patterns within the data by itself. c) reinforcement learning: the model learns to make decisions by interactions with its environment and updates its understanding by an integrated feedback loop.

Semi-supervised learning: Semi-supervised learning is a type of machine learning that combines supervised and unsupervised learning. The model is trained on a dataset that contains a small set of labeled data in combination with a larger set of unlabeled data.

Self-supervised learning: In Self-supervised learning, a model is trained to learn from input data with no given data labels. In comparison to unsupervised learning, the model is not able to cluster data in the dataset. Instead, the method is often used in applications where parts of data is missing.

2.1.2. Artificial neural networks

Artificial neural networks are algorithms which are inspired by biological neural networks of the animal's brains. Machine learning models are a subset of artificial intelligence, with the objective to learn from data without being explicitly programmed. Machine learning includes various types of models for individual applications, such as decision trees[30], support-vector machines[47], or Bayesian networks[7]. A central type of machine learning model are artificial neural networks (ANNs)[3]. Figure 2.2 illustrates the architecture of an artificial neural network.

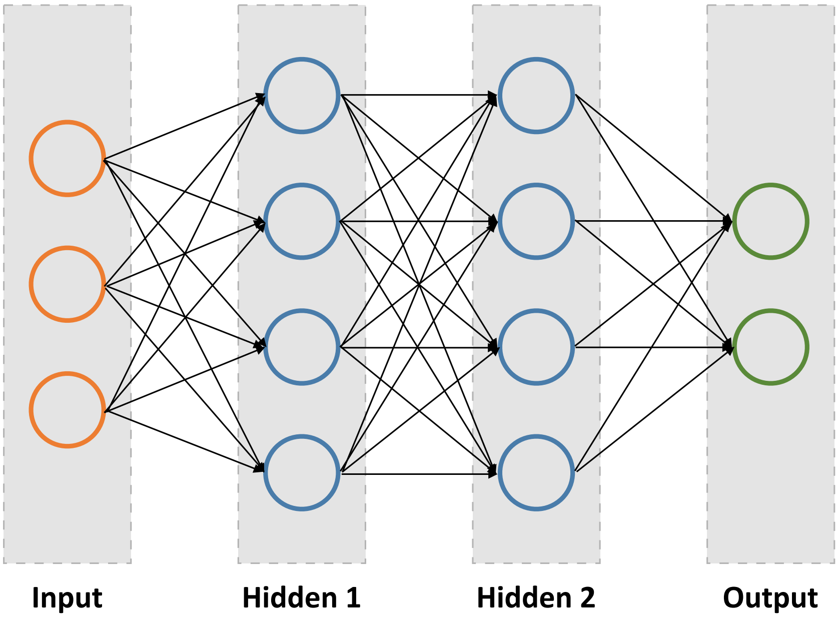


Fig. 2.2: Schematic of an artificial neural network[2]. The feedforward neural network consists of an input layer, an output layer and two hidden layers (deep neural network). Each layer contains several interconnected neurons. Each neuron in the ANN receives input from other neurons of previous layers. The neurons perform a computation on that input, and produce an output that is passed on to neurons in the next layer.

ANNs consists of several layers. An input layer, an output layer and one or several (deep neural network) hidden layers. These layers contain multiple interconnected artificial neurons. The connections between neurons are represented by weights, which have an associated value with it. The neurons are able to process and transmit information between neurons of consecutive layers.

A mathematical operation on that input is performed which includes multiplication of the neuron information with the weight value, summation of these output information and the the addition of a bias value for each neuron. The values of the weights can be updated by training the network. A demonstration of an artificial neural network is shown in figure 2.3.

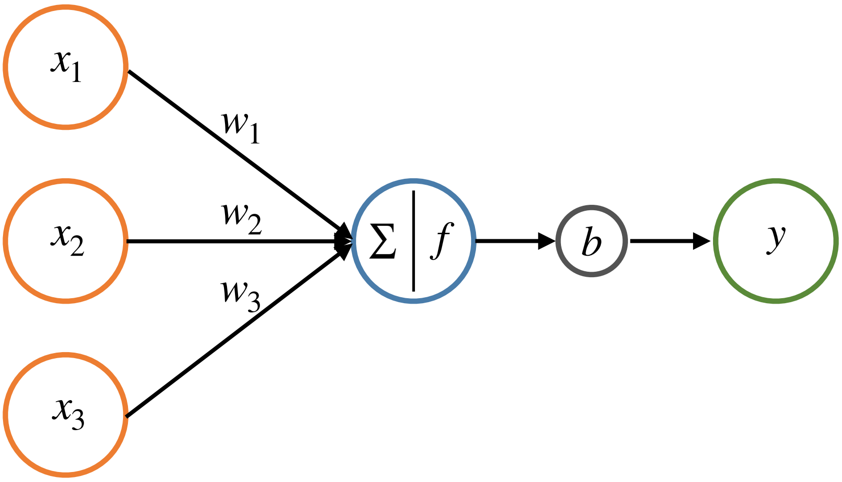


Fig. 2.3: Representation of an artificial neuron in a neural network[83]. A neuron receives the inputs x_1, x_2 and x_3 which are multiplied by weights w_1, w_2 and w_3. Multiplications of inputs and weights are summed up. An activation function is applied to the weighted sum. A bias unit b is added, which leads to an output y.

A common type of ANN is a feedforward neural network (FNN), where information is passed through the network in one direction, from the input to the output layer. A fully connected FNN is also referred to as multilayer perceptron (MLP)[33]. FNNs utilize a technique called backpropagation, which is widely used for training artificial neural networks[44,61]. By adjusting the weights and biases of each neuron between the output layer and the input layer, the error between input data and predicted output can be corrected. It enables the ANN to automatically improve its performance with each training cycle.

Another type of neural networks are recurrent neural networks[46] (RNNs) where the information flow happens in cycles. RNNs are often used to analyze sequential data like text, speech or time-series.

Deep neural networks (DNNs) are specific machine learning models which are composed of multiple layers of artificial neurons. Deep neural networks have been particularly successful in tasks such as image recognition or natural language processing[8,15,59].

2.1.3. Convolutional neural networks

Convolutional neural networks (CNNs) are a subclass of ANNs. CNNs are mostly applied to analyze images. They utilize a process called convolution, which involves sliding a small matrix of weights (called a kernel or filter) over the input data and computing a dot product at each position[49] (fig. 2.4). The information is then condensed in a pooling process, which reduces the dimensionality of the convoluted image. An activation function such as a rectified linear unit (ReLU) is applied on the resulting sum. The activation function acts as a filter to produce the final output value for that position in the

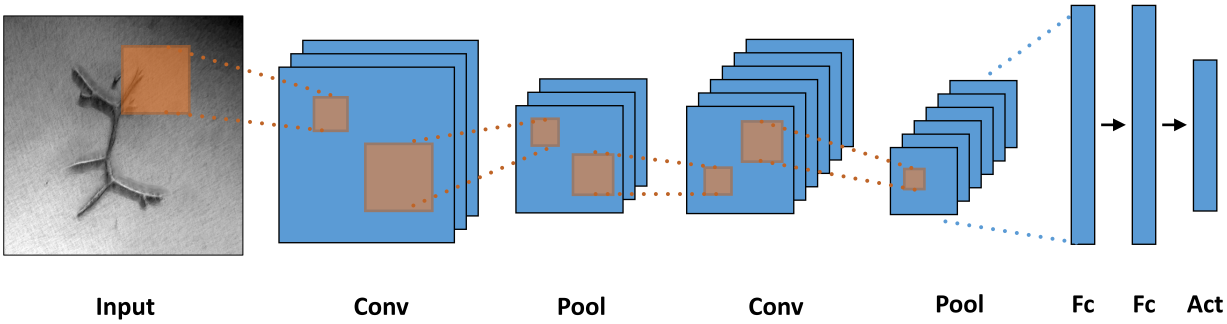


Fig. 2.4: Network architecture of a convolutional neural network[2]. The network uses several convolution and pooling layers to extract features from input images. In the fully connected layers (Fc), every input neuron is connected to every output neuron. The output is forwarded to an activation function of the CNN.

feature map[57]. For each position in the image, the same process is repeated, defined by the size of the filter. Due to this process, features or patterns from the input image can be extracted. In early training cycles, these features are often edges or textures, which develop more complex structures after multiple training cycles. These features can be used for tasks such as classification or segmentation. Figure 2.5 shows a schematic of a convolution process. A 3x3 kernel is used to extract essential information from an input image to an output image. To avoid a decrease of the overall pixel size of the image, padding of one and a stride of one is used.

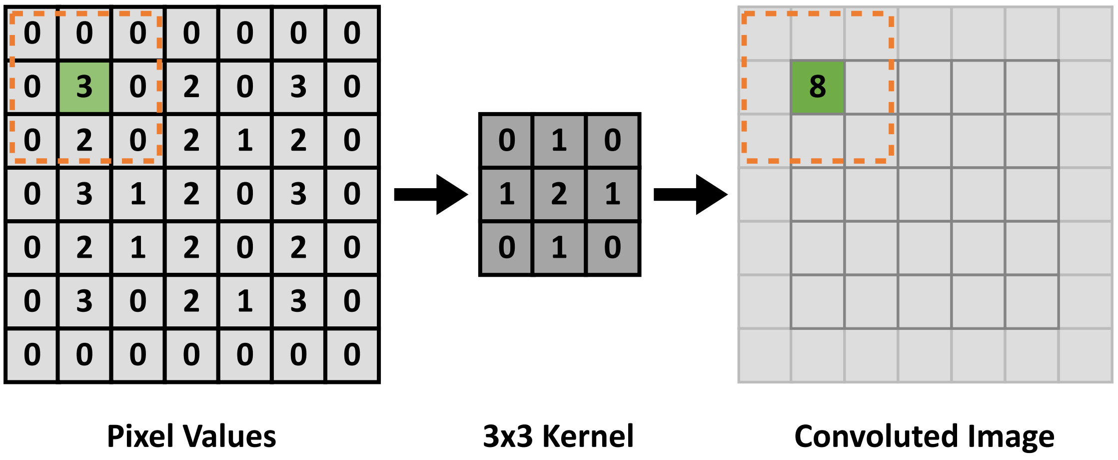
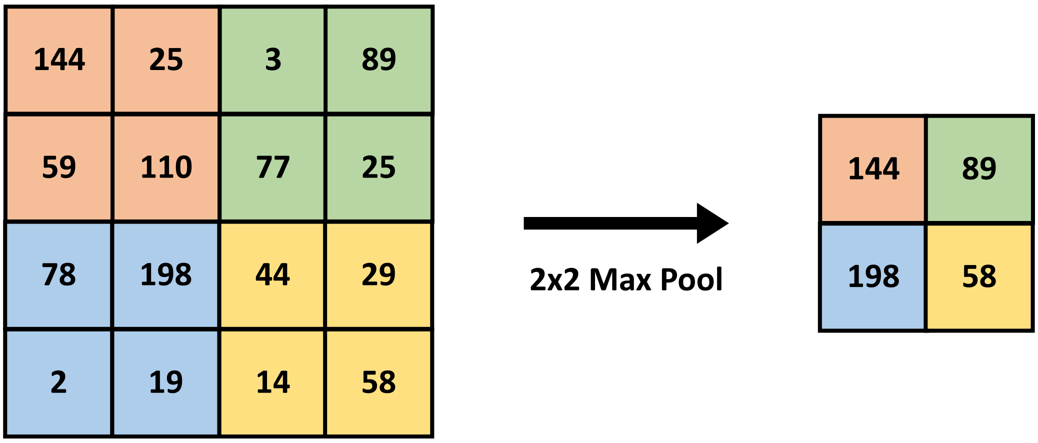


Fig. 2.5: Schematic of the convolution process in a convolutional neural network[24]. A filter / kernel is used to scan over an input image with a padding of 1. Each pixel value is multiplied by the values of the kernel and summed for all positions. The convolution results in an extraction of image features.

When using CNNs, padding is a frequently used technique in order to maintain spatial resolution when applying a convolutional filter on an image. Padding inserts an additional row pixels around the image to encounter the reduction of spatial resolution due to a convolution. The output feature map has the same size as the input image, when padding is used. A shrinking of the image and losing information on the border of the image is therefore prevented.

Stride refers to the step size which is used when applying a convolutional filter to an image. When the stride is increase, the output feature map is decreased and vice versa. Stride is often used to reduce the computational cost of the CNN.

Another way to reduce the size of a the output feature map is the use of pooling layers in CNNs. After a convolution, pooling is applied to extract relevant information or patterns from the images in the training dataset. Common types of pooling are max pooling, min pooling or average pooling. Max pooling selects the maximum value from a set of input values, while minimum pooling selects the smallest pixel value. Pooling layers not only reduce the spatial resolution of the input image but also make the network more robust to small transitions of objects in the input data and are able to prevent overfitting in the training process. Figure 2.6 shows the schematic of a max pooling layer with a stride of two in a convolutional neural network. Pooling leads to a decrease in the spatial dimension of the input image, while reducing the computational cost of processing the output image. After a repeating number of convolution and pooling layers, the extracted



144	25	3	89
59	110	77	25
78	198	44	29
2	19	14	58

144	89
198	58

Fig. 2.6: Schematic of max pooling in a convolutional neural network[30]. Maximum values from the input feature map are collected within a certain field of view. Values are passed to the output feature map. The most important information of the input feature map is contained in the output feature map. Pooling reduces the spacial dimension of the output layer in a CNN.

features are fed into a fully connected layer. A loss function is applied to the output of the fully connected layer.

A common challenge in machine learning is overfitting. Overfitting occurs when the machine learning model is representing the data in a too high level of detail. The resulting model is not able to make predictions on unseen data, because the level of detail learned from the training data is exaggerated. An overfitted model produces low accuracy results for data points that are not covered in the training dataset. An underfitted model on the other hand is not complex enough to recognize the patterns in the dataset. It usually has a high bias towards one output value because the network considers the variations of the input data as noise and generates similar outputs regardless of the given input[72].
Figure 2.7 shows an example of underfitting and overfitting[85] in data. In the case of underfitting on the on hand, the representation of the fit curve does not represent the appropriate level of detail of the dataset. Overfitting on the other hand leads to a fit curve where details in the training set are represented on a very high level. This leads to the effect, that it does not generalize well to a new unseen dataset. It is therefore important that the level of detail which is represented by the model is balanced.

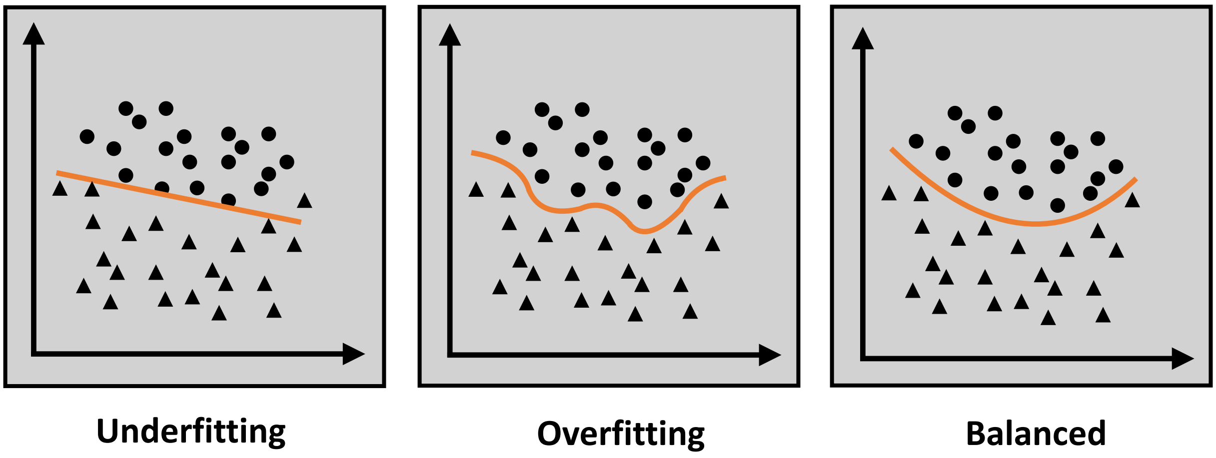


Fig. 2.7: Fitting of data when using machine learning models[2]. The separation line represents how well the trained model fits the dataset. An underfitted model does not capture essential details of the dataset. An overfitted model represents the dataset to a very high level of detail. Overall trends are not captured anymore. In the balanced model, the fit shows an ideal representation of the dataset.

There are several possible options to prevent overfitting. These include using a simpler model with fewer parameters, using methods such as dropout, regularization and early stopping, or using a large dataset with high diversity to make the training process more efficient.
A validation dataset in addition to a training and a test set is important to monitor the performance of the machine learning model. By using a validation set, hyperparameters such as the kernel size, padding, stride, or the number of filters of the model can be adjusted accordingly to prevent overfitting. To evaluate the ability of the deep learning algorithm to model the dataset, a loss function is used. Underfitting and overfitting can be identified by the associated loss curve after training the model.

Figure 2.8 illustrates different types of learning curves of a neural network. The learning rate is a hyperparameter of a neural network that affects the training loss of a dataset. The learning rate defines how quickly the neural network updates new patters learned from the dataset. Small learning rates require more training cycles, with an increase in

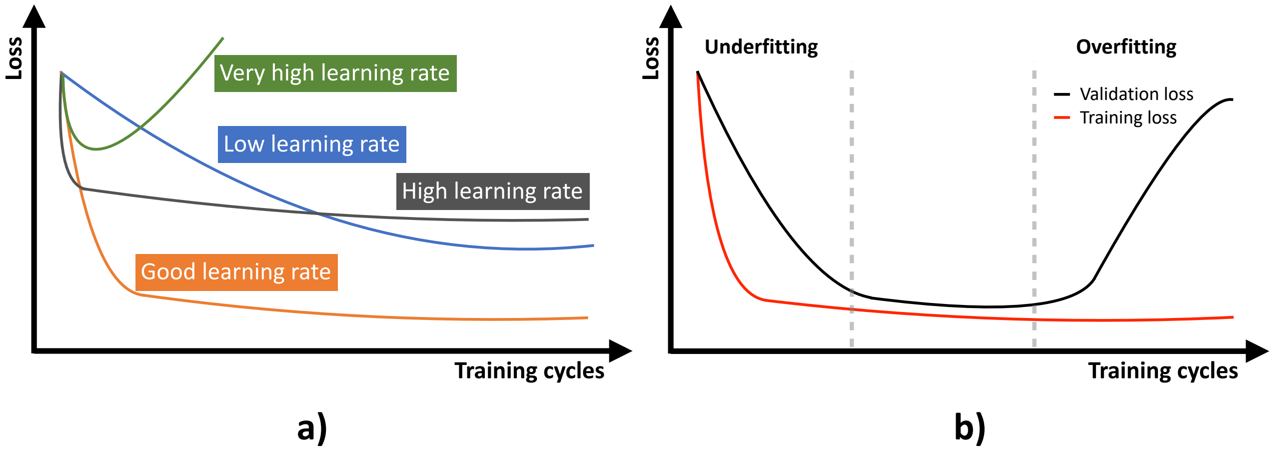


Fig. 2.8: Comparison of loss curves in machine learning[4,72]. a) The learning rate at which the model updates information, affects the loss of model. A low training loss can be obtained by an adequate selection of the learning rate. b) Validation and training loss curve for a dataset trained with a neural network. The model represents the dataset where the validation is minimal. Overfitting occurs when the validation loss increases. The model starts to represent noise in the dataset. At early training cycles the model is underfitting the training data. The model is unable to relate input data to output data.

computational cost. Larger learning rates require less training epochs, often with a decrease in the accuracy of the model. In order to validate if the model of a neural network represents a dataset, the information of the training loss is not sufficient. Dependent on the number of training cycles, the representation of the dataset changes. A low number of training cycles often results in a model that is underfitting the training data. The model is not able to obtain the relation between input and output data. The model is overfitting the training data when the model performs well on training data but poorly on the validation dataset. The level of detail obtained from the dataset is too high, leading to an increase in the validation loss error compared to the training loss error. The model represents the dataset well, when the validation loss is minimized. Early stopping is a method that is often implemented in machine learning models to automatically stop the training process when the validation loss is starting to increase.

2.1.4. Future frame prediction

Future frame prediction is a task in which an algorithm - often a machine learning model - is trained to predict the next frame(s) in the sequence of image data or a video[42,52,87]. Several deep learning algorithms have been proposed for this task. They include convolutional neural networks (CNNs), recurrent neural networks (RNNs), generative models (GANs) or flow-based models, which are a combination of CNNs and RNNs. The choice of algorithms is dependent on the specific task of prediction and the availability of computational resources.

Recurrent neural network: A recurrent neural network (RNN) is a certain type of artificial neural network used for processing and understanding sequential data, such as time series. Traditional deep neural networks such as feed forward neural networks process data where the inputs and outputs of the network are independent of each other. RNNs on the other hand are able to process prior information within the sequence (fig. 2.9). This allows some kind of “memory” within the network which enables to process “already known” information in order to influence the input and output of the network.

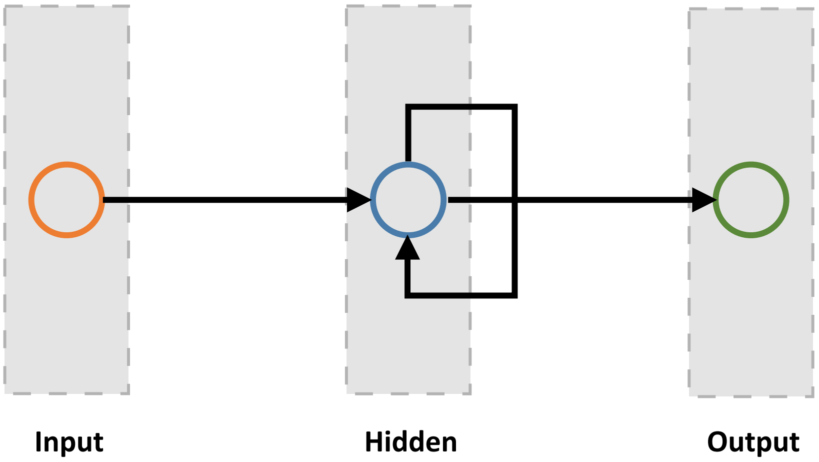


Fig. 2.9: Schematic of a recurrent neural network[69]. Information flow from neurons in the input layer via neurons in the hidden layer to neurons in the output layer. Neurons in the hidden layer are able to interact with the environment. Due to this feedback loop, future decision can be affected.

Generative adversarial neural network: A generative adversarial network (GAN) is a certain type of artificial neural network used to generate images[20]. These images are new data that is similar to the training dataset. A GAN therefore is able to generate variations of the input dataset.

Figure 2.10 illustrates the working principle of a GAN. A GAN typically consists of two networks, a generator network and a discriminator network. Both networks are trained in a process called adversarial training. The aim of the generator network is to generate realistic images. The discriminator network receives both generated and real images and its task is to determine which of both images generated by the generator network, and which is the real image. Both networks are trained simultaneously and are competing against each other[20].

The decision of the discriminator influences the loss of the discriminator and generator. The weights of both models are independently improved by back-propagation. This process continues until the generator generates data that is indistinguishable for the discriminator from real data.

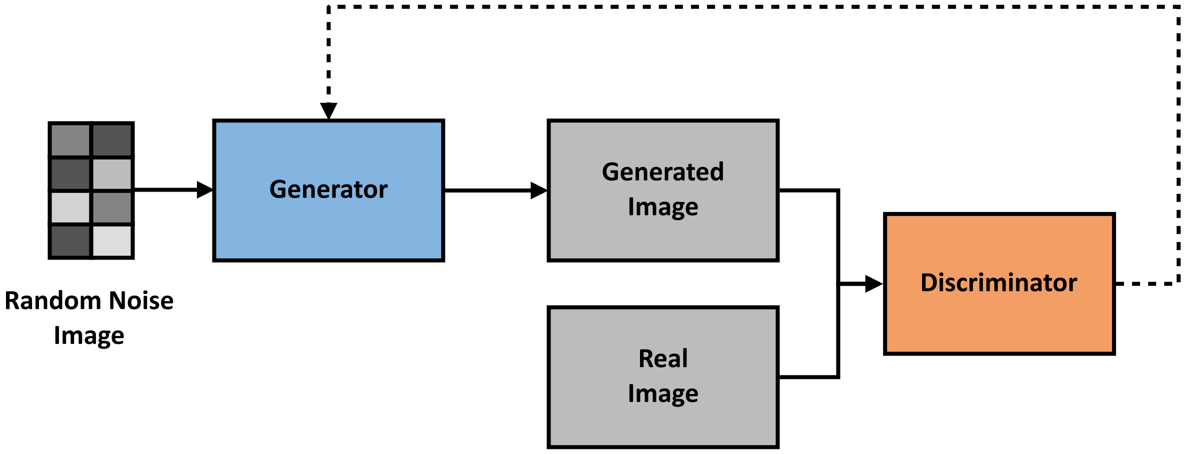


Fig. 2.10: Schematic of a generative adversarial network[27]. The model consists of two networks. A generator network and a discriminator network. The generator network generates an image from random noise. The discriminator tries to distinguish the generated image from a real input image. The discriminator sends a feedback to the generator based on its decision. Due to the feedback loop, the generator is able to generate more realistic images. At some point the discriminator is not able to distinguish generated images from real images.

2.2. Cell biology

2.2.1. Structure formation processes

Structure development in cell biology refers to the process through which cells and cell tissues self-assemble into distinct morphologies. The cultivation, manipulation and observation of cells and cell tissues is performed on 2D substrates or in a three-dimensional extracellular matrix (ECM)[13]. This reduced complexity of the system in vivo allows an in-depth study of the physical, biological and biochemical aspects that drive the structure formation process in organoids and cells.

Single cell migration in 2D refers to the process by which cells propagate on a flat surface of a tissue or a culture dish. The migration is driven by a polarized expression of the actin network in the direction of migration[73]. Thin filopodia-like protrusions or lamellipodia form on the leading edge of the cells, depending on the cell type and the properties of the substrate. Complex signaling cascades involving various proteins such as CDC42, PIP3, or RAC14 enhance this polarization. The cells form focal adhesions at the cell-substrate boundary[82] to increase adhesion to the substrate and to transmit regulatory signals and mechanical forces between the cells. When cells migrate, the adhesion at the back weakens, causing the cells to separate from the substrate and protrude forward. Cells exert traction forces during this phase in the opposing direction of cell movement on the substrate. This force application only affects the front of the migrating cell, while the force that was previously applied to the back of the cell is released[32]. During the force developed by actomyosin contractions and by cell-substrate adhesion via focal adhesions, cells detect the stiffness of the substrate. Based on this durotaxis, cells move in the direction with higher stiffness[13].

Cell-cell adhesion, which describes the coupling of neighboring cells, leads to collective cell migration in a confluent cell layer. Specific studies of collective cell migration and wound healing assays have previously been conducted[60]. These studies demonstrate a simplified wound healing process by monitoring the closure of a gap between two opposing epithelial cell layers (fig. 2.11). During the collective growth, all cells contribute to the overall migration of the cell layer. The individual cell traction forces of all cells add up as a result of the internal cellular coupling along their neighbors. This leads to an internal stress within the cell layer perpendicular to the substrate. The resulting internal stress depends on the height and width of the cell sheet. The directed cell migration and the coupling of the cells induce an internal stress within the epithelial sheet. With increasing distance, this stress within the cell layer continuously increases. The individual traction forces stay constant within the area. Even when the cell layer is confluent, after closing the gap between both layers, the cells are further migrating. The velocity of the cell movement decreases with an increase of the cell density and cell-substrate as well as cell-cell adhesion. The cells then undergo a jamming transition which leads to a stopping of the cell migration[28,54,13].

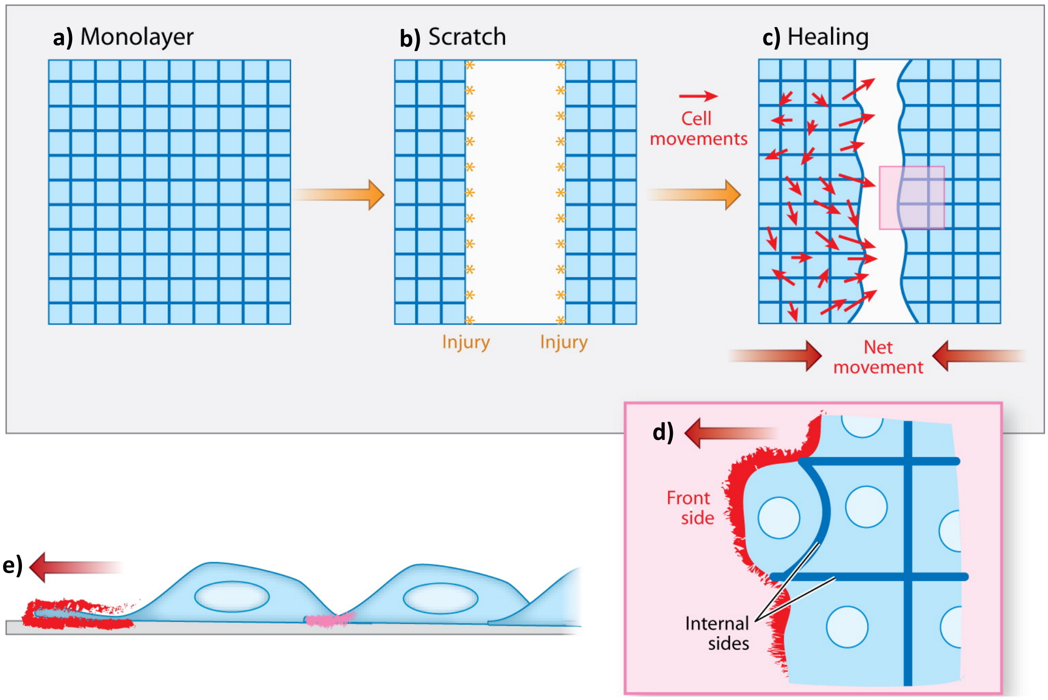


Fig. 2.11: Cell migration on a 2D substrate[60]. a) An unperturbed and confluent monolayer of (epithelial) cells. b) A mechanical scratch creates a gap and free space between cells. The edge cells may be injured (orange asterisks). c) The cell sheets move inwards to close the gap. Red arrows indicate displacement vectors for individual cells at one time point. d) Higher magnification of the area in c). A front cell automatically has polarity, a front (red) surface adjoining the free space and cell layer. e) Side view of cells moving inwards to fill the gap, with free surface extension of the front cell in red.

Compared to 2D cell culture, in 3D assays, cells are cultivated within a natural spatial confinement, such as collagen or matrigel, which is defined by the extracellular matrix (ECM). The migration of cells occurs through a dense network of collagen fibers, with various adhesion sites and different mechanical and biochemical properties. The movement of cells in a 3D environment occurs through mesenchymal, amoeboid and lobopodial migration[84] (fig. 2.12). Mesenchymal migration in particular demonstrates a strong cell-ECM adhesion via focal adhesions. During the invasion of the ECM, the cells are remodeling the ECM. Lamellipodia are generated due to actin polymerization at the leading edge. During this process, the fibers in the ECM get aligned in the direction of migration. During amoeboid cell migration, the adhesion between cells and the ECM is reduced. This leads to a round cellular morphology. The migration of the cells occurs through a squeezing of the cells through the pores of the ECM. The trajectory of the cells is therefore defined by the ECM. Lobopodial migration combines two modes of migration. The cells show a strong adhesion to the ECM, while the cell body is deformed due to internal asymmetric pressure originated from the nucleus position in combination with myosin-II contractility in the cell[55]. Three-dimensional cell migration is furthermore spatially defined by the structure, stiffness or composition of the surrounding ECM[84,13].

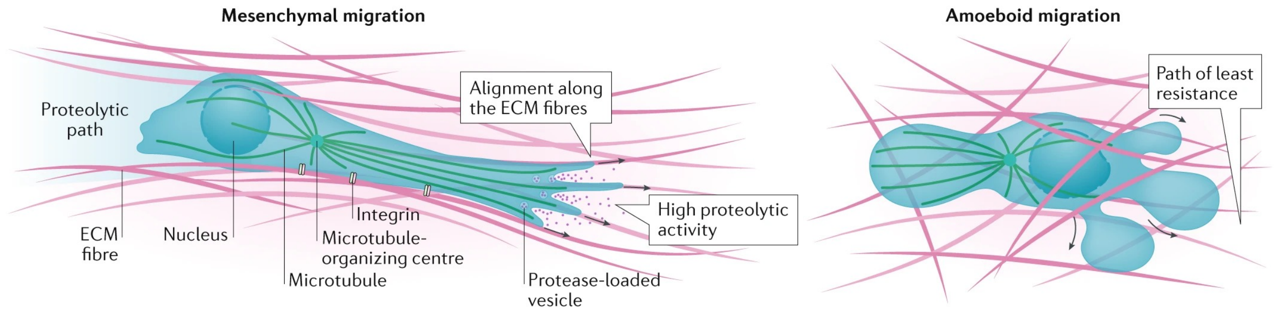


Fig. 2.12: Mesenchymal and amoeboid cell migration[84]. The mesenchymal cell aligns along the ECM fibers while at the same time creating a tunnel of proteolytically digested ECM. The amoeboid cell does not align along the ECM fibers but instead propagates by protruding through pores in the ECM.

2.2.2. Organoids

Within the last decade, 3D cell culture assays have significantly improved to replicate organ growth in an artificial environment outside of the normal biological context (in vitro)[19]. By the cultivation of patient- or animal-derived cells, embryonic stem cells or pluripotent stem cells in 3D matrices enabled the creation of organ-like cell clusters[13]. Through this process, cells self-organize into multicellular organoids which represent a simplified but realistic morphology of the real organ in vivo. Organoid models that replicate the brain, lung, pancreas, mammary gland, and many more have been developed during the past years[18]. Since organoids originate from patient- or animal-derived cells or tissue fragments, they have the potential to offer a patient-specific medication[6]. Organoids for instance can be utilized as model systems to create a drug medication that is patient-specific or to replace diseased tissue through regenerative medicine[64,13]. As an essential basis, the organoid assays and the respective developmental processes need to be fully analyzed and understood. In the following, the pancreatic ductal adenocarcinoma (PDAC) cells and organoids will be in focus.

The complexity of the structure formation process can be partly described by the appearance of different growth phases in organoids[58]. In fact, the branched organoid morphogenesis shows up four distinct developmental phases (fig. 2.13). An onset phase, where the cell proliferation from an initial single cell is exponential and evenly distributed in all dimensions in space. In the consecutive extension phase, the motion of the cells is mostly directed from the core towards the tips of the branches of the organoid. The third phase, also referred as “thickening phase“, is characterized by a contractile motion which results in a thickening of the branch tips into buds. Towards the end of this thickening phase, a fourth growth phase is observed, which is characterized by a lumen-formation in the branches of the organoid[58].

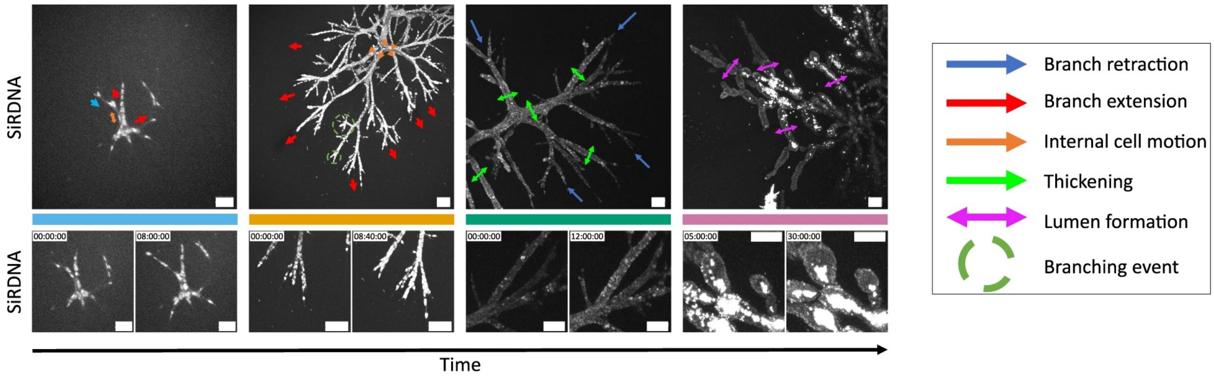


Fig. 2.13: Developmental phases of PDAC organoids[58]. Development phases are denoted by color bars which follow the color code - blue: onset phase, orange: extension phase, green: thickening phase, pink: lumen formation phase. All organoids are grown in collagen. Cellular motion patterns are observed with live confocal imaging for each development phase (*n* = 66 organoids). Cell nuclei are stained with SiRDNA (white). Scale bars: 100 µm. From left to right: Day 4 sum projection, and Day 7, Day 10, Day 13 maximum projections.

The occurrence of these growth patterns is observed in most PDAC organoids but the duration of these phases it is not synchronous between individual organoids. This leads to the appearance of organoid samples with highly individual structures, even if the seeding process and culturing duration of two samples are identical. This complexity in addition with multiple more subtle biological and environmental factors in the structure formation of organoids is a cause of the highly individual morphology.

3. Methodology

3.1. Cell culture and live-cell imaging

To obtain microscopy image data, cells were seeded and cultured for several days until a cell layer or cell tissue is formed. Compared to cells growing in a flat two-dimensional layer, organoids are usually grown in a three-dimensional matrix like collagen or matrigel. The culturing process is more complex and prone to complications in comparison to 2D cell culture methods. This leads to a bottleneck in the throughput of samples in organoid cell culture. At the same time, machine learning and especially deep learning models, require a high amount of data to understand underlying features within the dataset. To encounter this problem, a pipeline was introduced which covers the process from initial cell seeding and culturing of the cells to consecutive live-cell imaging methods. This increases the total amount of cell and organoid samples for generating biological image data.

3.1.1. Cell culture

The cells used for the experiments in this thesis are Madin-Darby Canine Kidney (MDCK) epithelial cells and pancreatic ductal adenocarcinoma (PDAC) cells from the 9591 mouse cell line provided by the research group of Prof. Dr. Med. Dieter Saur at Klinikum rechts der Isar. For cell cultivation, Dulbecco's Modified Eagle's Medium Mixture F-12 (DMEM F-12) by *Sigma* was used for MDCK cells. Dulbecco's Modified Eagle's Medium - high glucose was used for PDAC cells and organoids. The medium was supplemented with 10% fetal bovine serum (FBS) and 1% Penicillin / Streptomycin (Pen-Strep). Cells were passaged every 2-3 days to ensure sufficient nutrient supply for the cells. Depending on the experiment, samples were cultured in 2-well plates by *Ibidi* or 24-well plates by *Sarstedt AG & Co. KG*. For 3D experiments, PDAC cells were cultured in Collagen Type I of rat tail by *Corning* with a concentration of 1.3 mg/ml

The cells for the experiments demonstrated in this thesis were cultivated inside an *HERAcell 150i* incubator by *Thermo Scientific*. The temperature was kept constant at 37°C, the CO_2 level was set to 5%.

3.1.2. Live-cell imaging

Images for the experiments were captured with the Thunder Imager Live Cell & 3D Assay inverted DMi8 microscope by Leica. Datasets were acquired with a N PLAN 5x (0.12 NA) dry objective and a HC PL FLUOTAR L 20x (0.40 NA) dry objective, depending on the experiment. The microscope was used with the LAS X Life Science 3.7.6. software. An additional incubation chamber (Tokai Hit STX Stage Top Incubator) was used in combination with the microscope to maintain culture conditions of the cell culture incubator. Temperature was therefore maintained at 37 °C, with a humidity of 80% and a

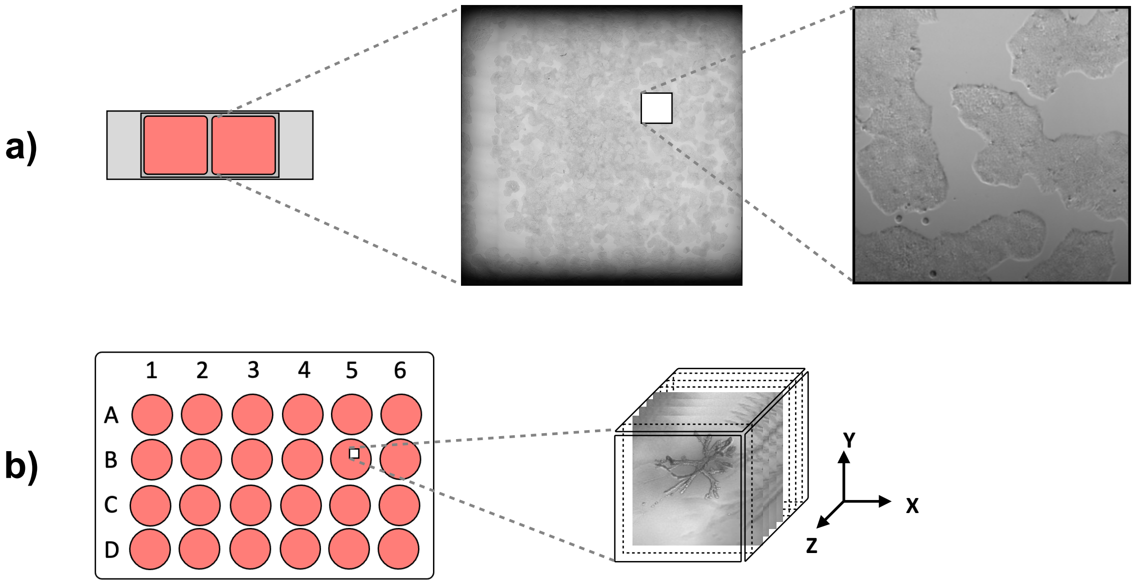


Fig. 3.1: 2D and 3D image acquisition. a) Example of a 2D image acquisition pipeline. For two-dimensional cell growth, PDAC cells are seeded in a 2 well plate. The wells are imaged by live-cell microscopy. The stitched microscopy image contains multiple subsample of image sequences. b) Example of a 3D image acquisition pipeline. PDAC organoids are grown in a 24-well plate. The positions of each organoid is registered in the microscopy software. Each measurement contains an organoid within a cubic imaging volume. Additionally time-series data of each organoid is acquired.

CO_2 level of 5%. Live-cell imaging was applied to all types of cells, varying in parameters specific to the individual experiment.

Figure 3.1 shows an imaging pipeline of biological samples. For the experiments in this thesis, 2-well plates were used for 2D image acquisition of PDAC sells. 24-well plates were used for 3D image acquisition of PDAC organoids. The imaging pipeline enables a high throughput of biological microscopy data.

3.2. Datasets

For the experiments in this thesis, several image datasets were generated. The datasets can be divided into 4 categories. Figure 3.2 shows a sample overview of the different classes of datasets. One dataset containing a cell layer of MDCK cells (2D) with fluorescent labeled nuclei. The second dataset includes time-series data of a PDAC cell layer (2D). The third dataset contains time-series data in combination with volumetric images of PDAC organoids. In the fourth class, two datasets of time-series image data of

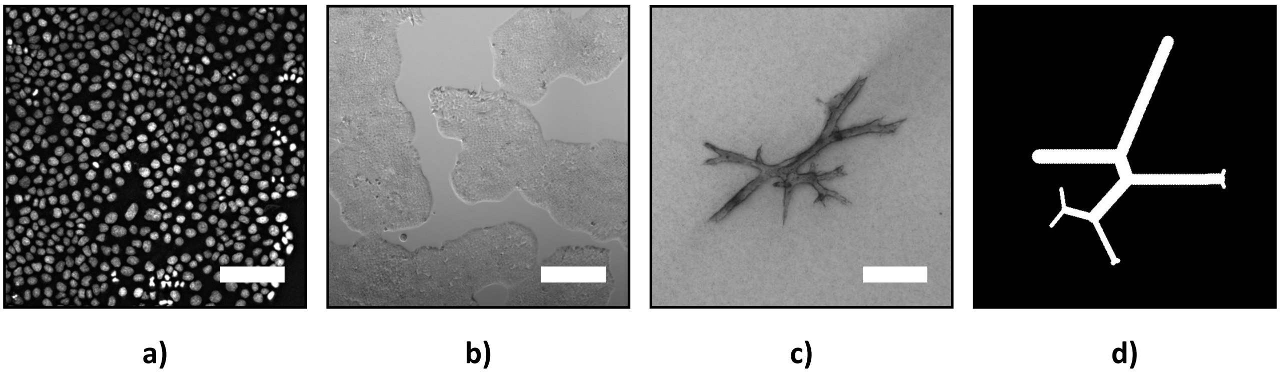


Fig. 3.2: Sample images of time-series datasets. a) MDCK cell nuclei dataset. The dataset contains microscopy data of fluorescent labeled MDCK cell nuclei. b) PDAC cell dataset (2D). The dataset contains brightfield microscopy images of a PDAC cell layer. c) PDAC organoid dataset (3D). The dataset contains brightfield images of multiple individual PDAC organoids. d) Organoid simulation dataset. The dataset contains simulation data of the structure formation of organoids. Corresponding segmentation masks are provided for all images in the microscopy image datasets. The image datasets differ in the amount of total images. Scale bars are not contained in the image dataset. Scale bars: a) 120 µm, b) 350 µm, c) 500 µm.

computationally simulated organoids are included. The data of that class only contains image masks of simulated organoids. The two simulation datasets differ in their growth patters. One dataset contains data of simulated organoids with an exponential growth including branch elongation, branch thickening as well as creation of new branches. The other datasets contain simulated organoids where growth happens in three different growth phases. The first phase simulates exponential branch elongation, where branching events are increased, and branch thickening is limited. The second phase simulates linear growth where branch elongation, new branching events and branch thickening is inhibited. The third phase simulates inverse exponential growth, where branch elongation and branching events are inhibited while branch thickening is promoted. This leads to characteristic growth patterns contained in the image sequences of both simulation datasets.

Datasets 1-3 contain microscopy images. Raw microscopy image data are provided in the Leica Image File Format (LIF). All data were pre-processed to make them available to the individual deep learning networks.

3.2.1. MDCK cell nuclei dataset

The dataset contains a 2D image sequence of fluorescently labeled nuclei of MDCK cells. The cells were stained with SiR-Hoechst, a far-red DNA stain[43]. Cells were incubated with Verapamil to improve fluorescence signal. Videos were acquired with a 20x (0.4 NA) air objective, a frame rate of 10 minutes and a total capture time of 15 hours. The training dataset consists of a total of 5 images containing 6409 nuclei. The test image contains a total of 792 nuclei. All nuclei were annotated manually in order to provide ground truth data. The training and test images were captured at different positions on the same sample. 16-Bit images with a 1:1 aspect ratio and a pixel size of 2048 x 2048 are used. Figure 3.3 shows a sample of the time series image data. Over time, the cell layer is growing to cell confluence. Growth of the cell tissue occurs due to cell divisions and movements of the cells. Cell division often shows an increased fluorescent signal.

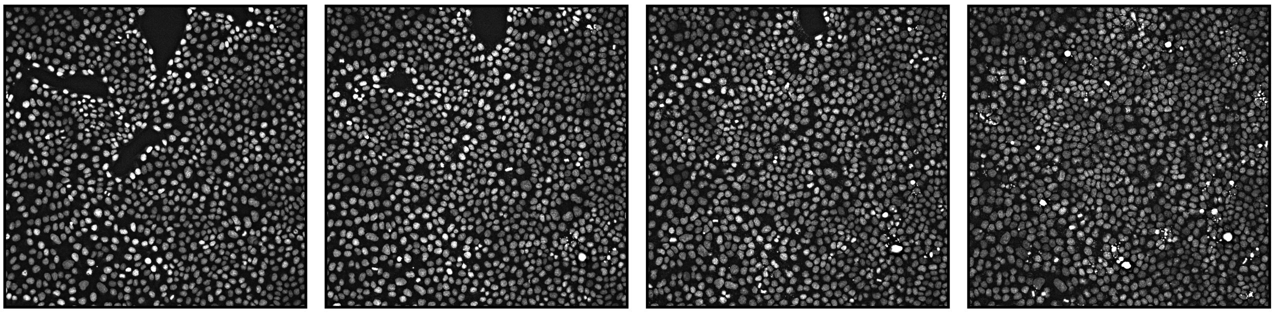


Fig. 3.3: Time series data of Madin-Darby Canine Kidney (MDCK) cells. The data contains microscopy images of fluorescently labeled nuclei of MDCK cell growth. The dataset covers the growth process and movement of cells up to confluence of the cell layer.

3.2.2. PDAC cell dataset (2D)

The dataset contains time-series image data of PDAC cells. The area of 2-well plates by *Ibidi* was imaged hourly. Images were acquired with a 5x (0.12 NA) air objective, a frame rate of 1 hour and a total capture time of 72 hours. Figure 3.4 shows sequential image data of PDAC cell growth. Over time, the cell layer grows to a confluent cell layer. Additional to the brightfield microscopy images, segmentation masks of the cell layer were generated by a python script.

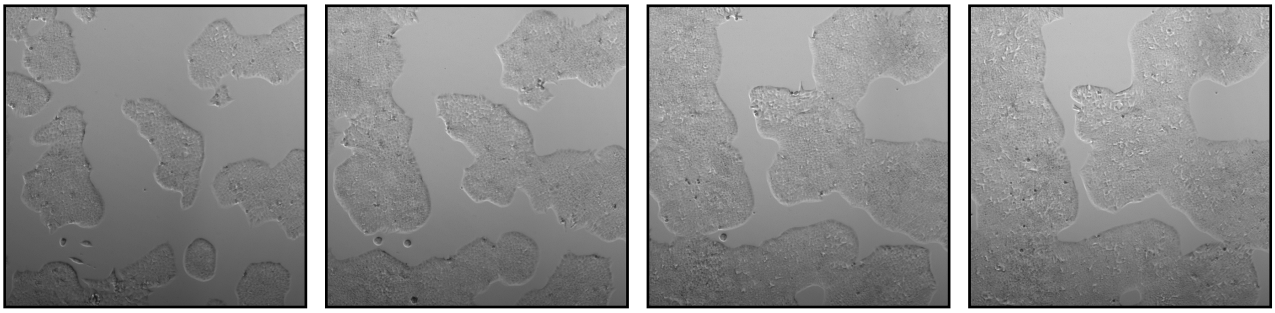


Fig. 3.4: Time series data of pancreatic ductal adenocarcinoma (PDAC) cells. The data contains brightfield microscopy images of PDAC cells.

3.2.3. PDAC organoid dataset (3D projection)

Dataset No. 3 contains time-series and volumetric image data of PDAC organoids. The dataset is split into 80% training and 20% test sets. Both datasets contain time series image data comprising 72 hours of live-cell imaging. The time interval of imaging is one hour. Cells were imaged by wide-field microscopy (Leica THUNDER Imager). For both datasets, a 5x (0.12 NA) air objective (Leica) was used. Images were captured with a 1:1 aspect ratio and a pixel depth of 8-bit to ensure high-throughput measurement of multiple positions with a time interval of one hour between the frames of the time series data. Measurements were conducted in a controlled environment at 37 °C, 5% CO_2 and 100% humidity, using an incubation system for live-cell imaging by TOKAI.

Figure 3.5 illustrates the sequential image data of one single organoid. The structural development of the organoid is captured over a time-frame of ~72 hours. The 2D images show 3D minimum projections of a volume containing the whole organoid in collagen. The distance between image slices in z-direction is ~15 µm. Brightfield microscopy images as well as segmentation masks are contained in the dataset. The segmentation of the organoids was performed manually due to the complexity of certain organoid structures.

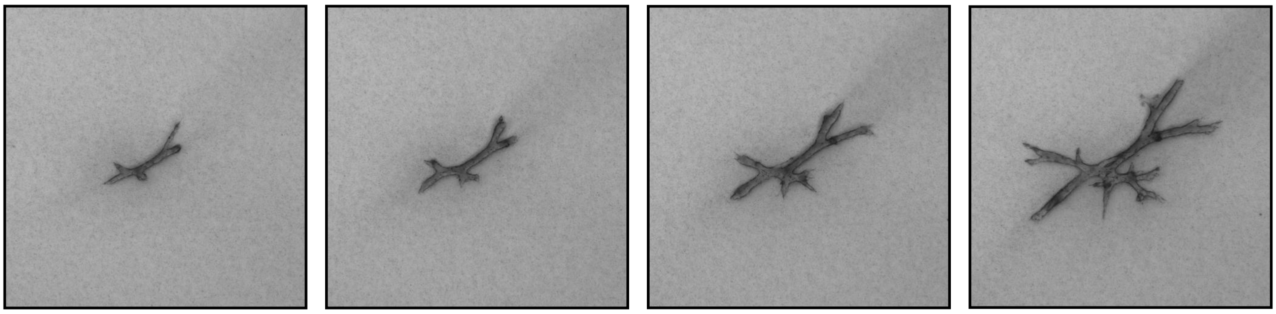


Fig. 3.5: Time series data of pancreatic ductal adenocarcinoma (PDAC) organoids. The data contains brightfield microscopy images of PDAC organoid growth.

3.2.4. Organoid simulation datasets

To avoid limitations in the training dataset size of a neural network, simulation data is used. The script to generate simulation data of organoid growth is based on the work provided by Edouard Hannezo[58]. The dataset contains time series data of structural growth. The idea of the dataset is to mimic the simplified structure formation processes in real organoids. Figure 3.6 shows a sequence of the simulated structure formation process with exponential growth. The simulation data includes elongation processes of branches, formation of new branches as well as thickening of branches. The rate or speed at which these events occur are adjustable, leading to individual growth formation processes. The growth formation process is randomized for individual organoids. This leads to different structures for individual organoids, each containing the same underlying growth pattern.

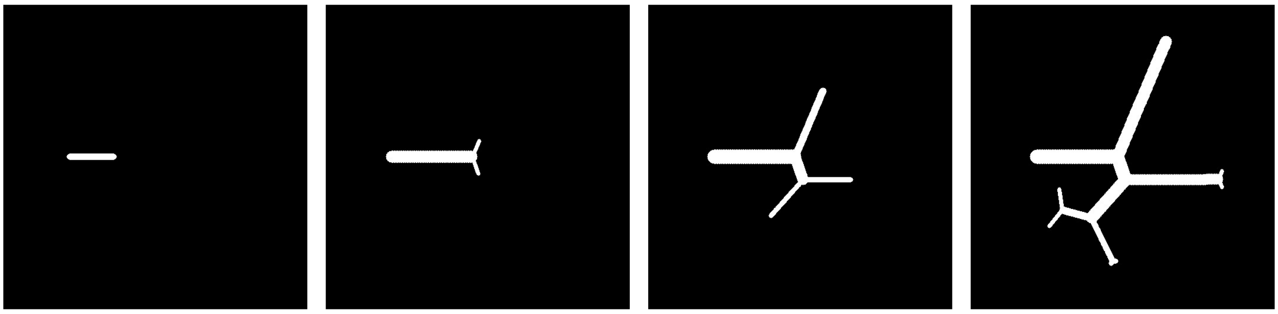


Fig. 3.6: Growth stages of simulated organoids. Four time points of the development stage of a simulated organoid image dataset. The image sequence of each simulated organoid contains a total of 70 time points.

To improve the analysis of the underlying growth processes, two datasets with different structure formation patterns are generated.

The first of both datasets contains simulation data of organoid development with a single growth pattern. The second dataset contains simulation data with three growth patters by containing the same number of images in an image sequence. The second dataset is thus containing a more complex structure formation process.

In order to train both networks with simulation data, image sequences of 1000 organoids are generated. Figure 3.7 shows a comparison of the image sequence of both simulation dataset. While the first dataset demonstrates exponential growth over the whole time sequence, the second dataset contains three different growth phases.

Phase 1 represents exponential growth including branch elongation, branching events while thickening is inhibited.

Phase 2 shows a linear growth with a slow rate of branch elongation, while branching events and branch thickening are minimized.

Phase 3 demonstrates inverse exponential growth, where branch thickening is promoted while new branching events and branch elongations are inhibited.

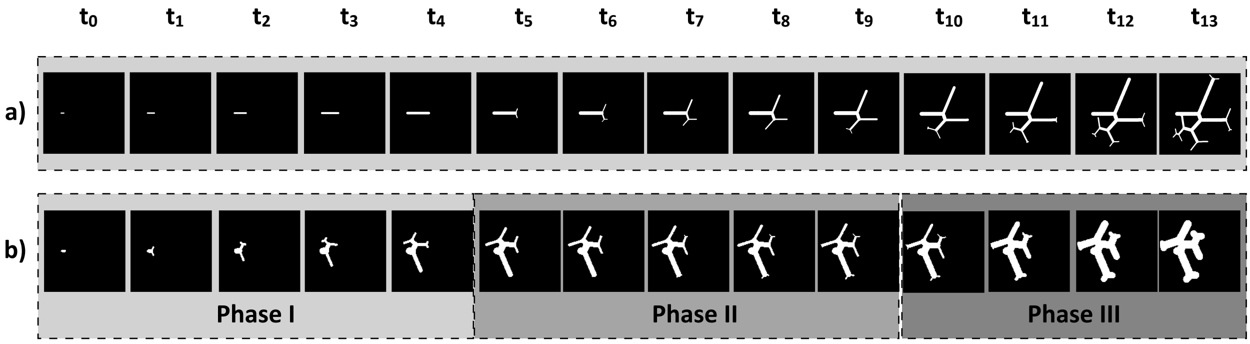


Fig. 3.7: Comparison of simulation test datasets. Image sequence data of simulated organoids is shown for 14 time points. Between every time point 6 frames are acquired. The test set contains a total of 90 images. a) Simulation data of exponential organoid growth. b) Simulation data of organoid growth with 3 different growth phases. Phase 1 shows exponential growth. Phase 2 shows linear growth at a small rate of structural change. Phase 3 represents an inverse exponential growth pattern.

Each sequence in the training and validation set covers the structure formation process of a simulated organoid by 70 frames. Thus, the whole dataset contains a total of 70,000 images. The test dataset includes the same amount of simulated organoids in 70 frames plus an additional 20 frames, resulting in a total of 90 frames. The 20 additional frames allow a more detailed analysis of far future predictions with the neural network. The training dataset only contains data between time points t_0 - t_{10}, representing a total of 70 frames per sequence. Each image is generated in a resolution of 512x512 pixels with a pixel depth of 8-bit.

3.3. Data processing

To be able to use the acquired datasets with deep neural networks, it is necessary to process the image data to meet the requirements of each individual network. Due to the amount of image data in each dataset, the processing has to be automated to provide a high quality of training, validation and test data in a reasonable amount of time. Image processing in cellular biophysics is typically performed with specific software such as *Imaris*, *Aivia* or *FIJI / imageJ*. These software tools are typically set up with powerful analysis and automation tools. For specific datasets, analysis methods and automation tasks these software tools are limited. Most of the image data processing was therefore set up and automated by python scripts, avoiding the possibilities of errors due to human interaction in this process.

3.3.1. Self-supervised learning

The first experiment of this thesis focuses on a self-supervised learning algorithm for nuclei segmentation.[25] Self-supervised in this context refers to the process, that an algorithm is automatically detecting cell nuclei in microscopy images and segmenting these. The automatically annotated nuclei can then be used as an additional input for the training dataset and improve the segmentation performance by retraining the deep neural network. This enables a self-improving system that allows a high data throughput in comparison to manual data labelling, which is limited to smaller datasets due to the sheer workload. Raw microscopy images in most cases do not offer high quality data for machine learning approaches. A quality standard for the image dataset is therefore important to guarantee reliable results from a neural network which is trained on these data. To provide high quality of training, test and validation data to a neural network, the image processing, beginning from raw microscopy data, has to be done carefully. For automation of the segmentation process, a neural network together with traditional image processing methods are used.

The input training data of the algorithm consists of microscopy images of fluorescent labeled MDCK cell nuclei, together with segmentation masks of the cell nuclei. To automate the segmentation process of the nuclei, several image processing steps are required. These are illustrated in figure 3.8. The automated data labelling process is compared to a manual annotation approach. While manual data annotation requires only simple pre-processing of raw microscopy images, automated labelling uses several image processing algorithms to achieve the same goal. The output of images and segmentation masks can be directly used as an input of training data for the neural network. The combination of the automated annotation together with a segmentation neural network allows a self-supervised learning approach.

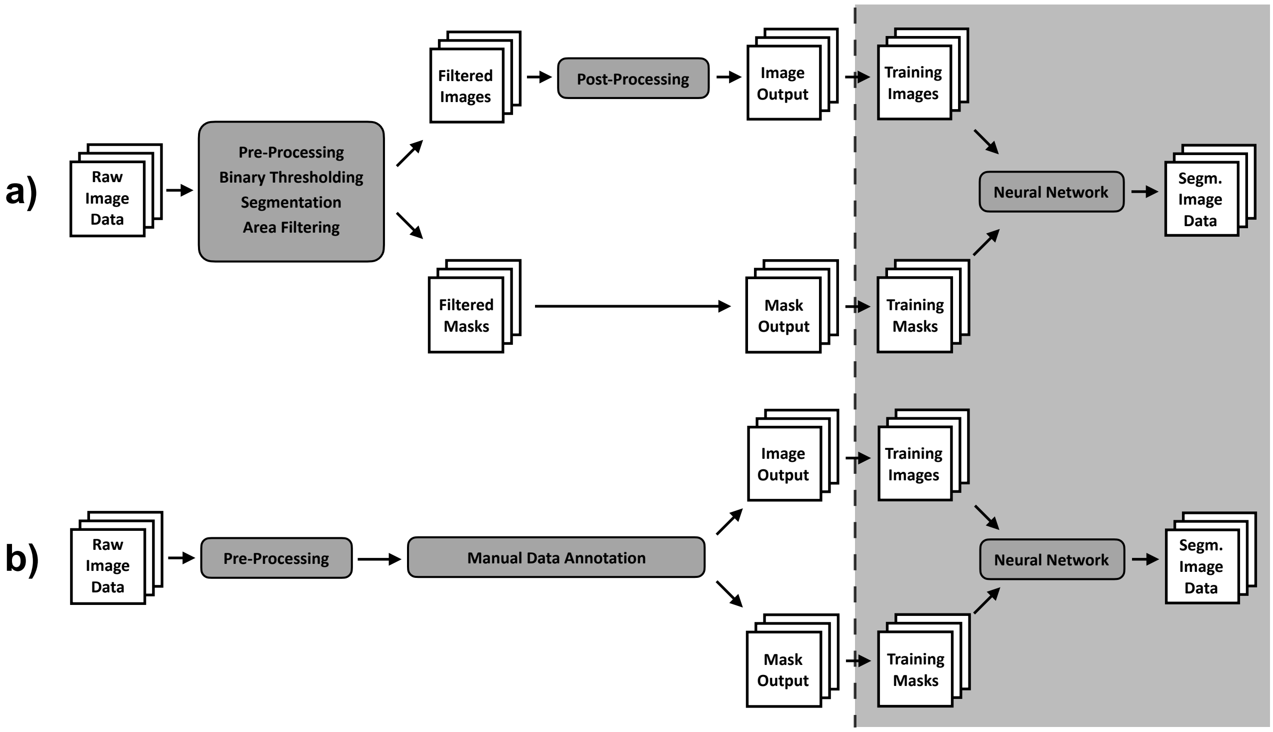


Fig. 3.8: Process of automated and manual nuclei annotation for segmentation with a neural network[25]. a) Automated data annotation considers image pre-processing, binary thresholding, watershed-segmentation, filtering and post-processing to provide a training dataset for nuclei segmentation with a neural network. b) Manual data annotation is applied on raw image data (after pre-processing) in order to provide a training dataset of images and annotation masks.

The following section demonstrates the image processing steps for automated labelling of cell nuclei.

Pre-processing: The processing step includes pixel value conversions, resizing or cropping of the microscopy images[11,62,74] to unify the unstructured dataset. This enables the image dataset to meet the training and testing requirements of the neural network.

Otsu's method: Automatic thresholding of binary images is done with the thresholding method by Nobuyuki Otsu et. al[53]. The algorithm provides an automated thresholding, based on the image histogram, to separate two object classes in microscopy datasets - specifically fluorescent labeled nuclei from its background[11,74]. The algorithm determines the threshold value that minimizes the weighted sum of variances of these two classes:

$$\sigma_w^2(t) = W_0(t)\sigma_0^2(t) + W_1(t)\sigma_1^2(t) \tag{1}$$

Watershed segmentation: To separate the image into foreground (cell nuclei) and background, watershed segmentation is applied[9,10]. It is a widely used segmentation algorithm which is able to add a label to each individual object of the microscopy image[11,35,45,53,74,75]. These objects include single nuclei, overlapping or touching nuclei, nuclei clusters, nuclei during cell division, as well as image artifacts such as weak fluorescence signal or other artifacts from imaging or culturing cells.

Pixel area filtering: To identify objects in the segmentation mask, which are other than single nuclei, the pixel area distribution of all objects in the dataset is assessed. The median pixel area of all segmented objects is selected. This results in a dominant object class (cell nuclei) and several outliers objects, such as touching nuclei, overlapping nuclei or image artifacts. A positive and negative threshold is applied in order to exclude objects larger (e.g. cell clusters) or smaller (e.g. cell debris particles, image artifacts) than 50% of the median pixel area[75]. As a result, outliers which are most certainly other than nuclei are filtered in the dataset. This results in a filtered dataset of images and annotation masks containing primarily single nuclei with a mean error of including non-single nuclei objects (false positives, false negatives) lower than 1%. Compared to the ground truth dataset, the new image dataset contains a total number ratio of 68% of the objects or nuclei.

Post-processing: The thresholded nuclei segmentation masks are subsequently separated into a foreground (cells) and a background. To improve the performance of the segmentation network[48], random noise is added to the background of the images before training with a neural network[11,17,74].

3.3.2. Automated segmentation

Accurate segmentation of organoids in microscopy image is a non-trivial task. Segmentation with a neural network requires several processing steps of the input image data to prepare a training, test and validation dataset for deep learning applications.

Classification: To use microscopy image data with a deep neural network, high quality of input data has to be insured. While a certain amount of variance within the training data is beneficial to the overall output, there are limitations. Objects in the image data which obviously do not represent the dataset such as artifacts while imaging or from cell culture can degrade the overall quality of the dataset. Classification can therefore be used in terms of a quality control for the dataset of the segmentation network. The *inception v3* network[68] is used maintain the quality of a training set of microscopy images for further processing.

Image registration: Due to minimal vibrations during microscope, there is a drift of the organoids in respect to the field of view of the microscopy. The effect appears with any biological sample during microscopy. Organoids are furthermore cultured in a 3D collagen matrix. The collagen is not attached to the well plate. It therefore results in floating gel, which increases this problem of a drift of the object in respect to the microcopy field of view. The result is a subtle but constant translation and rotation of the organoid within the field of view of the image sequence. To encounter this problem, the image registration algorithm *StackReg*[70] is used. Depending on the complexity of the drifting object, transversal or rotational movements can be corrected. In most cases, the rigid body transformation leads to the desired results. Rigid body transformation is calculated by where $\overrightarrow{x_0}$ is mapped to the output vector $\overrightarrow{x_1}$, and $\Delta\vec{x}$ refers to the amount of translation.

Projection in z-direction: Projection allows the visualization of a 3D dataset in two dimension. Projections help to reduce the complexity of a 3D dataset while maintaining relevant image information. Projections can be performed in various ways to enhance the image dataset. In this processing step, minimum intensity projection was used. The output of the minimum projection sums up minimum pixel values of microscopy slices in z-direction, providing improved structural information for automated segmentation with a deep neural network.

Post-processing: Conventional image processing methods are used for data preparation. These include pixel value conversions, resizing or cropping of the image dataset[11,62,74] to unify the image data for neural network applications.

Data Augmentation: Data augmentation is an image processing technique to enhance the size and quality of a training dataset by applying a set of geometric transformations to the images in the dataset. These transformations include rotation, shifting, flipping, zooming, and other operations. The goal of data augmentation is to artificially provide higher diversity in the training dataset while simultaneously increasing the training dataset to prevent overfitting. Data augmentation can help to provide enough data for machine learning applications in scenarios where the actual generation of data is complicated or even impossible. Especially in the field of cell biology, the cultivation of cells is time and labor intensive which limits the generation of image data[65,67].

3.3.3. Growth prediction

The third experiment of this thesis focuses on a future frame prediction model to predict organoid growth. Simulation and microscopy image sequences are used as input data of the machine learning model. These include Projection of the images, image registration, pre-processing and segmentation of the organoids. The data processing is required only for the biological image dataset. There is no processing required for the simulation dataset.

Projection: To obtain high quality imaging results, microscopy of organoids was done in three-dimensional space. The prediction networks however require two-dimensional image information. Minimum projection was used to reduce the three-dimensional data to two-dimensional images. The algorithm combines the minimum pixel values of images in z-direction and provides the output in a single 2D image.

Registration: Due to minimal vibrations during microscope, there is a drift of the organoids in respect to the field of view of the microscopy. The effect appears with any biological sample during microscopy. Organoids however are cultured in a 3D collagen matrix. The collagen is not attached to any border in the well plate. It therefore results in a floating gel, which increases this challenge. The result is a steady translation and rotation of the organoid between single frames. To encounter this problem, the rigid body transformation for image registration is used.

Pre-processing: Pixel value conversions, resizing, rotation or cropping of the image dataset was performed[11,62,74] to unify the dataset and make it applicable for training with the neural network used for image prediction.

Segmentation: Segmentation of organoids in the microscopy image was done by manual annotation of each organoid in the dataset. This ensures maximum accuracy for ground truth data. Manual annotation time of a single organoid was several minutes per organoid depending on the accuracy of annotation.

3.4. Deep learning algorithms

Deep learning algorithms have proven to exceed the results of data analysis compared to traditional analysis methods in many disciplines[31,34,50]. In this thesis, several deep learning algorithms are used to analyze microscopy images and image sequences of organoids in vitro. These include classification, segmentation and future frame prediction models. The networks receive high quality microscopy as well simulation image data. The raw image data require a processing to meet the requirements of the individual deep neural networks.

3.4.1. Classification

Image classification in context of this thesis is used for quality control of the input dataset. Microscopy images of cells or organoids can show image artifacts due to a variety of reasons. This can be biological reasons like cell apoptosis due to environmental reasons such as a drop of temperature due to leakage of the incubation chamber or phototoxicity during fluorescence microscopy.

The classification algorithm is based on the *inception v3* architecture by Christian Szegedy et al.[68]. It was published in 2016 as a successor of the *inception* architecture from 2014[66]. *Inception v3* uses a system, which allows the network to learn abstract and complex features from images. The architecture of the network is shown in figure 3.9

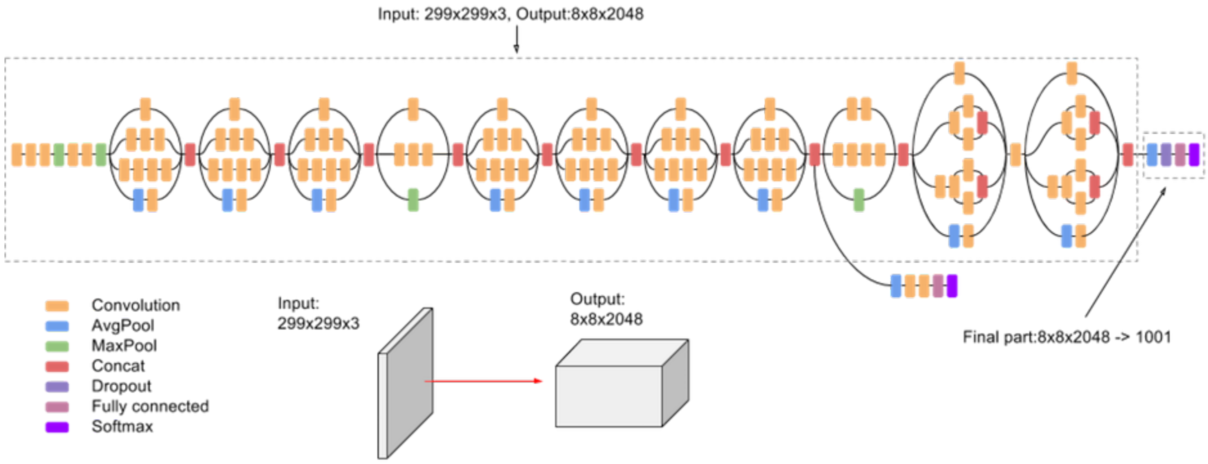


Fig. 3.9: Architecture of the *Inception v3* network for image classification[68]. The network consists in total of 42 layers. It is composed of a combination of convolutional layers, pooling layers together with 1x1 convolutional layers. To reduce the input dimensionality, 1x1 convolutional layers are used. This drastically lowers the amount of parameters in the network. The final layer of the network applies a softmax activation function that is used in order to make a classification prediction based on the input image.

The idea of the network is, instead of choosing specific filters for convolutions or pooling, the network instead applies multiple filters and concatenates the results. The network is in return able to learn which filters and which combinations work best for feature extraction. On the other hand, the computational cost due to this brute force method increases. To circumvent this issue, 1x1 convolutions are implemented to the input volume to reduce the dimensionality. This intermediate volume or bottleneck layer acts as an input for the actual convolution and in return lowers the computation cost to process the output volume.

The *Inception v3* classification model is pre-trained on the *ImageNet*[22] dataset. The dataset contains more than 14 million annotated images in over 20,000 categories. The network and the dataset are publicly available.

3.4.2. Segmentation

The segmentation algorithm is based on the *U-Net* architecture by Olaf Ronneberger et al.[26]. It is a convolutional neural network for biomedical image segmentation. Figure 3.10 shows the *U-net* network architecture. The network is composed of an encoder (contracting path) and a symmetric decoder (expanding path). The encoder takes an image tile as input and successively computes feature maps at multiple scales and abstraction levels yielding a multi-level, multi-resolution feature representation. The decoder takes the feature representation and classifies all pixels at original image resolution in parallel[26]. A main feature of *U-Net* is a flow of information between the encoder and decoder at the same level of image resolution. It is also referred as skip connection or shortcut connection and helps to increase the segmentation accuracy.

Based on the *U-Net* architecture, the *StarDist* network by Uwe Schmidt et al.[63] is used in combination with conventional image processing algorithms for self-supervised nuclei segmentation. *StarDist* is a neural network architecture for instance segmentation of 2D and 3D cell nuclei. In comparison to *U-Net, StarDist* is able to predict a star-convex polygon object classifier as a part of the network. This prediction of star-convex polygons allows an accurate localization and segmentation of cell nuclei. Especially in dense cell layers, nuclei on microscopy images can be very close to each other, thus hard to segment and separate. Even in this difficult segmentation scenarios the network demonstrates high performance. This method is particularly helpful in the context of cell tracking tasks[63].

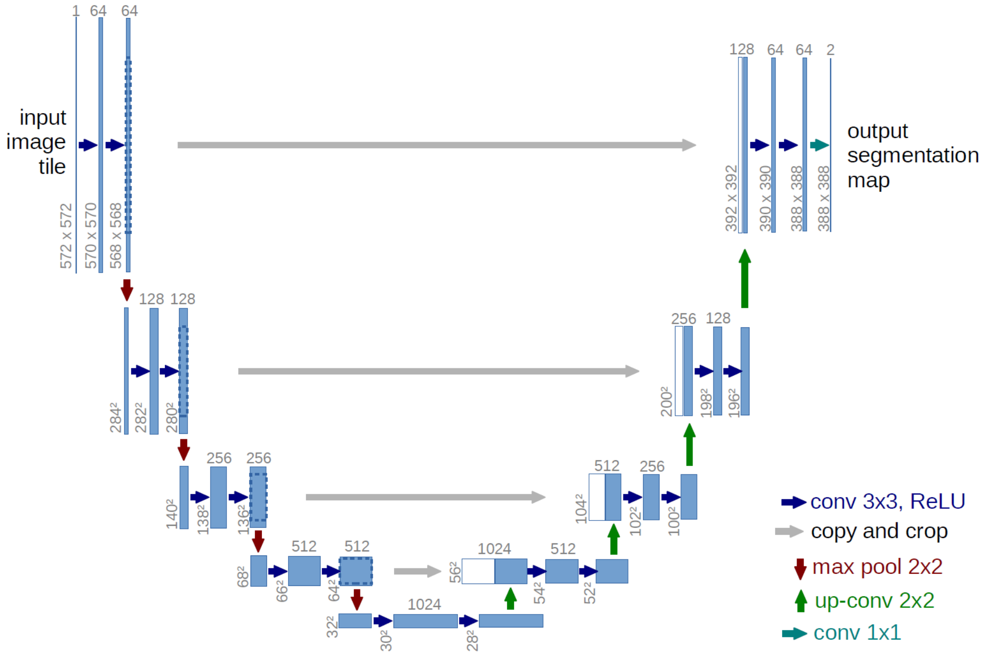


Fig. 3.10: Architecture of the *U-Net* network for image segmentation[26]. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Figure 3.11 illustrates the segmentation approach of the *StarDist* network. The training data contains pairs of input images and corresponding annotated masks. The model is trained to predict the boundary of fluorescently labeled nuclei based on a star-convex polygon and a probability map. This approach of the network is not only limited to 2D data. Also, 3D nuclei can be segmented, given volumetric input images together with annotated labels in the training dataset.

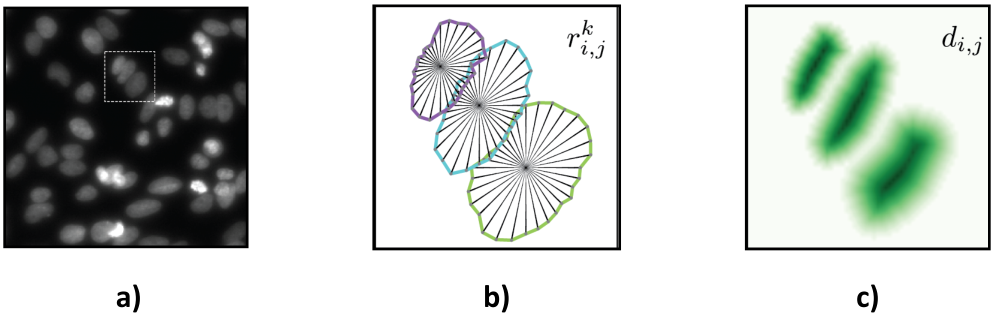


Fig. 3.11: Working principle of the *StarDist* segmentation network for cell nuclei[63]. a) Microscopy image of fluorescent labeled nuclei. The density of the cell nuclei promotes segmentation errors with traditional object segmentation algorithms. Identifying touching nuclei as separate objects requires a more sophisticated segmentation approach. (b) The *StarDist* method is able to fit star-convex polygons by the radial distances $r^k_{i,j}$ into objects to identify individual nuclei. c) Additional object probability maps $d_{i,j}$ are predicted by the network. The combination with the star-convex polygons allow separation of cell nuclei with high precision.

3.4.3. Future frame prediction

The algorithm for future frame prediction is based on the FutureGAN architecture by Sandra Aigner et al. (fig. 3.12)[1]. It is a generative adversarial network which is able to predict future images based on time-series sequence data. The network is an extension of the progressively growing GAN method by Karras et al.[39].

Progressive Growing of GANs (PGGAN) is a method for training a generative adversarial network to generate realistic images in high resolution. The main idea of PGGAN is to gradually increase the image resolution starting with low-resolution images by progressively adding layers in the encoder and decoder network. By this, the GAN is able to learn features and patterns from images of increasing complexity as the training progresses. The architecture was extended for the task of image sequence prediction. By using the FutureGAN architecture, the model is able to learn underlying patterns and dynamics of objects in an image sequence, and use the learned information to generate realistic image predictions of how the objects will propagate in the future[1].

Similar to a GAN model, the FutureGAN framework is based on the idea of training a generative model in an adversarial setting. It consists of two separate networks, a generator and a discriminator. The generator network is trained to predict a sequence of future image frames given a sequence of past frames. The discriminator network is trained to distinguish between the generated image sequence and a real sequence from the training dataset. The discriminator alternately receives real and fake / generated sequences as an input and determines whether the sequence appears real or not[1].

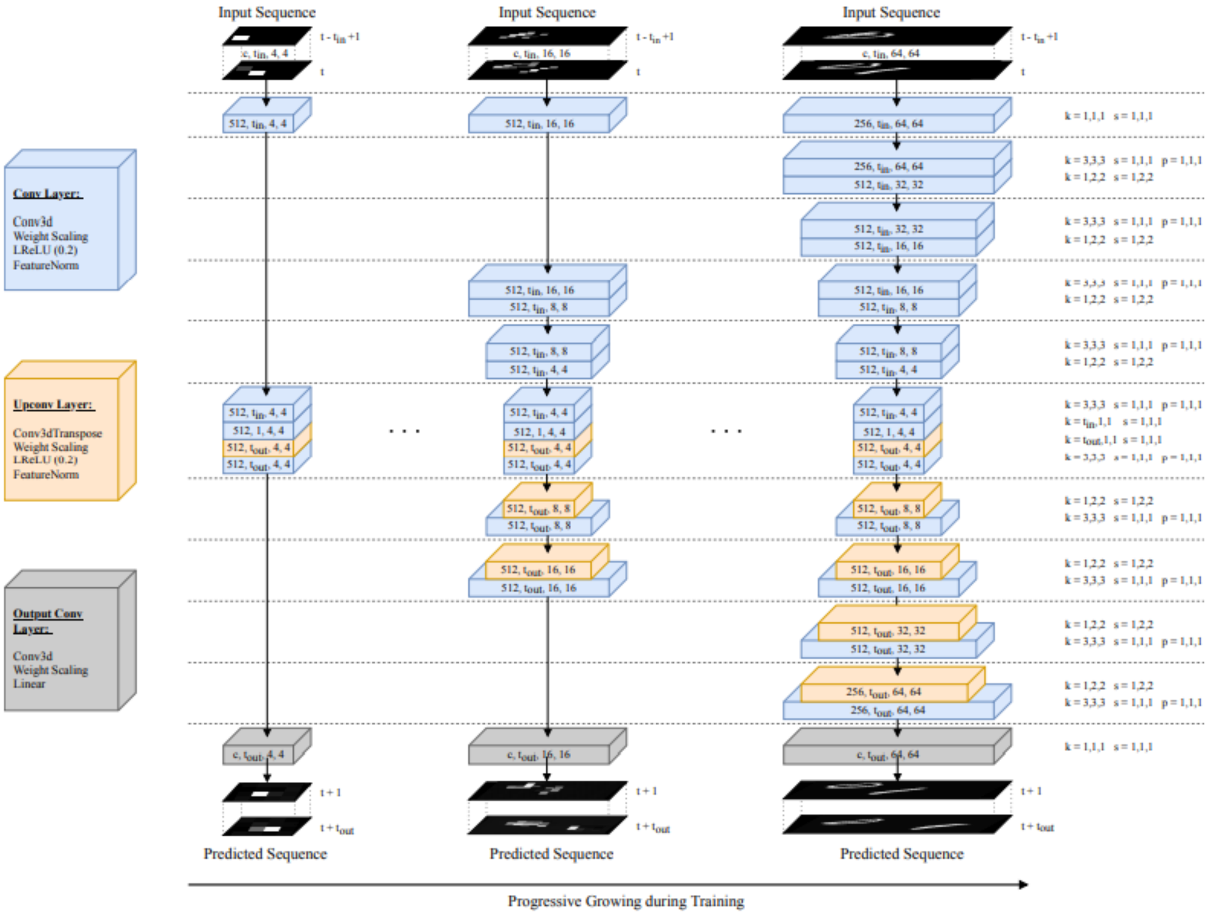


Fig. 3.12: Architecture of the *FutureGAN* generator for training a dataset[1]. The initial step of model uses a set of 4 × 4 px resolution frames as wells as output frames of the same resolution. Layers are added progressively added during training to increase the resolution after a certain number of iterations. The image resolution of the input frames always matches the resolution of the current state of the network. In this example the figure demonstrates the growth progress of the generator for the MovingMNIST dataset with a final resolution of 64 × 64 px. The full generator network furthermore includes 8 × 8 px as well as 32 × 32 px resolution steps.

3.5. Evaluation

To evaluate the performance of segmentation and prediction, microscopy images are segmented and converted into image masks. This allows pixel wise classification analysis between ground truth (GT) mask and segmentation or prediction masks. To evaluate segmentation and prediction performance, several evaluation metrics are used[25]. These include True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN), Precision, Recall / Sensitivity, Accuracy, Specificity as well as the Jaccard Index.

Jaccard Index: Jaccard similarity coefficient was used as a measure of similarity between two images[56]. It is calculated by using the Intersection over Union (IoU):

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{3}$$

For further analysis the confusion matrix parameters True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) are evaluated[77].

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

Table 3.1: Elements of a binary confusion matrix. Positive and negative classifications are described by true positives, false negatives, false positives, true negatives.

Precision: Proportion of true positive predictions, out of all the positive predictions.

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

Recall: Proportion of true positive predictions, out of all the actual positive cases. Recall is a synonym for sensitivity or the true positive rate (TPR).

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

Specificity: Proportion of true negative predictions, out of all the actual negative cases. It is a synonym for the true negative rate (TNR).

$$Specificity = \frac{TN}{TN + FP} \tag{6}$$

Accuracy: Proportion of correct predictions, out of all the predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

For evaluation of the future-frame prediction deep learning network, further evaluation was considered. These include the structural similarity (SSIM) index, mean squared error (MSE) as well as peak signal-to-noise ratio (PSNR). The measures are used to measure the quality of predicted images.

F1-Score: The F1-score is a measure for the performance of binary classification. It depends on the precision and recall of the model.

$$F_1 = \frac{2TP}{2TP + FP + FN} \tag{8}$$

Structural Similarity Index (SSIM): Measure of the structural similarity between two images by comparing their local means, standard deviations and cross-covariances[12]. A SSIM close to 1 indicates a high structural similarity between two images.

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{9}$$

Mean Squared Error (MSE): Measure of the average squared difference between the pixel values of two images[79]. Low MSE values indicate a high similarity between two images.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{10}$$

Peak Signal-to-Noise ratio (PSNR): Measure of the peak signal-to-noise ratio of two images[37]. High PSNR values indicate a better quality of the reconstructed image.

$$PSNR = 20 \times log_{10}\left(\frac{MAX_I^2}{\sqrt{MSE}}\right) \tag{11}$$

4. Self-supervised learning for biological image data

Annotating or labelling datasets for machine learning applications is a time-consuming and tedious task. Especially annotating objects in images can take several minutes per object depending on the complexity of the structure. Annotated datasets are often not publicly available. In those cases where annotated these datasets are available it is often the case that depending on the specifications of individual experiments, these dataset are often not suitable for obtaining precise results for deep learning algorithms.

This experiment demonstrates a self-supervised deep learning algorithm for automated labelling of fluorescent cell nuclei. The method can be applied to a variety of cell culture experiments. The *StarDist* deep learning network by Uwe Schmidt et al.[63] was used for automated segmentation of MDCK nuclei. To provide self-supervised learning of cell nuclei, the algorithm was implemented in a data processing pipeline. This enables to automatically annotate newly seen nuclei and provide this information to the training dataset. The training dataset increases, containing additional variations of cell nuclei. In return, more precise results after retraining the deep neural network are available. To test the capabilities of this method, tracking of individual nuclei based on the annotations was investigated. The results shown in this section have been published previously[25].

4.1. Data annotation

The idea of the self-supervised deep learning algorithm is to automate the cell nuclei annotation process. To evaluate the performance of the algorithm, it is necessary to compare the automatically annotated nuclei with ground truth data.

To provide ground truth data for the segmentation algorithm, cell nuclei of MDCK cells with a fluorescent were annotated manually. Due to the fluorescent signal, manual data labelling can be performed with high precision. Traditional segmentation algorithms for this task often lack in the precision of annotation. The challenges can be due to inhomogeneous signals of the fluorophor, separation of close nuclei or nuclei during cell division and also dealing with different sizes of nuclei. Figure 4.1 shows a microscopy image of fluorescent labeled MDCK cell nuclei. Each nucleus was manually annotated and acts as ground truth to test the annotation performance of a self-supervised deep

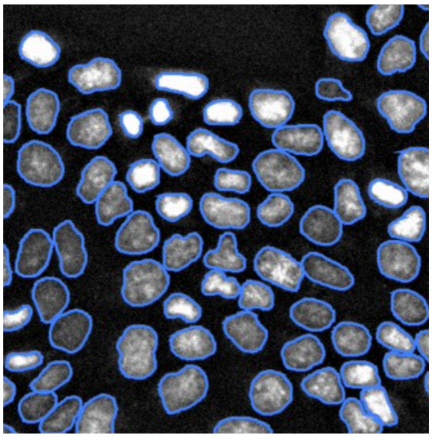


Fig. 4.1: Manual data labelling of MDCK cell nuclei. The annotated nuclei act as ground truth data for automated segmentation of cell nuclei. The fluorophor shows variations of signal intensity between nuclei.

learning segmentation network. To obtain precise results and a high variation of input data, a total amount of 6409 cell nuclei was annotated manually. The duration of annotation was ~15 hours for a single researcher.

4.2. Automated annotation

Segmentation performance of an automatically annotated training dataset after training a neural network was assessed. The results are compared to a manually annotated training dataset (fig 4.2). The neural network trained on automated annotation provides better segmentation of small nuclei (green arrows). Yet, it is only able to detect touching nuclei with low accuracy (red arrows), which are more efficiently segmented by the network trained on the manually annotated dataset (red arrows). This is attributed to the fact that touching nuclei are better annotated manually. Both training datasets - automatically and manually annotated - provide coequal overall segmentation performance after training a neural network, resulting in a mean IoU of 0.861 (precision: 0.982, recall: 0.984) and 0.863 (precision: 0.999, recall: 0.905).

The sensitivity of nuclei size and shape in our method is unfavorable if the dataset contains nuclei of big size in combination with high nuclei density (touching nuclei) and is favorable if the dataset contains nuclei of small size in combination with low nuclei density. In general, segmenting small nuclei accurately has a higher effect on overall segmentation performance compared to accurately segmenting touching nuclei[14] and should be considered when using the proposed method.

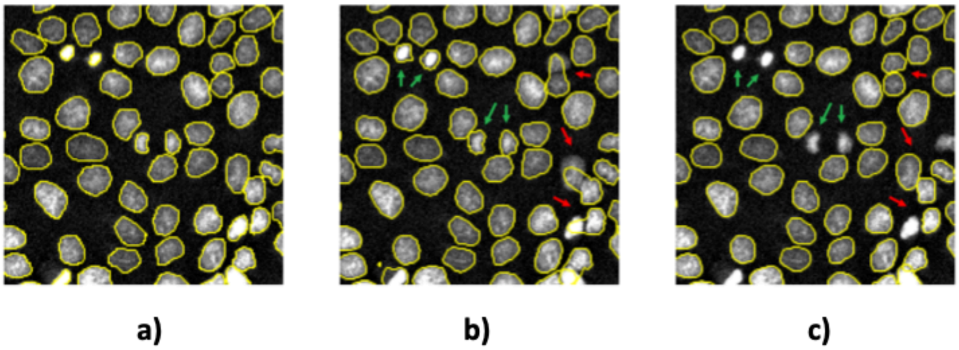


Fig. 4.2: Comparison of segmentation performance. a) Ground truth annotation data of cell nuclei. b) Segmentation of cell nuclei based on a neural network trained on automatically labeled data. c) Segmentation of cell nuclei based on a neural network trained on manually labeled data.

The adaptability of the method is demonstrated by applying the algorithm on a different cell nuclei dataset. The nuclei are imaged with a higher magnification objective (63x) compared to our previous results. Automated annotation on a new image dataset can quickly be performed and results in an accurate segmentation of the test images (mean IoU: 0.792). Results can be provided within minutes. Results are compared with the neural network which is trained on the manually annotated dataset at lower magnification

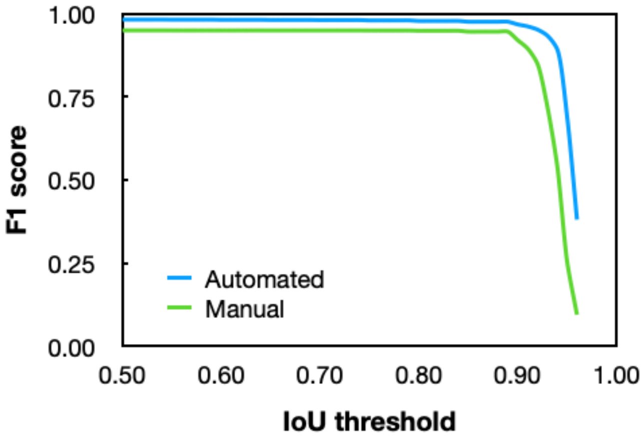


Fig. 4.3: F1 score of a segmentation network trained on a manually and automatically labeled dataset. Automated and manual annotation show equal segmentation performance after training a neural network.

(10x), which segments only poorly (mean IoU: 0.155). For better performance, a new time-consuming manual annotation of the dataset imaged at higher magnification would be required to train the neural network. Figure 4.3 shows the IoU threshold as a function of the F1 score for segmentation based on an automatically and manually segmented training dataset, confirming a high precision of segmentation with the neural network.

4.3. Conclusion and outlook

The proposed method demonstrates a self-learning machine learning algorithm to automatically detect and segment cell nuclei in fluorescent microscopy images. The segmented data can be used as additional training input for the segmentation network. This allows automated segmentation of nuclei with high performance. The network demonstrates equal segmentation results compared to segmentation based on manually annotated cell nuclei.

Segmentation of cell nuclei furthermore allows a tracking of cell movements. The method was used to track cell nuclei for a timeframe of ~8 hours. The software *Trackmate*[71] was

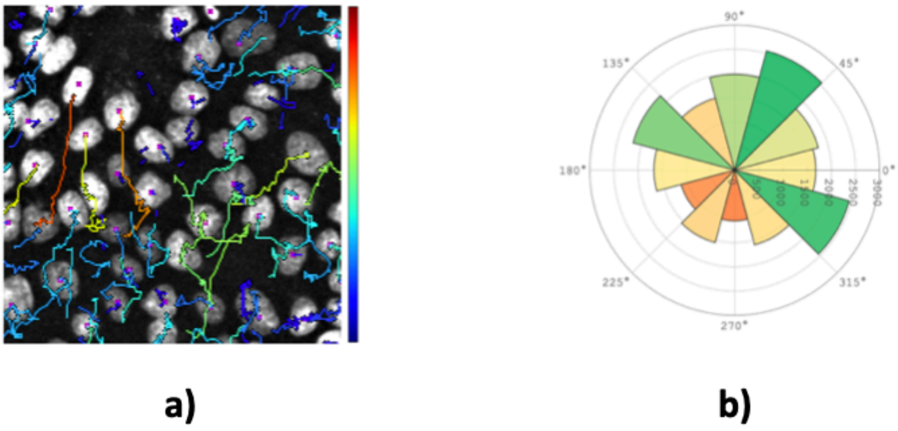


Fig. 4.4: Nuclei tracking results based on an automatically annotated dataset. a) Direction of migration of fluorescently labeled nuclei visualized by track-lines. The color of the tracks indicates the speed of each individual nucleus/cell (from blue: Low speed, to red: High speed). b) Distribution of the migration direction of cells within 8 hours 20 minutes. Predominant movement directions are obtained between 45°-75° and 315°-345° from its starting position.

used to monitor a trajectory from individual cells within an image time-series. *Trackmate* provides reliable tracking of fluorescent labeled nuclei. Results are shown in figure 4.4. To investigate cell migration, parameters such as density, speed, mitosis events, number of tracks or the migration direction of single nuclei can be readily obtained.

For this experiment a dataset containing 6409 nuclei of fluorescent labeled MDCK cell nuclei were manually annotated for providing ground truth. Even during annotation by a single researcher, the amount of decision processes for accurately labelling data– specifically including or excluding areas of interest, is prone to error. This introduces a bias to the dataset that should not be underestimated. Hence, the task of manual data annotation cannot be outsourced for biomedical applications. In fact, most of these datasets are split amongst several researchers for data annotation. A statistical error due to these decision processes amongst multiple annotators has to be considered[54]. The total annotation time of the manually annotated dataset is prohibitive for many applications. Moreover, the mean manual annotation accuracy varied for our experiment depending on the annotation duration. Automated annotation on the other hand provides a highly controllable and consistent way of nuclei annotation. Compared to manual nuclei annotation, automated annotation detects only a fraction (68%) of the total number of nuclei in an image. However, automated annotation can be applied easily to larger datasets. Therefore, the total number of annotated objects can easily be increased, which allows providing high-quality training data for a neural network.

To demonstrate the advantage and practicability of the fully automated process, we automatically annotated a large-scale microscopy image of a millimeter range containing more than 60000 nuclei, illustrated in figure 4.5. The nuclei are segmented with a neural network and consequently tracked over 8 hours and 20 minutes with a time resolution of 10 minutes. The whole process including our annotation method on the one hand as well as training the individual dataset on the other hand took less than one hour. The neural network was not pre-trained.

The method provides a solution to automate the data annotation process of cell nuclei in microscopy images. The automated annotation process reduces the time and labor-intensive manual data annotation to a minimum and can be adapted to individual datasets. Neural networks which are widely in use for biomedical or clinical applications

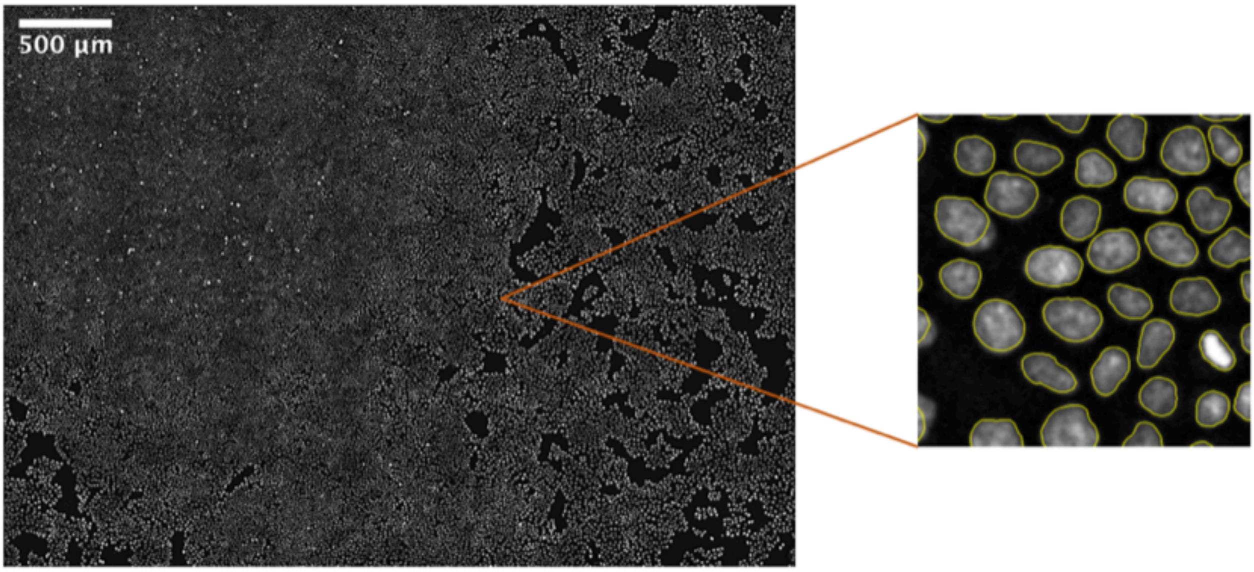


Fig. 4.5: Large scale nuclei segmentation of a widefield microscopy image trained on an automatically annotated dataset. The image has a field of view corresponding to 4.2 mm x 3 mm and containing more than 60000 single nuclei. Segmentation results are obtained within less than one hour, including (automatic) annotation and training with a neural network.

are used to train on these annotated datasets. Computer vision tasks for cell biology such as cell nuclei detection, segmentation or tracking can be analyzed in one single process with high segmentation precision. This allows instant access to experiment results and independence of pre-trained models (transfer learning)[86] or third-party datasets.

The presented automated annotation is based on the identification of one object class with low variance in the cell nuclei area (e.g. single nuclei of cells) and is independent of the object morphology. However, our system works best in combination with a neural network for segmenting star-convex objects, which improves the segmentation of touching nuclei. The method can be adapted for segmentation of more complex object morphologies, histopathological images, as well as 3D nuclei segmentation applications. The results demonstrate the effectiveness of automatic nuclei annotation for quantitatively analyzing microscopy time series images with high flexibility.

5. Automated segmentation of organoids

Segmentation of organoids in microscopy images is a non-trivial task. These annotations however are highly important in organoid research, as they allow a detailed analysis of organoid morphology. These range from volumetric analysis, obtaining the center of mass, to the number of branches in an organoid. With increasing complexity of the organoid morphology, this challenge of segmentation is even more difficult. To address this challenge, the performance of a neural network for segmentation was assessed. The network uses manually annotated image data of pancreatic ductal adenocarcinoma (PDAC) organoids at different developmental stages. Due to the limited throughput of biological microscopy data, the size of the dataset was artificially increased by data augmentation. The transformations include rotation, flipping, zooming and shifting of the original images.

5.1. Data annotation

Manual data annotation is a labor intensive and often tedious task. Data labelling often takes several minutes for a single object with a complex morphology. Annotated data however is mandatory for most machine learning and especially deep learning methods. In the case of organoids, their morphologies can range from relatively simple to complex structures, as shown in figure 5.1. The figure illustrates the variations of structural complexity of two organoids. Both organoids were imaged five days after seeding the cells in collagen and cell culture medium. The structure after five days is highly individual. To analyze the morphology of organoids it is helpful to assess segmentation masks of the microscopy images. These masks enable a more simple readout of organoid specific parameters such as the size of organoids, the elongation of branches, the total number of branches, the center of mass or the speed of growth. The data shows a projection of the organoids. Missing information in the z-direction has to be considered when analyzing the data.

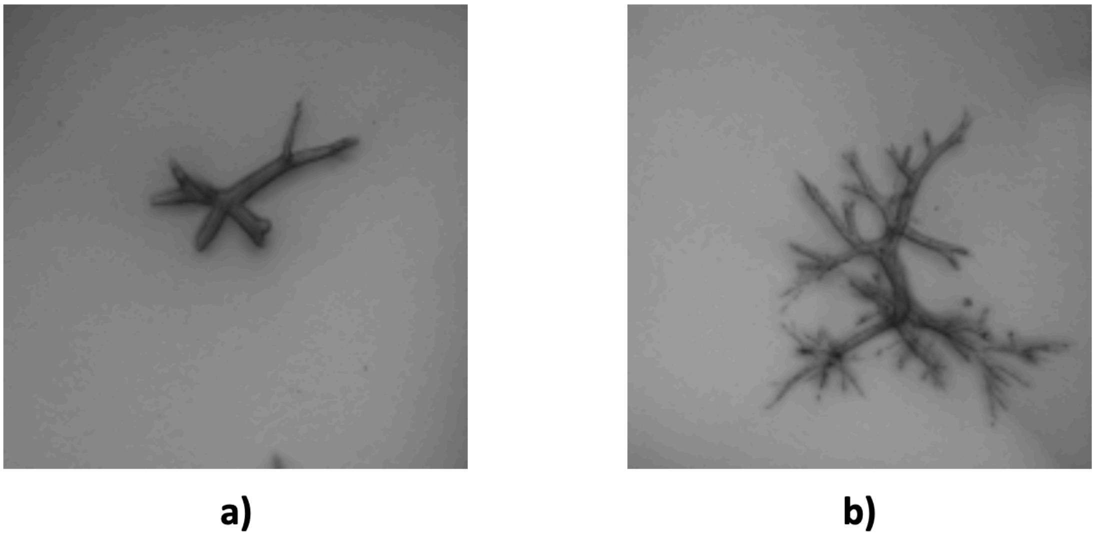


Fig. 5.1: Variations of structural complexity of pancreatic ductal adenocarcinoma organoids. a) Organoid with low morphological complexity. The microscopy image shows high contrast between the organoid structure and the background. b) Organoid with high morphological complexity. The organoid hast multiple very thin branches. Manual segmentation of the organoid with high precision takes several minutes and is complicated in regions where branches are thin and the contrast compared to the background is low.

In the specific case of future frame prediction, deep learning algorithms are most often able to process and predict colorized image data. The processing time however increases drastically compared to using image masks. Often it is not even necessary to use colorized image data, as all the relevant information is already contained in the image mask. In order to study growth formation processes in organoids, the morphology of the individual organoid is most relevant. This information is already contained in an image mask or binary image of an organoid.

Manual annotation was performed in collaboration with Samuel Randriamanantsoa, Sandra Andrusca and Sándor Battaglini-Fischer. Manual annotation of multiple people implies differences in the level of detail[54]. Samples of annotation masks are shown in figure 5.2. While manual segmentation of organoids can take several minutes, there are still differences in the level of detail of the segmentation mask. The training dataset contains images with variations in the overall level of detail of segmentation.

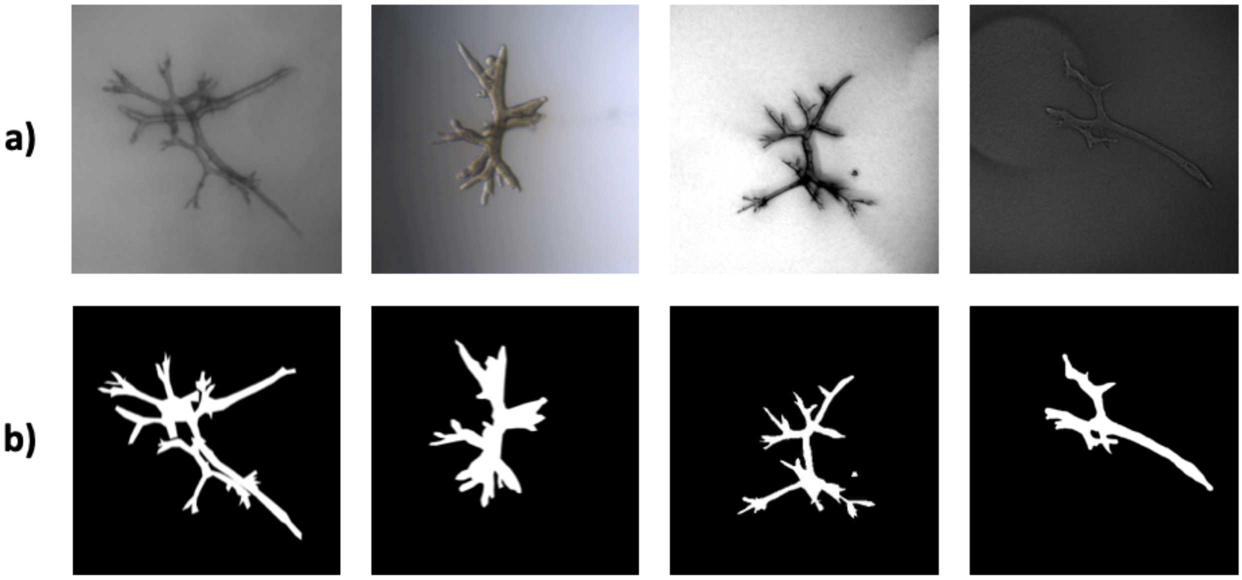


Fig. 5.2: Manual annotation of various microscopy images of organoids. a) Brightfield microscopy images of PDAC organoids. Microscopy images show differences in brightness and contrast of the images. b) Manual annotation masks of brightfield microscopy images. The segmentation masks show different levels of annotation detail and complexity of organoid morphology.

5.2. Automated segmentation

To test the performance of automated segmentation of organoids, several methods are used. These include segmentation by thresholding, semi-automated segmentation with a human-in-the-loop deep learning model, as well as an encoder-decoder system used for segmentation. Results are shown in figure 5.3, demonstrating different levels of segmentation accuracy. To measure the segmentation performance, the organoids are segmented manually. Masks are used as ground truth data. The image was segmented with four different methods. The first method uses segmentation by thresholding the brightfield image. This method sets a threshold to separate a foreground and a background at a certain image pixel value. In the case of grayscale image data, such as brightfield microscopy images, segmentation is a difficult challenge using a brightness threshold. Due to brightness gradients in the microscopy image, this segmentation is even more difficult. These brightness gradients however regularly occur, for example due to shadowing effects at regions close to the border of a well plate. For experiments where organoids are cultured in multi-well plates, this effect is even increased.

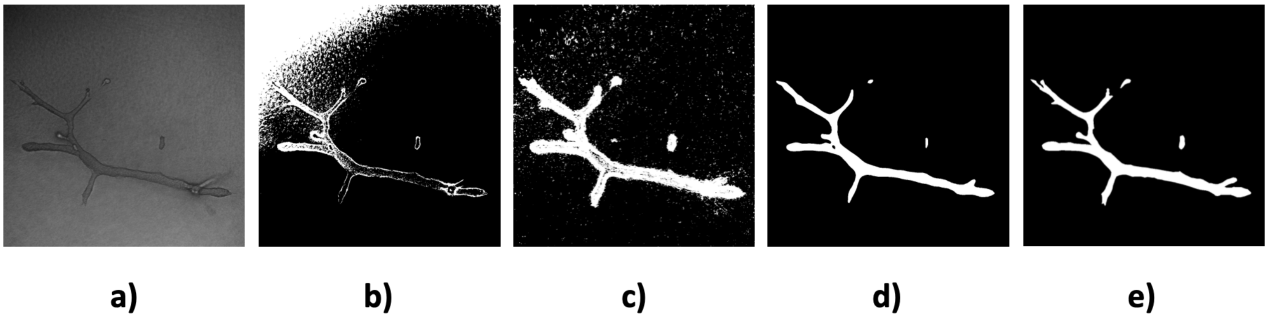


Fig. 5.3: Comparison of segmentation methods for microscopy images of organoids. a) Brightfield image of a PDAC organoid. b) Segmentation by thresholding. c) Segmentation with a semi-automated machine learning algorithm (human-in-the-loop machine learning). d) Segmentation with a deep neural network trained on organoid masks. e) Manual annotation mask (ground truth). Segmentation algorithms show different levels of accuracy from low segmentation performance (thresholding), to high segmentation accuracy (deep neural network).

Furthermore, a human-in-the-loop (HITL) machine learning model for segmentation was assessed. The segmentation results are shown in figure 5.3 c). The segmentation mask illustrates much more realistic segmentation of the organoid structure. For this example, the *Trainable Weka Segmentation*[5] method implemented in the *Fiji* software was used. For segmentation, the foreground (organoid) and background of the image are manually classified. The network is trained on the given image classifiers. The model can be refined, by adding regional information to the classifiers and retraining the model on this information. The method therefore requires human interaction to improve the segmentation results. The segmentation mask demonstrates improvements in segmentation accuracy compared to traditional segmentation methods. There are still many false positive predictions in the region of actual background. Depending on the application, the level of detail of the segmentation mask by using this method might not be sufficient.

U-Net, a neural network for segmentation is assessed. The encoder-decoder network was trained on manually annotated organoids with additional data augmentations of these images. The segmentation mask shows high structural precision (figure 5.3. d) compared to the ground truth (figure 5.3 e). The network allows automation of the segmentation with high accuracy without the need of human interaction. One requirement however is the availability of a manually labeled image dataset for training the network.

The evaluation of segmentation accuracy is shown in table 5.1. Segmentation masks obtained with conventional thresholding, a neural network (HITL) and a neural network (*U-Net*) are compared to ground truth segmentation data. The decoder-encoder system used for segmentation shows highest performance. It is trained on a dataset of organoid images and masks. To increase the variation and size of the training dataset, additional augmented data are added.

	Thresholding	Neural Network (HITL)	Neural Network
IoU	0.13	0.45	0.82
Precision	0.15	0.46	0.85
Recall	0.48	0.96	0.95

Table 5.1: Evaluation of different segmentation methods. The performance of three different segmentation methods was analyzed. These include, segmentation by conventional thresholding, segmentation with a human-in-the-loop neural network as well as segmentation with a neural network only. The intersection over union, Precision and Recall of segmented images compared to ground truth annotation data was assessed. The neural network without of a human shows segmentation results with highest accuracy.

Especially in cases with a contrast gradient, most traditional segmentation algorithms perform badly. Figure 5.4 illustrates an example where automatic segmentation is applied

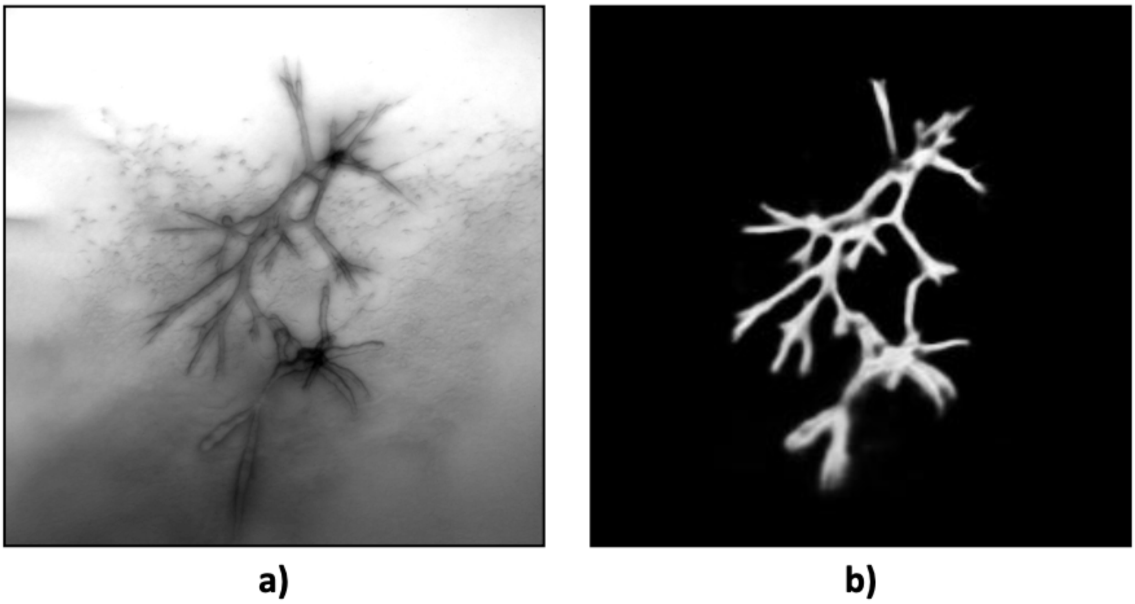


Fig. 5.4: Automated organoid segmentation of microscopy image with a brightness gradient. Even in cases where the illumination of the microscopy image is inhomogeneous the neural network demonstrates segmentation masks with high precision.

to a brightfield image with unfavorable brightness conditions in the image. The deep neural network demonstrates a high level of precision in detecting the overall structure of the organoid even in these more difficult scenarios.

5.3. Conclusion and outlook

Automated segmentation of organoids in microscopy images is an important task. Deep learning algorithms for future frame prediction often require high quality input data to guarantee high prediction performance. By using image masks instead of colorized images, the processing time for each training cycle can be reduced. In other words, for the same amount of processing time, much more images can be trained and analyzed using image masks.

The results of this experiment show that the segmentation accuracy of a machine learning architecture such as *U-Net* outperforms traditional segmentation algorithms or other machine learning algorithms where human interaction for refinement is required. There are limitations of this method, however. Depending on the complexity of the organoid structure, there are still noticeable differences between the precision of ground truth or manual segmentation and the prediction of *U-Net*. For less complex morphologies, the Jaccard index provides reliable results above 0.80. For more complex morphologies, the index can drop below values of 0.80. This effect can partly be compensated by increasing the amount and variability of input data.

Figure 5.5 illustrates the use of *U-net* for segmentation of organoids with different structural complexity. Visually, the segmentation demonstrates high segmentation precision. Numerical analysis shows, that the accuracy of segmentation lowers in cases of high morphologic complexity of the organoid. Increasing the dataset with additional organoid images and masks of this kind will most likely improve the performance of the segmentation network. The applications are therefore limited depending on the individual experiment.

Additionally, the use of colorized image data, such as with fluorescent microscopy, is highly promising for future applications. In addition to the morphological image information of organoids, the localization of cells or cell nuclei in the organoid for example can provide a better understanding of cell migration and structure formation in organoids.

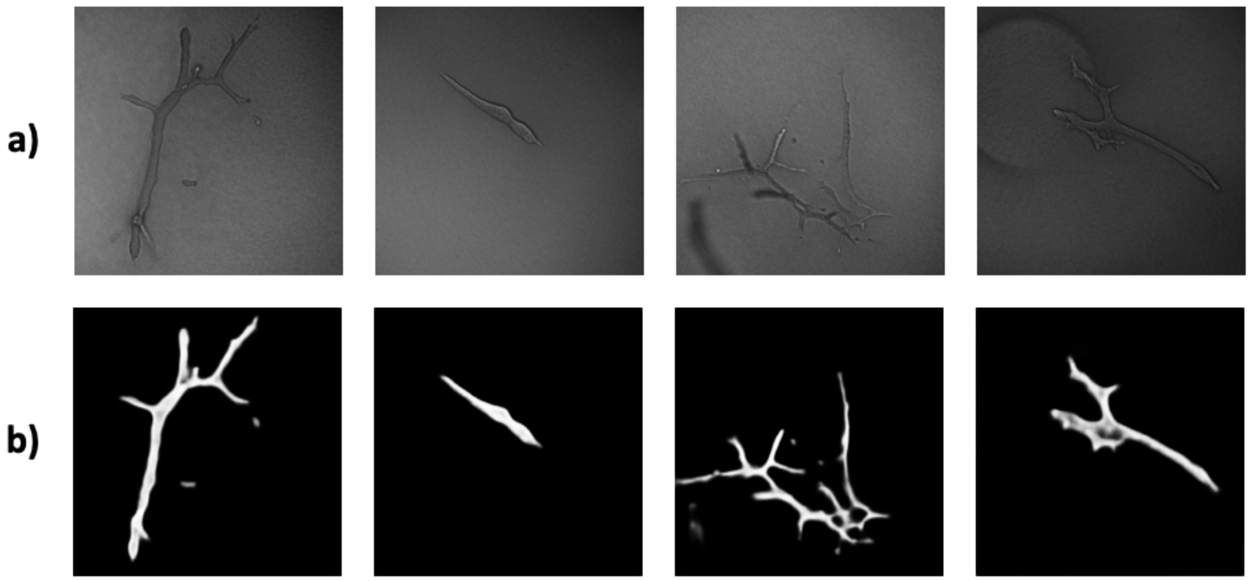


Fig. 5.5: Organoid segmentation results of microscopy images with a neural network. a) Brightfield microscopy images of PDAC organoids at different developmental stages. b) Segmentation masks of organoids generated with a deep neural network. The network provides segmentation masks with high level of detail. The network is trained on microscopy images and masks (including image augmentation) of PDAC organoids.

A limitation of this segmentation method is the specific use of two-dimensional data. Organoids however grow in a three-dimensional environment. 3D segmentation of organoids is still a complicated task which is prone to error in many cases. To simplify the segmentation and later prediction, 3D projections are used. To capture full information of organoid morphology, three-dimensional segmentation is required. Figure 5.6 illustrates an example of a three-dimensional representation of a PDAC organoid. The use of 3D image information would allow a more detailed analysis of organoids considering additional volumetric parameters. Implementations of 3D segmentation are already available to this date[16]. Considering structural predictions of organoids with a neural network, the use of volumetric data however exceeds the scope of this thesis.

Another limitation is the accessibility of biological data. Culturing and growing organoids is a process that takes several days up to weeks. This limits the overall amount of data throughput. High throughput applications in cell culture are more and more established, however often still not available in many research environments.

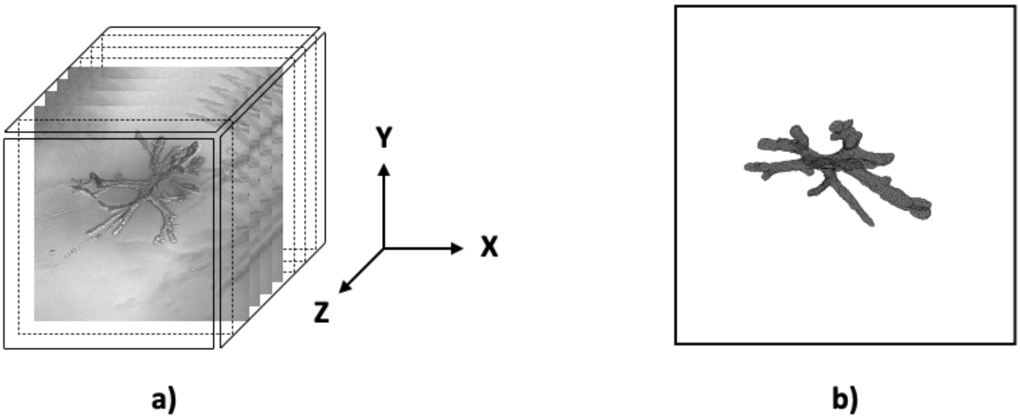


Fig. 5.6: Three-dimensional representation of volumetric microscopy images of an organoid. a) Volumetric image data of a PDAC organoid. b) 3D render of a PDAC organoid based in volumetric microscopy image data.

The proposed segmentation method provides many advantages compared to other traditional segmentation methods. Whether to use this method in a research environment or not depends on the fact if the segmentation performance is accurate enough for the individual experiment. The accuracy of segmentation compared to ground truth data can still be improved by increasing the training dataset and retraining the network. The method can be improved by using simulation data of organoids in combination with real microscopy data. The use of simulation data bypasses the limitations of input data of biological samples. To encounter this challenge, data augmentation is often used to artificially increase the amount of training data[65].

6. Growth prediction of complex structure formation processes

The following experimental results demonstrate the capabilities and limitations of a deep learning algorithm for the prediction of future organoid morphology. Organoids undergo a complex structure formation process when cultivating them in cell culture. In this experiment pancreatic ductal adenocarcinoma (PDAC) cells are seeded and grown in a collagen matrix. Five days after seeding the cells and cultivating them under cell culture conditions, organoids of different structural complexity have eventually grown. Figure 6.1 illustrates samples of microscopy images of PDAC organoids. Even though the duration of cultivation after seeding the PDAC cells is identical, the organoids show highly individual morphologies. These individual structures are based on an underlying structure formation process. The structure formation is dependent on multiple factors. These can be either from biological nature or environmentally driven.

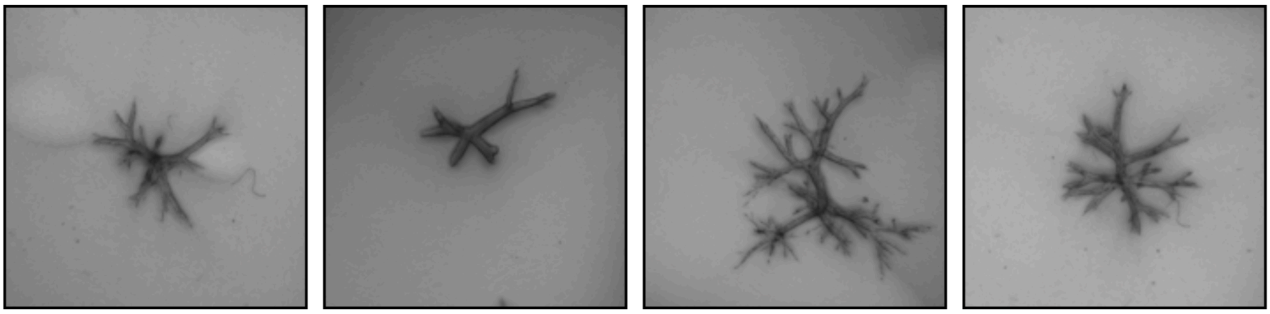


Figure 6.1: Microscopy images of pancreatic ductal adenocarcinoma (PDAC) organoids in collagen. All organoids are imaged 5 days after seeding while developing highly individual structures. The underlying growth process of organoids is non-trivial as it depends on multiple biological and environmental factors.

To get a better understanding of the structure formation process in organoids, a deep neural network, specifically a generative adversarial network for future frame prediction was used. The idea of the network is to learn and understand the structural development of organoids, by analyzing image sequence data of individual organoids. With this understanding, the network is able to perform predictions of the future morphology of an organoid. The neural network was trained and tested with biological image data of real organoids, as well as simulation data of organoids to avoid limitations of training image data.

To investigate the morphology of different organoids it is inevitable to understand the structure formation processes that lead to these individual morphologies. Organoids have shown to grow in multiple developmental phases[58]. These include an initiation phase at an early developmental stage, an extension phase, where the growth of an organoid is dominated by the longitudinal extension of its branches, as well as a thickening phase of the organoid and a lumen-formation phase at a late developmental stage of the organoid. In this experiment, simulation data is used to test the performance of the prediction network in scenarios with different growth phases. The fact that organoids are not synchronously following these growth patterns, increases the challenge of investigating the growth processes.

6.1. Prediction of structure formation

A generative adversarial network was used to predict the future morphological structure of individual organoids. The network was trained on a dataset including time-series image data of multiple individual organoids. The experiments demonstrate the prediction results of four different training datasets. The first set of image data includes simulation data of organoids where the growth of organoids (size of the organoid, including elongation, thickening, branching) increases exponentially. The second one includes simulation data, where the growth of the organoids includes instead of one, but three growth phases, thus containing a more complex growth pattern. The third dataset includes biological data of MDCK cells in a 2D layer. In the fourth dataset, biological image data of PDAC organoids (3D) are included. The datasets also differ in their total size of image data. The use of these different datasets allows an investigation of the capabilities and limitations of the network for future predictions.

6.1.1. Organoid simulation dataset - exponential growth

The neural network was trained on image sequence data of simulated organoids. The total dataset contains image sequences of 1000 simulated organoids. Each image sequence contains 70 frames. Thus resulting in a total dataset of 70000 frames. The network automatically splits the dataset into 80% training data and 20% validation data. The test set includes 90 instead of 70 frames to allow a more detailed assessment of far-future predictions.

After training the dataset, the network learns patterns of the structure formation of multiple individual organoids. These patterns can then be used to predict a given structure based on the information of the network. Figure 6.2 shows the growth of a simulated organoid in comparison to future frame predictions of the deep neural network. The test image sequence containing 90 frames was split into 14 time points for better

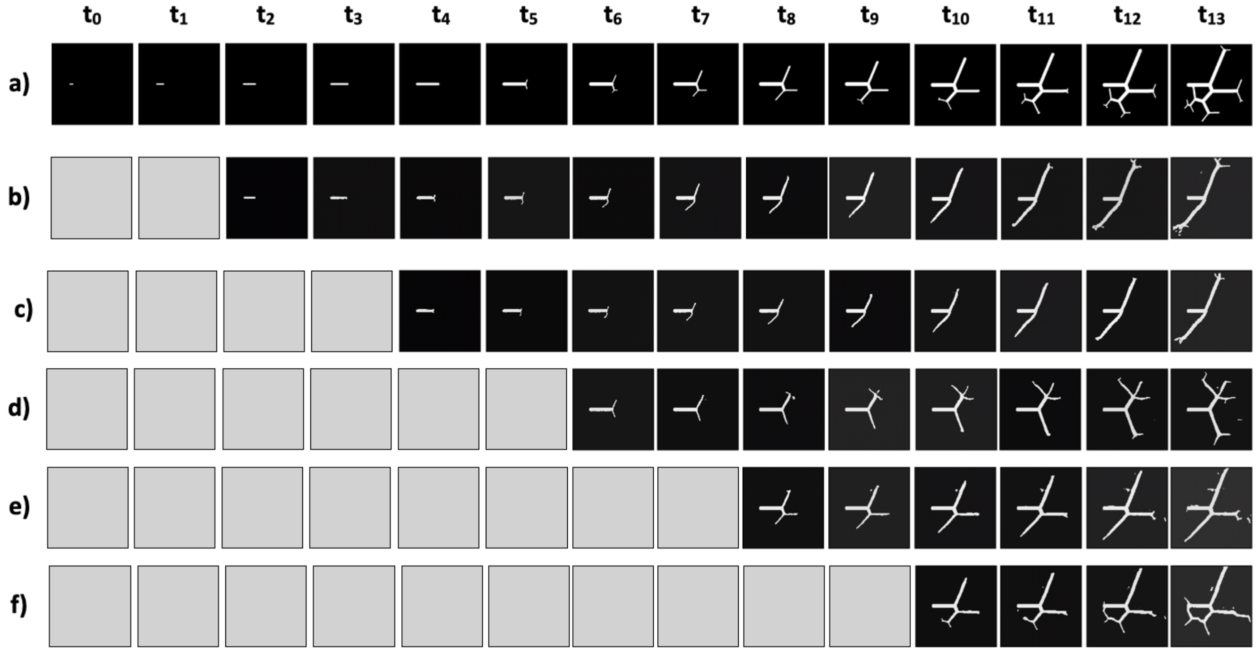


Fig. 6.2: Comparison of simulation data and long-term predictions at different time points. a) Simulation image sequence of the structure formation on an organoid. b) Predicted structure formation starting from time point t_1. Due to the minimal structural information at time point t_1, the final predicted structure highly differs from the final simulated structure. c) Predicted structure formation starting from time point t_3. d) Predicted structure formation starting from time point t_5. e) Predicted structure formation starting from time point t_7. f) Predicted structure formation starting from time point t_9.

visualization. The difference between each time point includes 6 frames in the actual dataset. The prediction at each time point therefore corresponds to a future prediction of 6 frames. The predictions start at different time points in respect to the ground truth data. Dependent on the time point from which the network performs a prediction, the structure of the organoid is affected. While predictions starting at early time points are based on low structural initial information, predictions at later time points are based on more individually developed structures. The initial image information affects the precision of far-future predictions at late time points as shown in figure 6.2. b)-f).

On the other hand, in many experimental cases, these long-term predictions are not necessary. In the case of drug treatments for example, the short-term effects of the drug on the structure of cells or organoids are often more important. Figure 6.3 compares simulation data to its corresponding short-term structure predictions for every time point. Short-term in this regard refers to the prediction of a single time point or 6 frames in the dataset. The predicted images demonstrate a high structural similarity between actual data and predicted data.

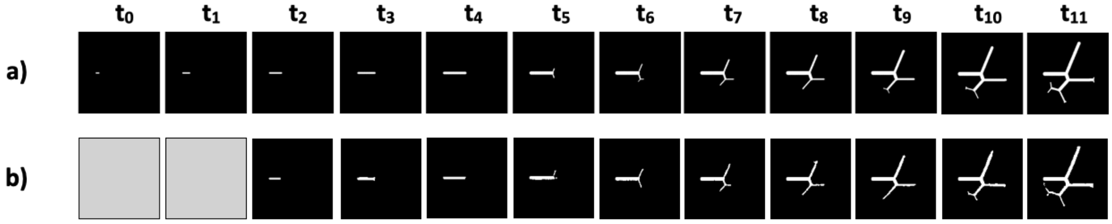


Fig. 6.3: Comparison of simulation data and short term predictions. a) Simulation image sequence of the structure formation on an organoid. b) Predicted structure formation starting from every previous time point in the ground truth data. The predictions show high structural correlation in comparison to the actual structure.

To further assess the structure formation of the simulation data, the overall growth of the organoids (increase of pixels between each time point) is analyzed. Figure 6.4 illustrates the growth curve of a simulated organoid in comparison to short-term as well as long-term predictions. The plot shows an exponential pixel growth of the simulation data. Short-term prediction results are highly aligned with this process. Results of long-term prediction show that even if the overall trend is not accurately followed, an exponential increase of the structures is predicted.

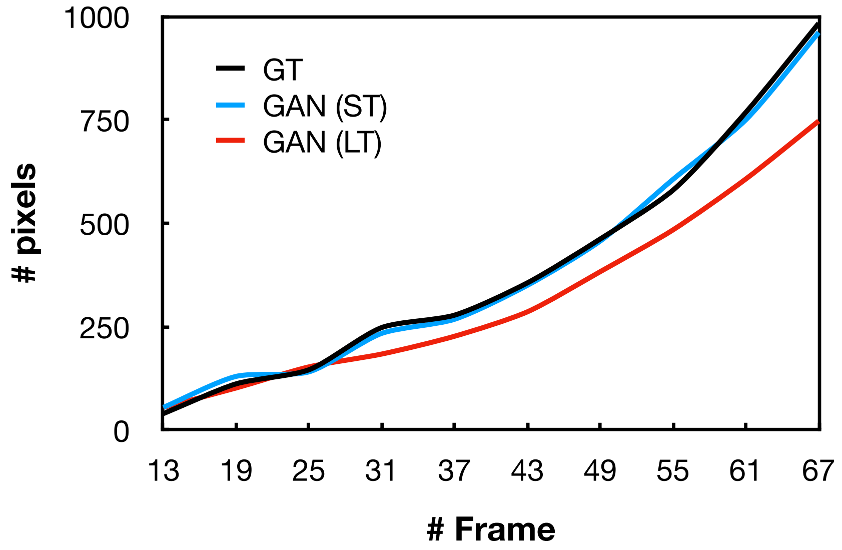


Fig. 6.4: Growth curves of ground truth and prediction data. The GAN network shows high precision of structural prediction for short term future predictions. Long term future prediction shows a similar trend of the growth curve.

It is important to highlight, that even if the structure prediction starting at early time points (figure 6.2. b) is not equivalent to the actual structure of the organoid at a late time point, the underlying parameters of structure formation are still preserved. These include any kind of structural information, such as the growth rate, the number of branches, the size, or the elongation of the simulated organoid.

6.1.2. Organoid simulation dataset - growth phases

In this experiment, the neural network was trained on simulation data of organoid structures. In comparison to the previous experiment, the growth of the structures is not exponential but divided into three different growth phases. The dataset thus contains more complexity in the formation of its structure. Growth in that sense includes longitudinal growth of branches (elongation), transversal growth of branches (thickening), as well as the development of new branches.

Figure 6.5 shows the prediction results of the neural network in comparison to the simulation dataset. The simulation dataset shows a structure formation process including the three growth phases.

The growth in the first phase is exponential, covered between time points t_0 - t_4. It corresponds to an increased rate of branching events as well as an increased speed of branch elongations.

The second phase shows linear growth of the simulated organoids between t_5 - t_9. Branching events are very low, the speed of branch elongation is slow.

The third phase covers time points t_{10} - t_{13}. This phase shows a thickening of branches with a decreased rate of branching events and a low speed of branch elongation. It results in an inverse exponential growth rate.

Predictions were realized at different time points of the simulation data. Based on the structure of each specific time point the predictions are affected. The more structural

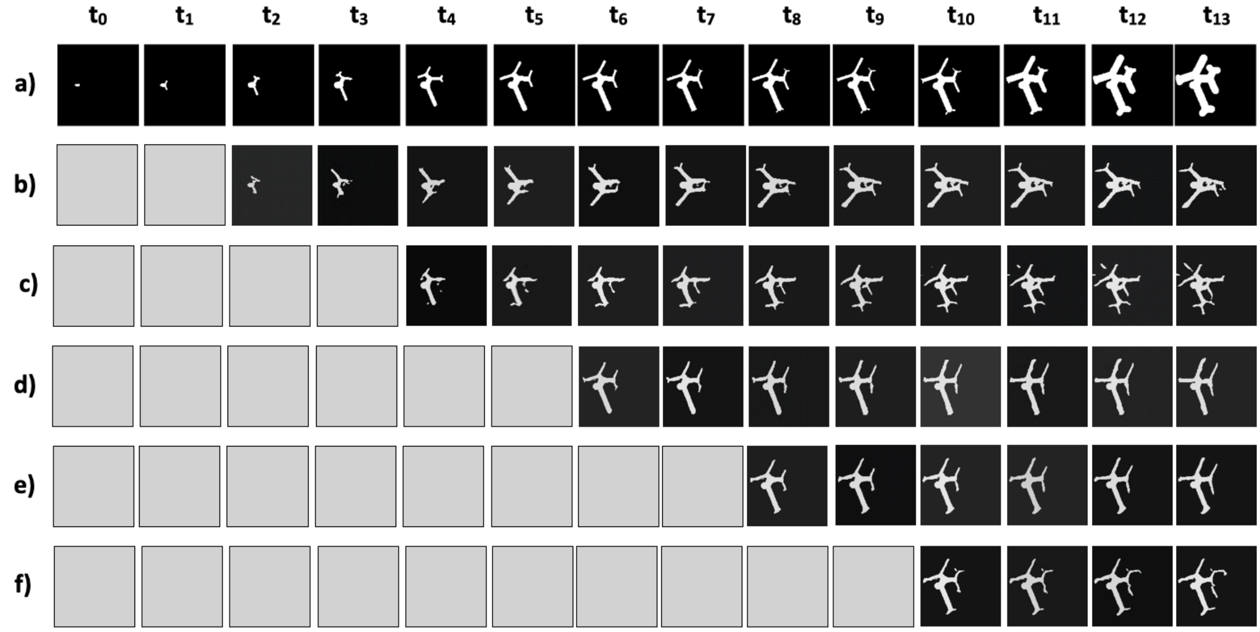


Fig. 6.5: Comparison of simulation data and long-term predictions at different time points. a) Simulation image sequence of the structure formation on an organoid. The image sequence represents three growth phases of the simulated organoid structure. b) Predicted structure formation starting from time point t_1. Due to the minimal structural information at time point t_1, the final predicted structure highly differs from the final simulated structure. c) Predicted structure formation starting from time point t_3. d) Predicted structure formation starting from time point t_5. e) Predicted structure formation starting from time point t_7. f) Predicted structure formation starting from time point t_9.

information given at the specific time point, the more similar is the predicted structure compared to the simulated structure. For long-term predictions, the final structures at time point t_{13} highly differ from the simulated structures, as shown in figure 6.5 b). It furthermore shows that the third phase, which mainly represents the thickening of branches is poorly predicted. It is important to mention that only the test dataset contains the full structural information captured in the shown 14 time points or 90 frames. The training dataset only represents 12 time points. A decreased structural similarity of the prediction compared to the ground truth is therefore expected.

For short-term predictions, on the other hand, predictions are preserved with a high level of structural similarity. Prediction results in comparison to ground truth data are shown in figure 6.6.

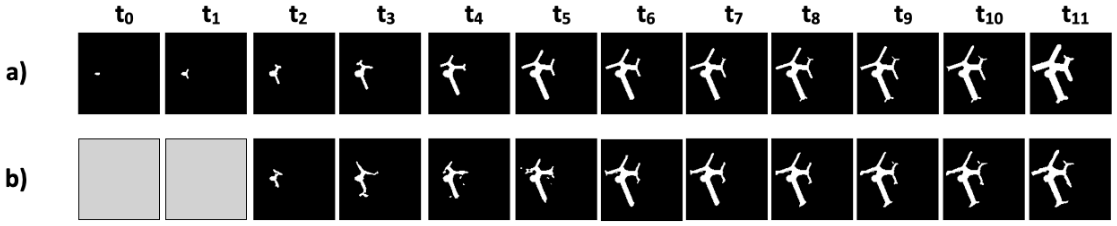


Fig. 6.6: Comparison of simulation data and short term predictions containing information of multiple growth phases. a) Simulation image sequence of the structure formation on an organoid with three growth phases between t_0 - t_4, t_5 - t_9 and t_{10} - t_{13} b) Predicted structure formation starting from every previous time point in the ground truth data. The predictions show high structural correlation in comparison to the actual structure.

In the case of short-term predictions, the structural accuracy of the third growth phase is predicted with low accuracy, where branch thickening should be dominant. Figure 6.7 shows the growth curve of the simulation data in comparison to short-term and long-term predictions of the GAN network.

The plot illustrates a representation of the three growth phases in the ground truth data. Exponential growth is shown between frames 1-30, linear growth between frames 31-60, inverse exponential growth starts beginning from frame 61.

In the case of short-term predictions, the plot also shows a representation of the growth phases in the predictions. The trend however is not accurately following the growth curve of the ground truth data. This effect occurs due to the additional complexity in the training

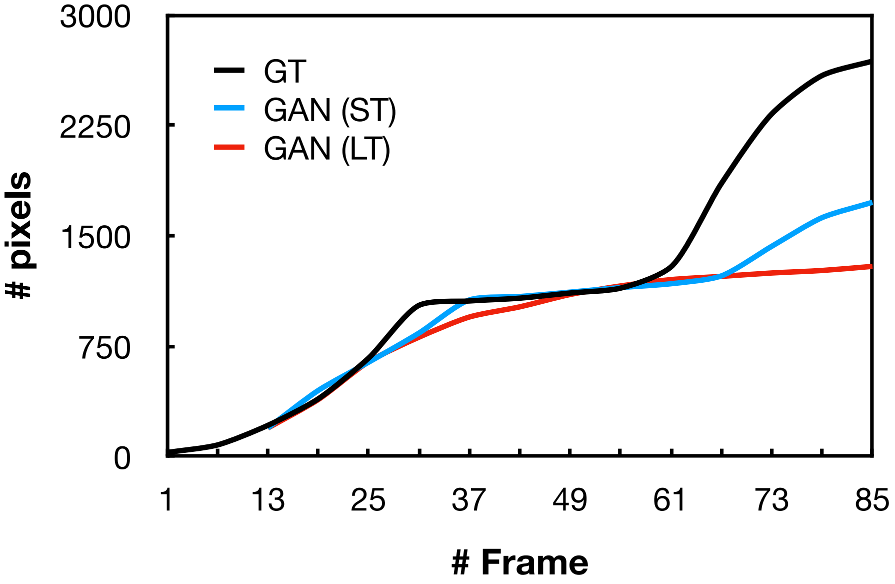


Fig. 6.7: Growth curves of ground truth and prediction data based on three growth phases. The GAN network shows high precision of structural prediction for short term future predictions in growth phases 1 and 2. Inverse exponential growth in phase 3 is maintained. Long term future prediction results maintain the overall trend in growth phase 1 and 2. The inverse exponential growth in the third phase was not maintained.

dataset due to the implementation of growth phases. Furthermore, the lack of an equal amount of training data in the third growth phase (branch thickening) results in a decrease of the accuracy of growth prediction. The trend still illustrates an inverse exponential growth in this growth phase.

The trend for long term growth prediction only follows the actual trend in first and second developmental stage. The inverse exponential trend of the third growth phase on the other hand was not predicted.

It is important to highlight that in the case of long-term predictions, the model tries to fit the ground truth data. By increasing the amount of training data, the model thus is able to provide more accurate results in prediction of the growth.

This equivalently translates to any parameter that defines the simulated organoid structures, for example the number of branches or the position of a new branching event.

6.1.3. PDAC cell dataset (2D)

This experiment focuses on structure formation processes of real biological data. The dataset contains segmentation masks of live-cell image data of MDCK cells. Due to the minimal amount of biological training data in comparison to the simulation dataset, only short term predictions are considered. Prediction results are illustrated in figure 6.8. After training the GAN network, an input sequence of 6 test frames is provided to the network. The network predicts 6 subsequent future frames based on the trained dataset and the short input sequence. The predicted sequence is compared to actual ground truth data of the segmented MDCK cell layer. The predicted sequence contains low contrast images. The structural information however is well-preserved. Predictions show a high level of structural similarity compared to ground truth data. The network is able to understand the structural formation of the growing PDAC cell layer. This results in a realistic prediction of future areas of migrations of the cell layer. The morphological changes between each time point are small in this case of short-term prediction.

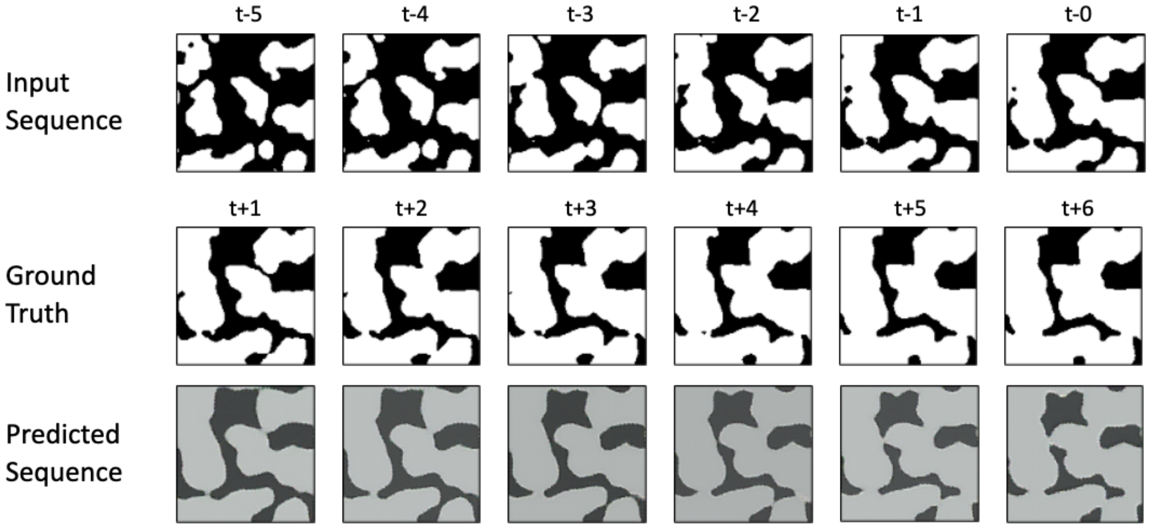


Fig. 6.8: Prediction of a segmented layer of a MDCD cell layer image sequence. The neural network makes future predictions based on the training dataset and the input sequence containing 6 image masks. The predicted image sequence is compared to ground truth data at for 6 subsequent frames.

Figure 6.9 demonstrates a more detailed illustration of a predicted image containing erroneous areas of predictions. The prediction is compared to the corresponding brightfield microscopy image of a MDCK cell layer, as well as a segmentation mask of the cell layer that acts as ground truth. The prediction shows errors in regions where cell clusters are close to each other. Regions at the border of the field of view show false positive predictions. In the biological image dataset, the growth of the cell layer at certain positions at the image border into the field of view can actually occur. This effect makes the erroneous predictions in the regions at the border of the field of view more reasonable.

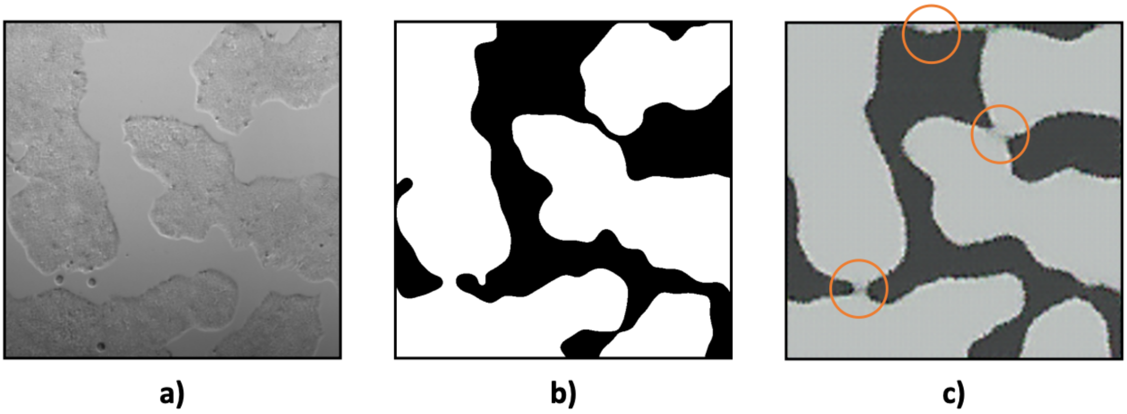


Fig. 6.9: Comparison between brightfield image, ground truth and predicted image. a) Brightfield microscopy image of the MDCK cell layer. b) Segmentation mask which acts as ground truth for the prediction. c) Predicted mask. Marked regions show areas where predictions are erroneous.

6.1.4. PDAC organoid dataset (3D projection)

The last experiment of this thesis focuses on the prediction of actual organoid data. The organoids are cultured in a three-dimensional collagen matrix. Organoids are imaged in 3D. The volumetric information is reduced by using 3D minimum-projections. This results in 2D images of organoids. The prediction network is trained on time-series image sequence data multiple organoids. Due to the culturing time of organoids, the throughput of organoid measurements is limited. The minimal amount of biological training data in comparison to the simulation dataset limits the prediction capabilities of the network. Therefore, only short-term predictions are considered.

Figure 6.10 illustrates the prediction results after training the network. The prediction is based on an input sequence of 6 frames (t_{-5} to t_0). The ground truth data of 6 subsequent frames is compared to the predictions of the neural network (t_{+1} to t_{+6}). The difference between each time point corresponds to one hour of actual organoid growth. In the case of PDAC organoids the predictions show a realistic representation of the actual structure. Structural changes between each frame however are small.

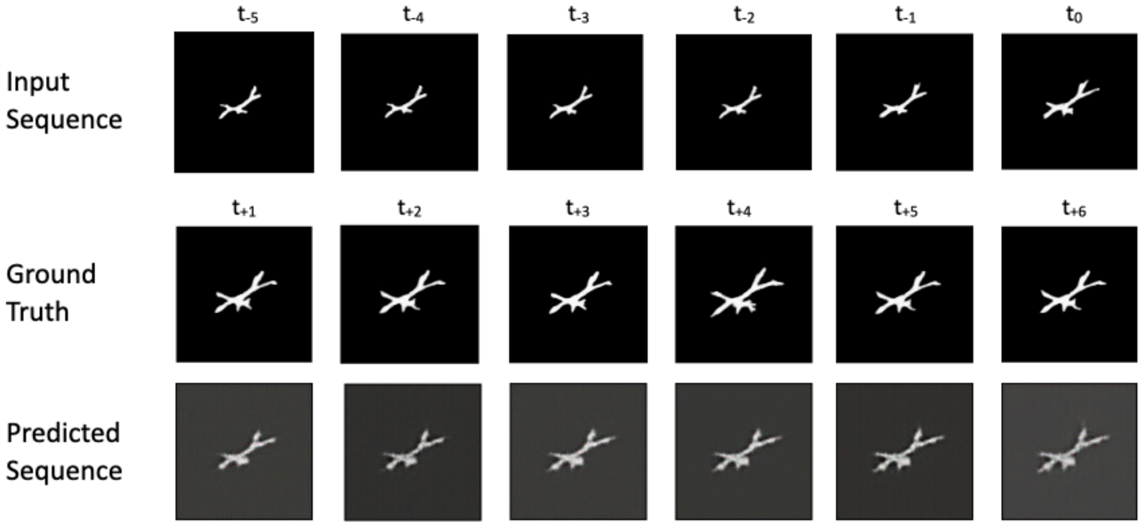


Fig. 6.10: Prediction of annotated masks of a PDAC organoid image sequence. The neural network makes future predictions based on the training dataset and the input sequence containing 6 image masks (t_{-5} to t_0). The predicted image sequence is compared to ground truth data for 6 subsequent frames (t_{+1} to t_{+6}).

6.2. Evaluation of neural network predictions

The image sequence predictions provided by the neural network have demonstrated a high accuracy in predicting the actual structure for short-term predictions. In the case of long-term prediction, parameters which are affected by the underlying structure formation process are maintained. To obtain a more precise understanding of the results, the predicted images were evaluated and compared between the individual results.

Figure 6.11 illustrates the prediction results of a sequence of 6 future images (t_{+1} to t_{+6}) compared to ground truth segmentation masks. The frames show an overlay of predicted masks and manually annotated ground truth masks of a 2D PDAC cell layer. The colors illustrate the accuracy of the prediction, with true positives (white), true negatives (black), false positives (cyan) as well as false negative pixels (magenta). The image sequence shows an increase of false positive and false negative predictions. While the prediction is able to represent a realistic representation of the cell layer, the overall accuracy of obtaining the exact future morphology is not preserved. The distribution of false positive and false negative predictions however seem to be in balance. This correlates with the results shown in chapter 6.1.1 and 6.1.2 that overall parameters learned from the dataset, such as the development of the cell area are contained.

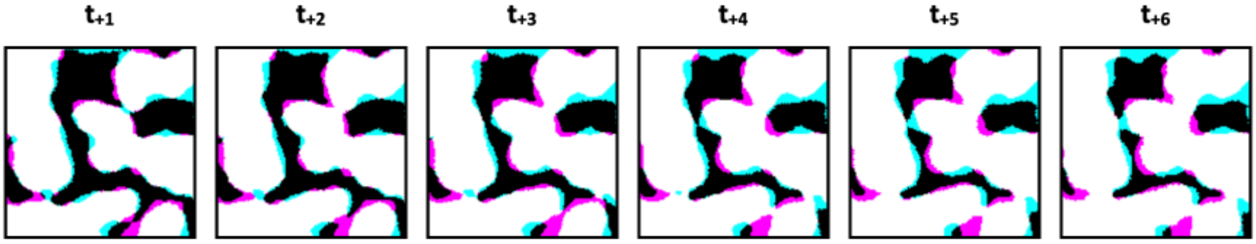


Fig. 6.11: Structural accuracy of a future prediction image sequence. Assessment of the structural accuracy of six predicted frames (t_{+1} to t_{+6}). The images show an overlay of ground truth and predicted data. True positives (white), true negatives (black), false positives (cyan) and false negatives (magenta) are illustrated.

Figure 6.12 illustrates that the overall pixel area is well-preserved over the whole structure formation process of a simulated organoid for short-term predictions. The dataset contains simulated organoids with an exponential growth pattern. The predicted masks demonstrate a similar exponential trend, based on the patterns it learned from the training dataset. The intersection over union lowers at a later developmental stage. This confirms the result, that the prediction does not preserve the actual structure of the organoids and is affected due to an increased motility. Patterns learned from the dataset such as the growth of the organoids are represented in the predictions however. Further evaluation results are shown in the appendix.

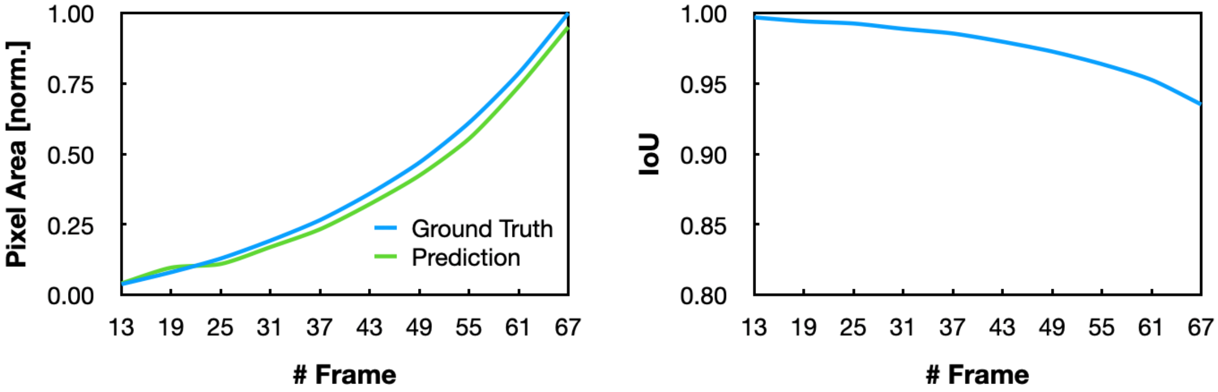


Fig. 6.12: Prediction performance of simulated exponential growth. a) Development of the pixel area of simulated organoids compared between ground truth and predicted data. The overall growth shows an exponential trend. b) Intersection of union of the prediction compared to ground truth data. The accuracy of the network to predict the actual morphology of the simulated organoids increasingly lowers at later developmental stages.

Exponential growth is one of the most simple pattern growth patterns and therefore easily to predict. It is not necessary to us a machine learning or deep learning approach to predict such trend. To increase the complexity of the dataset, three growth phases instead of a single one is introduced. The effect of the increase in complexity is shown in figure 6.13. The growth patterns between frames 1-30, 31-60 and 61-90 affect the pixel area of the segmentation masks in the simulation dataset. The accuracy of the growth prediction is lowered compared to a single exponential growth pattern. Especially between frames 61-85, the predicted area is hardly preserved. This is due to the fact that the training dataset contains only 70 frames instead of 90 frames per image sequence. The third growth phase is trained on only one third of the training data compared to the first and second growth phase. The amount of training data therefore highly affects the accuracy of the predictions. This is even more clearly illustrated in the plot showing the intersection over union. Compared to the single exponential growth pattern, this increase in complexity in the structure formation process directly affects the ability of the network to predict the actual structure. Due to short-term predictions of the dataset, the overall predicted growth curve however, provides a good representation of the actual growth based on the image dataset.

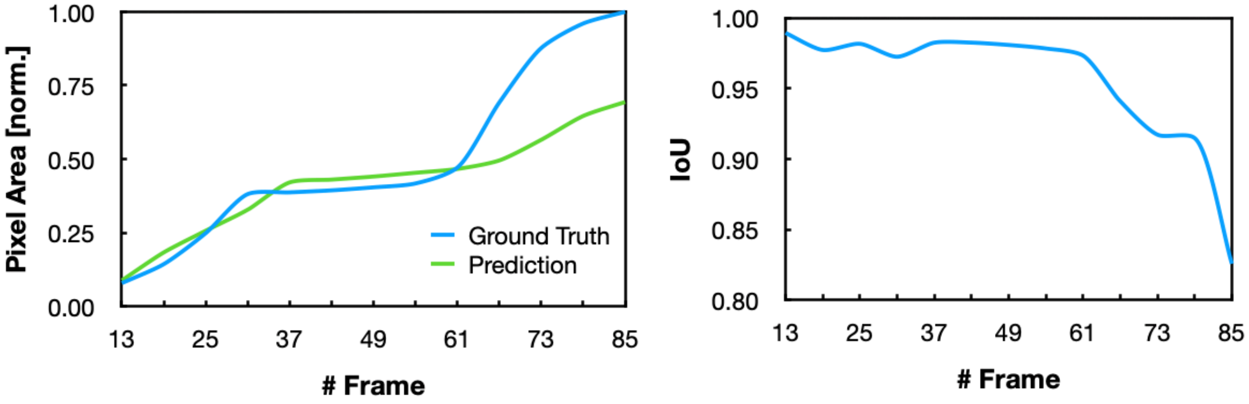


Fig. 6.13: Prediction performance of simulated growth phases. a) Development of the pixel area of simulated organoids compared between ground truth and predicted data. The overall growth shows three different growth phases. b) Intersection of union of the prediction compared to ground truth data. The ability of the network to predict the actual morphology of the simulated organoids is hindered due to the increase in data complexity.

It is important to note that even in the case of long term predictions these learned patterns of the dataset are preserved as shown in the results of this experiment. Further evaluation of simulated growth phases is shown in the appendix.

6.3. Conclusion and outlook

This work and specifically the proposed method has demonstrated the capabilities and limitations of a generative adversarial network for future frame prediction. The predictions show a high level of accuracy for short-term predictions. In the case of long-term predictions, overall structure formation parameters are well obtained. The predictions enable a new approach for understanding complex growth processes in cell biology - specifically in the case of organoids - where the information contained in microscopy images and the throughput of biological samples are limited. The prediction of the simulation data has shown that the network learns to understand the structure formation process and is able to generate a realistic prediction of the future structure based on this information. The predictions of the network can be improved at any time by increasing the size of the dataset and retraining the network.

Based on the results of this work, there are many opportunities to improve the performance and the accuracy of prediction results for future applications. To this end, the prediction of structure formation in organoids is limited to two-dimensional image data. A three-dimensional approach however, would give a more general view of organoid growth.

There are several applications where the use of future frame prediction networks is highly promising. In particular, the detailed analysis of structure formation processes in combination with sophisticated biological analysis techniques of organoids, such as RNA sequencing or gene expression profiling will most likely provide a holistic understanding of why organoids develop these highly individual complex morphologies. For applications like personalized drug treatments on organoids, the use of future frame prediction algorithms is also highly relevant. Due to the individual complex morphology of organoids, the prediction could be used as a digital control - also referred to as digital twin - of the real organoid. Figure 6.14 illustrates the effect of a calyculin treatment on a PDAC organoid. Calyculin A is a marine toxin isolated from disodermia calyx which is an inhibitor of protein phosphatase 1[36]. It can be used as a drug component for treating certain types of cancer. The drug was added to the organoid after time point t_4. Between time point t_0 - t_4 the organoid shows normal growth behavior. After calyculin A was added, the organoid shows a contractile behavior within a short period of time (t_5 - t_9). Together with a segmentation mask of the brightfield microscopy data, the image sequence can be used to train a neural network for future frame prediction. This directly leads to two possible future applications of the method.

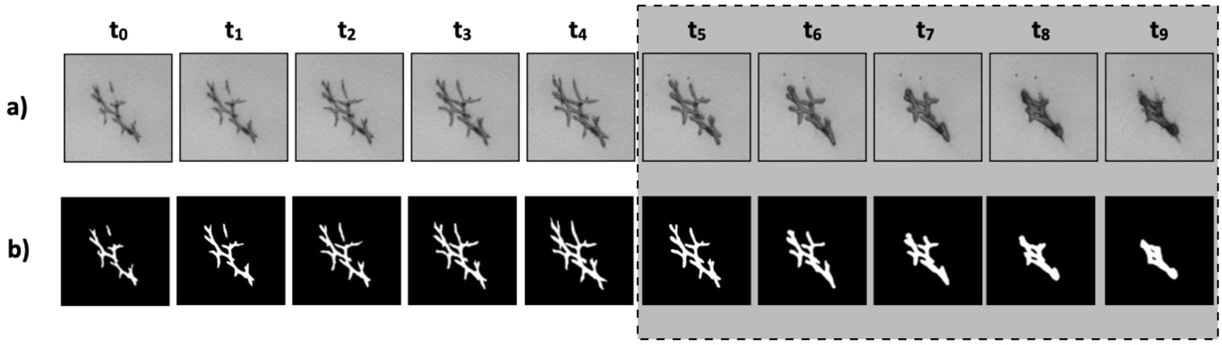


Fig. 6.14: Calyculin A treatment of a PDAC organoid. a) Brightfield microscopy images of the development of a PDAC organoid at 10 time points. The organoid was treated with calyculin after time point t_4. The treatment with calyculin leads to a contractility of the organoid. b) Segmentation masks of the PDAC organoid. Brightfield images and segmentation mask can be used as training input to predict the future structure, based on the information of the morphological change of the organoid due to the drug treatment.

The network can be trained on specific drug treatments. The information can be used to predict an organoid (which was not treated) to obtain the effects of a drug on a normal growing organoid. This can be useful for example in the field of drug development.

Another application is to use the method as a control experiment or “digital twin”. A prediction network trained on normal growing organoid image sequences is considered. In the example of figure 6.14, calyculin was added after time point t_4. The network can be used to make future predictions based on time point t_4. The predicted sequence would show a continuing organoid growth without the effects of the drug treatment. Compared to the actual image data with the treatment, the predictions can be used as a control to the calyculin treatment. This would help to better understand the effects of a drug on the morphology of an organoid. Especially when focusing on different length scales during imaging, such as monitoring the migration of individual cells with a higher magnification objective would provide a detailed insight of the effect of the drug also on a cellular level.

Apart from using a GAN network to perform future frame predictions, other deep learning methods should also be considered to accomplish this task. Especially the performance of recently promising transformer models[76] apparently show high potential at prediction tasks of sequential data.

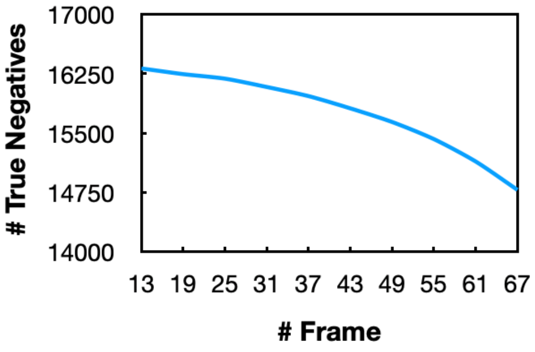
Yet, it has to be noted that the structure formation processes in organoids are far more complex than the assessed simulation examples. A realistic representation not only requires a large dataset of organoid image sequences but also an in-depth analysis of the biological mechanisms that drive the formation processes in organoids.

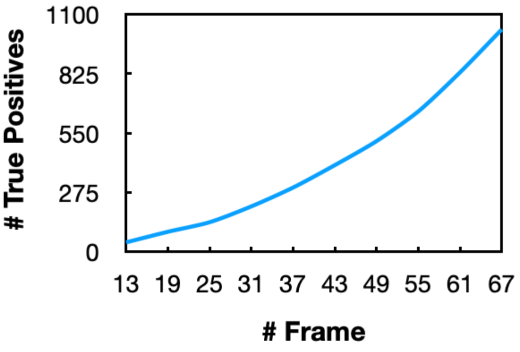
The proposed method, is not limited to the analysis of the overall structure of organoids but can be directly applied to study the growth behavior of organoids on different levels of scale, such as the migration of individual branches or even cells within the organoid. To this end, the assessed results for future frame prediction of organoids to study the fundamental growth behavior and structure formation processes in organoids sets a relevant benchmark for future organoid research.

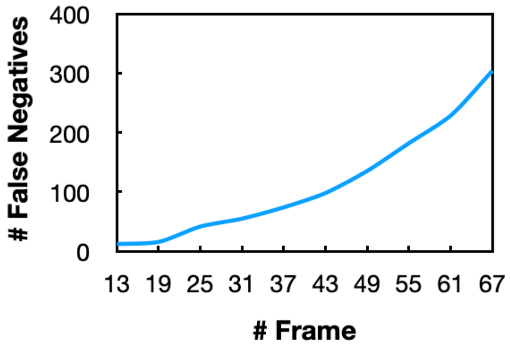
7. Appendix

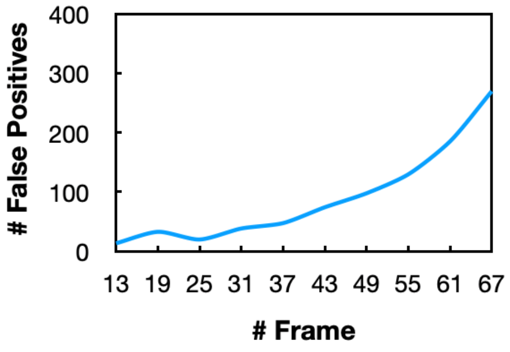
7.1. Evaluation data

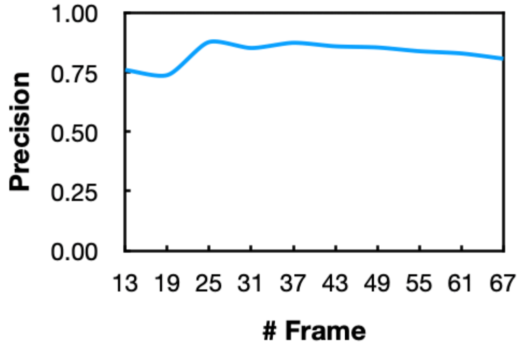
7.1.1. Simulation data - exponential growth

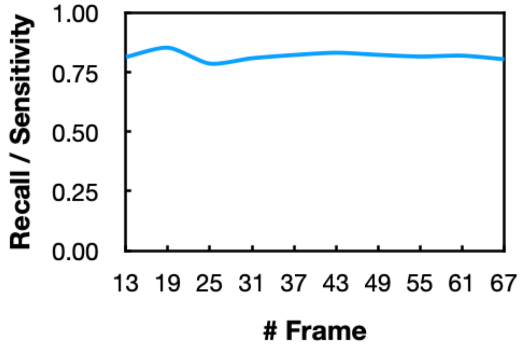


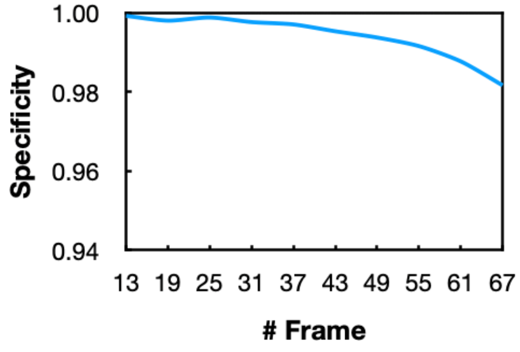


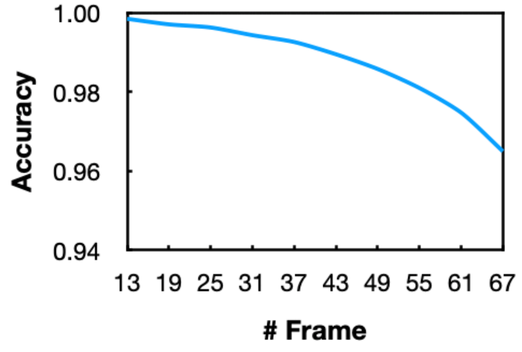












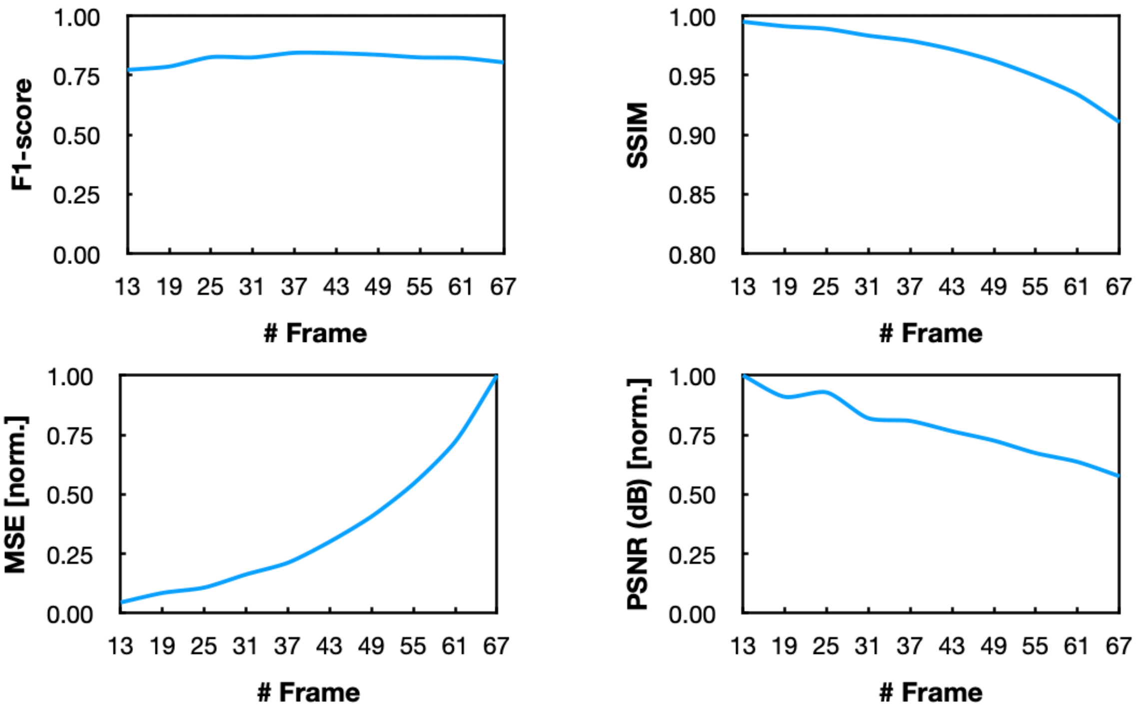
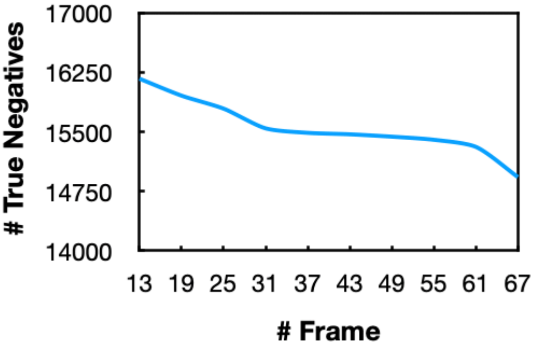


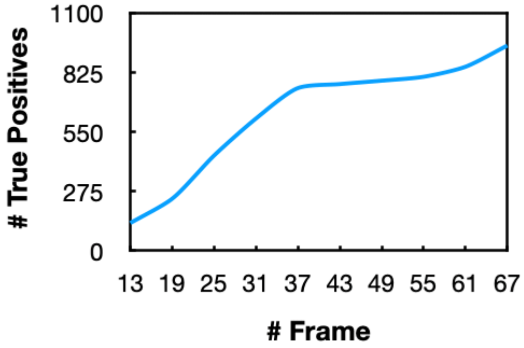
Fig. 7.1: Evaluation of short-term predictions of simulation data (exponential growth).

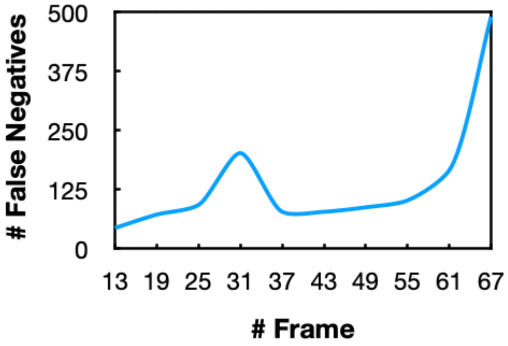
# Frame	13	19	25	31	37	43	49	55	61	67
TN	16316	16244	16187	16081	15966	15811	15639	15423	15140	14780
FP	13	33	20	38	47	74	98	130	185	270
FN	12	16	42	55	74	98	135	182	229	304
TP	43	92	136	210	297	401	512	650	831	1030
Precision	0.76	0.74	0.88	0.85	0.87	0.86	0.85	0.84	0.83	0.81
Recall	0.81	0.85	0.79	0.81	0.82	0.83	0.82	0.82	0.82	0.80
Sensitivity	0.81	0.85	0.79	0.81	0.82	0.83	0.82	0.82	0.82	0.80
Specificity	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.99	0.98
Accuracy	1.00	0.99	0.99	0.99	0.99	0.98	0.97	0.96	0.95	0.94
IoU	1.00	0.99	0.99	0.99	0.99	0.98	0.97	0.96	0.95	0.94
F1-score	0.77	0.79	0.83	0.83	0.84	0.84	0.84	0.82	0.82	0.80
SSIM	1.00	0.99	0.99	0.98	0.98	0.97	0.96	0.95	0.93	0.91
MSE	100	191	243	369	481	683	924	1237	1642	2277
PSNR	29	26	26	23	23	22	21	19	18	16
Area (GT)	51	109	175	261	361	488	638	830	1069	1360
Area (PRED)	55	131	149	230	316	437	575	753	1004	1293

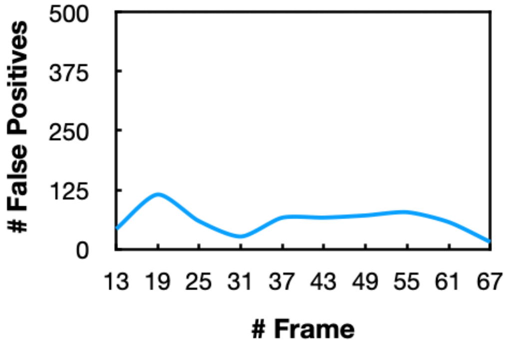
Table 7.1: Evaluation of short-term predictions of simulation data (exponential growth).

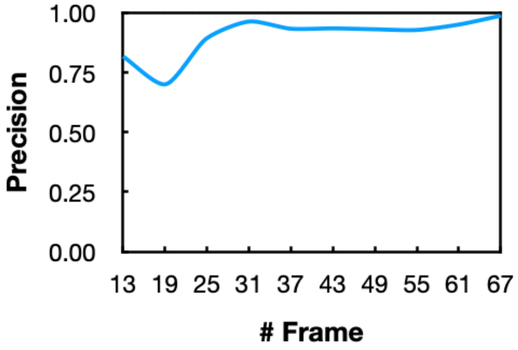
7.1.2. Simulation data - growth phases

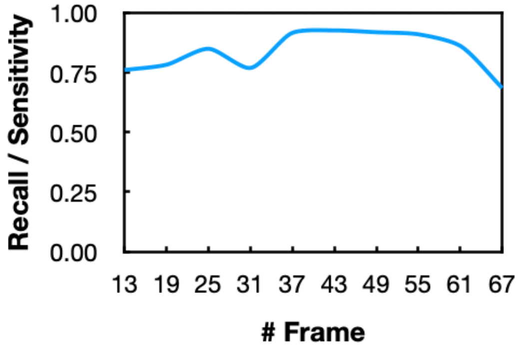


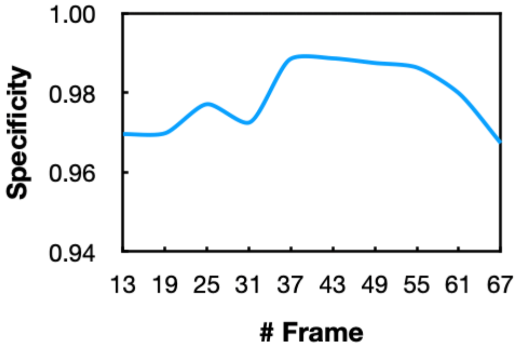


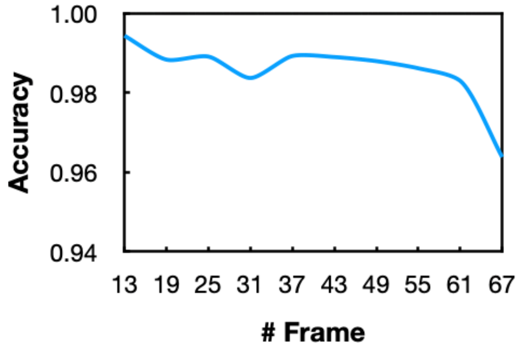












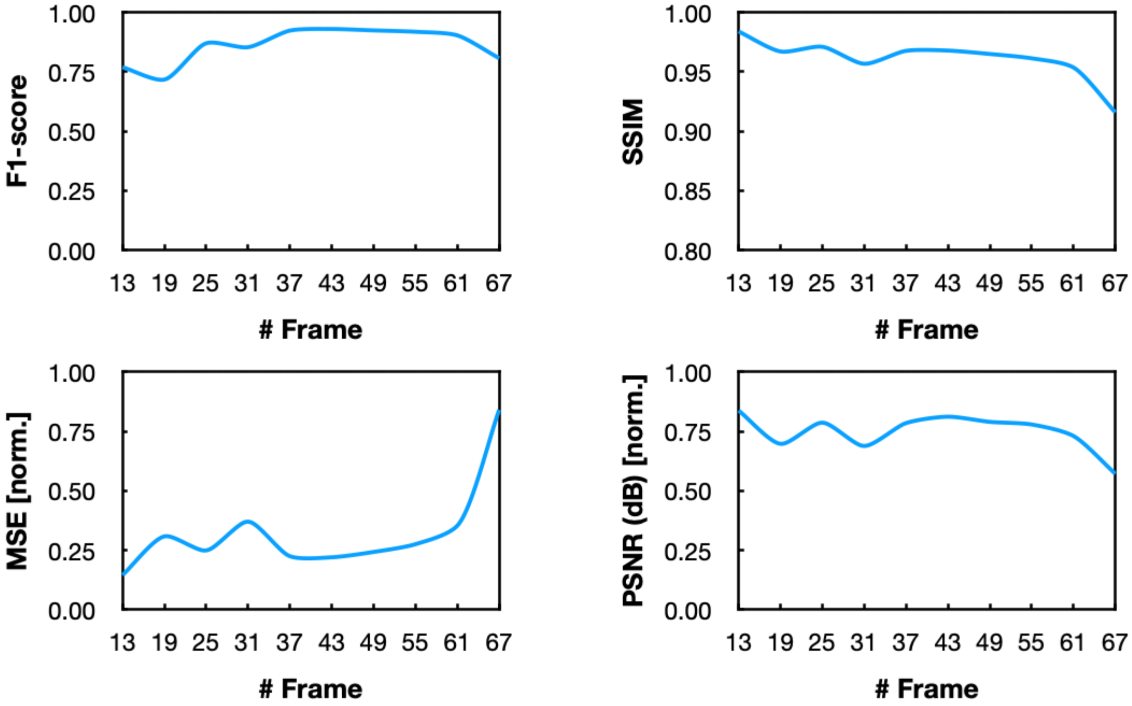


Fig. 7.2: Evaluation of short-term predictions of simulation data (growth phases).

# Frame	13	19	25	31	37	43	49	55	61	67
TN	16169	15955	15791	15543	15487	15468	15438	15399	15309	14929
FP	43	115	59	27	67	67	71	78	57	15
FN	44	73	93	202	78	78	88	102	166	490
TP	128	241	440	612	752	771	786	804	851	949
Precision	0.82	0.70	0.89	0.96	0.93	0.93	0.93	0.93	0.95	0.99
Recall	0.76	0.78	0.85	0.77	0.92	0.93	0.92	0.91	0.86	0.68
Sensitivity	0.76	0.78	0.85	0.77	0.92	0.93	0.92	0.91	0.86	0.68
Specificity	0.97	0.97	0.98	0.97	0.99	0.99	0.99	0.99	0.98	0.97
Accuracy	0.99	0.99	0.99	0.98	0.99	0.99	0.99	0.99	0.98	0.96
IoU	0.99	0.98	0.98	0.97	0.98	0.98	0.98	0.98	0.97	0.94
F1-score	0.77	0.72	0.87	0.85	0.92	0.93	0.92	0.92	0.90	0.81
SSIM	0.98	0.97	0.97	0.96	0.97	0.97	0.96	0.96	0.95	0.92
MSE	329	700	565	841	510	499	550	625	803	1913
PSNR	24	20	22	20	22	23	23	22	21	16
Area (GT)	153	278	483	741	753	766	786	812	917	1347
Area (PRED)	171	356	499	638	819	837	858	882	908	964

Table 7.2: Evaluation of short-term predictions of simulation data (growth phases).

7.2. List of figures

Figure 1.1: Microscopy images of pancreatic ductal adenocarcinoma organoids. All organoids are imaged five days after seeding and culturing the tumor cells, while developing highly individual structures. The underlying growth process of organoids is complex and depends on multiple biological and environmental factors.

Fig. 2.1: Types of machine learning algorithms[2,3]. Traditional classification of machine learning algorithms with a) supervised learning: the model is trained on a labeled dataset. For each output, a correct input is provided. b) unsupervised learning: the model is trained on unlabeled data. It finds patterns within the data by itself. c) reinforcement learning: the model learns to make decisions by interactions with its environment and updates its understanding by an integrated feedback loop.

Fig. 2.2: Schematic of an artificial neural network[2]. The feedforward neural network consists of an input layer, an output layer and two hidden layers (deep neural network). Each layer contains several interconnected neurons. Each neuron in the ANN receives input from other neurons of previous layers. The neurons perform a computation on that input, and produce an output that is passed on to neurons in the next layer.

Fig. 2.3: Representation of an artificial neuron in a neural network[83]. A neuron receives the inputs x_1, x_2 and x_3 which are multiplied by weights w_1, w_2 and w_3. Multiplications of inputs and weights are summed up. An activation function is applied to the weighted sum. A bias unit b is added, which leads to an output y.

Fig. 2.4: Network architecture of a convolutional neural network[2]. The network uses several convolution and pooling layers to extract features from input images. In the fully connected layers (Fc), every input neuron is connected to every output neuron. The output is forwarded to an activation function of the CNN.

Fig. 2.5: Schematic of the convolution process in a convolutional neural network[24]. A filter / kernel is used to scan over an input image with a padding of 1. Each pixel value is multiplied by the values of the kernel and summed for all positions. The convolution results in an extraction of image features.

Fig. 2.6: Schematic of max pooling in a convolutional neural network[30]. Maximum values from the input feature map are collected within a certain field of view. Values are passed to the output feature map. The most important information of the input feature map is contained in the output feature map. Pooling reduces the spacial dimension of the output layer in a CNN.

Fig. 2.7: Fitting of data when using machine learning models[2]. The separation line represents how well the trained model fits the dataset. An underfitted model does not capture essential details of

the dataset. An overfitted model represents the dataset to a very high level of detail. Overall trends are not captured anymore. In the balanced model, the fit shows an ideal representation of the dataset.

Fig. 2.8: Comparison of loss curves in machine learning[4,72]. a) The learning rate at which the model updates information, affects the loss of model. A low training loss can be obtained by an adequate selection of the learning rate. b) Validation and training loss curve for a dataset trained with a neural network. The model represents the dataset where the validation is minimal. Overfitting occurs when the validation loss increases. The model starts to represent noise in the dataset. At early training cycles the model is underfitting the training data. The model is unable to relate input data to output data.

Fig. 2.9: Schematic of a recurrent neural network[69]. Information flow from neurons in the input layer via neurons in the hidden layer to neurons in the output layer. Neurons in the hidden layer are able to interact with the environment. Due to this feedback loop, future decision can be affected.

Fig. 2.10: Schematic of a generative adversarial network[27]. The model consists of two networks. A generator network and a discriminator network. The generator network generates an image from random noise. The discriminator tries to distinguish the generated image from a real input image. The discriminator sends a feedback to the generator based on its decision. Due to the feedback loop, the generator is able to generate more realistic images. At some point the discriminator is not able to distinguish generated images from real images.

Fig. 2.11: Cell migration on a 2D substrate[60]. a) An unperturbed and confluent monolayer of (epithelial) cells. b) A mechanical scratch creates a gap and free space between cells. The edge cells may be injured (orange asterisks). c) The cell sheets move inwards to close the gap. Red arrows indicate displacement vectors for individual cells at one time point. d) Higher magnification of the area in c). A front cell automatically has polarity, a front (red) surface adjoining the free space and cell layer. e) Side view of cells moving inwards to fill the gap, with free surface extension of the front cell in red.

Fig. 2.12: Mesenchymal and amoeboid cell migration[84]. The mesenchymal cell aligns along the ECM fibers while at the same time creating a tunnel of proteolytically digested ECM. The amoeboid cell does not align along the ECM fibers but instead propagates by protruding through pores in the ECM.

Fig. 2.13: Developmental phases of PDAC organoids[58]. Development phases are denoted by color bars which follow the color code - blue: onset phase, orange: extension phase, green: thickening phase, pink: lumen formation phase. All organoids are grown in collagen. Cellular motion patterns are observed with live confocal imaging for each development phase (*n* = 66 organoids). Cell nuclei are stained with SiRDNA (white). Scale bars: 100 µm. From left to right: Day 4 sum projection, and Day 7, Day 10, Day 13 maximum projections.

Fig. 3.1: 2D and 3D image acquisition. a) Example of a 2D image acquisition pipeline. For two-dimensional cell growth, PDAC cells are seeded in a 2 well plate. The wells are imaged by live-cell

microscopy. The stitched microscopy image contains multiple subsample of image sequences. b) Example of a 3D image acquisition pipeline. PDAC organoids are grown in a 24-well plate. The positions of each organoid is registered in the microscopy software. Each measurement contains an organoid within a cubic imaging volume. Additionally time-series data of each organoid is acquired.

Fig. 3.2: Sample images of time-series datasets. a) MDCK cell nuclei dataset. The dataset contains microscopy data of fluorescent labeled MDCK cell nuclei. b) PDAC cell dataset (2D). The dataset contains brightfield microscopy images of a PDAC cell layer. c) PDAC organoid dataset (3D). The dataset contains brightfield images of multiple individual PDAC organoids. d) Organoid simulation dataset. The dataset contains simulation data of the structure formation of organoids. Corresponding segmentation masks are provided for all images in the microscopy image datasets. The image datasets differ in the amount of total images. Scale bars are not contained in the image dataset. Scale bars: a) 120 µm, b) 350 µm, c) 500 µm.

Fig. 3.3: Time series data of Madin-Darby Canine Kidney (MDCK) cells. The data contains microscopy images of fluorescently labeled nuclei of MDCK cell growth. The dataset covers the growth process and movement of cells up to confluence of the cell layer.

Fig. 3.4: Time series data of pancreatic ductal adenocarcinoma (PDAC) cells. The data contains brightfield microscopy images of PDAC cells.

Fig. 3.5: Time series data of pancreatic ductal adenocarcinoma (PDAC) organoids. The data contains brightfield microscopy images of PDAC organoid growth.

Fig. 3.6: Growth stages of simulated organoids. Four time points of the development stage of a simulated organoid image dataset. The image sequence of each simulated organoid contains a total of 70 time points.

Fig. 3.7: Comparison of simulation test datasets. Image sequence data of simulated organoids is shown for 14 time points. Between every time point 6 frames are acquired. The test set contains a total of 90 images. a) Simulation data of exponential organoid growth. b) Simulation data of organoid growth with 3 different growth phases. Phase 1 shows exponential growth. Phase 2 shows linear growth at a small rate of structural change. Phase 3 represents an inverse exponential growth pattern.

Fig. 3.8: Process of automated and manual nuclei annotation for segmentation with a neural network. a) Automated data annotation considers image pre-processing, binary thresholding, watershed-segmentation, filtering and post-processing to provide a training dataset for nuclei segmentation with a neural network. b) Manual data annotation is applied on raw image data (after pre-processing) in order to provide a training dataset of images and annotation masks.

Fig. 3.9: Architecture of the *Inception v3* network for image classification. The network consists in total of 42 layers. It is composed of a combination of convolutional layers, pooling layers together with 1x1 convolutional layers. To reduce the input dimensionality, 1x1 convolutional layers are

used. This drastically lowers the amount of parameters in the network. The final layer of the network applies a softmax activation function that is used in order to make a classification prediction based on the input image.

Fig. 3.10: Architecture of the *U-Net* network for image segmentation. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Fig. 3.11: Working principle of the *StarDist* segmentation network for cell nuclei. a) Microscopy image of fluorescent labeled nuclei. The density of the cell nuclei promotes segmentation errors with traditional object segmentation algorithms. Identifying touching nuclei as separate objects requires a more sophisticated segmentation approach. (b) The *StarDist* method is able to fit star-convex polygons by the radial distances $r^k_{i,j}$ into objects to identify individual nuclei. c) Additional object probability maps $d_{i,j}$ are predicted by the network. The combination with the star-convex polygons allow separation of cell nuclei with high precision.

Fig. 3.12: Architecture of the *FutureGAN* generator for training a dataset. The initial step of model uses a set of 4 × 4 px resolution frames as wells as output frames of the same resolution. Layers are added progressively added during training to increase the resolution after a certain number of iterations. The image resolution of the input frames always matches the resolution of the current state of the network. In this example the figure demonstrates the growth progress of the generator for the MovingMNIST dataset with a final resolution of 64 × 64 px. The full generator network furthermore includes 8 × 8 px as well as 32 × 32 px resolution steps.

Fig. 4.1: Manual data labelling of MDCK cell nuclei. The annotated nuclei act as ground truth data for automated segmentation of cell nuclei. The fluorophor shows variations of signal intensity between nuclei.

Fig. 4.2: Comparison of segmentation performance. a) Ground truth annotation data of cell nuclei. b) Segmentation of cell nuclei based on a neural network trained on automatically labeled data. c) Segmentation of cell nuclei based on a neural network trained on manually labeled data.

Fig. 4.3: F1 score of a segmentation network trained on a manually and automatically labeled dataset. Automated and manual annotation show equal segmentation performance after training a neural network.

Fig. 4.4: Nuclei tracking results based on an automatically annotated dataset. a) Direction of migration of fluorescently labeled nuclei visualized by track-lines. The color of the tracks indicates the speed of each individual nucleus/cell (from blue: Low speed, to red: High speed). b) Distribution of the migration direction of cells within 8 hours 20 minutes. Predominant movement directions are obtained between 45°-75° and 315°-345° from its starting position.

Fig. 4.5: Large scale nuclei segmentation of a widefield microscopy image trained on an automatically annotated dataset. The image has a field of view corresponding to 4.2 mm x 3 mm and containing more than 60000 single nuclei. Segmentation results are obtained within less than one hour, including (automatic) annotation and training with a neural network.

Fig. 5.1: Variations of structural complexity of pancreatic ductal adenocarcinoma organoids. a) Organoid with low morphological complexity. The microscopy image shows high contrast between the organoid structure and the background. b) Organoid with high morphological complexity. The organoid hast multiple very thin branches. Manual segmentation of the organoid with high precision takes several minutes and is complicated in regions where branches are thin and the contrast compared to the background is low.

Fig. 5.2: Manual annotation of various microscopy images of organoids. a) Brightfield microscopy images of PDAC organoids. Microscopy images show differences in brightness and contrast of the images. b) Manual annotation masks of brightfield microscopy images. The segmentation masks show different levels of annotation detail and complexity of organoid morphology.

Fig. 5.3: Comparison of segmentation methods for microscopy images of organoids. a) Brightfield image of a PDAC organoid. b) Segmentation by thresholding. c) Segmentation with a semi-automated machine learning algorithm (human-in-the-loop machine learning). d) Segmentation with a deep neural network trained on organoid masks. e) Manual annotation mask (ground truth). Segmentation algorithms show different levels of accuracy from low segmentation performance (thresholding), to high segmentation accuracy (deep neural network).

Fig. 5.4: Automated organoid segmentation of microscopy image with a brightness gradient. Even in cases where the illumination of the microscopy image is inhomogeneous the neural network demonstrates segmentation masks with high precision.

Fig. 5.5: Organoid segmentation results of microscopy images with a neural network. a) Brightfield microscopy images of PDAC organoids at different developmental stages. b) Segmentation masks of organoids generated with a deep neural network. The network provides segmentation masks with high level of detail. The network is trained on microscopy images and masks (including image augmentation) of PDAC organoids.

Fig. 5.6: Three-dimensional representation of volumetric microscopy images of an organoid. a) Volumetric image data of a PDAC organoid. b) 3D render of a PDAC organoid based in volumetric microscopy image data.

Fig. 5.7: Three-dimensional representation of volumetric microscopy images of an organoid. a) Volumetric image data of a PDAC organoid. b) 3D render of a PDAC organoid based in volumetric microscopy image data.

Figure 6.1: Microscopy images of pancreatic ductal adenocarcinoma (PDAC) organoids in collagen. All organoids are imaged 5 days after seeding while developing highly individual structures. The underlying growth process of organoids is non-trivial as it depends on multiple biological and environmental factors.

Fig. 6.2: Comparison of simulation data and long-term predictions at different time points. a) Simulation image sequence of the structure formation on an organoid. b) Predicted structure formation starting from time point t_1. Due to the minimal structural information at time point t_1, the final predicted structure highly differs from the final simulated structure. c) Predicted structure formation starting from time point t_3. d) Predicted structure formation starting from time point t_5. e) Predicted structure formation starting from time point t_7. f) Predicted structure formation starting from time point t_9.

Fig. 6.3: Comparison of simulation data and short term predictions. a) Simulation image sequence of the structure formation on an organoid. b) Predicted structure formation starting from every previous time point in the ground truth data. The predictions show high structural correlation in comparison to the actual structure.

Fig. 6.4: Growth curves of ground truth and prediction data. The GAN network shows high precision of structural prediction for short term future predictions. Long term future prediction shows a similar trend of the growth curve.

Fig. 6.5: Comparison of simulation data and long-term predictions at different time points. a) Simulation image sequence of the structure formation on an organoid. The image sequence represents three growth phases of the simulated organoid structure. b) Predicted structure formation starting from time point t_1. Due to the minimal structural information at time point t_1, the final predicted structure highly differs from the final simulated structure. c) Predicted structure formation starting from time point t_3. d) Predicted structure formation starting from time point t_5. e) Predicted structure formation starting from time point t_7. f) Predicted structure formation starting from time point t_9.

Fig. 6.6: Comparison of simulation data and short term predictions containing information of multiple growth phases. a) Simulation image sequence of the structure formation on an organoid with three growth phases between t_0 - t_4, t_5 - t_9 and t_{10} - t_{13} b) Predicted structure formation starting from every previous time point in the ground truth data. The predictions show high structural correlation in comparison to the actual structure.

Fig. 6.7: Growth curves of ground truth and prediction data based on three growth phases. The GAN network shows high precision of structural prediction for short term future predictions in growth phases 1 and 2. Inverse exponential growth in phase 3 is maintained. Long term future prediction results maintain the overall trend in growth phase 1 and 2. The inverse exponential growth in the third phase was not maintained.

Fig. 6.8: Prediction of a segmented layer of a MDCD cell layer image sequence. The neural network makes future predictions based on the training dataset and the input sequence containing 6 image masks. The predicted image sequence is compared to ground truth data at for 6 subsequent frames.

Fig. 6.9: Comparison between brightfield image, ground truth and predicted image. a) Brightfield microscopy image of the MDCK cell layer. b) Segmentation mask which acts as ground truth for the prediction. c) Predicted mask. Marked regions show areas where predictions are erroneous.

Fig. 6.10: Prediction of annotated masks of a PDAC organoid image sequence. The neural network makes future predictions based on the training dataset and the input sequence containing 6 image masks (t_{-5} to t_0). The predicted image sequence is compared to ground truth data for 6 subsequent frames (t_{+1} to t_{+6}).

Fig. 6.11: Structural accuracy of a future prediction image sequence. Assessment of the structural accuracy of six predicted frames (t_{+1} to t_{+6}). The images show an overlay of ground truth and predicted data. True positives (white), true negatives (black), false positives (cyan) and false negatives (magenta) are illustrated.

Fig. 6.12: Prediction performance of simulated exponential growth. a) Development of the pixel area of simulated organoids compared between ground truth and predicted data. The overall growth shows an exponential trend. b) Intersection of union of the prediction compared to ground truth data. The accuracy of the network to predict the actual morphology of the simulated organoids increasingly lowers at later developmental stages.

Fig. 6.13: Prediction performance of simulated growth phases. a) Development of the pixel area of simulated organoids compared between ground truth and predicted data. The overall growth shows three different growth phases. b) Intersection of union of the prediction compared to ground truth data. The ability of the network to predict the actual morphology of the simulated organoids is hindered due to the increase in data complexity.

Fig. 6.14: Calyculin A treatment of a PDAC organoid. a) Brightfield microscopy images of the development of a PDAC organoid at 10 time points. The organoid was treated with calyculin after time point t_4. The treatment with calyculin leads to a contractility of the organoid. b) Segmentation masks of the PDAC organoid. Brightfield images and segmentation mask can be used as training input to predict the future structure, based on the information of the morphological change of the organoid due to the drug treatment.

7.3. List of tables

Table 3.1: Elements of a binary confusion matrix. Positive and negative classifications are described by true positives, false negatives, false positives, true negatives.

Table 5.1: Evaluation of different segmentation methods. The performance of three different segmentation methods was analyzed. These include, segmentation by conventional thresholding, segmentation with a human-in-the-loop neural network as well as segmentation with a neural network only. The intersection over union, Precision and Recall of segmented images compared to ground truth annotation data was assessed. The neural network without of a human shows segmentation results with highest accuracy.

7.4. List of acronyms

AI	Artificial Intelligence
ANN	Artificial Neural Network
AUC	Area Under Curve
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CV	Computer Vision
DMEM F-12	Dulbecco’s Modified Eagle’s Medium Mixture F-12
DNN	Deep Neural Network
ECM	Extra-Cellular Matrix
FBS	Fetal Bovine Serum
FN	False Negatives
FNN	Feedforward Neural Network
FP	False Positives
FPR	False Positive Rate
GAN	Generative Adversarial Network
GPU	Graphics Processing Unit
GT	Ground Truth
HITL	Human-In-The-Loop
IoU	Intersection over Union
LIF	Leica Image File Format
LSTM	Long-Short Term Memory
MDCK	Madin-Darby Canine Kidney
ML	Machine Learning
MLP	Multilayer Perceptron
MSE	Mean Squared Error
PDAC	Pancreatic Ductal Adenocarcinoma
Pen-Strep	Penicillin / Streptomycin
PGGAN	Progressive Growing Generative Adversarial Network

PR	Precision Recall
PSNR	Peak Signal-to-Noise Ratio
RAM	Random-Access Memory
ReLU	Rectified Linear Unit
RGB	red, green, blue
RL	Reinforcement Learning
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristics
SSIM	Structural Similarity
TIFF, TIF	Tagged Image File Format
TN	True Negatives
TP	True Positives

8. References

1 Aigner, S., & Körner, M. (2018). Futuregan: Anticipating the future frames of video sequences using spatio-temporal 3d convolutions in progressively growing gans. arXiv preprint arXiv:1810.01325.

2 Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., ... & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of big Data, 8, 1-74.

3 An Introduction to Reinforcement learning & its methods – Towards AI. (n.d.-b). Retrieved March 1, 2023, from https://www.towardsai.blog/an-introduction-to-reinforcement-learning-its-methods/

4 An introduction to reinforcement learning & its methods. (n.d.). Retrieved March 1, 2023, from https://www.towardsai.blog/an-introduction-to-reinforcement-learning-its-methods/

5 Arganda-Carreras, I., Kaynig, V., Rueden, C., Eliceiri, K. W., Schindelin, J., Cardona, A., & Sebastian Seung, H. (2017). Trainable Weka Segmentation: a machine learning tool for microscopy pixel classification. Bioinformatics, 33(15), 2424-2426.

6 Bartfeld, S., & Clevers, H. (2017). Stem cell-derived organoids and their application for medical research and patient treatment. Journal of molecular medicine, 95, 729-738.

7 Ben-Gal, I. (2008). Bayesian networks. Encyclopedia of statistics in quality and reliability, 1.

8 Benzeghiba, M., De Mori, R., Deroo, O., Dupont, S., Erbes, T., Jouvet, D., ... & Wellekens, C. (2007). Automatic speech recognition and speech variability: A review. Speech communication, 49(10-11), 763-786.

9 Beucher, S. (1992). The watershed transformation applied to image segmentation. Scanning Microscopy, 1992(6), 28.

10 Beucher, S., & Lantuejoul, C. (1979). Use of Watersheds in Contour Detection," International Workshop on Image Processing: Real-Time Edge and Motion Detection/Estimation. Rennes.

11 Bradski, G. (2000). The openCV library. Dr. Dobb's Journal: Software Tools for the Professional Programmer, 25(11), 120-123.

12 Brunet, D., Vrscay, E. R., & Wang, Z. (2011). On the mathematical properties of the structural similarity index. IEEE Transactions on Image Processing, 21(4), 1488-1499.

13 Buchmann, B., Engelbrecht, L. K., Fernandez, P., Hutterer, F. P., Raich, M. K., Scheel, C. H., & Bausch, A. R. (2021). Mechanical plasticity of collagen directs branch elongation in human mammary gland organoids. Nature Communications, 12(1), 2759.

14 Caicedo, J. C., Roth, J., Goodman, A., Becker, T., Karhohs, K. W., Broisin, M., ... & Carpenter, A. E. (2019). Evaluation of deep learning strategies for nucleus segmentation in fluorescence images. Cytometry Part A, 95(9), 952-965.

15 Chowdhary, K., & Chowdhary, K. R. (2020). Natural language processing. Fundamentals of artificial intelligence, 603-649.

16 Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016). 3D U-Net: learning dense volumetric segmentation from sparse annotation. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19 (pp. 424-432). Springer International Publishing.

17 Clark, A. (2015). Pillow (pil fork) documentation. readthedocs.

18 Clevers, H. (2016). Modeling development and disease with organoids. Cell, 165(7), 1586-1597.

19 Corrò, C., Novellasdemunt, L., & Li, V. S. (2020). A brief history of organoids. American Journal of Physiology-Cell Physiology, 319(1), C151-C165.

20 Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A. (2018). Generative adversarial networks: An overview. IEEE signal processing magazine, 35(1), 53-65.

21 Cunningham, P., Cord, M., & Delany, S. J. (2008). Supervised learning. Machine learning techniques for multimedia: case studies on organization and retrieval, 21-49.

22 Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.

23 Dong, C., Loy, C. C., He, K., & Tang, X. (2015). Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence, 38(2), 295-307.

24 Dumoulin, V., & Visin, F. (2016). A guide to convolution arithmetic for deep learning. arXiv preprint arXiv:1603.07285.

25 Englbrecht, F., Ruider, I. E., & Bausch, A. R. (2021). Automatic image annotation for fluorescent cell nuclei segmentation. PloS one, 16(4), e0250093.

26 Falk, T., Mai, D., Bensch, R., Çiçek, Ö., Abdulkadir, A., Marrakchi, Y., ... & Ronneberger, O. (2019). U-Net: deep learning for cell counting, detection, and morphometry. Nature methods, 16(1), 67-70.

27 Feng, J., Feng, X., Chen, J., Cao, X., Zhang, X., Jiao, L., & Yu, T. (2020). Generative adversarial networks based on collaborative learning and attention mechanism for hyperspectral image classification. Remote Sensing, 12(7), 1149.

28 Garcia, S., Hannezo, E., Elgeti, J., Joanny, J. F., Silberzan, P., & Gov, N. S. (2015). Physics of active jamming during collective cellular motion in a monolayer. Proceedings of the National Academy of Sciences, 112(50), 15314-15319.

29 Ghahramani, Z. (2004). Unsupervised learning. Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures, 72-112.

30 Gholamalinezhad, H., & Khosravi, H. (2020). Pooling methods in deep neural networks, a review. arXiv preprint arXiv:2009.07485.

31 Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).

32 Han, S. J., Rodriguez, M. L., Al-Rekabi, Z., & Sniadecki, N. J. (2016). Spatial and temporal coordination of traction forces in one-dimensional cell migration. Cell Adhesion & Migration, 10(5), 529-539.

33 Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction (Vol. 2, pp. 1-758). New York: springer.

34 Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., & Blunsom, P. (2015). Teaching machines to read and comprehend. Advances in neural information processing systems, 28.

35 Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in science & engineering, 9(03), 90-95.

36 Ishihara, H., Martin, B. L., Brautigan, D. L., Karaki, H., Ozaki, H., Kato, Y., ... & Hartshorne, D. J. (1989). Calyculin A and okadaic acid: inhibitors of protein phosphatase activity. Biochemical and biophysical research communications, 159(3), 871-877.

37 Johnson, D. H. (2006). Signal-to-noise ratio. Scholarpedia, 1(12), 2088.

38 Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., ... & Hassabis, D. (2021). Highly accurate protein structure prediction with AlphaFold. Nature, 596(7873), 583-589.

39 Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2017). Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196.

40 Kingsford, C., & Salzberg, S. L. (2008). What are decision trees?. Nature biotechnology, 26(9), 1011-1013.

41 Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84-90.

42 Liu, W., Luo, W., Lian, D., & Gao, S. (2018). Future frame prediction for anomaly detection–a new baseline. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 6536-6545).

43 Lukinavičius, G., Blaukopf, C., Pershagen, E., Schena, A., Reymond, L., Derivery, E., ... & Johnsson, K. (2015). SiR–Hoechst is a far-red DNA stain for live-cell nanoscopy. Nature communications, 6(1), 8497.

44 McClelland, J. L., Rumelhart, D. E., & PDP Research Group. (1987). Parallel Distributed Processing, Volume 2: Explorations in the Microstructure of Cognition: Psychological and Biological Models (Vol. 2). MIT press.

45 McKinney, W. (2010, June). Data structures for statistical computing in python. In Proceedings of the 9th Python in Science Conference (Vol. 445, No. 1, pp. 51-56).

46 Medsker, L. R., & Jain, L. C. (2001). Recurrent neural networks. Design and Applications, 5, 64-67.

47 Noble, W. S. (2006). What is a support vector machine?. Nature biotechnology, 24(12), 1565-1567.

48 Noh, H., You, T., Mun, J., & Han, B. (2017). Regularizing deep neural networks by noise: Its interpretation and optimization. Advances in Neural Information Processing Systems, 30.

49 O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458.

50 Oord, A. V. D., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., ... & Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. arXiv preprint arXiv:1609.03499.

51 OpenAI (2023) CHATGPT: Optimizing language models for dialogue, OpenAI. OpenAI. Available at: https://openai.com/blog/chatgpt/ (Accessed: February 6, 2023).

52 Oprea, S., Martinez-Gonzalez, P., Garcia-Garcia, A., Castro-Vargas, J. A., Orts-Escolano, S., Garcia-Rodriguez, J., & Argyros, A. (2020). A review on deep learning techniques for video prediction. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(6), 2806-2826.

53 Otsu, N. (1979). A threshold selection method from gray-level histograms. IEEE transactions on systems, man, and cybernetics, 9(1), 62-66.

54 Pajic-Lijakovic, I., & Milivojevic, M. (2019). Jamming state transition and collective cell migration. Journal of Biological Engineering, 13, 1-12.

55 Petrie, R. J., Koo, H., & Yamada, K. M. (2014). Generation of compartmentalized pressure by a nuclear piston governs cell motility in a 3D matrix. Science, 345(6200), 1062-1065.

56 Rahman, M. A., & Wang, Y. (2016). Optimizing intersection-over-union in deep neural networks for image segmentation. In International symposium on visual computing (pp. 234-244). Springer, Cham.

57 Ramachandran, P., Zoph, B., & Le, Q. V. (2017). Searching for activation functions. arXiv preprint arXiv:1710.05941.

58 Randriamanantsoa, S., Papargyriou, A., Maurer, H. C., Peschke, K., Schuster, M., Zecchin, G., ... & Bausch, A. R. (2022). Spatiotemporal dynamics of self-organized branching in pancreas-derived organoids. Nature communications, 13(1), 5219.

59 Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

60 Rørth, P. (2009). Collective cell migration. Annual review of cell and developmental, 25, 407-429.

61 Rosenblatt, F. (1961). Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Cornell Aeronautical Lab Inc Buffalo NY.

62 Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., ... & Cardona, A. (2012). Fiji: an open-source platform for biological-image analysis. Nature methods, 9(7), 676-682.

63 Schmidt, U., Weigert, M., Broaddus, C., & Myers, G. (2018). 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 (pp. 265-273). Springer International Publishing.

64 Schutgens, F., & Clevers, H. (2020). Human organoids: tools for understanding biology and treating diseases. Annual Review of Pathology: Mechanisms of Disease, 15, 211-234.

65 Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of big data, 6(1), 1-48.

66 Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).

67 Szegedy, C., Toshev, A., & Erhan, D. (2013). Deep neural networks for object detection. Advances in neural information processing systems, 26.

68 Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).

69 Talathi, S. S., & Vartak, A. (2015). Improving performance of recurrent neural network with relu nonlinearity. arXiv preprint arXiv:1511.03771.

70 Thevenaz, P., Ruttimann, U. E., & Unser, M. (1998). A pyramid approach to subpixel registration based on intensity. IEEE transactions on image processing, 7(1), 27-41.

71 Tinevez, J. Y., Perry, N., Schindelin, J., Hoopes, G. M., Reynolds, G. D., Laplantine, E., ... & Eliceiri, K. W. (2017). TrackMate: An open and extensible platform for single-particle tracking. Methods, 115, 80-90.

72 Tokuç, W. (2022, November 06). Underfitting and overfitting in machine learning. Retrieved March 1, 2023, from https://www.baeldung.com/cs/ml-underfitting-overfitting

73 Trepat, X., Chen, Z., & Jacobson, K. (2012). Cell migration. Comprehensive Physiology, 2(4), 2369.

74 Van Der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The NumPy array: a structure for efficient numerical computation. Computing in science & engineering, 13(2), 22-30.

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

76 Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

77 Visa, S., Ramsay, B., Ralescu, A. L., & Van Der Knaap, E. (2011). Confusion matrix-based feature selection. Maics, 710(1), 120-127.

78 Vittayakorn, S., & Hays, J. (2011, September). Quality Assessment for Crowdsourced Object Annotations. In BMVC (pp. 1-11).

79 Wallach, D., & Goffinet, B. (1989). Mean squared error of prediction as a criterion for evaluating and comparing system models. Ecological modelling, 44(3-4), 299-306.

80 Wang, Y., Yao, H., & Zhao, S. (2016). Auto-encoder based dimensionality reduction. Neurocomputing, 184, 232-242.

81 Wiering, M. A., & Van Otterlo, M. (2012). Reinforcement learning. Adaptation, learning, and optimization, 12(3), 729.

82 Wozniak, M. A., Modzelewska, K., Kwong, L., & Keely, P. J. (2004). Focal adhesion regulation of cell behavior. Biochimica et Biophysica Acta (BBA)-Molecular Cell Research, 1692(2-3), 103-119.

83 Yacim, J. A., & Boshoff, D. G. B. (2018). Impact of artificial neural networks training algorithms on accurate prediction of property values. Journal of Real Estate Research, 40(3), 375-418.

84 Yamada, K. M., & Sixt, M. (2019). Mechanisms of 3D cell migration. Nature Reviews molecular cell biology, 20(12), 738-752.

85 Ying, X. (2019, February). An overview of overfitting and its solutions. In Journal of physics: Conference series (Vol. 1168, p. 022022). IOP Publishing.

86 Zhang, W., Li, R., Zeng, T., Sun, Q., Kumar, S., Ye, J., & Ji, S. (2015, August). Deep model based transfer and multi-task learning for biological image analysis. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1475-1484).

87 Zhou, Y., Dong, H., & El Saddik, A. (2020). Deep learning in next-frame prediction: A benchmark review. IEEE Access, 8, 69273-69283.