Heinrich Heine
University
Düsseldorf

Applications of Supervised Deep (Transfer) Learning for Medical Image Classification

Inaugural-Dissertation

zur Erlangung des Doktorgrades
der Mathematisch-Naturwissenschaftlichen Fakultät
der Heinrich-Heine-Universität Düsseldorf

vorgelegt von

Raphael Marvin Kronberg

geboren in

Düsseldorf

Düsseldorf, Dezember 2021

aus dem Institut für Informatik
der Heinrich-Heine-Universität Düsseldorf

Gedruckt mit der Genehmigung der
Mathematisch-Naturwissenschaftlichen Fakultät der
Heinrich-Heine-Universität Düsseldorf

Berichterstatter:

1. Prof. Markus Kollmann

2. Prof. Philipp Lang

Tag der mündlichen Prüfung: 15.03.2022

Dedicated to my family

Abstract

Medical imaging procedures are central components in the diagnosis of fractures and tumors. Deep Learning, a subfield of Machine Learning, has already established itself in radiology, while other fields such as pathology are also increasingly discovering the power of Deep Learning-based image classification for their own workflows. The automated analysis of many medical images saves time, resources, and money. In medicine, Deep Learning has various applications including the diagnosis of diseases, the more rapid development of drugs, and the personalization of treatments.

In the first application of Machine Learning that we include in the field of diagnosis of diseases, we used Deep Transfer Learning to analyze scanned histological hematoxylin and eosin (H&E) stained tissue sections. We showed that our neural network is able to identify and localize pancreatic metastases in healthy lymph nodes. The network can thus be used to assist pathologists with the automated evaluation of numerous tissue sections by having the algorithm pre-filter the data and alert the pathologist to certain sections that require a more detailed investigation.

The second application of Machine Learning presented in this thesis is related to the faster development of drugs via Deep Transfer Learning. Using light images of virus-infected cells, we can automatically classify the effectiveness of drugs against a given virus and evaluate the toxicity of the drugs. This approach, which has only been tested under laboratory conditions to date, allows for the rapid, automated analysis of many different drugs.

The final application of Machine Learning that we cover is in the area of treatment personalization. When brain tumors are suspected, the protocol includes collecting four MRI sequences (T1, T1CE, T2, and FLAIR). Since patients are often claustrophobic or in poor physical condition, not all four sequences can always be acquired at all. Therefore, we calculated the optimal approach for acquiring the MRI sequences with the maximum information gain as measured by the F1 score of the segmentation neural network and we present a proposal for a shortened acquisition sequence for this type of patient.

Our work can be extended in many ways and opens up the possibility of automating time-consuming and cost-intensive processes in clinical routine and basic research in the analysis of medical imaging.

Zusammenfassung

Medizinische Bildgebungsverfahren sind zentrale Bestandteile in der Diagnostik von Frakturen und Tumoren. Deep Learning, ein Teilgebiet von Machine Learning, hat sich daher bereits in der Radiologie etabliert. Weitere Bereiche wie die Pathologie entdecken derzeit auf Deep Learning basierende Bilderklassifizierung für die eigenen Workflows. Die automatisierte Auswertung einer großen Zahl von medizinischen Bilder spart Zeit, Ressourcen und Geld. Deep Learning in der Medizin hat verschiedene Anwendungen, wie die Diagnose von Krankheiten, die schnellere Entwicklung von Medikamenten und die Personalisierung von Behandlungen.

In der ersten Anwendung von Deep Learning, die wir dem Bereich der Krankheitsdiagnose zuordnen, haben wir Deep Transfer Learning eingesetzt, um gescannte histologische, mit Hämatoxylin und Eosin (H&E) gefärbte Gewebeschnitte zu analysieren. Wir haben gezeigt, dass unser neuronales Netzwerk in der Lage ist, Pankreasmetastasen in gesunden Lymphknoten zu identifizieren und zu lokalisieren. Das Netzwerk kann Pathologen bei der automatisierten Auswertung zahlreicher Gewebeschnitte unterstützen, indem der Algorithmus die Daten vorfiltert und den Pathologen auf bestimmte Schnitte hinweist, die er sich genauer ansehen sollte.

Die zweite Anwendung von Deep Learning, die wir in dieser Arbeit vorstellen werden, betrifft die schnellere Entwicklung von Medikamenten mittels Deep Transfer Learning. Anhand von Lichtbildern virusinfizierter Zellen können wir automatisch die Wirksamkeit von Medikamenten gegen das Virus und die Toxizität des betreffenden Medikaments klassifizieren. So wird eine schnelle, automatisierte Analyse vieler verschiedener Medikamente ermöglicht. Dies wurde bisher nur unter Laborbedingungen getestet.

Die letzte Anwendung von Deep Learning, welche wir behandeln werden, betrifft die Personalisierung von Behandlungen. Bei Verdacht auf einen Hirntumor werden nach dem Protokoll die folgenden vier MRT-Sequenzen (T1, T1CE, T2 und FLAIR) aufgenommen. Da die Patienten oft klaustrophobisch oder in schlechter körperlicher Verfassung sind, können nicht alle vier Sequenzen aufgenommen werden. Daher berechnen wir die optimale Reihenfolge für die Erfassung der Sequenzen mit dem maximalen Informationsgewinn (gemessen durch den F1-Score des neuronalen Segmentierungsnetzes). Damit können wir eine verkürzte Aufnahmezeit für die oben beschriebenen Patienten mit maximalem Informationsgehalt vorschlagen.

Unsere Arbeit ist vielseitig erweiterbar und eröffnet die Möglichkeit der Automatisierung von zeit- und kostenintensiven Prozessen im klinischen Alltag und der Grundlagenforschung bei der Analyse von medizinischer Bildgebung.

Acknowledgements

Throughout this research and the writing of my thesis, I was supported by many people who I would like to thank for their support for making this work possible. Thanks to Markus B., who provided me with the layout for the thesis, and a thorough and helpful proofreading of the thesis.

I would like to thank Markus K. and Philipp for giving me the opportunity to work in their groups. They helped me to establish valuable contacts with medical research groups and were always interested in my work and ideas. A big thank you goes to Markus K., my supervisor, from whom I could learn a great deal concerning mathematical understanding and the interpretation of objective functions, representation learning, contrastive learning, and deep learning algorithms in lectures, seminars, and bilateral talks. I am very grateful to Philipp for the discussion about new projects and ideas on the white board in my office. I also would like to thank Michael for being my mentor, and taking me in the operating room with him. My thanks also go to Stefan H., Holger S., Peter A. and Marcel S. for their interesting seminars and lectures on Applied Statistics, Data Science and Machine Learning.

Thanks to my MMBS lab mates Julius, Nikolas, Nima, Rahil, Simon, and Tim, with whom I could share ideas during the WebEx conference (papers-worth-reading) and discuss new interesting papers and the tricks behind them each Wednesday evening. Thanks to you, Dieter, Julia, Melanie and Pawel, my colleagues from the MM2 lab, and our master students for many fruitful discussions during the morning meetings. Thank you, Christian S., Philipp H. R., and Stephan R., for helping out with all the technical and organizational tasks. You were always quick to help when I faced software problems on the HPC or administration questions. Furthermore, I would like to thank my colleagues from the Neurosurgery and Radiology departments, Christian R., Igor, and Dziugas for the exciting joint projects. Moreover, it was a great experience to be part of the Heine Center for Artificial Intelligence and Data Science (HeiCAD) and the Interdisciplinary Graduate and Research Academy Düsseldorf (iGRAD). Special thanks go to the Anton Betz Stiftung of the Rheinische Post for funding my doctoral studies. I am thankful for two more great years with my friends Christian, Sina, Markus, Tobias, and Pascal at the Heinrich Heine University in Düsseldorf. Thank you Ellen for the pleasant cooperation in the iGRAD seminar and for spending my last day at university with me.

Finally, I would like to thank my mother Petra, Mario and my brother Fabian and other friends who have supported me throughout my studies.

Contents

Chapter 1

Introduction and Motivation

Due to constant improvements in computers, processors, and graphics cards, and large amounts of available data, Machine Learning algorithms can take on increasingly complex tasks. From beating the best players in chess (Hsu, 2002), Go (Granter et al., 2017), and various video games (Vinyals et al., 2019) to recommendation systems used by Netflix and Amazon (Gomez-Uribe and Hunt, 2016, Smith and Linden, 2017) and self-driving cars like Tesla (Tian et al., 2018), Machine Learning is finding its way into our everyday lives through an increasing number of applications. Search engines (e.g., Google) and translation software (e.g., DeepL) (Rescigno et al., 2020) also rely on the use of Machine Learning. Google and Facebook have developed their own Deep Learning frameworks with Tensorflow (Martin Abadi et al., 2015) and Pytorch (Paszke et al., 2019), which are constantly being expanded with functionalities and models.

Machine Learning in medicine has also gained some traction in recent years (Wang and Summers, 2012). Particularly in medical imaging, Deep Learning, a subarea of Machine Learning based on the use of so-called artificial neural networks, is increasingly being used. One of the pioneers in the use of Machine Learning in medicine is certainly the field of radiology (Wang and Summers, 2012) and typical applications include the segmentation of tumors or organs based on Magnetic Resonance Imaging (MRI) (Mazurowski et al., 2019), X-Ray (Hemdan et al., 2020), or other imaging techniques. In addition to radiology, the development and exploitation of new use cases in pathology is also increasing (Janowczyk and Madabhushi, 2016). This new field is called digital pathology, where Machine or Deep Learning is used to determine tissue types, and predict mutations and survival times (Echle et al., 2021; Kather et al., 2019a,b). With the help of Deep Learning it is also possible to predict the 3D structures of ribonucleic acid (RNA) (Ramakers et al., 2021) and Deoxyribonucleic acid (DNA) (Jumper et al., 2021) from their sequences.

For this work, we follow the division of Machine Learning use cases under the following three categories: Applications that improve diagnoses of diseases (Fatima, Pasha, et al., 2017; Rambhajani et al., 2015; Sajda, 2006), applications that lead to the more rapid and easier development of new drugs (Ekins et al., 2019; Murphy, 2011; Vamathevan et al., 2019), and applications whereby treatment can be personalized (Ahamed and Farid, 2018). In this thesis, we present three such use cases that together cover all three areas.

This thesis is intended to address mainly computer scientists in the field of Machine and Deep Learning as well as medical doctors with an affinity for Artificial intelligence (AI) and Mathematics so that Machine Learning can be more strongly integrated into everyday clinical practice in the future.

1.1 Research Questions

How can Deep Learning improve medical research and clinical routine? This question was divided into three sub-questions, whereby each of the questions covers one of the previously mentioned topics, namely the diagnosis of diseases, development of new drugs, and personalized treatment.

RQ1: How could researchers automatically detect metastasis of pancreatic ductal adenocarcinoma in the lymph node and how can data labeling be improved?

Relevance: With the advent of BigData and the use of AI in medicine, it is increasingly important to develop data pipelines that can direct the focus of physicians to the difficult and special cases to accelerate diagnosis and support physicians in analyzing the flood of data they are faced with (Fatima, Pasha, et al., 2017; Rambhajani et al., 2015; Sajda, 2006).

Our Approach: We used a two step approach for answering the questions. First, we provide an automatic Deep Transfer Learning approach for data cleaning to provide a better ground truth by iterative classification with a pair of networks called the communicator. Second the data analysis part, we trained a neural network on histology images to detect primary and metastatic pancreatic ductal adenocarcinoma (Kronberg et al., 2022a). See Section 3.2.

RQ2: How could researchers automatically evaluate drug screenings against viruses with a Deep Learning approach?

Relevance: Drug screening is an essential part of drug development. Normally, drug development is a time-consuming, labor-intensive, and costly process which is why faster, cost-effective, and (partially) automated approaches are increasingly being sought (Ekins et al., 2019; Murphy, 2011; Vamathevan et al., 2019).

Our Approach: We introduced an automatic evaluation approach for drug screening against the SARS-CoV-2 by using a Deep Transfer Learning approach. As training data, we used a bright field image of cells with the following three conditions: infected with SARS-CoV-2, controls, and infected cells with drug compounds. In our experimental set-up, our algorithms were able to classify in three classes, namely cythopatic effect, toxicity, and control (Werner et al., 2021). See Section 4.2.

RQ3: Which MRI Sequence should researchers acquire under a fixed time budget (depending on the patient's health status) for a good segmentation result for brain tumors?

Relevance: The current clinical practice is that doctors mostly prescribe medicines by trial and error and use a one-size-fits-all approach (Ahamed and Farid, 2018). Therefore new personalized treatments based on patients data, e.g. maximum acquisition time can add valuable information for the diagnoses.

Our Approach: We evaluated different combinations of orders of MRI sequences and found the best ordering for a fixed time budget. We also show – for the length of three sequences – the significant performance increase with our proposed method. The measure of the performance was a neural network based segmentation algorithm (Kronberg et al., 2022b). See Section 5.2.

1.2 Contributions

We selected the following three use cases for this thesis:

1. Personalized treatment: Communicators improve ground truth during deep transfer learning
2. Development of drugs: Deep Transfer Learning Approach for Automatic Recognition of SARS-CoV-2
3. Diagnosis of diseases: Optimal Acquisition Sequence for AI-assisted Brain Tumor segmentation

1.3 Outline of this Thesis

We give a short overview of the topics Mathematical Modelling and Deep Learning. In addition, we introduce the standard metrics, the training, validation and testing process and methods we later used in our publications.

In Chapter 3, we look at applications for improving the diagnosis of diseases. We introduce our data label clean up application called communicators to optimize noisy labeled or due heritage from meta labels by distinguishing fat tissue from lymph nodes. Later, we use a fine-tuned neural network to detect and classify metastasis of pancreatic ductal adenocarcinoma in the

lymph nodes.

In Chapter 4, we introduce a Deep Learning use case to improve the development of novel drugs. In this context, we propose an Automatic Recognition of Drug Toxicity and Inhibition of SARS-CoV-2, by measuring the virus-induced cytopathic effects in brightfield images.

Chapter 5 deals with a use case that shows how Deep Learning can improve treatment personalization. Our application provides the best ordering for the acquisition of MRI scans for patients with malignant brain tumors if the time frame of acquisition is limited due the individual patient's fears or health situation. We test different combinations of MRI sequences and validate the F1 score from our network's segmentation task.

Last of all, Chapter 6 summarizes our main results and provides an outlook for future research.

Chapter 2

Background

In this chapter, we give a short introduction in Mathematical Modelling, Deep Learning, and Transfer Learning. This introduction is a guide for master students and newly started Ph.D. students to understand the concepts of applied supervised Deep Learning on medical data. Therefore, we assume mathematical and computer science basics to be known. It is essential to understand the ideas and the mathematics behind the algorithms and Python implementations. To better understand, we simplify some definitions and mainly focus on the methods important for this thesis, knowing that there is so much more about Deep Learning.

For more detailed information and an overview of the whole field, we recommend the following three books: (Marc Peter Deisenroth et al. (2020). *Mathematics for machine learning.* Cambridge University Press, Christopher M. Bishop (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics).* Berlin, Heidelberg: Springer-Verlag and Ian J. Goodfellow et al. (2016). *Deep Learning.* Cambridge, MA, USA: MIT Press). In Addition, there is a Computer Science Course on Deep Learning CS231n (Li et al., 2016) and Machine Learning CS540 (Telgarsky., 2008).

2.1 Mathematical Modelling

Complex systems and nested processes can usually not be mapped precisely because, for example, the individual components of the system and their interaction are not known or subprocesses and dependencies are not yet understood or researched. If only the input and the output are known, we often speak of so-called *black boxes.* In order to make predictions about the output for a given input, we must therefore model the system. The model only approximate the mapped system. When modelling, one simplifies assumptions and approximates a solution. Therefore, every model has an error. This error is the difference from the model output to the output of the system. Let's start with basic mathematical modelling.

We want to motivate this chapter with the quote of a Principe called Occam's Razor, which simply says: The simplest solution is the best solution (Hamilton, 1855).

2.1.1 Basic Definitions: Model, Data set, Parameters

First, we define some terms we will use for the mathematical modelling. We start by defining the core of Deep Learning: the data often called the data set.

Definition 2.1.1 *(Data set/Data)*
A data set consists of the feature characteristics, the input X and the targets/labels/output y. We set

$$D = (X, y) = \{x_i, y_i\}_{i=1}^{N}$$

The input data X is often given or transformed as a tensor, that can be rewritten as a high dimensional matrix. The tensor $X = (x_1, \cdots, x_N)$ with $x_i \in \mathbb{R}^d$ and the output $y = (y_1, \cdots, y_N)$ with $y_i \in \mathbb{R}$, here N is the number of samples and d the feature dimension of x.

Often X is also called the (input) data $X = \{x_i\}_{i=1}^{N}, x_i \in \mathbb{R}^d$ and $y = \{y_i\}_{i=1}^{N}, y_i \in \mathbb{R}$ is often called output.

Example 2.1.1 *(Image data)*
For image input data the shape is (N, C, H, W), where N is the number of samples, C is the number of color channels, H is the height of the image and W the width of the image. If our data set consists of 100 colored images with a resolution of 224×224 pixels, the shape of X is given by $(100, 3, 224, 224)$. Corresponding labels are provided in a vector y of size 100. Often, labels are binary (i.e., each entry of y is either 0 or 1) and could, e.g., indicate whether an image of a brain corresponds to a healthy patient or not. A colored image has typically three channels and a black and white image has only one channel.

We define Parameters as follows:

Definition 2.1.2 *(Parameters)*
The parameter vector is given as

$$\Theta = (\theta_0, \ldots, \theta_R),$$

where R is the number of parameters and $\theta_i \in \mathbb{R}, i \in \{0, \ldots, R\}$ are the individual parameters.

A function given through the parameters is called *model*, which is the next term we define.

Definition 2.1.3 *(Model)*
A model is an objective or a function with parameters Θ, which based on input data D predicts output data y, called predictions $\hat{y}$.

$$M(X, \Theta) = \hat{y}.$$

Remark: The aim is to find the parameters, that $\hat{y}$ as closed as possible to the desired ground truth y.

Example 2.1.2 *(Doctor as model)*
By this definition a doctor can be seen as a model $M(\cdot, \Theta)$, where the parameters Θ of this doctor are learned through his education and experience. Based on the image input data X from the data set described in Example 2.1.1, the doctor can predict the patients outcomes $\hat{y}$.

We end this subsection with one definition for later use:

Definition 2.1.4 *(Probability)*
The probability that event A occurs is given by $P(A)$, where $P(A) \in [0, 1]$.

2.1.2 Likelihood

The likelihood, $L(\Theta|D)$, is a function of parameters given the data. For regression problems we typically assume $L(\Theta|D) = N(y|\mu(X, \Theta), \Sigma(X, \Theta))$, which models the output y as normal distribution.

We are looking for a model $M = M(\cdot, \Theta)$ which can make a prediction based on the data X with parameter Θ (see Definition 2.1.1).

We want to determine a model which maximizes the probability

$$\max_M P(M|D).$$

To find such $P(M|D)$ we can use Bayes' Theorem (Deisenroth et al., 2020).

$$P(M|D) = \frac{P(D|M)P(M)}{\sum_{\mathrm{M}} P(D|M)P(M)} = \frac{P(D|M)P(M)}{P(D)},$$

where $P(D|M)$ is called the *likelihood*, $P(M)$ is called the *prior* and $\sum_{\mathrm{M}} P(D|M)P(M)$ is called the *evidence*. The second equality

$$\sum_{\mathrm{M}} P(D|M)P(M) = P(D)$$

only holds because of our normal distribution assumption (Deisenroth et al., 2020). The aim to find the "best" model can be realized by maximizing the term

$$\max_M P(M|D) = \max_M \frac{P(D|M)P(M)}{P(D)} \tag{2.1}$$

$$\propto \max_M P(D|M)P(M) \quad \text{(Maximum a posterior (MAP))} \tag{2.2}$$

$$\propto \max_M P(D|M) \quad \text{(Maximum Likelihood (MLE))}. \tag{2.3}$$

We can use the universal function approximators, called Deep Neural Networks as our models. We give a short excursion about the theory of Deep Learning, the home of Deep Neural Networks. After that we will finish our Likelihood approach.

2.2 Deep Learning

Before we classify Deep Learning thematically and introduce the concepts relevant to us and give a brief overview of the state of research, we would like to describe Deep Learning as what it actually is, mathematics or more precisely statistics and not magic.

2.2.1 Artificial intelligence, Machine Learning and Deep Learning

Artificial Intelligence (AI) is the collective term for applications in which computers perform tasks like learning, making decisions and solving problems. A large and well-known subfield of AI is Machine Learning (ML), where an intersection of mathematics, more specifically applied statistics and probability, and applied computer science has created powerful computer-based methods.

Advances in hardware (e.g. processors, GPUs, TPUs etc.), software (e.g. frameworks like Tensorflow and Pytorch), the development of new algorithms and the availability of Big Data have led to the rise of a new sub-domain under called Deep Learning (LeCun, 2019). Therefore, both areas are associated with the discipline of Artificial Intelligence, see Figure 2.1.

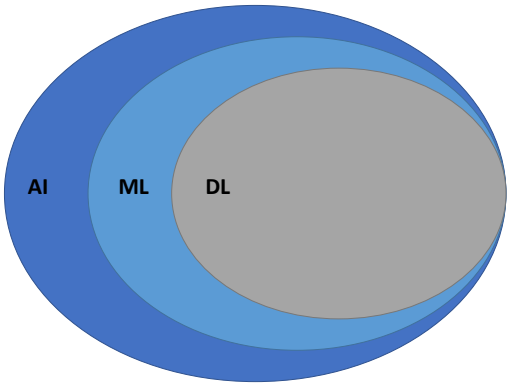


Figure 2.1: A Venn diagram showing how Deep Learning (DL) is a subset of Machine Learning (ML), which is subset of Artificial Intelligence (AI). Adapted and simplified from (Goodfellow et al., 2016)

Here we start with Deep Learning. Deep Neural Networks have been used with great success for computer vision (Dosovitskiy et al., 2020; He et al., 2016; Huang et al., 2017; Kolesnikov et al., 2020; Krizhevsky et al., 2012; Pham et al., 2021; Simonyan and Zisserman, 2014; Szegedy et al., 2015, 2016; Tan and Le, 2019; Xie et al., 2020).

2.2.2 Deep Neural Network Layers

A Deep Neural Network layer consists of a linear transformation, followed by a nonlinear transformation.

Definition 2.2.1 *(Activation functions)*
A element-wise nonlinearity function $\rho(\cdot)$ is called activation function.

Some common activation functions are

$$\rho(x_i) = (1 + \exp(-x_i))^{-1} \qquad \text{(sigmoid function)} \tag{2.4}$$

$$\rho(x_i) = \max(0, x_i) \qquad \text{(rectified linear unit (ReLU))} \tag{2.5}$$

$$\rho(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}} \qquad \text{(Softmax)} \tag{2.6}$$

Definition 2.2.2 *(Fully Connected layer)*
A Fully Connected layer is given by a function

$$\Phi(x, W, b) = \rho(Wx + b), \qquad W \in \mathbb{R}^{d_l \times d_x}, b \in \mathbb{R}^{d_l},$$

where d_x is the input dimension, d_l is dimension of the linear transformation, W are the weights, b the bias, and ρ is an activation function.

Definition 2.2.3 *(Residual layer) A residual layer is defined as*

$$\Phi(x, \Psi) = \Psi(x) + x,$$

where $\Psi(x)$ is any deep neural network layer.

Definition 2.2.4 *(Convolutions layer)*
A convolutional layer is defined as

$$\Phi(x, W, b) = \rho(W * x + b),$$

where W are the weights, b the bias and $(\cdot * \cdot)$ is the discrete convolution operator.

Convolutional Neural Networks, consisting mainly of convolutions layers, are used in computer vision tasks because of the convolution operator's implicit spatial weight sharing (Kilcher, 2021; Krizhevsky et al., 2012).

A ResNet Architecture combines Convolutional Neural Networks and Residual layers and activation functions (He et al., 2016).

Definition 2.2.5 *(Normalization layer for 2D Image)*
A 2D image data set has the tensor shape of (N, C, H, W), where N is Batch axis, C is the channel axis, and H, W are the spatial axis Example 2.1.1. Most common normalization performs the same calculations, but on different axes:

$$z_i = \frac{1}{\sigma_i}(x_i - \mu_i),$$

where

$$\mu_i = \frac{1}{m}\sum_{k \in S_i} x_k,$$

and

$$\sigma_i = \sqrt{\frac{1}{m}\sum_{k \in S_i}(x_k - \mu_i)^2 + \epsilon},$$

where ϵ is a small constant, $i = (iN, iC, iH, iW)$ is a 4D vector indexing the features in (N, C, H, W) order and S_i is the set of pixels in which the mean μ and and the standard deviation σ are calculated (Ioffe and Szegedy, 2015; Wu and He, 2018)

Definition 2.2.6 *Batch Norm (BN)*
For the Batch Norm (BN) the set of pixels is given by

$$S_i = \{k | k_C = i_C\},$$

where i_C (and k_C) denote the sub-index of i (and k) along the C-axis. BN computes μ and σ along the (N, H, W)-axis (Ioffe and Szegedy, 2015; Wu and He, 2018). This is a special choice for S_i in Definition 2.2.5.

Definition 2.2.7 *Group Norm (GN)*
Let be G the number of groups (group of channels), for the Group Norm (GN) the set of pixels is given by

$$S_i = \{k | k_N = i_N, \left\lfloor \frac{k_C}{C/G} \right\rfloor = \left\lfloor \frac{i_C}{C/G} \right\rfloor\},$$

where i_C and k_C denote the sub-index of i and k along the C-axes (Ioffe and Szegedy, 2015; Wu and He, 2018). This is a special choice for S_i in Definition 2.2.5.

The number of channels per group is C/G. The Operation $\lfloor \cdot \rfloor$ is the floor operation, and $\left\lfloor \frac{k_C}{C/G} \right\rfloor = \left\lfloor \frac{i_C}{C/G} \right\rfloor$ means that the indexes i and k are in the same group of channels, assuming each group of channels are stored in a sequential order along the C axis. GN calculates μ and σ along the (H, W) axis and along a group of C/G channels (Ioffe and Szegedy, 2015; Wu and He, 2018).

For a small batch size, the use of Group Norm is preferable (Wu and He, 2018). We used the Group Norm in Kronberg et al., 2022b, because our batch size was only 8.

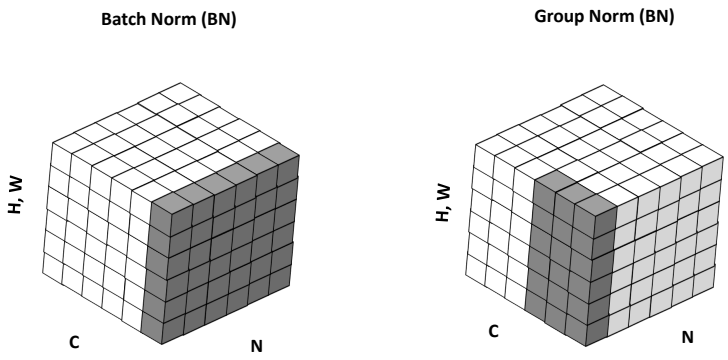


Figure 2.2: Visualization of the Batch Norm and Group Norm from a tensor with shape (N, C, H, W), where N is the batch axis, C the channel axis and the third spatial axis combines height and width axes. The pixels in dark gray are normalized by the same mean values and standard deviation, calculated by aggregating these pixels. Adapted and simplified from (Wu and He, 2018)

2.2.3 Deep Neural Network Task Types

Here we want to introduce the two Deep Neural Network Task Types we will in this thesis. Therefore, we continue with our Likelihood approach from Section 2.1.2, now that we know what a Neural Network is. From (2.1) we got:

$$\max_M P(M|D) \propto \max_M P(D|M).$$

Now we can use our Deep Neural Network as model. We set

$$M = M_{\text{DNN}}(\Theta),$$

where $M_{\text{DNN}}(\Theta)$ is a Deep Neural Network with parameter Θ.

This leads to the following equation to find the model which maximizes the probability $P(M|D)$. The Model is defined by its parameter vector Θ, so we get

$$\Theta = \text{argmax}_\Theta P(D|M(\Theta)) \tag{2.7}$$

Here, we need the probability

$$P(D|M(\Theta)) := P_M(D|\Theta).$$

To solve the maximization (2.7) we use a parameter optimizing trick.

Definition 2.2.8 *(Loss function)*
A loss function $L(M(x), y)$ is a measure of the difference between the actual output $M(x)$ and the desired output y.

Here we list two commonly used loss functions, which are cross-entropy loss (discrete targets) and $L2$-loss (continuous targets).

Definition 2.2.9 *(Cross-entropy loss)*
The Cross-entropy loss (CEL) is given by

$$L(M(x), y) = -\sum_{i}^{K} y_i \log(M(x_i)),$$

where x is the input, K the number of classes, M the Deep Neural Network and y contains a onehot encoding of the target label.

Definition 2.2.10 *(L2-loss)*
The L2-loss is given by

$$L(M(x), y) = \frac{1}{2}||y - M(x)||_2^2,$$

where x is the input, M the Deep Neural Network and y contains the continuous target values.

Optimizer

Deep Neural Networks are nonlinear and complex nested functions, so there is no closed-form solution that can be used for optimizing a particular loss function. In this subsection our model M is a Deep Neural Network called $M(\cdot)$.

To maximize a continuous loss function $L(w, y)$ by optimizing the parameter w, we can use the gradient information and use the following iteration algorithm, called **vanilla gradient descent**

$$w_{t+1} = w_t - \alpha * \nabla w_t, \tag{2.8}$$

where w is the parameter, α is the step size and t the iteration step.

In this thesis, we used the following three different optimizers.

The SDG use mini-batches and looks very similar to our vanilla gradient descent.

Definition 2.2.11 *(SGD) The parameter update by the Stochastic Gradient Descent optimizer is defined by*

$$w_t = w_{t-1} - \alpha \frac{1}{Q} \sum_{n=1}^{Q} \frac{\partial L(M(x_n), y_n)}{\partial w},$$

where $\{x_n, y_n\}_{n=1}^{Q}$ are small sets of data points and corresponding labels, also called mini-batch, sampled from the training data, and $\alpha \in \mathbb{R}$ is a step size parameter (Robbins and Monro, 1951).

The central concept of RMSprop, can described in two steps. First keeping the moving average of the squared gradients for each weight. Second, normalize the gradient by root mean square.

Definition 2.2.12 *(RMSprop) The parameter update by the RMSprop optimizer is defined by*

$$w_t = w_{t-1} - \frac{\alpha}{\sqrt{E[g^2]_t}} \frac{\partial L(M)}{\partial w},$$
$$E[g^2]_t = \beta E[g^2]_{t-1} + (1-\beta)\left(\frac{\partial L(M)}{\partial w}\right)^2,$$

where α is the learning rate, β the moving average parameter, $E[g^2]_t$ the moving average of squared gradients where g_t is the gradient of the loss function w.r.t w (the parameters) at timestep t. (Tieleman, Hinton, et al., 2012)

The Adam optimizer is kind of combination of RMSprop and Stochastic Gradient Descent with momentum.

Definition 2.2.13 *(Adam) The parameter update by the Adam optimizer is defined by*

$$w_t = w_{t-1} - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon},$$
$$\hat{v} = \frac{v_t}{(1-\beta_2^t)},$$
$$\hat{m} = \frac{m_t}{(1-\beta_1^t)},$$
$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2,$$
$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t,$$
$$g_t = \frac{\partial L_t(M, w_{t-1})}{\partial w_t},$$

where g is the gradient w.r.t the loss function L_t, m the biased first moment estimate, v the biased second raw moment estimate, $\hat{m}$ the bias-corrected first moment estimate, $\hat{v}$ the bias-corrected second raw moment estimate and α the learning rate. (Kingma and Ba, 2014)

Now we need to define a suitable loss function. Here we make a two case distinctions, depending on the target/label y. Is y a vector/tensor with real-value elements, we call the task **Regression**, if y has discrete elements, called classes, we call the task **Classification**.

Classification

Starting with Classification we use the softmax activation function (2.6). Assume we have $x \in X$ and K is the number of classes, then

$$P_M(\text{class} = l|x) = \frac{e^{f_l}}{\sum_{k=1}^{K} e^{f_k}},$$

where $f_k(x)$ are the logits, the output from the last fully connected layer. Since $P_M(\cdot)$ is a probability we have

$$\sum_{k=1}^{K} P_M(l|x) = 1.$$

Using the Kullback–Leibler divergence (Ay et al., 2018), which measured the difference between two probability distributions, we kind of construct the Cross-entropy loss. The parameter update is given by

$$\Theta_{i+1} = \Theta_i - \alpha \nabla_\Theta \mathbb{E}_{x,l\sim D}[D_{\mathrm{KL}}(P_D(l|x)||P_M(l|x)],$$

where α is the learning rate and i the iteration step.

Remark: In practice we approximate the expectation $\mathbb{E}$ over all data by an average over mini-batches.

Now we only focus on the gradient of the "loss function part" and drop the iteration index.

$$\begin{aligned}
&\nabla_\Theta D_{\mathrm{KL}}(P_D(l|x)||P_M(l|x)) \\
&= \nabla_\Theta \sum_l P_D(l|x) \log\left(\frac{P_D(l|x)}{P_M(l|x)}\right) \\
&= \nabla_\Theta \sum_l P_D(l|x) \log(P_D(l|x)) \\
&- \nabla_\Theta \sum_l P_D(l|x) \log(P_M(l|x)) \\
&= -\nabla_\Theta \sum_l P_D(l|x) \log(P_M(l|x))
\end{aligned}$$

So we get

$$\nabla_\Theta \mathbb{E}_{x,l\sim D} D_{\mathrm{KL}} = -\nabla_\Theta \mathbb{E}_{x,l\sim D} \log(P_M(l|x))$$

Plugin this term into (2.7) leads to

$$\begin{aligned}
\Theta &= \mathrm{argmax}_\Theta P(D|M(\Theta)) \\
\Theta &= \mathrm{argmin}_\Theta \mathbb{E}_{x,l\sim D} \log(P_M(l|x)),
\end{aligned}$$

what is the definition of the Cross-entropy loss.

The Cross-entropy loss is commonly used for classification tasks. In this thesis we used this loss for the work (Kronberg et al., 2022a; Werner et al., 2021).

Regression

We use the Maximum Likelihood Estimation approach, to get the $L2$-loss. For our targets y_i, the predictions from our Deep Neural Network $M(x_i, \Theta) = \hat{y}_i$, with the Central Limit Theorem (Grinstead and Snell, 2012) we can write:

$$\begin{aligned} y_i &= \hat{y}_i + \epsilon, \\ y_i - M(x_i, \Theta) &= \epsilon \sim N(\mu_0, \sigma_0^2), \end{aligned}$$

where σ_0 is the variance. With no loss of generality we assume that the expectation is $\mu_0 = 0$.

By using the logarithms on both sides and the assuming independence of the data samples we get

$$\begin{aligned} P_M(D|\Theta) &= \prod_{i=1}^{N} P_\epsilon(y_i - M(x_i, \Theta)) \\ \Leftrightarrow \log P_M(D|\Theta) &= \log \prod_{i=1}^{N} P_\epsilon(y_i - M(x_i, \Theta)) \\ \Leftrightarrow \log P_M(D|\Theta) &= \sum_{i=1}^{N} \log \left(\frac{1}{\sqrt{2\pi}\sigma_0} e^{-\frac{(y_i - M(x_i,\Theta))^2}{2\sigma_0^2}} \right) \\ \Leftrightarrow \log P_M(D|\Theta) &= C_0 - \frac{1}{2\sigma_0^2} \sum_{i=1}^{N} (y_i - M(x_i, \Theta))^2, \end{aligned}$$

where C_0 is a constant which does not matter for optimization and can be dropped. Plugging this term into (2.7) leads to

$$\begin{aligned} \Theta &= \text{argmax}_\Theta P(D|M(\Theta)) \\ \Leftrightarrow \Theta &= \text{argmax}_\Theta \left(-\frac{1}{2\sigma_0^2} \sum_{i=1}^{N} (y_i - M(x_i, \Theta))^2 \right) \\ \Leftrightarrow \Theta &= \text{argmin}_\Theta \frac{1}{2\sigma_0^2} \sum_{i=1}^{N} (y_i - M(x_i, \Theta))^2 \\ \Leftrightarrow \Theta &= \text{argmin}_\Theta \frac{1}{2} \sum_{i=1}^{N} (y_i - M(x_i, \Theta))^2, \end{aligned}$$

which is the definition of the $L2$-loss.
The $L2$-loss is commonly used for regression and reconstruction tasks. We use this loss in (Kronberg et al., 2022b).

Data set Splitting: Train, Validation and Test

In order to avoid overfitting (the model memorizes the training data too well and badly works on new data), we split our data set, see Definition 2.1.1 into up to three parts, depending on

whether we want to perform hyperparameter tuning or not (see Section 2.2.6). Commonly used data set splits are (train: 0.7, val: 0.15, test: 0.15) and (train: 0.75, test: 0.25).

2.2.4 Training of Deep Neural Networks

To perform the gradient calculation/approximation in Equation (2.8) (for advanced optimizers see 2.2.3), we use the backpropagation algorithm introduced by (Rumelhart et al., 1986). The process of parameters/weights optimization is called *training*.

Definition 2.2.14 *Chain rule*
Let f be depend on the variable y, which itself depends on the variable x, then f depends on x as well, via the intermediate variable y. Then the following equations holds:

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial y}\frac{\partial y}{\partial x}. \tag{2.9}$$

Layerwise application of the chainrule (2.9) constructs an algorithm for efficient computation of the parameter/weights updates.

Definition 2.2.15 *Backpropagation*
Given a loss function L, a Deep Neural Network M with the layerwise representations of the input signal $\{h_0, ..., h_L\}$ with the input $x = h_0$ and the output $y = h_L$, then the derivative of the loss L with respect to parameter w_l of layer l is given by

$$\frac{\partial L}{\partial w_l} = \frac{\partial L}{\partial h_L}\prod_{k=l+1}^{L}\left[\frac{\partial h_k}{\partial h_{k-1}}\right]\frac{\partial h_l}{\partial w_l},$$

This schematic has a straightforward extension to higher dimensions (Kilcher, 2021).

The efficient computation of derivatives with respect to all network parameters, allowed by the application of the chain rule, can effectively be re-used by starting from the last layer and passing derivatives down the layer hierarchy in a successive pattern (Kilcher, 2021).

Backpropagation is implemented in Python's Deep Learning frameworks (e.g. Pytorch (Paszke et al., 2019), Tensorflow (Martin Abadi et al., 2015)) to allow automated derivative computation for nearly all types of loss functions.

2.2.5 Metrics for Classification and Regression

To compare two configuration of Deep Neural Networks or models we need a measurement. In this section, we will introduce so-called metrics for the two task types.

Metrics for Classification

We will start with the Classification metrics:

Definition 2.2.16 *True Positive, True Negative, False Positive and False Negative The True Positive (TP) [hit], True Negative(TN) [correct rejection], False Positive (FP) [false alarm] and False Negative (FN) [miss] can be easy evaluated by comparing the prediction with the ground truth.*

We can calculate the followings metrics 2.2.17. We only show the formulas for the binary case.

Confusion Matrix A common graphic to show the True Positive, True Negative, False Positive and False Negative is the confusion matrix see Figure 2.3.

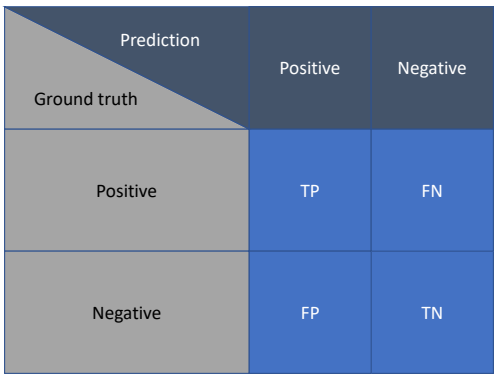


Figure 2.3: Confusion Matrix, True Positives, True Negatives, False Positives and False Negatives for the binary case.

Definition 2.2.17 *(Accuracy, Precision, Recall, F1-score, Jaccard-score.)*
With TP, TN, FP and FN from 2.2.16 we define:

$$\begin{aligned} Accuracy &= \frac{TP+TN}{TP+TN+FN+FP}, \\ Precision &= \frac{TP}{TP+FP}, \\ Recall &= \frac{TP}{TP+FN}, \\ F1\text{-}score &= \frac{2TP}{2TP+FN+FP}, \\ Jaccard\text{-}score &= \frac{TP}{TP+FN+FP}. \end{aligned}$$

The best possible score for the above metrics is 1.0, the worst is 0.0.

Metrics for Regression.

We only use following metric for Regression Tasks in our thesis:

Definition 2.2.18 *(R^2)*
Let be $\hat{y}_i$ the prediction of the i-th data sample and y_i is the corresponding ground truth for total n samples, the estimated R^2 is given by

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

where $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ and $\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} \epsilon_i^2$. (Pedregosa et al., 2011).

R^2 represents the proportion of variance (of y) that has been explained by the independent variables in the model. The score value satisfy $R^2 \in [-\infty, 1]$ and $R^2 = 0.0$ if independent of the input features (Pedregosa et al., 2011).

2.2.6 Validation and Hyper-Parameter-Tuning

Definition 2.2.19 *(Hyper-Parameter)*
Hyper-Parameters are the non-learnable parameters of Deep Learning algorithms. In contrast, learnable parameters are, for example, the weights.

The process of finding the most effective set of non-learnable parameters, such as learning rate, batch size, model architecture or weight decay, is called Hyper-Parameter Tuning. Selecting the best Hyper-Parameters is not an easy task and there is no good recipe for it. However, there exist different strategies to address it and even some computational approaches, e.g. Brute force, Grid search, Random search (Rodríguez-Barroso et al., 2019).

Definition 2.2.20 *(Brute force) Brute force consists of the complete evaluation of all possible values of all the hyper-parameters.*

This approach is not feasible for the most Deep Neural Networks, because of limitation of time and computational resources.

Definition 2.2.21 *(Grid search) Grid search is a brute force approach but constrained by a predefined set of hyper-parameters values.*

On the one hand this can be a feasible method because the number of evaluations is lower in comparison with brute force and it allows to reach good results as show in [Kim, 2014]. But on the other hand, the Hyper-Parameter values must be defined by hand, so there is a bias included.

Definition 2.2.22 *(Random search) Random search is brute force approach but constrained by a randomly chosen set of hyper-parameters values.*

Random search of the values of the hyper-parameters allows the Deep Neural Network to reach good results (Bergstra and Bengio, 2012), but the random search cannot assure to find out the values that optimise the performance of the network.

We used the Grid search approach in (Figure 6 Kronberg et al., 2022a) to visualize different clusters of good performing architectures.

2.2.7 Testing a Deep Neural Network

One of the last steps in the Deep Learning pipeline is to test the Deep Neural Network on the test data set. The performance of the network and the metrics (Section 2.2.5) are reported on the test data set. It is really important that the algorithm doesn't see this data before, neither at training, or at validation time. If possible one should compare the results on the test data of a benchmark data set to other research groups.

2.3 Deep Transfer Learning

Limited amount of data is very challenging for the most Deep Learning architectures. For the discovery of the data patterns, the capacity (e.g., number of parameters) of the model should be large enough. Deep Neural Networks can detect features as follows: The first layers can detect top-level features of the data, and the following layers can detect more low-level features (Tan et al., 2018).

In practice, there is often a lack of large data sets. As an example, consider medical data set of rare diseases with low incidences. Due to various reasons, for example that certain diseases occur only rarely or there is a lack of healthy samples for the control group, in addition to high collection costs (for example MRI), the annotation of the data is often time and cost intensive. In addition, many medical institutions lack sufficient digitization and automation of data management.

The knowledge transfer is from the large data set, namely the source data set to the minor data set called the target data set. The data dependency issue, can be solved by Deep Transfer Learning, because the target domain model doesn't require training from scratch, which can reduce the training time and the training data requirement (Figure 2.4 Tan et al., 2018).

Remark In this section we will use a slightly different notation. We will adapt to the notation of (Pan and Yang, 2009).

First, we give the definitions of the terms domain and task.

Definition 2.3.1 *(Domain)*
A domain is given by the tuple $\mathcal{D} = \{\mathcal{X}, P(X)\}$, the feature space $\mathcal{X}$ and the probability distribution $P(X)$, where $X = \{x_1, \ldots, x_n\} \in \mathcal{X}$ (Pan and Yang, 2009).

Definition 2.3.2 *(Task)*
A task is given by the tuple $\mathcal{T} = \{\mathcal{Y}, f(X)\}$, where $\mathcal{Y}$ is the label space and $f(X)$ is the target prediction function (Pan and Yang, 2009).

Definition 2.3.3 *(Transfer Learning)*
Given a learning task $\mathcal{T}_t$ based on $\mathcal{D}_t$ (target dataset), and we can get the help from $\mathcal{D}_s$(source dataset) for the learning task $\mathcal{T}_s$. Transfer learning aims to improve the performance of the predictive function $f_{\mathcal{T}}(\cdot)$ for learning task $\mathcal{T}_t$ by discovering and transfering latent knowledge from $\mathcal{D}_s$ and $\mathcal{T}_s$, where $\mathcal{D}_s \neq \mathcal{D}_t$ and/or $\mathcal{T}_s \neq \mathcal{T}_t$. In addition, in most cases, the size of $\mathcal{D}_s$ is much larger than the size of $\mathcal{D}_t$, $N_s >> N_t$ (Pan and Yang, 2009).

Definition 2.3.4 *(Deep Transfer Learning) A task defined by $\{\mathcal{D}_s, \mathcal{T}_s, \mathcal{D}_t, \mathcal{T}_t, f_T(\cdot)\}$. is called a Deep Transfer Learning task, where $f_T(\cdot)$ is a non-linear function that reflects a Deep Neural Network (Tan et al., 2018).*

In this thesis we only focus on network-based Deep Transfer Learning. It relates to the reuse of the weights of a Deep Neural Network, that was trained in the source domain, and then transferred/copied to the Deep Neural Network weights which are used in the target domain (Tan et al., 2018). Both Deep Neural Networks share the same architecture.

For our Deep Transfer Learning approaches we use a variety of pretrained Deep Neural Networks including ResNet18, ResNet50, ResNet101 (He et al., 2016), Vgg-16, Vgg-19 (Simonyan

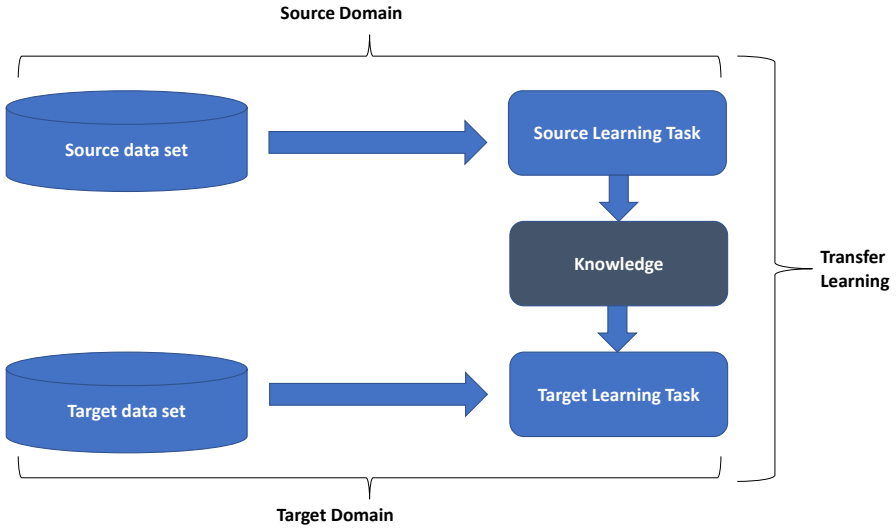


Figure 2.4: Learning process of Transfer Learning, adapted from Tan et al., 2018

and Zisserman, 2014), Alexnet (Krizhevsky et al., 2012), DenseNet (Huang et al., 2017) and SqueezeNet (Iandola et al., 2016). These networks are trained on very large datasets, with millions of sample data (e.g., Imagenet (Deng et al., 2009)), and their weights are stored. If we have the pretrained network, we copy all model parameters, but replace the output layer with a fitting output layer for our target dataset (number of classes). Then we freeze the weights for the first to the $L-1-N$-th Layer and just update the weights for the $L-N$-th to the output layer Figure 2.5. The output layer is initialized randomly. This is also called fine-tuning.

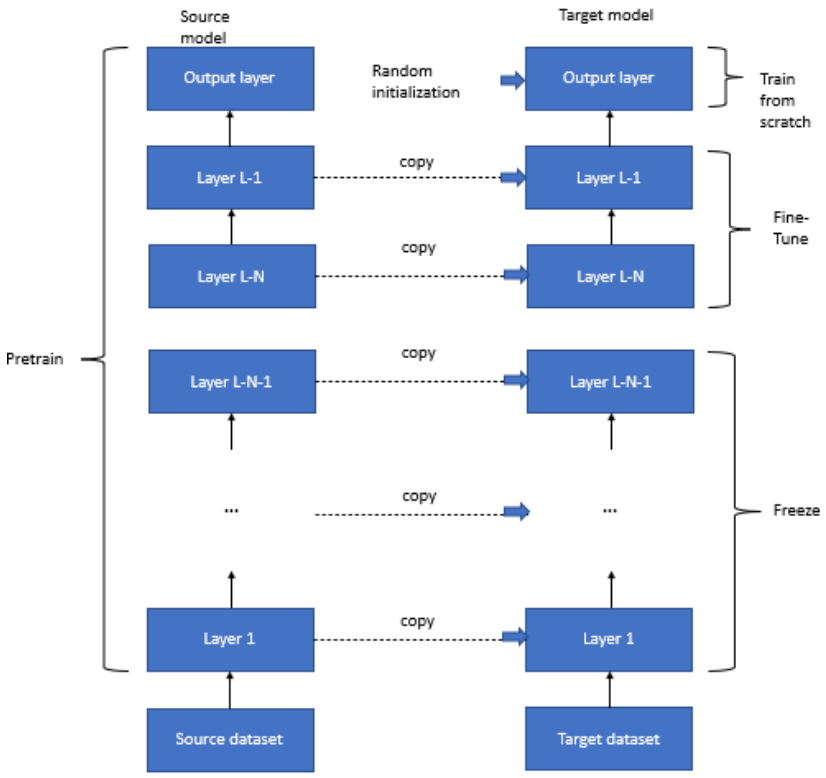


Figure 2.5: Fine-Tuning of a neural network. Coping the parameters of the Source model with L layers to the Target model, randomly initializing the parameter of the Output layer (L-th) and freezing (don't update) the parameters of the first $L-N-1$ layers and retrain the other layers.

Chapter 3

Improving the Diagnosis of Diseases using Deep Learning

In this chapter, we present our use case concerning the diagnosis of diseases: Communicators improve ground truth during deep transfer learning. We briefly introduce the topic "Improved Diagnosis of diseases using Machine Learning" before we present our approach to detect pancreatic cancer in lymph nodes as highlighted in Figure 3.

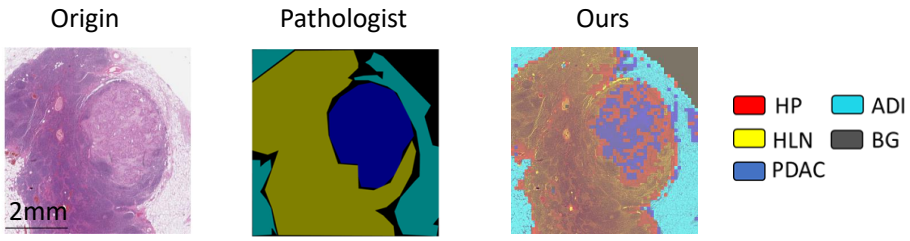


Figure 3.1: Origin, Prediction (Ours), and Ground truth of an H&E stained Whole Image Slide (WIS) of Lymph node annotated by a pathologist that was used to validate the neuronal network's performance in detecting PDAC (scalebar = 2 mm). Image based on Figure 5 in Kronberg et al., 2022a.

3.1 Improving the Diagnosis of Diseases using Machine Learning

Digitization has offered comparatively low-cost and accessible means for the collection and storage of data and created an entry point for effective diagnosis and cost-effective management by Machine Learning-based decision support systems. Clinical routine generates Big Data concerning clinical assessments, reports regarding patients, cures, follow-ups, and prescribed medication. Machine Learning and Deep Learning offer a principled approach for developing automated and objective algorithms for the analysis of high-dimensional and multimodal

data. Correct diagnostic data, the ground truth (outcomes), are presented using data based on previous cases, which in the supervised case, the algorithm can learn from. Predictions can automatically be obtained from the previously solved cases that were annotated by the physicians. As a result, physicians are assisted by this derived Machine Learning algorithm while pre-diagnosing new patients quickly and with enhanced accuracy, thereby allowing them to focus their skills and efforts on the more complex and unusual cases. Machine Learning is capable of managing Big Data and combining data from dissimilar resources (Fatima, Pasha, et al., 2017; Rambhajani et al., 2015; Sajda, 2006).

3.2 Communicator-driven Data Preprocessing Improves Deep Transfer Learning of Histopathological Prediction of Pancreatic Ductal Adenocarcinoma.

In this section, we give an overview of the contributions and impact of our paper Kronberg et al., 2022a:

Raphael M. Kronberg, Lena Härberle, Melanie Pfaus, Karina S. Krings, Haifeng C. Xu, Martin Schlensog, Tilman Rau, Aleksandra A. Pandyra, Karl. S. Lang, Irene Esposito and Philipp A. Lang

"Communicator-Driven Data Preprocessing Improves Deep Transfer Learning of Histopathological Prediction of Pancreatic Ductal Adenocarcinoma"

In: *Cancers* 2022, 14(8), 1964.

Main Results in Simple Terms

In the cited paper, using the example of identifying pancreatic cancer in the pancreas (primary tumors) or in other tissues (metastases) with the help of images from routine moderately collected histological sections, we showed that Machine Learning can improve the diagnosis of diseases. To more clearly define the term "improve" in relation to this use case, we hereby briefly explain what is meant by it. The neural network we developed can support the medical practitioner by filtering relevant data, i.e., a large number of images of the histological sections can be analysed automatically and in a short time by the neural network. The results of the neural network can point out critical images to the medical practitioner and direct the focus from the images that are easy to classify to the difficult cases. This gives the physician more time to look at the difficult cases.

In simple terms, we scanned and digitized the existing tissue sections and the pathologist annotated them. There were always several tissue spots on a scan page, which we then cut out individually and sorted according to tissue. This gave us a data set that the neural network could use to learn what the different tissue types look like.

Since the number of images were limited, we did not train a neural network from the beginning, but used transfer learning and further trained a neural network that can already recognise images to be able to classify our tissue types. We then processed the data, including a uniform resolution of the scans. In addition, due to the staining agent used, the staining of the sections varies fairly strongly between different analyses. We were able to put this into perspective with an appropriate pre-processing step. Unfortunately, the tissues were not all homogeneous, i.e., not only of a given named tissue type and some tissue spots contained other tissues. Therefore, the neural network would learn the wrong patterns, and we had to correct these defects in the

labels. For this task, we again used a Deep Learning neural network, whereby we added an additional tissue type from another data set and pre-sorted our data, so that we got five tissue types (including background) from our previous three.

After the training, we tested the neural network on unknown tissue samples. To do this, we first had the three known tissue types analyzed, and the neural network was able to analyze these images well. Furthermore, we tested the extent to which the neural network generalizes by taking larger tissue sections from which no spots were punched and which, for example, also show metastases from the pancreas in the lymph node. The pathologists annotated the external data again. This time we even got a segmentation map, i.e., we got a pixel-precise ground truth. From this, we could compare the analysis of the neural network with the ground truth of the pathologists by calculating and comparing the percentage of the corresponding tissue. This generalization (learning on the spots cut from the whole image slides and then analyse the whole image slides) works relatively well, which was not to be granted by theory of deep learning generalization.

Afterwards, we tried to test and optimise our neural network with different conditions and subsequently tested a total of 72 different configurations. In consequence we could find the best configuration for our data set from the 72. In Addition we show clusters of performance, group by architecture, optimizer and learning rate.

Remark: Due to the very specific laboratory data of patients with a particular disease, there are no suitable benchmark data sets to compare our method with other research groups.

Summary/Abstract

Pancreatic cancer is a fatal malignancy with poor prognosis and limited treatment options. Early detection in primary and secondary locations is critical, but fraught with challenges. While digital pathology can assist with the classification of histopathological images, the training of such networks always relies on a ground truth, which is frequently compromised as tissue sections contain several types of tissue entities. Here we show that pancreatic cancer can be detected on hematoxylin and eosin (H&E) sections by convolutional neural networks using deep transfer learning. To improve the ground truth, we describe a preprocessing data cleanup process using two communicators that were generated through existing and new datasets. Specifically, the communicators moved image tiles containing adipose tissue and background to a new data class. Hence, the original dataset exhibited an improved labelling and consequently a higher ground truth accuracy. Deep transfer learning of a ResNet18 network resulted in a five-class accuracy of about 94% on test data images. The network was validated on independent tissue sections composed of healthy pancreatic tissue, pancreatic ductal adenocarcinoma, and pancreatic cancer lymph node metastases. Screening of different models and

hyperparameter fine tuning was performed to optimize the performance on the independent tissue sections. Taken to-gether, we introduce a data preprocessing via communicators step as a means of improving the ground truth during deep transfer learning and hyperparameter tuning to identify pancreatic ductal adenocarcinoma primary tumors and metastases in histological tissue sections. (Kronberg et al., 2022a).

Personal Contribution to the Research

Formulated sentences

Raphael Marvin Kronberg (R.M.K.) performed computational experiments and data analysis, e.g. he calculated the metrics for the different Deep Neural Networks. He discussed the data and wrote the draft of the paper, including providing data for the figures. The implementation of the Deep Neural Networks and the pipeline in Python using Pytorch as framework was carried out by R.M.K.. In Addition, he supported Mr. Prof. Dr. Lang with the project administration.

Bullet points (CRediT version)

Conceptualization, I.E. and P.A.L; methodology, R.M.K. and M.P. and L.H. and H.C.X. and K.S.K. and M.S. and A.A.P and K.S.L. and I.E. and P.A.L.; software, R.M.K.; validation, L.H. and I.E.; formal analysis, R.M:K. and P.A.L.; investigation, R.M.K. and M.P. and L.H. and I.E. and P.A.L.; resources, I.E. and P.A.L.; data curation, R.M.K. and M.P. and L.H. and M.S. and P.A.L.; writing—original draft preparation, R.M.K; writing—review and editing, R.M.K. and M.P. and L.H. and T.R. and A.A.P and K.S.L. and I.E. and P.A.L; visualization, R.M.K. and M.P. and H.C.X. and K.S.K; supervision, I.E. and P.A.L.; project administration, R.M.K. and P.A.L.; funding acquisition, I.E. and P.A.L. . All authors have read and agreed to the published version of the manuscript.(Kronberg et al., 2022a)

Importance of the Research and Contribution to this Thesis

The automated Deep Learning-based classification of pancreatic cancer in histology images serves as an example of how artificial intelligence can improve the diagnosis of diseases. It answers our second research question: How could researchers automatically detect metastasis of pancreatic ductal adenocarcinoma in the lymph node and how can data labeling be improved?

cancers

MDPI

Article

Communicator-Driven Data Preprocessing Improves Deep Transfer Learning of Histopathological Prediction of Pancreatic Ductal Adenocarcinoma

Raphael M. Kronberg [1,2,†], Lena Haeberle [3,†], Melanie Pfaus [1], Haifeng C. Xu [1], Karina S. Krings [1], Martin Schlensog [3], Tilman Rau [3], Aleksandra A. Pandyra [4], Karl S. Lang [5], Irene Esposito [3,‡] and Philipp A. Lang [1,*,‡]

[1] Department of Molecular Medicine II, Medical Faculty, Heinrich-Heine-University, Universitätsstrasse 1, 40225 Düsseldorf, Germany; raphael.kronberg@hhu.de (R.M.K.); melanie.pfaus@hhu.de (M.P.); xuh@uni-duesseldorf.de (H.C.X.); kakri104@uni-duesseldorf.de (K.S.K.)
[2] Mathematical Modelling of Biological Systems, Heinrich-Heine University, Universitätsstrasse 1, 40225 Düsseldorf, Germany
[3] Institute of Pathology, Medical Faculty, Heinrich-Heine University and University Hospital of Duesseldorf, Moorenstr. 5, 40225 Düsseldorf, Germany; lenajulia.haeberle@med.uni-duesseldorf.de (L.H.); martin.schlensog@med.uni-duesseldorf.de (M.S.); Tilman.Rau@med.uni-duesseldorf.de (T.R.); irene.esposito@med.uni-duesseldorf.de (I.E.)
[4] Department of Pediatric Oncology, Hematology and Clinical Immunology, Medical Faculty, Center of Child and Adolescent Health, Heinrich-Heine-University, Moorenstrasse 5, 40225 Düsseldorf, Germany; aleksandra.pandyra@uni-duesseldorf.de
[5] Institute of Immunology, Medical Faculty, University of Duisburg-Essen, Hufelandstr. 55, 45147 Essen, Germany; KarlSebastian.Lang@uk-essen.de
* Correspondence: langp@uni-duesseldorf.de
† These authors share equal contribution.
‡ These authors share equal senior authorship.

check for updates

Citation: Kronberg, R.M.; Haeberle, L.; Pfaus, M.; Xu, H.C.; Krings, K.S.; Schlensog, M.; Rau, T.; Pandyra, A.A.; Lang, K.S.; Esposito, I.; et al. Communicator-Driven Data Preprocessing Improves Deep Transfer Learning of Histopathological Prediction of Pancreatic Ductal Adenocarcinoma. *Cancers* **2022**, *14*, 1964. https://doi.org/10.3390/cancers14081964

Academic Editor: Atsushi Masamune

Received: 14 March 2022
Accepted: 1 April 2022
Published: 13 April 2022

Simple Summary: Pancreatic cancer has a dismal prognosis and its diagnosis can be challenging. Histopathological slides can be digitalized and their analysis can then be supported by computer algorithms. For this purpose, computer algorithms (neural networks) need to be trained to detect the desired tissue type (e.g., pancreatic cancer). However, raw training data often contain many different tissue types. Here we show a preprocessing step using two communicators that sort unfitting tissue tiles into a new dataset class. Using the improved dataset neural networks distinguished pancreatic cancer from other tissue types on digitalized histopathological slides including lymph node metastases.

Abstract: Pancreatic cancer is a fatal malignancy with poor prognosis and limited treatment options. Early detection in primary and secondary locations is critical, but fraught with challenges. While digital pathology can assist with the classification of histopathological images, the training of such networks always relies on a ground truth, which is frequently compromised as tissue sections contain several types of tissue entities. Here we show that pancreatic cancer can be detected on hematoxylin and eosin (H&E) sections by convolutional neural networks using deep transfer learning. To improve the ground truth, we describe a preprocessing data clean-up process using two communicators that were generated through existing and new datasets. Specifically, the communicators moved image tiles containing adipose tissue and background to a new data class. Hence, the original dataset exhibited improved labeling and, consequently, a higher ground truth accuracy. Deep transfer learning of a ResNet18 network resulted in a five-class accuracy of about 94% on test data images. The network was validated with independent tissue sections composed of healthy pancreatic tissue, pancreatic ductal adenocarcinoma, and pancreatic cancer lymph node metastases. The screening of different models and hyperparameter fine tuning were performed to optimize the performance with the independent tissue sections. Taken together, we introduce a step of data preprocessing via communicators as a means of improving the ground truth during deep transfer learning and

Cancers **2022**, *14*, 1964

hyperparameter tuning to identify pancreatic ductal adenocarcinoma primary tumors and metastases in histological tissue sections.

Keywords: computer vision; deep learning; metastases; pancreatic cancer; pancreatic ductal adenocarcinoma; transfer learning

1. Introduction

In histopathological diagnostics, malignant neoplasms are detected and classified based on the analysis of microscopic tissue slides stained with hematoxylin and eosin (H&E) under a bright-field microscope. A precise classification of malignant neoplasms is pivotal for adequate patient stratification and therapy. In some cases, a histopathological diagnosis can be challenging, even when ancillary techniques for tissue characterization, such as immunohistochemistry (IH) or molecular analyses, are applied. Pancreatic ductal adenocarcinoma (PDAC) is a highly aggressive cancer type arising from the epithelial cells of the pancreatobiliary system. PDAC is usually recognized at an advanced stage [1] when it has already metastasized to the lymph nodes, peritoneum, liver or lungs [1,2]. Surgical resection is currently the only curative therapy for patients with PDAC. However, as the majority of patients present with locally advanced disease or distant metastases, there is a lack of effective treatment options [2,3]. In patients undergoing surgery, a definitive diagnosis of PDAC is achieved by a histopathological evaluation of surgical resection specimens. If a patient is not eligible for surgery, diagnostic confirmation is reached through a histopathological assessment of biopsy samples obtained during an endosonographic ultrasonography.

Deep neural networks can be used for the classification of images. Specifically, convolutional neural networks are multilayered and trained with a back-propagation algorithm to classify shapes [4]. In medicine, convolutional neural networks are used to classify images to predict clinical parameters and outcomes [5,6]. Deep neural networks can also be used to identify histological patterns [7]. Studies have shown that tissue sections from non-small lung cancer can be classified and their mutational profile predicted using deep transfer learning [8]. Patient outcomes can also be predicted from histology images. This has been demonstrated in studies of colorectal cancer [9,10], as well as for hepatocellular carcinoma patients following liver resection [11]. RNA-Seq profiles and prognostic features, such as microsatellite instability, can also be predicted from slide images of gastrointestinal cancers [12,13]. Importantly, using deep transfer learning of the model inception v3 and The Cancer Genome Atlas image database, most cancer types can be predicted from histological images [14]. One issue with data preparation for deep learning is that histological images are composed of multiple tissue components. Some datasets divide a histological image into subgroups, such as adipose tissue, mucosa and lymphoid tissue [10]. Currently, although a variety of networks are used for histological classification, including AlexNet, DenseNet, ResNet18, ResNet50, SqueezeNet, VGG-16 and VGG-19 [15–18], it is challenging to find a network with the ability to effectively filter out confounding histological tissue entities.

Using a new dataset consisting of healthy pancreases, healthy lymph nodes and PDAC, we show that histological material can be purified using two communicating neural networks, which we termed "Communicators". Based on an existing dataset, we added one class of our data which filtered the training data of Communicator 2. The purified dataset provided the training data for a convolutional neural network to classify these labels. The network was validated on further independent histological sections. Interestingly, the trained network was able to identify PDAC metastases in lymph nodes. Further, extensive hyperparameter testing suggests that the Resnet fine-tuned network with the ADAM Optimizer and a learning-rate of 0.0001 was efficient in this setting.

2. Materials and Methods

Patient Data: Histological images of PDAC and healthy pancreatic tissue were obtained from tissue micro arrays (TMAs) [19]. For the dataset, we used a cohort of well-characterized PDAC patients (n = 229). Two hundred and twenty-three PDAC tissue spots (one per patient) and 161 healthy pancreas tissue spots (one per patient) were used. A second anonymized TMA cohort contained healthy lymph node samples (n = 78), of which 76 spots were used (Supplementary Table S1). All tissue samples were obtained from patients who underwent surgical cancer resection at the University Hospital of Düsseldorf, Germany. Additionally, a third cohort contained whole-slide tissue images with different tissue types for validation. We used four evaluation sets with 10 patients: PDAC consisting of 15 images, healthy pancreas (HP) consisting of 3 images, lymph node (LN) with PDAC having 6 images, and healthy lymph node (HLN) with 5 images. To establish adequate ground truth for validation, the digitalized whole-slide images were annotated manually on the *regional level*, distinguishing healthy pancreas, normal lymphatic tissue, PDAC, adipose tissue and other "background tissues", such as blood vessels [20].

Tissue acquisition and preparation: Tissue samples were acquired from the routine diagnostic archive of the Institute of Pathology, Düsseldorf, Germany. All tissue samples were fixed in 4% buffered formaldehyde and embedded in paraffin blocks. For the preparation of tissue microarrays (TMAs), samples with a 1-mm core size from primary tumors (PDAC), lymph node metastases and corresponding normal tissue were selected and assembled into the respective TMA (Manual Tissue Arrayer MTA-1, Beecher Instruments, Inc., Sun Prairie, WI, USA). Hematoxylin & eosin staining was prepared from 2-µm thick tissue sections of the TMA blocks and whole-slide tissue blocks according to the protocol established in the routine diagnostic laboratory of the Institute of Pathology of Düsseldorf, Düsseldorf, Germany.

2.1. Digitalization of H&E Tissue Slides

H&E tissue slides were digitalized using the Aperio AT2 microscopic slide scanner (Leica Biosystems, Wetzlar, Germany). H&E slides were scanned using either the 40× magnification (TMA slides) or the 20× magnification (whole-tissue slides). Microscopic image files were saved as Aperio ScanScope Virtual Slide (.SVS) files and displayed using Aperio ImageScope software 12.3.3 (Leica Biosystems, Wetzlar, Germany). Tissue spots were extracted from the TMAs using the Aperio Imagescope software. The images were resized to 50% of the pixel size with Image Resizer for Windows (version 3.1.1.) when scanned with a 40× magnification. In addition to tissue slides acquired as described above, we also obtained a previously described dataset composed of the following tissue type: adipose tissue (ADI, 10.407 images) [10].

2.2. Deep Transfer Learning

The preprocessing pipeline included a 50% zoom on Unpatched Images, and normalization [21]. Images were dissected into image tiles, fitting the input size of the neural networks.

Architecture: We used a deep transfer learning approach for the network architecture [22]. We chose to fine-tune and adapt the residual neural network Resnet18 [23], as previously described [24]. In addition to the transformations, we added a Gaussian Blur for training as augmentation. We retrained the last three layers of the Resnet18. Adam [25] was used as the optimizer for this deep transfer learning approach. A square image patch size of 224 pixels was used. We trained the network with the batch sizes of 150 and 100 epochs, early stopping of 5 on the images of 80% of the samples from the dataset using the pathologist's label as ground truth. We balanced the dataset by random doubling of the images in the underrepresented classes. The predicted probability for each image patch to contain each of the labels (HLN, HP, PDAC, ADI, BG) was used as the objective/loss function (Cross Entropy Loss) in the training. We used an initial learning rate of 0.0001 and a decrease by 5% every five epochs. Evaluation was carried out by applying the previously

trained model to the remaining, previously unseen 20% of the dataset for each sequence set separately and comparing the results with the ground truth. In addition to the accuracy, we calculated the confusion matrix, the precision, recall, Jaccard index and the F1-score for each class. We used early stopping, based on the loss of the validation learning, with early stopping equaling 5 [26]. For further evaluation, the algorithm classified the tissue type by patch labeling of separate validation images. For visualization, we colored each image patch in the color of the predicted class.

Metrics: For comparison and evaluation of our models, we used the following five metrics; metrics for the binary case are shown.

The scores of the metrics are in the Interval (0,1) and, therefore, the greater the score, the better.

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

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

$$F1 - Score = \frac{2TP}{2TP + FP + FN}$$

$$Jaccard - Score = \frac{TP}{TP + FP + FN}$$

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

For the multiclass (non-binary) case, the positive is the target class and the other classes are the negative class. With this definition, separate metrics were obtained for TP, FP, TN and FN.

Classification score vector: A classification score summing up the classification labels of each patch of an image and pointing to the percentual portion of this class was determined.

$$c := (c_1, \ldots, c_i, \ldots, c_N),$$

where $i \in 1, \ldots, N$ and N is the number of classes. With the definition of the patch vector

$$p := (p_1, \ldots, p_j, \ldots, p_M),$$

where $j \in 1, \ldots, M$ and M is the number of patches for this image. Then we defined

$$c_i := \frac{\sum_{j=1}^{M} 1_{f(p_j)=i}}{\sum_{j=1}^{M} 1},$$

where f is the prediction function of the neural network. The dominator guarantees that the sum of the vector entries is equal to one. We used the classification vectors to determine the image label by argmax of the patch labels, if not otherwise stated.

Three score: For a second score to rank the networks, we calculated the percentages of the right label prediction. The average of image tiles of a group (HLN, HP, PDAC) on the test data was determined.

Four score: A segmentation tool was used to rank different networks on the validation dataset, by a pathologist [27]. The average of image tiles of a group (HLN, HP, PDAC and LNPM) was determined and compared to the pathologist's label as ground truth. Images with insufficient labeling were excluded, as indicated.

The Four score is defined by

$$fourscore = 1 - \frac{1}{4} \sum_{1}^{4} m_i$$

Cancers **2022**, *14*, 1964

where the m_is are given by

$$m_i = \text{abs}\left(\sum_{j=1}^{N_i} c_i^{(j)} - p_i\right),$$

where the $c_i^{(j)}$ is ith entry of the classification vector for the jth extern validation image, and p_i is the average over all images of one class prediction, annotated by the pathologists. The m_is are called HLN-score, HP-score, PDAC-score and LNPM-score.

2.3. Software & Hardware

Training and validation was performed on a Nvidia A100 of the high performance cluster (HPC, Hilbert) of the HHU, and on Quadro T2000 with Max-Q Design (Nvidia Corp., Santa Clara, CA, USA), depending on the computational power needed.

On the workstation, we used the Python VERSION:3.8.8 [MSC v.1916 64 bit (AMD64)] software (pyTorch VERSION:1.9.0.dev20210423, CUDNN VERSION:8005). On the high-performance cluster we used the following software: Python VERSION:3.6.5 [GCC Intel(R)\\ C++ gcc 4.8.5 mode] (including pyTorch VERSION:1.8.0.dev20201102+cu110, CUDNN VERSION:8004).

3. Results

3.1. Communicating Neural Networks Enrich New Datasets for Parenchymal Tissue

To investigate whether PDAC can be detected by convolutional neural networks, we obtained histological images of healthy pancreatic tissue, healthy lymph node (HLN) tissue and pancreatic ductal adenocarcinoma (PDAC) tissue. Each tissue section was extracted from scanned images of tumor microarrays (TMAs) for further data preprocessing (Figure 1a, Supplementary Figure S1). However, tissue samples and, consequently, histological images did not contain only image tiles attributed to their respective label. Specifically, adipose tissue was observed in some images (Figure 1b). Furthermore, artefacts could be observed in tissue images from TMAs (Figure 1c). Accordingly, when tissue sections were dissected into 224 × 224-pixel image squares to match the size of the input layer of the convolutional neural network ResNet18 [24], the image tiles showed a variety of tissue identities, including adipose tissue and background, which did not match the respective label (Figure 1d). Overall, we obtained 17,842 image patches for HP, 9954 patches for HLN tissue and 25,650 patches for PDAC. We therefore speculated that the ground truth was not ideal in this setting, necessitating further data preprocessing.

To purify the image tiles within each label, we made use of deep transfer learning on the ImageNet database's pretrained network, ResNet18 [22,23]. Specifically, an existing dataset containing labeled image tiles of adipose tissue was associated with tiles from 20 images of a new dataset class labeled Data A_i [10,28]. Since tissue sections were obtained from different image slides, we normalized the H&E staining intensity on image tiles, as previously described (Figure 2a) [21]. This dataset was used to train Communicator 1, which then removed image tiles from 20 different images of the new dataset class that were not classified as the new dataset class, resulting in a dataset labeled Data B_i (i-th iteration of the process) (Figure 2b). The selected Data B_i image dataset was used along with the existing dataset for the training of Communicator 2 (Figure 2b). Communicator 2 removed confounding images from the dataset Data A_i images, resulting in an improved dataset Data A_{i+1} (Figure 2b). This process was repeated through several cycles, i, to remove other tissue types, such as adipose tissue from the new datasets. Using this process, we reduced the number of tiles for the labels and purified the ground truth (Figure 2c). Notably, other network architectures, such as VGG11 or Densenet, can also be used for communicator-based purification of dataset classes (Supplementary Figure S2). The final Resnet18 communicators were used to remove all image tiles that were not classifiable on all input data images with a threshold of 0.55 on the softmax output. Since processing via the communicators relied on the normalization of image tiles to make use of a labeled dataset, we mapped image tiles to tiles generated from images which were normalized

Cancers **2022**, *14*, 1964

in toto (Figure 2d). Image tiles related to tiles the communicators labeled as adipose tissue or background were moved into a new dataset class (Figure 2d). Accordingly, the clean-up process through the communicators resulted in 13,261 image tiles for the healthy pancreas, 19,313 image tiles for PDAC, 8264 image tiles for HLN, 9952 images tiles for BG and 1235 image tiles for ADI (Figure 2e). Notably, the tissue patches selected by the communicators were not homogeneous as, for instance, the class label PDAC also included cancer-associated stromata, and inflamed/necrotic tissue (Figure 2e).

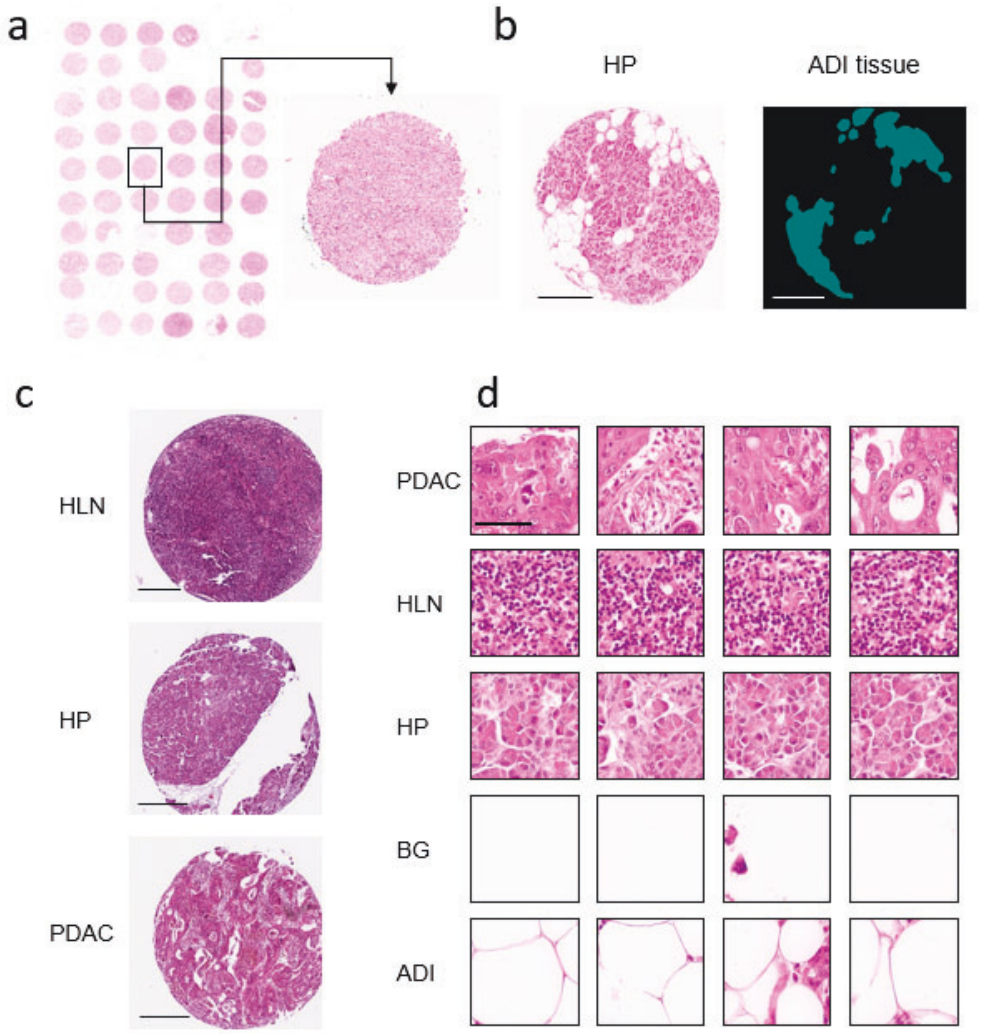


Figure 1. Data Pre-processing Pipeline for H&E-stained Tissue Micro Arrays provide datasets for Deep Transfer Learning. (**a**) Patients' data were extracted from tissue micro arrays (TMAs) and annotated. (**b**) A representative HP Spot with adipose tissue and a segmentation of the adipose tissue are shown (scale bar = 300 µm). (**c**) Spots from healthy lymph node (HLN), healthy pancreas (HP) and pancreatic ductal adenocarcinoma (PDAC) (scale bar = 300 µm). (**d**) Whole images were cut into square patches with 224 × 224 pixel sizes (scale bar = 60 µm). PDAC, HLN, HP, Background (BG), and Adipose Tissue (ADI) sample image tiles are shown.

Cancers **2022**, *14*, 1964

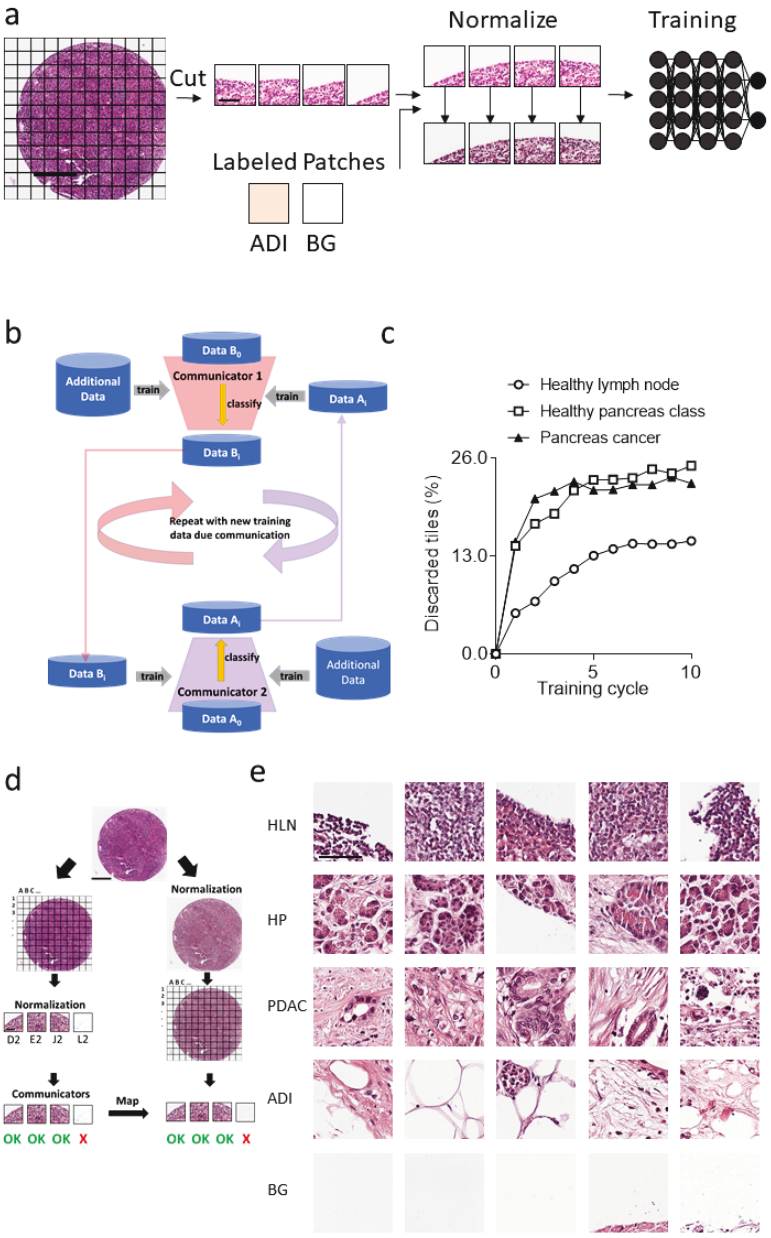


Figure 2. Data clean-up via communicators improves ground truth by introducing more labels. (**a**) The schematic view of preprocessing and training of the CNNs (Spot: Scale bar = 300 µm, Patch: Scale bar = 60 µm). (**b**) Schematic set up of the communicators used for data clean-up. (**c**) Percentage of discarded image patches of the different tissue types during the clean-up process from healthy lymph nodes, healthy pancreas and pancreatic ductal adenocarcinoma is indicated. (**d**) Selection of the normalized tissue patches based on the classification of the communicator CNNs is illustrated (Spot: Scale bar = 300 µm, Patch: Scale bar = 60 µm). (**e**) Representative communicators sorted tissue patches from three cleaned-up tissue classes and the two extracted new classes are shown (scale bar = 60 µm). Tissue patches of healthy lymph nodes (HLN), healthy pancreases (HP), pancreatic ductal adenocarcinoma (PDAC), background (BG), and adipose tissue (ADI) labels are presented.

Cancers **2022**, *14*, 1964

3.2. Dataset Clean-Up Improves Performance during Image Recognition

Next, we used the obtained image tiles for the retraining of a convolutional neural network. Hence, the patient cohort was divided into training (80%), validation (10%), and test (10%) datasets. The image tiles in the different dataset groups were taken from different patients. Deep transfer learning was performed in retraining the last 3 blocks (18 layers) of the network ResNet18 using a learning rate of 0.0001, Adam loss function, and an early stopping of 5, as previously described [24]. The neural network trained on the raw dataset ((test: 1690, train: 14,450, val: 1702) image patches for healthy pancreases, (874, 8275, 805) patches for HLN tissue, (2454, 20,694, 2502) patches for PDAC) achieved a weighted accuracy over all classes of 90%, a weighted Jaccard score of 81% and a weighted F1-score of 90% (Figure 3a, Table 1). For the single classes, the F1-score was 86% (HP) and 92% (PDAC). The Jaccard score was 82% (HLN) and 85% (PDAC) (Figure 3a, Table 1).

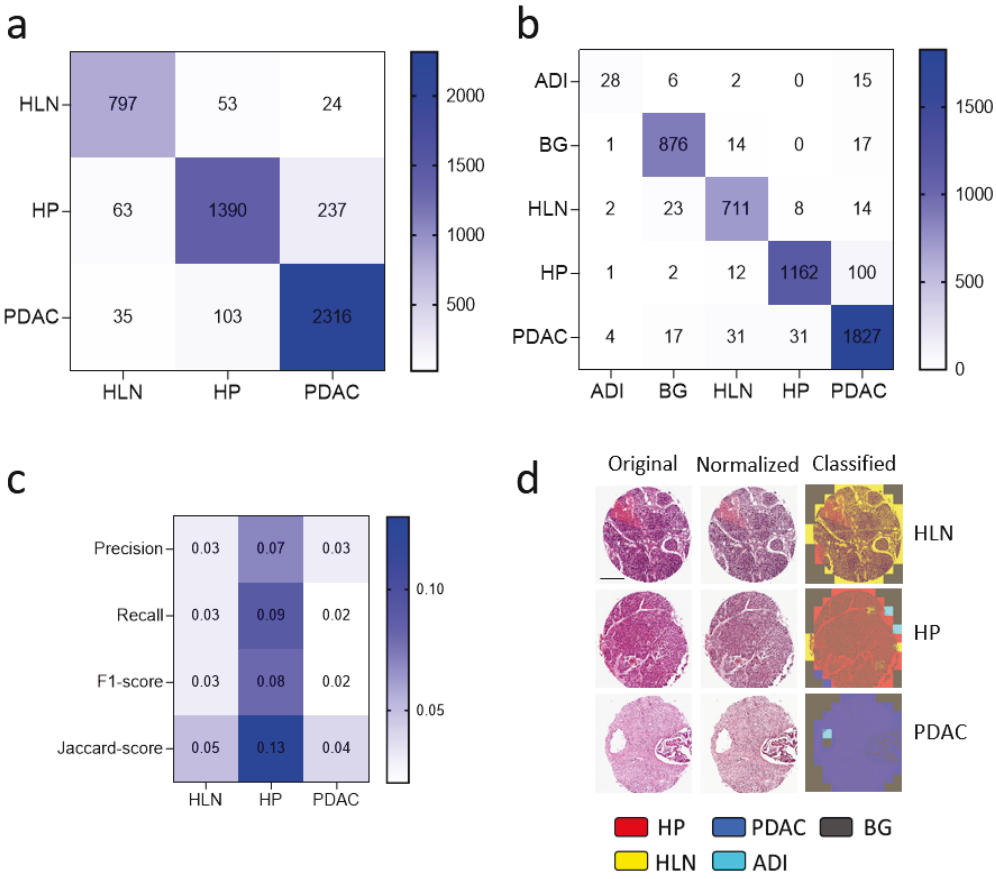


Figure 3. Data clean-up using communicators improves network's performance. Confusion Matrices from retraining the neural network with (**a**) original test data or (**b**) with data post-clean-up via communicators are shown. (**c**) Heatmap of the performance difference between the network trained on data with and without the data clean-up via the communicators is shown. (**d**) Representative test spots from healthy lymph nodes (HLN), healthy pancreases (HP) and pancreatic ductal adenocarcinoma (PDAC) classified with the neural network are shown (scale bar = 300 µm). HLN (yellow), HP (red), PDAC (blue), background (BG, grey) and adipose tissue (ADI, cyan) were predicted by the retrained CNN.

Cancers **2022**, *14*, 1964

Table 1. Metrics of the uncleaned network (ResNet18): Precision, Recall, F1-Score and Jaccard score for the classes healthy pancreas (HP), healthy lymph node (HLN) and pancreatic ductal adenocarcinoma (PDAC).

Class	Precision	Recall	F1-Score	Jaccard Score	Support
HLN	0.89	0.91	0.9	0.82	874
HP	0.9	0.82	0.86	0.75	1690
PDAC	0.9	0.94	0.92	0.85	2454
Accuracy			0.9		5018
Macro avg	0.9	0.89	0.89	0.81	5018
Weighted avg	0.9	0.9	0.9	0.81	5018

When we used the purified image data training set ((test: 1277, train: 10,767, val: 1217) image tiles for healthy pancreases, (1910, 15, 605, 1798) image tiles for PDAC, (758, 6848, 668) image tiles for HLN, (908, 7971, 908) image tiles for BG and (51, 1049, 135) image tiles for ADI), we observed an improvement in the confusion matrix (Figure 3b). Specifically, the neural network showed an increased performance for the HP class of the recall of 9% (up to 91%), a Jaccard score of 13% (88%) and F1-score of 8% (94%) (Figure 3c, Tables 1 and 2, Supplementary Tables S2 and S3). In addition, we visualized the patch-class labels in the tissue sections from the test dataset (Figure 3d). Notably, when we used the communicators for only 3 data clean-up cycles, we still observed an improved performance (Supplementary Table S3, Supplementary Figure S3). These data indicate that the neural network based on ResNet18 could be retrained to classify PDAC from images of the H&E slide sections. Furthermore, the performance was improved by dataset preprocessing involving two communicators that purified parenchymal image tiles.

Table 2. Metrics of the cleaned network (ResNet18): Precision, Recall, F1-Score and Jaccard score for the classes healthy pancreas (HP), healthy lymph node (HLN), pancreatic ductal adenocarcinoma (PDAC) and Adipose tissue (ADI).

Class	Precision	Recall	F1-Score	Jaccard	Support
ADI	0.78	0.55	0.64	0.47	51
BG	0.95	0.96	0.96	0.92	908
HLN	0.92	0.94	0.93	0.87	758
HP	0.97	0.91	0.94	0.88	1277
PDAC	0.93	0.96	0.94	0.89	1910
Accuracy			0.94		4904
Macro avg	0.91	0.86	0.88	0.81	4904
Weighted avg	0.94	0.94	0.94	0.89	4904

3.3. Convolutional Neural Networks (CNN) Classification of Histological Images of Primary Tumors and Lymph Node Metastases Can Be Improved through Hyperparameter Tuning during Training and Classification

To validate the retrained ResNet18, we used tissue sections of healthy pancreatic and PDAC tissue. Each image was normalized and divided into image tiles, which were classified according to the training labels (Figure 4a). The ground truth of this cohort was established by a pathologist, who labeled the histological images (Figure 4b). We noted that the majority of image tiles of histologically healthy pancreas tissue were labeled correctly. PDAC images were also correctly classified (Figure 4c–e). However, we also observed other classes appearing in healthy pancreas images (Figure 4c–e). This confusion likely resulted

Cancers **2022**, *14*, 1964

from other labels, including background, being present in pancreatic tissue that were not fed into the communicators.

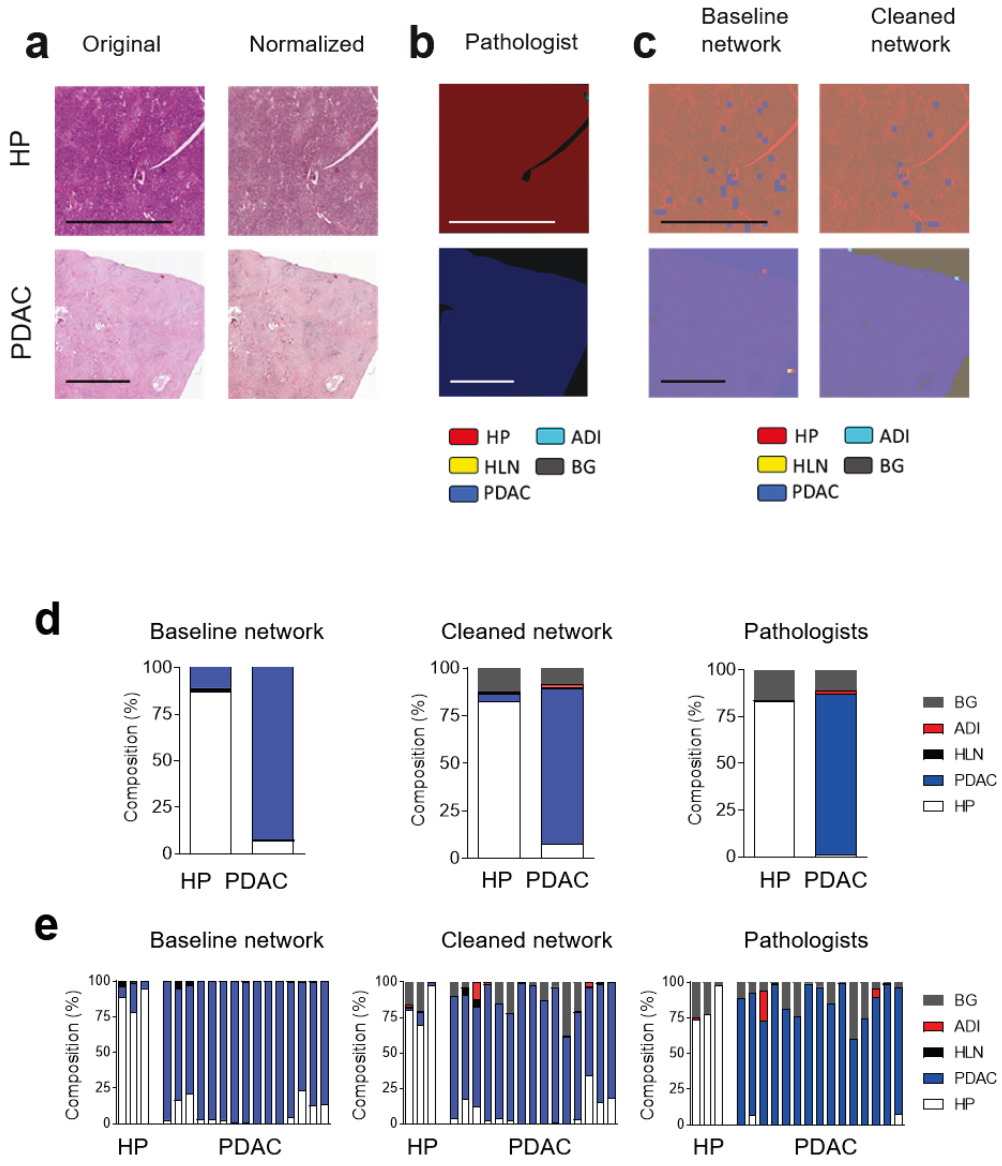


Figure 4. Convolutional Neural Network can classify healthy pancreas tissue and pancreatic ductal adenocarcinoma. (**a**) Sections from independent H&E-stained whole images from healthy pancreases (HP) and pancreatic ductal adenocarcinoma (PDAC) are shown (scale bar = 2 mm). (**b**) Expert label (ground truth), as determined by a pathologist and (**c**) classified with the baseline and cleaned network, are shown. HLN (yellow), HP (red), PDAC (blue), background (BG, grey) and adipose tissue (ADI, cyan) are indicated by the pathologist (**b**) or the CNNs (**c**). The pooled (**d**) and individual classification (**e**), as determined using a baseline and cleaned network, as well as by a pathologist, of whole-image slides from healthy pancreases (HP) (n = 3) and pancreatic ductal adenocarcinoma (PDAC) (n = 15) are shown.

Cancers **2022**, *14*, 1964

To further validate our findings, we classified images from healthy and PDAC metastatic lymph nodes (Figure 5a). To compare different CNNs, a pathologist labeled images from different tissue types (Figure 5b). Following normalization, the images were labeled by the network trained with the cleaned or uncleaned dataset (Figure 5c). As expected, following the dataset clean-up, labeling by the CNN better reflected the labeling done by the pathologist (Figure 5c). Although the HLN was detected, we found a considerable amount of misclassified image tiles (Figure 5d). However, when we analyzed these images using the CNN trained with the purified dataset, the labeling improved significantly (Figure 5d). Furthermore, in image data from PDAC metastatic lymph nodes, a proportion of image tiles was classified as PDAC (Figure 5d,e). Notably, a substantial amount of misclassified tiles in the baseline model was due to background that was not eliminated during the data preprocessing. To evaluate whether the communicators demonstrated a beneficial effect, we removed background tiles with a pixel cutoff at 239, thereby removing most of the image tiles (Supplementary Figure S4a). However, when we purified the dataset after the pixel cutoff via the communicators, we still found improved labeling with the cleaned-up network (Supplementary Table S3, Supplementary Figure S4b–d). These data show that the retrained ResNet18 can detect PDAC in primary tumors (Supplementary Figure S5) and lymph node metastases and that the data clean-up process via communicators improved the labeling of histological images.

To investigate whether different models or hyperparameters affected the CNNs' performance, we trained 72 networks based on different network architectures, including ResNet18 [23], ResNet50 [23], ResNet101 [23], Vgg-16 [29], Vgg-19 [29], Alexnet [30], DenseNet [31] and SqueezeNet [32]. We also performed the training using different learning rates (ranging from 10^{-4} to 10^{-6}) and optimizers (SGD, Adam [25], RMSprop). We evaluated the accuracy, Jaccard Score, F1-Score, and the classification of HP tissue, PDAC, HLN tissue and PDAC metastatic lymph nodes on independent images. The results of the networks were compared to the ground truth based on labeling by a pathologist (Figures 4b and 5b, Supplementary Table S4). As expected, the networks showed a wide variety of performances dependent on the different training parameters (Figure 6a). The best performance in this setting was seen in the Resnet_1 network, which had a four-score of 97.8% of the pathologist's labeling (Figure 6a, Supplementary Table S4). We observed a clear correlation between the performance on the test dataset and the independent validation dataset (Figure 6b). The different model architectures achieved a better performance with different optimizers (Figure 6c). While all network architectures were able to classify the validation images (Figure 6d), a clear dependence of the performance was associated with the learning rate (Figure 6d). Notably, a learning rate of 10^{-6} was not preferable in this setting compared to the other values (Figure 6d). Different models demonstrated different performances, and the gap to the annotated labels from the pathologist shows the performance as measured by the components of the four-score (Figure 6e). Taken together, these data indicate that dataset preprocessing, image classification stratification, and hyperparameter tuning can have an impact on the recognition of PDAC in lymph node tissue from H&E images.

3.4. Communicator Based Preprocessing Can Be Transferred to Other Input Sizes

Next, we wondered whether we could use the data preprocessing to purify the dataset for CNNs using another input size. We hypothesized that by using the clean-up process with the $224 \times 224 \times 3$ labeled image dataset, we could extract a cleaned $299 \times 299 \times 3$ image tile dataset needed to train an inceptionv3 CNN [33]. Specifically, we mapped the $299 \times 299 \times 3$ image tiles and classified a cropped section ($224 \times 224 \times 3$) via the communicators (Figure 7a). The labels were transferred to normalized image tiles to establish an improved ground truth (Figure 7a). The performance of the cleaned-up inceptionv3 CNN was increased compared to the baseline model (Figure 7b, Supplementary Table S3). Furthermore, the communicator preprocessed network was able to better label the independent validation dataset when compared to the baseline model (Figure 7c–e). These

Cancers **2022**, *14*, 1964

data indicate that a communicator-based clean-up process can potentially be transferred to CNNs with unmatching input sizes.

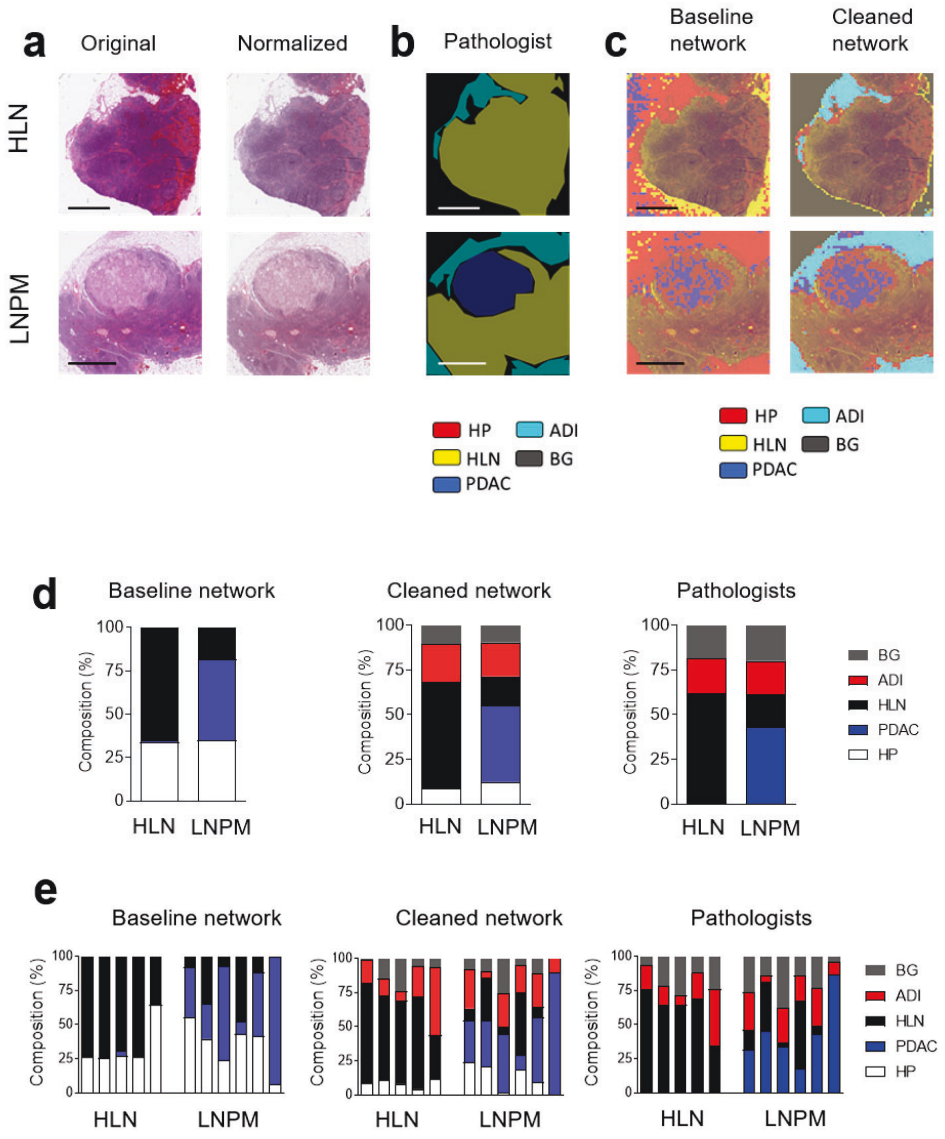


Figure 5. Convolutional Neural Network can classify metastases from pancreatic ductal adenocarcinoma in lymph nodes. (**a**) Sections from H&E-stained whole images from healthy lymph nodes (HLN) and lymph nodes with pancreatic ductal adenocarcinoma metastases (LNPM) are shown (scale bar = 2 mm). (**b**) Expert label (ground truth) as determined by a pathologist and (**c**) classified with the baseline and cleaned network are shown. HLN (yellow), HP (red), PDAC (blue), background (BG, grey) and adipose tissue (ADI, cyan) are indicated by the pathologist (**b**) or the CNNs (**c**). The pooled (**d**) and individual classification (**e**), as determined using a baseline and cleaned network, as well as by a pathologist, of whole-image slides from healthy lymph nodes (HLN) (n = 5) and lymph nodes with pancreatic ductal adenocarcinoma metastases (LNPM) (n = 6) are shown (scale bar = 2 mm).

Cancers **2022**, *14*, 1964

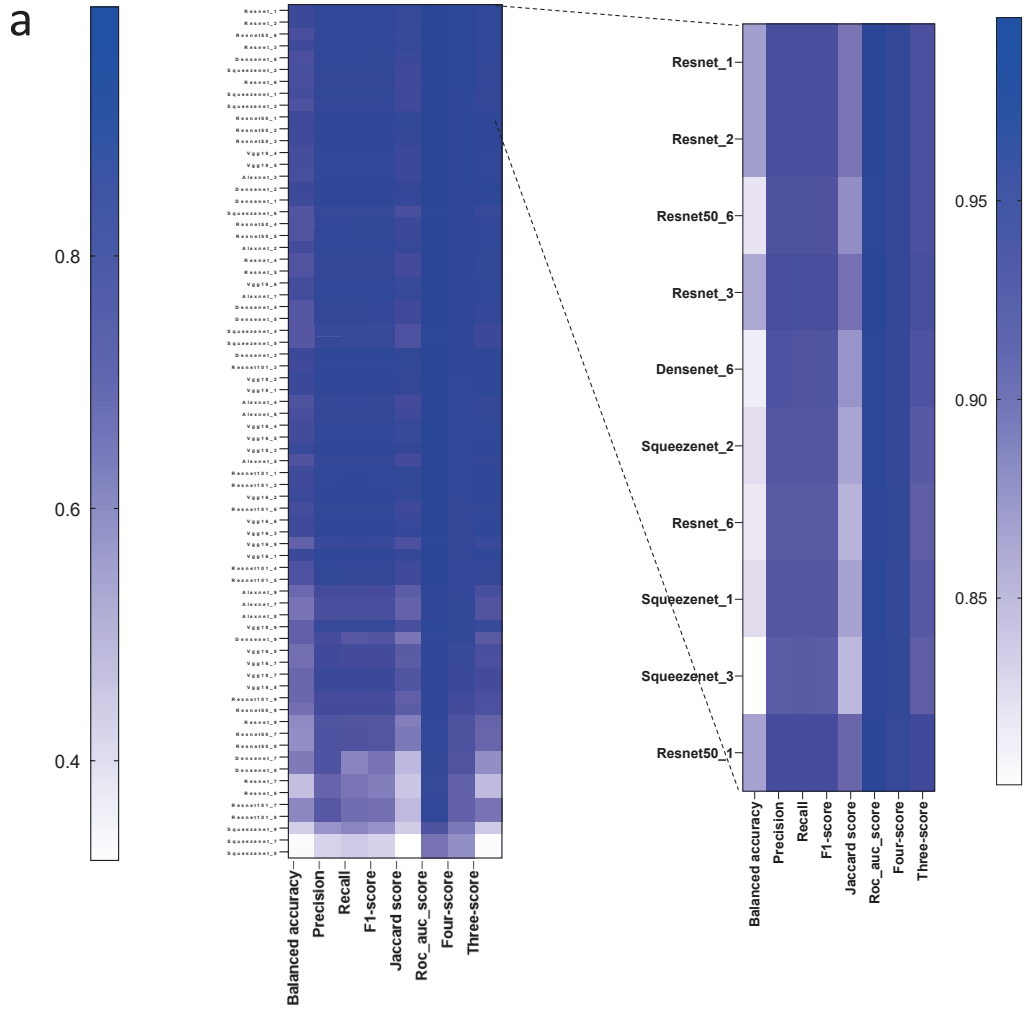


Figure 6. *Cont.*

Cancers **2022**, *14*, 1964

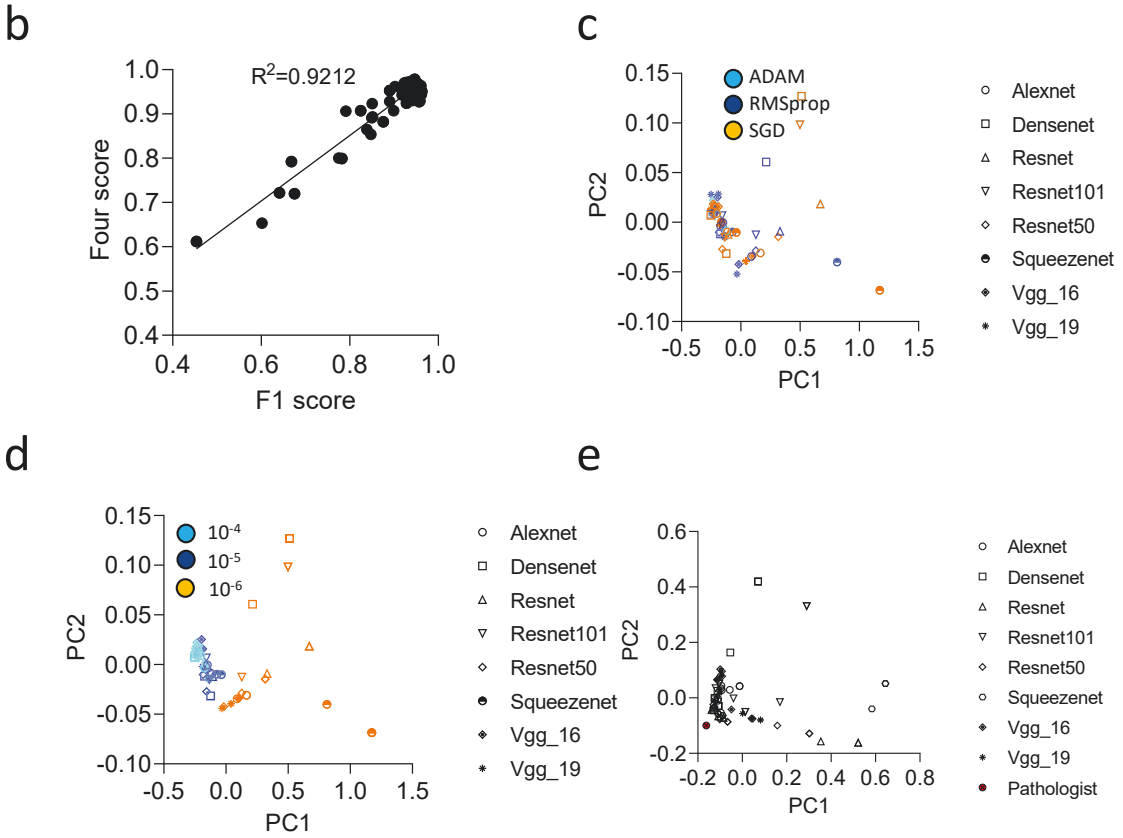


Figure 6. Hyperparameter Tuning illustrates the performance of different network architectures trained with variable learning rates and optimizers. (**a**) Performance of 72 trained and validated neuronal networks were ranked regarding the four-score, highlighting the best 10 network configurations. (**b**) Correlations were tested between F1-score and four-score via r2-score. (**c**,**d**) PCA (linear kernel) of the network metrics from hyperparameter tuning colored by the (**c**) different optimizers and architectures, (**d**) learning rates and architectures, and (**e**) PCA (linear kernel) of the modified four-score parts: HLN-score, HP-score, PDAC-score and LNMP-score (vs. the pathologist annotations over all 29 validation images).

Cancers **2022**, *14*, 1964

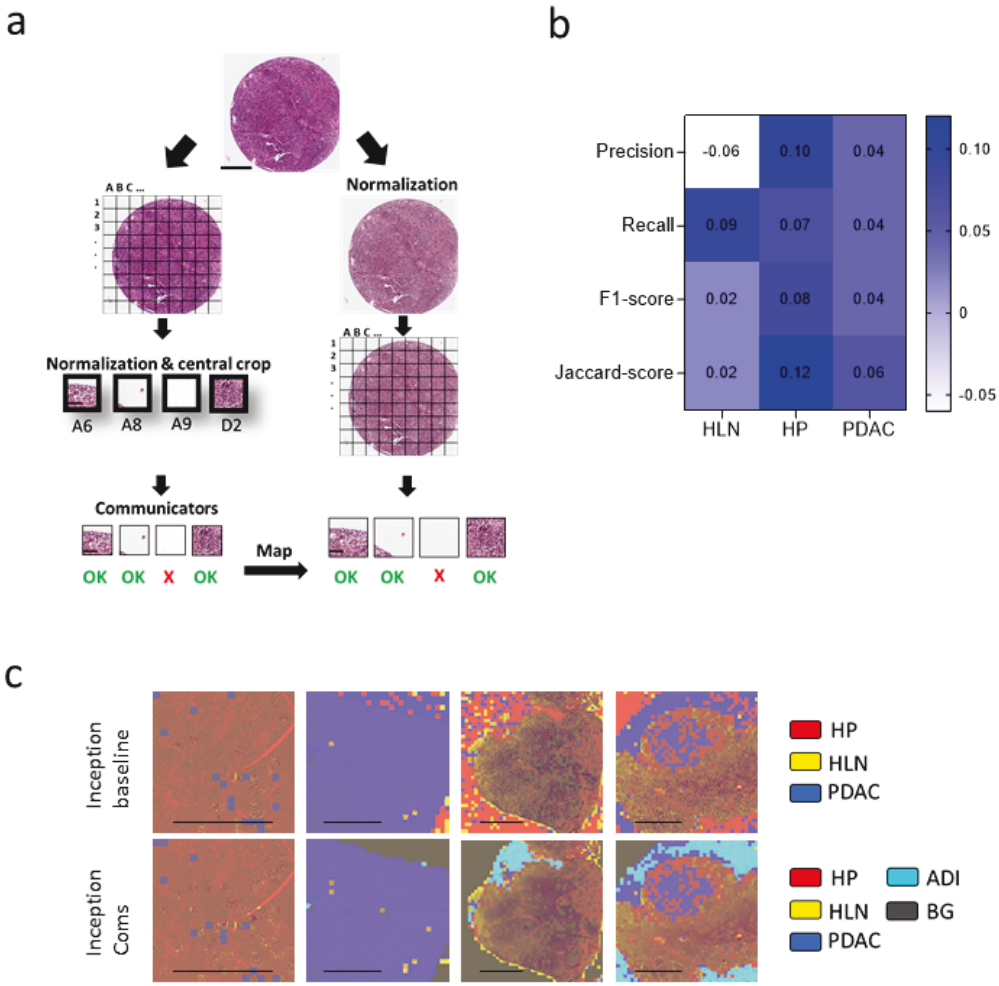


Figure 7. *Cont.*

Cancers **2022**, *14*, 1964

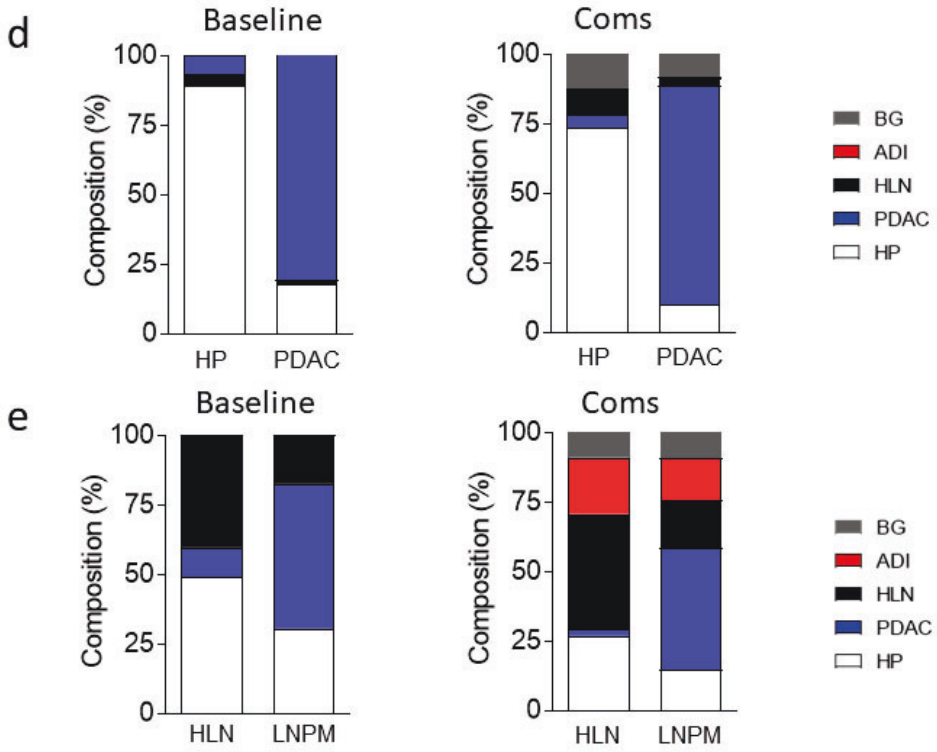


Figure 7. Communicator-based clean-up can be transferred to CNNs using a different input size. (**a**) Selection of the normalized 299 × 299 × 3 tissue patches based on the classification of the communicator CNNs using a 224 × 224 × 3 crop of the image tiles is illustrated (Spot: Scale bar = 300 µm, Patch: Scale bar = 60 µm). (**b**) Heatmap of the performance difference between the inceptionnetv3 trained on data with and without the data clean-up via the communicators is shown. (**c**) Visualization of the classified validation data by the baseline or cleaned-up inceptionv3 net. Healthy lymph node (HLN, yellow), healthy pancreas (HP, red), pancreatic ductal adenocarcinoma (PDAC, blue), background (BG, grey) and adipose tissue (ADI, cyan) classification is illustrated (scale bar = 2 mm). (**d**) Pooled classification, as determined using a baseline and cleaned inceptionv3 network from healthy pancreas (HP) (n = 3) and pancreatic ductal adenocarcinoma (PDAC) images (n = 15), is illustrated. (**e**) Average of classification, as determined using a baseline and cleaned inceptionv3 network of images showing healthy lymph nodes (HLN) (n = 5) and lymph nodes with pancreatic ductal adenocarcinoma metastasis (LNPM) (n = 6), is presented (ADI= adipose tissue, BG = background).

4. Discussion

In the current investigation, we show that PDAC can be detected with the help of convolutional neural networks using deep transfer learning. We introduced a dataset preprocessing step to purify dataset classes according to new labels via two communicators. As a result of this purification step, we increased the ground truth and, therefore, the performance of image classification on an independent validation dataset. Furthermore, we titrated several networks and hyperparameters to optimize their performance.

In daily diagnostic practice, carcinomas are classified on the basis of their characteristic histomorphology and immunohistochemical marker profiles. While different cancer types can be distinguished by deep learning algorithms based on data retrieved from the Cancer Genome Atlas [14], the diagnosis of PDAC metastases can be challenging due to overlapping features with other entities, such as biliary cancer. Here, we show that, based on a dataset of 460 tissue spots (223 PDAC, 161HP, 76 HLN), tissue entities could be correctly

labeled in images from independent tissue sections. The short time taken to classify an image might be useful to potentially aid pathologists during tissue evaluation. If several cases/slides have to be evaluated, the algorithm could potentially be employed to highlight areas of interest for the pathologist. This could be achieved, for example, by annotating the cases/slides and/or by flagging unclear cases. For example, the algorithm could flag areas of interest (i.e., areas of suspected cancer infiltration, e.g., in lymph nodes) that should be examined first by the pathologist. It is also imaginable that the algorithm could be exploited to aid pathologists with measurements (e.g., measuring the diameter of tumor formations, or measuring distances from the tumor to resection margins). However, it remains essential that a trained pathologist examines histopathological images and makes decisions involving the diagnosis, treatment regimens, and prognosis. Deep-learning-based algorithms carry the risk of methodical biases, such as overfitting, imperfect ground truth, variation in reproducible staining patterns, and confusion with untrained tissue types. Moreover, installation costs, such as histological slide digitalization and computational capacities, apply, although, overall, the use of machine learning algorithms is cost-effective. In the current state, our algorithm and the underlying program needs further development before being potentially applied for clinical use. Future development using data from large multicentered cohorts with solid labeled ground truths might improve CNNs in their role to help in the classification and quantification of histopathological images. The question of whether the described communicator approach can help with establishing a ground truth also in other datasets, including for different cancer types, needs more exploration. For a training dataset with more class labels, for example, a cancer-associated stroma or inflamed/necrotic tissue, the clean-up process could be potentially further improved. Although biopsy samples enable pathologists to make a definite diagnosis in most cases, contexts in which a primary tumor cannot be determined are known to exist both in PDAC diagnostics and in the diagnostics of other tumors [34]. Therefore, future studies should also focus on where the gaps are and which type of diagnostic-setting deep-learning-based algorithms can best be used to maximize its utility. Furthermore, whether the communicator approach can be used for other cancer identities or detect cancer tissue in different organs needs to be further evaluated. In principle, the data clean-up procedure can be transferred to different tasks. However, whether other cancer types can benefit from the use of communicator-based pre-processing needs to be shown for individual datasets to support this speculation.

Importantly, we demonstrate the ability to correctly classify image tiles derived from healthy or metastatic lymph node tissues. However, we also observed a proportion of mislabeled image tiles in these datasets. Specifically, these areas showed other tissue types, such as vasculature, which caused confusion in the labeling network. This indicates that further datasets are required to increase the performance of neural networks and that therapeutic decisions, ultimately, are dependent on the physicians.

Dataset purification can improve the performance of convolutional neural networks. Digital pathology can assist pathologists with classifying histopathological images [35]. These networks are trained on large datasets from various public sources, including PubMed and The Cancer Genome Atlas [14,35]. However, automated software-supported analysis of histological slides is often hampered by the presence of different tissue types on the histology slide. Hence, dataset preprocessing can help to increase the quality of the ground truth. In this study, we used an existing dataset containing adipose tissue to eliminate tissue tiles from our new dataset [10]. This was performed using two communicators, which cleaned up the dataset in cycles. The purified dataset could improve the performance of the convolutional neural network. This automated process might be useful to identify and label pathologic tissue identities. The correct identification of adipose tissue in particular is an important aspect of the deep-learning-based analysis of histological slides. Locally advanced invasive cancer will often infiltrate organ-surrounding adipose tissue. In order to use deep-learning-based analyses of histologic slides to determine classical prognostic parameters, such as the tumor diameter or the minimal distance of the

tumor to the resection margins, a precise distinction between tumor tissue and adipose tissue is crucial. This distinction between tumor and fatty tissue is also important in the detection of the extracapsular extension of lymph node metastases into the lymph-node-surrounding adipose tissue, which has been shown to be a prognostic factor in various solid cancers [36–38]. Other, more experimental approaches, such as the detection of so-called Stroma AReactive Invasion Front Areas (SARIFA) as a potential prognostic factor in gastrointestinal cancers, also strongly depend on the distinction between the tumor's invasive front and its inconspicuous surrounding fatty tissue [39]. Whether deep-learning-based algorithms and the communicator-based approach can be successfully used to aid in the distinction between adipose and tumor tissue remains to be determined.

Hyperparameter tuning can determine the performance of neural networks. A variety of convolutional neural networks are used to analyze histological images. Specifically, a ResNet-50 architecture was used to classify large histological datasets [35]. Furthermore, other architectures, including GoogLeNet, AlexNet, and Vgg-16, were successfully used for classifying histopathological images [40]. Since all these network architectures share the same input size of 224 × 224, our hyperparameter tuning was focused on these models. Our data show that several convolutional neural networks were able to distinguish between PDAC, healthy lymph nodes, adipose tissue and healthy pancreas tissue. However, when we tested several networks and the hyperparameters during training, we found that VGG19 with a learning rate of 10^{-5} and ADAM as an optimizer was ideal for our task. Future studies should investigate whether these differences are task specific. Notably, the use of inception v3, which performed very well in other tasks using H&E tissue sections [8,14], relies on an input data size of 299 × 299. Using the communicator approach, the ground truth was improved by transferring the image tile classification of the communicators to 299 × 299 image tiles.

5. Conclusions

In conclusion, our study shows that dataset preprocessing via two communicators and hyperparameter tuning can improve classification performance to identify PDAC on H&E tissue sections. Further studies applying this approach to metastases from different primaries are needed for validation.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/cancers14081964/s1: Supplementary Figure S1: Tissue Micro Arrays enable staining and presentation of multiple patient tissue sections on one histological slide.; Supplementary Figure S2: Percentage of discarded image patches of the different tissue types during the cleanup process; Supplementary Figure S3: ComCylce3 on the extern validation data shows improvement with only 3 cycles; Supplementary Figure S4: Baseline and Coms with Pixelcutoff instead of background class. Coms still outperform Baseline on the extern validation data with $n = 10$ Cycles for the Communicators; Supplementary Figure S5: Receiver operating characteristic for the different tissue classes; Supplementary Table S1: Patients Data; Supplementary Table S2: Differences of the metrics of the cleaned and uncleaned network; Supplementary Table S3: Overview of the seven CNN configuration used for the experiments; Supplementary Table S4: Network parameters and metrics from the 72 nets from Hyperparameter-Tuning are shown.

Author Contributions: Conceptualization, I.E. and P.A.L.; methodology, R.M.K., M.P., L.H., H.C.X., K.S.K., M.S., A.A.P., K.S.L., I.E. and P.A.L.; software, R.M.K.; validation, L.H. and I.E.; formal analysis, R.M.K. and P.A.L.; investigation, R.M.K., M.P., L.H., I.E. and P.A.L.; resources, I.E. and P.A.L.; data curation, R.M.K., M.P., L.H., M.S. and P.A.L.; writing—original draft preparation, R.M.K.; writing—review and editing, R.M.K., M.P., L.H., T.R., A.A.P., K.S.L., I.E. and P.A.L.; visualization, R.M.K., M.P., H.C.X. and K.S.K.; supervision, I.E. and P.A.L.; project administration, R.M.K. and P.A.L.; funding acquisition, I.E. and P.A.L. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the Deutsche Forschungsgemeinschaft (DFG, SFB974, GRK1949), the Jürgen Manchot Graduate School (MOI IV), the Forschungskommission (2021-41), and Anton-Betz Foundation (22/2020).

Institutional Review Board Statement: The use of human tissue samples was approved by the local ethics committee at the University Hospital of Düsseldorf, Germany (study numbers 3821 and 5387).

Informed Consent Statement: All human material used in this study is from the diagnostic archive of the Institute of Pathology at the University Hospital of Düsseldorf. Samples were acquired for diagnostic purposes, patients underwent no additional procedure. Samples are no longer needed for diagnostic purposes and may therefore be used for research purposes as permitted by the vote by the local ethics board cited in the paper. All patient data were anonymized.

Data Availability Statement: The source code is available at: https://github.com/MolecularMedicine2/pypdac (accessed on 13 March 2022).

Acknowledgments: Computational infrastructure and support were provided by the Center for Information and Media Technology at the Heinrich Heine University Düsseldorf. We acknowledge support by the Open Access Publication Fund of the University of Duisburg-Essen. We acknowledge the support by the Biobank of the University Hospital of Düsseldorf.

Conflicts of Interest: The authors declare no conflict of interest.

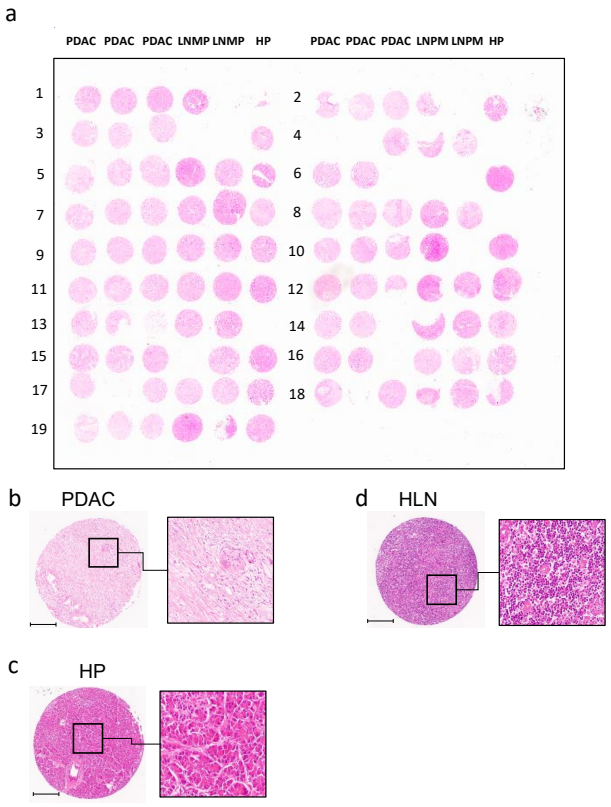
References

1. Ryan, D.P.; Hong, T.S.; Bardeesy, N. Pancreatic adenocarcinoma. *N. Engl. J. Med.* **2014**, *371*, 1039–1049. [CrossRef] [PubMed]
2. Park, W.; Chawla, A.; O'Reilly, E.M. Pancreatic Cancer: A Review. *JAMA* **2021**, *326*, 851–862. [CrossRef] [PubMed]
3. Orth, M.; Metzger, P.; Gerum, S.; Mayerle, J.; Schneider, G.; Belka, C.; Schnurr, M.; Lauber, K. Pancreatic ductal adenocarcinoma: Biological hallmarks, current status, and future perspectives of combined modality treatment approaches. *Radiat. Oncol.* **2019**, *14*, 141. [CrossRef] [PubMed]
4. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
5. Zhang, K.; Liu, X.; Shen, J.; Li, Z.; Sang, Y.; Wu, X.; Zha, Y.; Liang, W.; Wang, C.; Wang, K. Clinically Applicable AI System for Accurate Diagnosis, Quantitative Measurements, and Prognosis of COVID-19 Pneumonia Using Computed Tomography. *Cell* **2020**, *181*, 1423–1433.e1411. [CrossRef]
6. Harmon, S.A.; Sanford, T.H.; Xu, S.; Turkbey, E.B.; Roth, H.; Xu, Z.; Yang, D.; Myronenko, A.; Anderson, V.; Amalou, A. Artificial intelligence for the detection of COVID-19 pneumonia on chest CT using multinational datasets. *Nat. Commun.* **2020**, *11*, 4080. [CrossRef]
7. Barisoni, L.; Lafata, K.J.; Hewitt, S.M.; Madabhushi, A.; Balis, U.G. Digital pathology and computational image analysis in nephropathology. *Nat. Rev. Nephrol.* **2020**, *16*, 669–685. [CrossRef]
8. Coudray, N.; Ocampo, P.S.; Sakellaropoulos, T.; Narula, N.; Snuderl, M.; Fenyö, D.; Moreira, A.L.; Razavian, N.; Tsirigos, A. Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning. *Nat. Med.* **2018**, *24*, 1559–1567. [CrossRef]
9. Mobadersany, P.; Yousefi, S.; Amgad, M.; Gutman, D.A.; Barnholtz-Sloan, J.S.; Vega, J.E.V.; Brat, D.J.; Cooper, L.A.D. Predicting cancer outcomes from histology and genomics using convolutional networks. *Proc. Natl. Acad. Sci. USA* **2018**, *115*, E2970–E2979. [CrossRef]
10. Kather, J.N.; Krisam, J.; Charoentong, P.; Luedde, T.; Herpel, E.; Weis, C.-A.; Gaiser, T.; Marx, A.; Valous, N.A.; Ferber, D. Predicting survival from colorectal cancer histology slides using deep learning: A retrospective multicenter study. *PLoS Med.* **2019**, *16*, e1002730. [CrossRef]
11. Saillard, C.; Saillard, C.; Schmauch, B.; Laifa, O.; Moarii, M.; Toldo, S.; Zaslavskiy, M.; Pronier, E.; Laurent, A.; Amaddeo, G.; et al. Predicting survival after hepatocellular carcinoma resection using deep-learning on histological slides. *Hepatology* **2020**, *72*, 2000–2013. [CrossRef] [PubMed]
12. Schmauch, B.; Romagnoni, A.; Pronier, E.; Saillard, C.; Maillé, P.; Calderaro, J.; Kamoun, A.; Sefta, M.; Toldo, S.; Zaslavskiy, M. A deep learning model to predict RNA-Seq expression of tumours from whole slide images. *Nat. Commun.* **2020**, *11*, 3877. [CrossRef] [PubMed]
13. Kather, J.N.; Pearson, A.T.; Halama, N.; Jäger, D.; Krause, J.; Loosen, S.H.; Marx, A.; Boor, P.; Tacke, F.; Neumann, U.P. Deep learning can predict microsatellite instability directly from histology in gastrointestinal cancer. *Nat. Med.* **2019**, *25*, 1054–1056. [CrossRef] [PubMed]
14. Noorbakhsh, J.; Farahmand, S.; Pour, A.F.; Namburi, S.; Caruana, D.; Rimm, D.; Soltanieh-Ha, M.; Zarringhalam, K.; Chuang, J.H. Deep learning-based cross-classifications reveal conserved spatial behaviors within tumor histological images. *Nat. Commun.* **2020**, *11*, 6367. [CrossRef] [PubMed]
15. Abbas, M.A.; Bukhari, S.U.K.; Syed, A.; Shah, S.S.H. The Histopathological Diagnosis of Adenocarcinoma & Squamous Cells Carcinoma of Lungs by Artificial intelligence: A comparative study of convolutional neural networks. *medRxiv* **2020**. [CrossRef]
16. Talo, M. Automated classification of histopathology images using transfer learning. *Artif. Intell. Med.* **2019**, *101*, 101743. [CrossRef]
17. Saxena, S.; Shukla, S.; Gyanchandani, M. Pre-trained convolutional neural networks as feature extractors for diagnosis of breast cancer using histopathology. *Int. J. Imaging Syst. Technol.* **2020**, *30*, 577–591. [CrossRef]

18. Wang, L.; Jiao, Y.; Qiao, Y.; Zeng, N.; Yu, R. A novel approach combined transfer learning and deep learning to predict TMB from histology image. *Pattern Recognit. Lett.* **2020**, *135*, 244–248. [CrossRef]
19. Haeberle, L.; Steiger, K.; Schlitter, A.M.; Safi, S.A.; Knoefel, W.T.; Erkan, M.; Esposito, I. Stromal heterogeneity in pancreatic cancer and chronic pancreatitis. *Pancreatology* **2018**, *18*, 536–549. [CrossRef]
20. Wahab, N.; Miligy, I.M.; Dodd, K.; Sahota, H.; Toss, M.; Lu, W.; Jahanifar, M.; Bilal, M.; Graham, S.; Park, Y. Semantic annotation for computational pathology: Multidisciplinary experience and best practice recommendations. *J. Pathol. Clin. Res.* **2021**, *8*, 116–128. [CrossRef]
21. Macenko, M.; Niethammer, M.; Marron, J.S.; Borland, D.; Woosley, J.T.; Guan, X.; Schmitt, C.; Thomas, N.E. A method for normalizing histology slides for quantitative analysis. In Proceedings of the 2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Boston, MA, USA, 28 June–1 July 2009; IEEE: New York, NY, USA, 2009; pp. 1107–1110.
22. Tan, C.; Sun, F.; Kong, T.; Zhang, W.; Yang, C.; Liu, C. A survey on deep transfer learning. In Proceedings of the International Conference on Artificial Neural Networks, Rhodes, Greece, 4–7 October 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 270–279.
23. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
24. Werner, J.; Kronberg, R.M.; Stachura, P.; Ostermann, P.N.; Müller, L.; Schaal, H.; Bhatia, S.; Kather, J.N.; Borkhardt, A.; Pandyra, A.A.; et al. Deep Transfer Learning Approach for Automatic Recognition of Drug Toxicity and Inhibition of SARS-CoV-2. *Viruses* **2021**, *13*, 610. [CrossRef] [PubMed]
25. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
26. Prechelt, L. Early Stopping—But When? In *Neural Networks: Tricks of the Trade*; Orr, G.B., Müller, K.-R., Eds.; Springer: Berlin/Heidelberg, Germany, 1998; pp. 55–69.
27. Wada, K. Labelme: Image Polygonal Annotation with Python. 2016. Available online: https://github.com/wkentaro/labelme (accessed on 1 November 2021).
28. Kather, J.N.; Halama, N.; Marx, A. 100,000 Histological Images of Human Colorectal Cancer and Healthy Tissue. Zenodo10. 2018. Available online: https://zenodo.org/record/1214456#.YlU2AMjMJPZ (accessed on 1 November 2021).
29. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
30. Krizhevsky, A. One weird trick for parallelizing convolutional neural networks. *arXiv* **2014**, arXiv:1404.5997.
31. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
32. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
33. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
34. Hashimoto, K.; Nishimura, S.; Ito, T.; Oka, N.; Akagi, M. Limitations and usefulness of biopsy techniques for the diagnosis of metastatic bone and soft tissue tumors. *Ann. Med. Surg.* **2021**, *68*, 102581. [CrossRef] [PubMed]
35. Schaumberg, A.J.; Juarez-Nicanor, W.C.; Choudhury, S.J.; Pastrian, L.G.; Pritt, B.S.; Prieto Pozuelo, M.; Sotillo Sanchez, R.; Ho, K.; Zahra, N.; Sener, B.D.; et al. Interpretable multimodal deep learning for real-time pan-tissue pan-disease pathology search on social media. *Mod. Pathol.* **2020**, *33*, 2169–2185. [CrossRef]
36. Amit, M.; Liu, C.; Gleber-Netto, F.O.; Kini, S.; Tam, S.; Benov, A.; Aashiq, M.; El-Naggar, A.K.; Moreno, A.C.; Rosenthal, D.I.; et al. Inclusion of extranodal extension in the lymph node classification of cutaneous squamous cell carcinoma of the head and neck. *Cancer* **2021**, *127*, 1238–1245. [CrossRef]
37. Gruber, G.; Cole, B.F.; Castiglione-Gertsch, M.; Holmberg, S.B.; Lindtner, J.; Golouh, R.; Collins, J.; Crivellari, D.; Thürlimann, B.; Simoncini, E.; et al. Extracapsular tumor spread and the risk of local, axillary and supraclavicular recurrence in node-positive, premenopausal patients with breast cancer. *Ann. Oncol.* **2008**, *19*, 1393–1401. [CrossRef]
38. Luchini, C.; Fleischmann, A.; Boormans, J.L.; Fassan, M.; Nottegar, A.; Lucato, P.; Stubbs, B.; Solmi, M.; Porcaro, A.; Veronese, N.; et al. Extranodal extension of lymph node metastasis influences recurrence in prostate cancer: A systematic review and meta-analysis. *Sci. Rep.* **2017**, *7*, 2374.
39. Grosser, B.; Glückstein, M.; Dhillon, C.; Schiele, S.; Dintner, S.; VanSchoiack, A.; Kroeppler, D.; Martin, B.; Probst, A.; Vlasenko, D.; et al. Stroma A Reactive I nvasion F ront A reas (SARIFA)—A new prognostic biomarker in gastric cancer related to tumor-promoting adipocytes. *J. Pathol.* **2022**, *256*, 71–82. [CrossRef] [PubMed]
40. Yu, K.H.; Wang, F.; Berry, G.J.; Ré, C.; Altman, R.B.; Snyder, M.; Kohane, I.S. Classifying non-small cell lung cancer types and transcriptomic subtypes using convolutional neural networks. *J. Am. Med. Inform. Assoc.* **2020**, *27*, 757–769. [CrossRef] [PubMed]

Supplementary Data

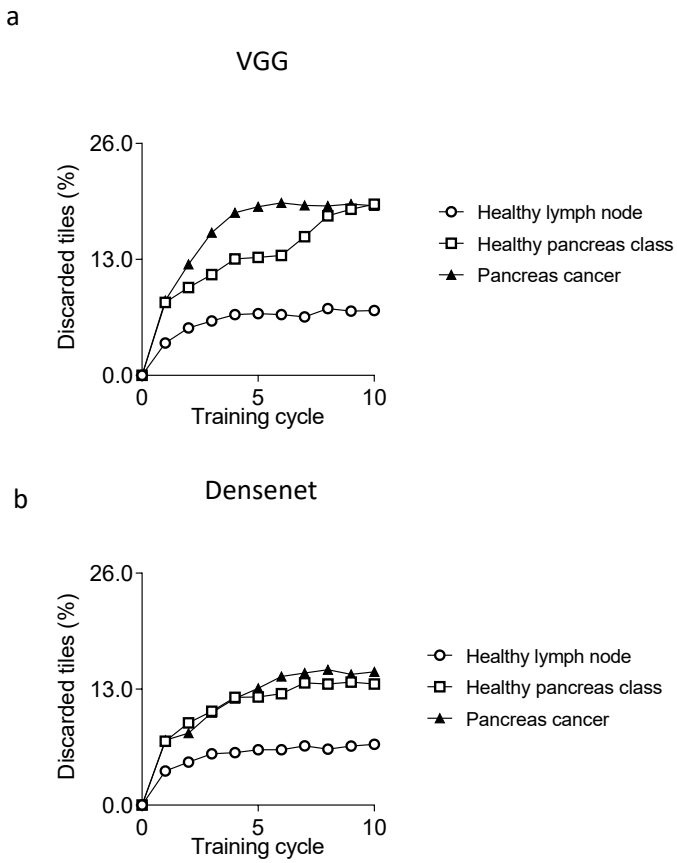
Supplementary Figure S1



Supplementary Figure S1: Tissue Micro Arrays enable staining and presentation of multiple patient tissue sections on one histological slide. (a) TMA with three spots of pancreatic ductal adenocarcinoma (PDAC), two lymph nodes with metastasis from pancreatic ductal adenocarcinoma (LNPM) and one healthy pancreas (HP) per patient are shown. Healthy lymph nodes are on different TMAs. Representative images and zoom from H&E-stained samples of **(b)** pancreatic ductal adenocarcinoma (PDAC), **(c)** healthy pancreas (HP) and **(d)** healthy lymph node (HLN) are shown (Scalebar = 300 µm).

Supplementary Data

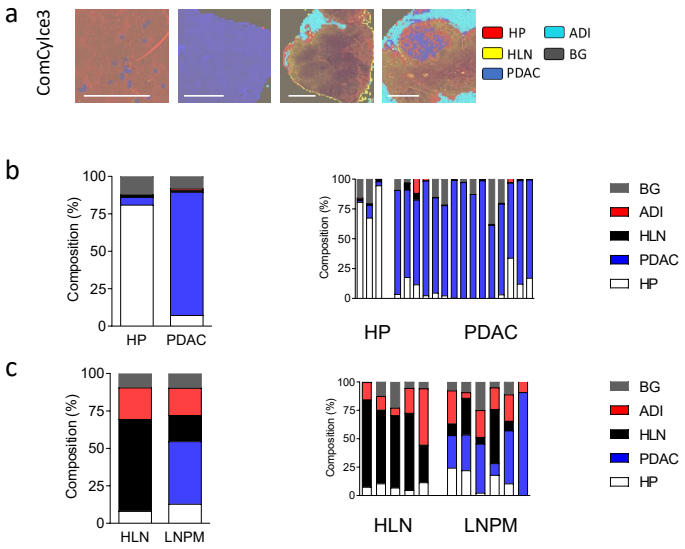
Supplementary Figure S2



Supplementary Figure S2: Percentage of discarded image patches of the different tissue types during the cleanup process from healthy lymph nodes, healthy pancreas and pancreatic ductal adenocarcinoma is indicated for **(a)** VGG and **(b)** Densenet are shown.

Supplementary Data

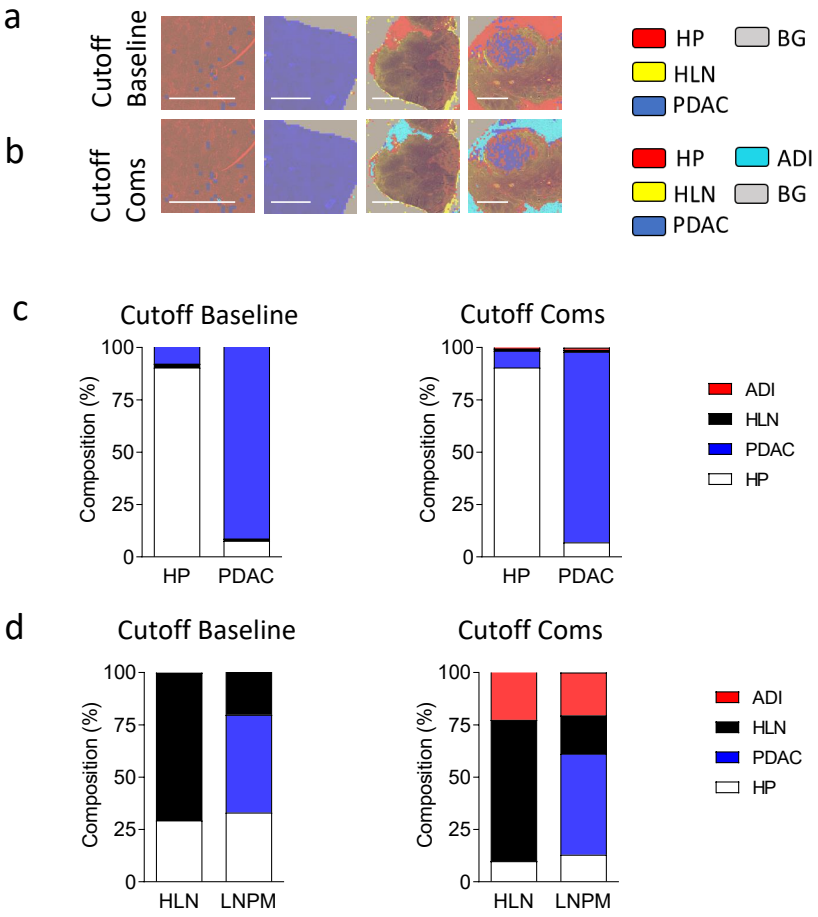
Supplementary Figure S3



Supplementary Figure S3: ComCylce3 on the extern validation data shows improvement with only 3 cycles. Colored external validation images with the **(a)** Baseline model and with the Cutoff Communicators model are shown. (**b)** Pooled and Individual classification as determined using an cutoff baseline and cutoff cleaned of whole images slides from healthy pancreas (HP) (n=3) and pancreatic ductal adenocarcinoma (PDAC) (n=15) **(c)** Pooled and Individual classification as determined using an cutoff baseline and cutoff cleaned network of whole images slides from healthy lymph nodes (HLN) (n=5) and lymph nodes with metastasis from pancreatic ductal adenocarcinoma (LNPM) (n=6) are shown.

Supplementary Data

Supplementary Figure S4

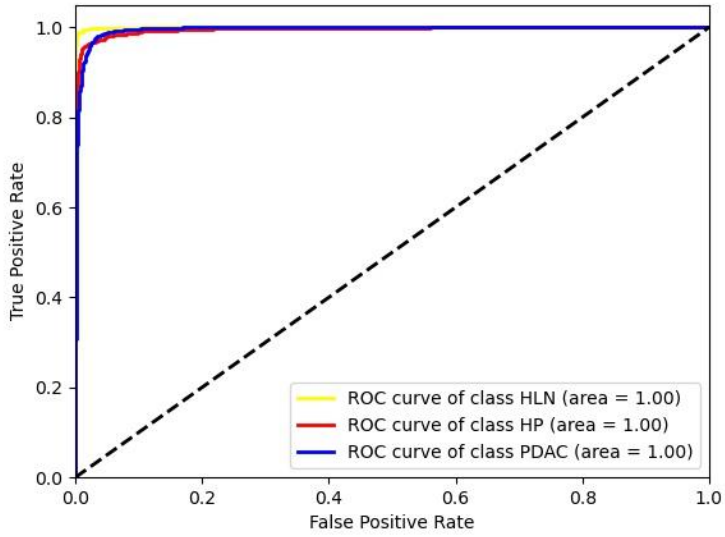


Supplementary Figure S4: Baseline and Coms with Pixelcutoff instead of background class. Coms still outperform Baseline on the extern validation data with n =10 Cycles for the Communicators. Colored external validation images with the **(a)** Cutoff Baseline model and **(b)** with the Cutoff Communicators model are shown. **(c)** Pooled classification as determined using an cutoff baseline and cutoff cleaned of whole images slides from healthy pancreas (HP) (n=3) and pancreatic ductal adenocarcinoma (PDAC) (n=15) are shown. **(d)** Pooled classification as determined using an cutoff baseline and cutoff cleaned network of whole images slides from healthy lymph nodes (HLN) (n=5) and lymph nodes with metastasis from pancreatic ductal adenocarcinoma (LNPM) (n=6) are shown.

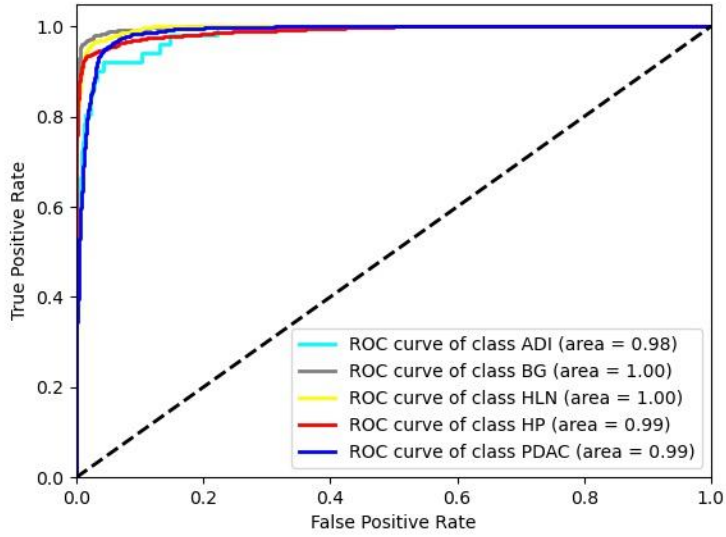
Supplementary Data

Supplementary Figure S5

a



b



Supplementary Figure S5: Receiver operating characteristic for the different tissue classes for **(a)** Baseline evaluated on the cleaned dataset and for the **(b)** Cleaned network evaluated on the cleaned dataset are shown.

Supplementary Data

Supplementary Table S1

class	Sex (male(%)/female(%))	Median Age (range)	Number of Spots	Number of patients
PDAC	52.9 / 47.1	68 (41-90)	223	223
HLN	anonym	anonym	76	78
HP	52.9 / 47.1	68 (41-90)	161	164

Supplementary Table S1: Patients Data: Age, Gender and Number of Spots for the three classes healthy pancreas (HP), healthy lymph node (HLN) and Pancreatic ductal adenocarcinoma (PDAC) are provided.

Supplementary Data

Supplementary Data

Supplementary Table S2

class	precision	recall	f1-score	jaccard-score
HLN	0.03	0.03	0.07	0.05
HP	0.07	0.09	0.11	0.13
PDAC	0.03	0.02	0.05	0.04
accuracy			0.04	
macro avg	0.01	-0.03	-0.01	0
weighted avg	0.04	0.04	0.04	0.08

Supplementary Table S2: Differences of the metrics of the cleaned and uncleaned network: Accuracy, Precision, Recall, F1-Score and Jaccard score for the classes healthy pancreas (HP), healthy lymph node (HLN) and Pancreatic ductal adenocarcinoma (PDAC) are shown.

Supplementary Data

Supplementary Table S3

Name	Model	Patchsize	Batchsize	Optimizer	Learning rate	Cutoff	Com Cycles
Baseline	resnet18	224x224x3	150	ADAM	0.0001	No	-
Cleaned network	resnet18	224x224x3	150	ADAM	0.0001	No	10
Inception Baseline	inception	299x299x3	75	RMSprop	0.01	No	-
Inception Coms	inception	299x299x3	75	RMSprop	0.01	No	10
Cutoff Baseline	resnet18	224x224x3	150	ADAM	0.0001	239	-
Cutoff Coms	resnet18	224x224x3	150	ADAM	0.0001	239	10
ComCylce3	resnet18	224x224x3	150	ADAM	0.0001	No	3

Supplementary Table S3: Overview of the seven CNN configuration used for the experiments. (Excluding the models from hyperparamter tuning.)

Supplementary Data

Supplementary Table S4

model_pre	model_name	lr	opt	balanced accuracy	precision	recall	f1 score	jaccard score	roc_auc_score	HLN score	PDAC score	HP score	LNPM score	test HLN	test PDAC	test HP	four_score	three_score
alexnet_1	alexnet	0.0001	ADAM	0.857	0.944	0.945	0.944	0.896	0.994	0.969	0.89	0.978	0.997	0.945	0.959	0.928	0.958	0.944
alexnet_2	alexnet	0.0001	SGD	0.857	0.945	0.945	0.944	0.897	0.994	0.968	0.898	0.979	0.997	0.945	0.962	0.924	0.96	0.944
alexnet_3	alexnet	0.0001	RMSprop	0.831	0.937	0.937	0.935	0.881	0.988	0.944	0.942	0.985	0.989	0.943	0.965	0.898	0.965	0.935
alexnet_4	alexnet	0.00001	ADAM	0.801	0.92	0.921	0.92	0.855	0.987	0.962	0.851	0.979	0.991	0.918	0.939	0.894	0.946	0.917
alexnet_5	alexnet	0.00001	SGD	0.797	0.919	0.92	0.918	0.853	0.987	0.963	0.842	0.978	0.989	0.918	0.935	0.896	0.943	0.916
alexnet_6	alexnet	0.00001	RMSprop	0.823	0.929	0.93	0.929	0.869	0.99	0.959	0.858	0.977	0.988	0.927	0.942	0.911	0.946	0.927
alexnet_7	alexnet	0.000001	ADAM	0.666	0.824	0.83	0.825	0.71	0.955	0.894	0.855	0.987	0.883	0.71	0.853	0.809	0.907	0.791
alexnet_8	alexnet	0.000001	SGD	0.666	0.824	0.83	0.825	0.71	0.955	0.894	0.855	0.997	0.883	0.71	0.853	0.809	0.907	0.791
alexnet_9	alexnet	0.000001	RMSprop	0.691	0.847	0.856	0.851	0.747	0.965	0.95	0.881	0.995	0.866	0.805	0.878	0.814	0.923	0.832
densenet_1	densenet	0.0001	ADAM	0.875	0.96	0.961	0.96	0.925	0.997	0.988	0.891	0.978	0.995	0.985	0.968	0.955	0.963	0.969
densenet_2	densenet	0.0001	SGD	0.876	0.962	0.962	0.961	0.927	0.997	0.987	0.895	0.977	0.996	0.987	0.97	0.956	0.964	0.971
densenet_3	densenet	0.0001	RMSprop	0.877	0.964	0.964	0.963	0.931	0.997	0.982	0.861	0.97	0.996	0.984	0.975	0.952	0.952	0.97
densenet_4	densenet	0.00001	ADAM	0.773	0.928	0.926	0.923	0.862	0.986	0.984	0.939	0.992	0.911	0.914	0.952	0.905	0.956	0.924
densenet_5	densenet	0.00001	SGD	0.773	0.928	0.926	0.923	0.862	0.986	0.984	0.939	0.992	0.911	0.914	0.951	0.905	0.956	0.923
densenet_6	densenet	0.00001	RMSprop	0.82	0.938	0.937	0.936	0.882	0.99	0.989	0.947	0.992	0.959	0.938	0.962	0.913	0.972	0.938
densenet_7	densenet	0.000001	ADAM	0.649	0.794	0.629	0.668	0.523	0.905	0.901	0.457	0.951	0.86	0.679	0.363	0.793	0.792	0.612
densenet_8	densenet	0.000001	SGD	0.649	0.794	0.629	0.668	0.523	0.905	0.901	0.457	0.951	0.86	0.679	0.363	0.793	0.792	0.612
densenet_9	densenet	0.000001	RMSprop	0.729	0.839	0.765	0.791	0.662	0.939	0.934	0.714	0.99	0.986	0.782	0.635	0.841	0.906	0.753
resnet_1	resnet	0.0001	ADAM	0.877	0.947	0.947	0.947	0.9	0.994	0.99	0.942	0.998	0.981	0.946	0.962	0.93	0.978	0.946
resnet_2	resnet	0.0001	SGD	0.877	0.947	0.947	0.947	0.9	0.994	0.99	0.942	0.998	0.981	0.946	0.962	0.93	0.978	0.946
resnet_3	resnet	0.0001	RMSprop	0.871	0.948	0.949	0.948	0.903	0.995	0.98	0.933	0.999	0.981	0.95	0.963	0.935	0.973	0.949
resnet_4	resnet	0.00001	ADAM	0.792	0.917	0.916	0.914	0.844	0.983	0.944	0.963	0.97	0.962	0.906	0.939	0.878	0.96	0.908
resnet_5	resnet	0.00001	SGD	0.792	0.917	0.916	0.914	0.844	0.983	0.944	0.963	0.97	0.962	0.906	0.939	0.878	0.96	0.908
resnet_6	resnet	0.00001	RMSprop	0.824	0.927	0.927	0.926	0.864	0.987	0.958	0.958	0.983	0.986	0.921	0.949	0.894	0.971	0.921
resnet_7	resnet	0.000001	ADAM	0.506	0.705	0.659	0.641	0.489	0.899	0.402	0.892	0.6	0.994	0.203	0.735	0.616	0.722	0.518
resnet_8	resnet	0.000001	SGD	0.506	0.705	0.659	0.641	0.489	0.899	0.402	0.892	0.6	0.994	0.203	0.735	0.616	0.722	0.518
resnet_9	resnet	0.000001	RMSprop	0.619	0.787	0.782	0.775	0.64	0.937	0.518	0.954	0.796	0.932	0.545	0.829	0.748	0.8	0.707
resnet101_1	resnet101	0.0001	ADAM	0.872	0.951	0.951	0.951	0.908	0.994	0.992	0.849	0.982	0.938	0.971	0.953	0.949	0.94	0.958
resnet101_2	resnet101	0.0001	SGD	0.872	0.951	0.952	0.951	0.909	0.994	0.992	0.849	0.981	0.938	0.971	0.954	0.95	0.94	0.958
resnet101_3	resnet101	0.0001	RMSprop	0.861	0.951	0.951	0.951	0.909	0.994	0.999	0.869	0.996	0.944	0.976	0.958	0.943	0.952	0.959
resnet101_4	resnet101	0.00001	ADAM	0.794	0.932	0.93	0.928	0.869	0.989	0.824	0.923	0.987	0.964	0.931	0.964	0.895	0.924	0.93
resnet101_5	resnet101	0.00001	SGD	0.794	0.932	0.93	0.928	0.869	0.989	0.824	0.923	0.987	0.964	0.931	0.964	0.895	0.924	0.93
resnet101_6	resnet101	0.00001	RMSprop	0.833	0.933	0.933	0.932	0.875	0.991	0.903	0.89	0.989	0.939	0.943	0.954	0.906	0.93	0.934
resnet101_7	resnet101	0.000001	ADAM	0.624	0.764	0.681	0.675	0.522	0.942	0.68	0.551	0.976	0.675	0.715	0.438	0.847	0.72	0.667
resnet101_8	resnet101	0.000001	SGD	0.624	0.764	0.681	0.675	0.522	0.942	0.68	0.551	0.976	0.675	0.715	0.438	0.847	0.72	0.667
resnet101_9	resnet101	0.000001	RMSprop	0.694	0.843	0.842	0.839	0.729	0.959	0.644	0.853	0.989	0.974	0.794	0.808	0.865	0.865	0.822
resnet50_1	resnet50	0.0001	ADAM	0.874	0.952	0.952	0.952	0.91	0.996	0.985	0.926	0.984	0.982	0.97	0.972	0.93	0.969	0.957
resnet50_2	resnet50	0.0001	SGD	0.866	0.951	0.951	0.951	0.908	0.996	0.986	0.93	0.984	0.976	0.971	0.966	0.939	0.969	0.959
resnet50_3	resnet50	0.0001	RMSprop	0.872	0.955	0.956	0.955	0.916	0.996	0.999	0.923	0.981	0.968	0.968	0.974	0.945	0.968	0.962
resnet50_4	resnet50	0.00001	ADAM	0.792	0.937	0.935	0.933	0.878	0.988	0.903	0.973	0.993	0.975	0.943	0.961	0.909	0.961	0.938
resnet50_5	resnet50	0.00001	SGD	0.792	0.937	0.935	0.933	0.878	0.988	0.903	0.973	0.993	0.975	0.943	0.961	0.909	0.961	0.938
resnet50_6	resnet50	0.00001	RMSprop	0.827	0.94	0.94	0.939	0.887	0.991	0.937	0.965	0.993	0.999	0.954	0.962	0.914	0.974	0.943
resnet50_7	resnet50	0.000001	ADAM	0.611	0.808	0.796	0.782	0.653	0.944	0.626	0.979	0.826	0.765	0.475	0.949	0.679	0.799	0.701
resnet50_8	resnet50	0.000001	SGD	0.611	0.808	0.796	0.782	0.653	0.944	0.626	0.979	0.826	0.765	0.475	0.949	0.679	0.799	0.701
resnet50_9	resnet50	0.000001	RMSprop	0.674	0.859	0.855	0.848	0.743	0.958	0.756	0.982	0.901	0.779	0.701	0.955	0.76	0.854	0.805
squeezenet_1	squeezenet	0.0001	ADAM	0.833	0.933	0.933	0.931	0.874	0.99	0.953	0.945	0.988	0.996	0.938	0.954	0.895	0.97	0.929
squeezenet_2	squeezenet	0.0001	SGD	0.832	0.932	0.932	0.931	0.873	0.99	0.952	0.947	0.988	0.999	0.937	0.954	0.894	0.972	0.928
squeezenet_3	squeezenet	0.0001	RMSprop	0.803	0.924	0.925	0.923	0.861	0.987	0.999	0.943	0.993	0.945	0.927	0.949	0.887	0.97	0.921
squeezenet_4	squeezenet	0.00001	ADAM	0.768	0.895	0.888	0.89	0.807	0.971	0.973	0.959	0.941	0.938	0.877	0.907	0.846	0.953	0.877
squeezenet_5	squeezenet	0.00001	SGD	0.768	0.895	0.889	0.89	0.807	0.971	0.973	0.959	0.941	0.938	0.877	0.907	0.846	0.953	0.877
squeezenet_6	squeezenet	0.00001	RMSprop	0.784	0.904	0.902	0.902	0.825	0.977	0.969	0.959	0.944	0.978	0.88	0.931	0.856	0.962	0.889
squeezenet_7	squeezenet	0.000001	ADAM	0.342	0.453	0.47	0.454	0.321	0.665	0.384	0.727	0.635	0.701	0.005	0.666	0.324	0.612	0.332
squeezenet_8	squeezenet	0.000001	SGD	0.343	0.453	0.471	0.454	0.321	0.665	0.384	0.727	0.635	0.701	0.005	0.666	0.325	0.612	0.332
squeezenet_9	squeezenet	0.000001	RMSprop	0.464	0.597	0.626	0.602	0.466	0.801	0.428	0.854	0.696	0.634	0.09	0.77	0.555	0.653	0.472
vgg16_1	vgg16	0.0001	ADAM	0.884	0.957	0.957	0.957	0.919	0.997	0.987	0.797	0.971	0.952	0.963	0.953	0.967	0.927	0.961
vgg16_2	vgg16	0.0001	SGD	0.881	0.957	0.958	0.957	0.92	0.997	0.988	0.811	0.973	0.96	0.963	0.954	0.966	0.933	0.961
vgg16_3	vgg16	0.0001	RMSprop	0.871	0.959	0.96	0.959	0.924	0.997	0.986	0.818	0.972	0.939	0.971	0.963	0.96	0.929	0.965
vgg16_4	vgg16	0.00001	ADAM	0.859	0.939	0.94	0.939	0.887	0.992	0.991	0.823	0.974	0.99	0.96	0.948	0.922	0.945	0.943
vgg16_5	vgg16	0.00001	SGD	0.858	0.939	0.939	0.939	0.886	0.992	0.992	0.823	0.974	0.99	0.96	0.948	0.922	0.945	0.943
vgg16_6	vgg16	0.00001	RMSprop	0.872	0.942	0.942	0.942	0.892	0.994	0.989	0.785	0.973	0.974	0.96	0.947	0.93	0.93	0.946
vgg16_7	vgg16	0.000001	ADAM	0.676	0.862	0.856	0.85	0.746	0.961	0.875	0.991	0.949	0.748	0.761	0.979	0.717	0.891	0.819
vgg16_8	vgg16	0.000001	SGD	0.678	0.864	0.858	0.852	0.749	0.962	0.88	0.992	0.952	0.749	0.772	0.979	0.719	0.893	0.823
vgg16_9	vgg16	0.000001	RMSprop	0.72	0.891	0.895	0.89	0.809	0.974	0.951	0.966	0.992	0.805	0.884	0.954	0.814	0.928	0.884
vgg19_1	vgg19	0.0001	ADAM	0.876	0.962	0.962	0.962	0.929	0.997	0.975	0.867	0.978	0.976	0.975	0.97	0.962	0.949	0.969
vgg19_2	vgg19	0.0001	SGD	0.884	0.963	0.963	0.963	0.93	0.997	0.97	0.873	0.98	0.977	0.972	0.971	0.96	0.95	0.968
vgg19_3	vgg19	0.0001	RMSprop	0.886	0.962	0.962	0.962	0.928	0.997	0.98	0.864	0.974	0.96	0.971	0.975	0.954	0.944	0.967
vgg19_4	vgg19	0.00001	ADAM	0.833	0.938	0.938	0.937	0.884	0.993	0.985	0.902	0.986	0.994	0.943	0.952	0.924	0.967	0.94
vgg19_5	vgg19	0.00001	SGD	0.833	0.937	0.938	0.937	0.883	0.993	0.985	0.901	0.986	0.993	0.943	0.952	0.924	0.966	0.94
vgg19_6	vgg19	0.00001	RMSprop	0.837	0.942	0.942	0.941	0.892	0.995	0.971	0.884	0.984	0.997	0.955	0.954	0.93	0.959	0.946
vgg19_7	vgg19	0.000001	ADAM	0.697	0.878	0.881	0.876	0.786	0.969	0.827	0.987	0.954	0.759	0.768	0.96	0.839	0.882	0.856
vgg19_8	vgg19	0.000001	SGD	0.697	0.878	0.881	0.876	0.786	0.969	0.827	0.987	0.954	0.759	0.768	0.96	0.839	0.882	0.856
vgg19_9	vgg19	0.000001	RMSprop	0.722	0.898	0.904	0.899	0.824	0.978	0.915	0.98	0.971	0.763	0.852	0.962	0.863	0.907	0.892

Supplementary Table S4: Network parameters and metrics from the 72 nets from Hyperparameter-Tuning are shown. We kept the following parameters constant for all the networks: batch size =256, epoch =100, early stopping 10, learning rate decay 0.95, learning step schedule = 5, retrain the last three blocks / layers.

Chapter 4

Improving Development of Novel Drugs using Deep Learning

In this chapter, we present our use case concerning the development of drugs: Deep Transfer Learning Approach for Automatic Recognition of SARS-CoV-2. We first briefly introduce the topic "Improved development of drugs using Machine Learning" and subsequently describe our approach to detect cytopatic effects in brightfield images as shown in Figure 4.1.

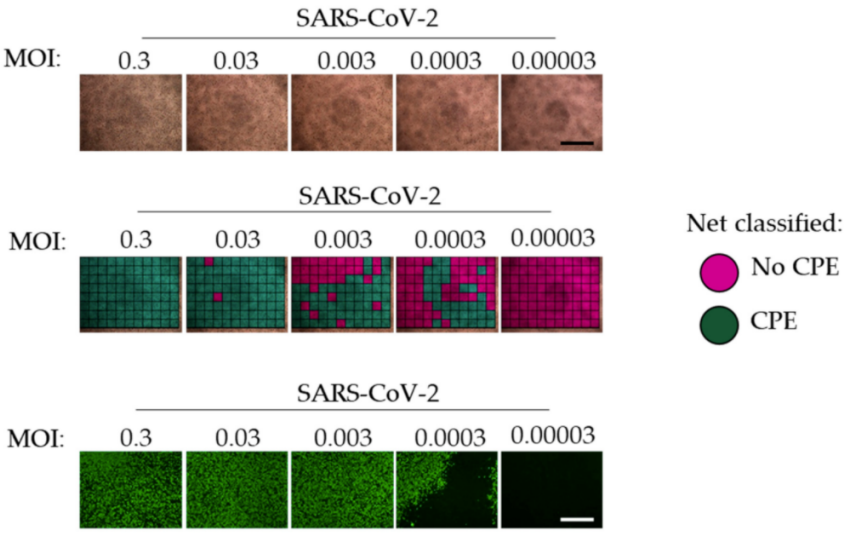


Figure 4.1: Neural net can detect in brightfield images. Image based on Figure 1 in Werner et al., 2021.

4.1 Improving the Development of Drugs Using Machine Learning

Drug development pipelines are complex, time-consuming, costly. Due to the complexity of biological systems, Machine Learning methods will be critical for future drug development. In particular, computer vision methods to extract detailed information from imaging assays to guide experimentation will be required to overcome the dimensionality problem in drug development. The exploitation of the potential of these Machine Learning techniques could fundamentally change the research process for identifying new molecules and/or repurposing old drugs.Machine Learning can speed up drug development, by reducing failure rates, which also reduces costs. The development and integration of such Machine Learning based models for end-to-end applications has a broad relevance and considerable implications for future drug development (Ekins et al., 2019; Murphy, 2011; Vamathevan et al., 2019).

4.2 Deep Transfer Learning Approach for Automatic Recognition of Drug Toxicity and Inhibition of SARS-CoV-2

In this section, we provide an overview of the contribution and impact of our paper Werner et al., 2021:

Julia Werner, Raphael M. Kronberg, Philipp N. Ostermann, Lisa Müller, Heiner Schaal, Jakob N. Kather, Arndt Borkhardt, Aleksandra A. Pandyra, Karl S. Lang, and Philipp A. Lang

"Deep Transfer Learning approach for automatic recognition of drugtoxicity and inhibition of SARS-CoV-2"

In: *Viruses* 13, no. 4: 610

Main Results in Simple Terms

In this paper, we investigated the application of Deep Transfer Learning to the development of drugs against the novel SARS-CoV-2 virus. More specifically, we developed a method to accelerate drug development under laboratory conditions. In order to develop active substances against viruses, the viruses first have to be cultivated in experiments and then exposed to the active substances.

The desired effect of the active substance on the virus can be demonstrated with various tests, some of which are very time-consuming while others are labor-intensive and costly. For example, parameters such as the virus titer or toxicity are determined or measured experimentally. Our approach relies on an automated evaluation of the experiments without requiring additional experiments, as the only input that we need for our approach is light field images.

The so-called cytopatic effect can already be seen with the naked eye, and the differentiation between a control and a toxic sample is also clearly recognisable for the biologist. Hence, we set up several experiments and photographed them over time. Nowadays, this can even be done automatically with the help of a photo table, which then photographs all the samples automatically at a set distance. From these images we created a training data set so that our neural network had examples for all three classes (control, cyptopathic effect, and toxicity).

Due to the rather limited data, we relied on Deep Transfer Learning and adapted and fine-tuned a pre-trained network for our purposes. The trained network was able to calculate a score that tells us how strong the effect or toxicity is. We compared this score to the established experiments and calculated the adjustment. The results of our neural network were

highly correlated with the results from the experiments.

Furthermore, we analyzed and correctly classified three known compounds and published the developed software so that it can be further improved and used in an experimental setup. With the above-mentioned automatic photo table and an automatic drug printer, one can create a very simple and effective pipeline for drug development. The advantages of our work are time saving and cost saving, as no further experiments have to be set up for evaluation and the neural network can analyse images within seconds.

Remark: Because of the novelty of this method, there are no suitable benchmark data sets to compare our method with other research groups.

Summary/Abstract

Severe acute respiratory syndrome coronavirus-2 (SARS-CoV-2) causes COVID-19 and is responsible for the ongoing pandemic that was declared by the WHO in 2020. The screening of potential antiviral drugs against SARS-CoV-2 depend on in vitro experiments that are based on the quantification of the virus titer. Here, we used virus-induced cytopathic effects (CPE) in brightfield microscopy of SARS-CoV-2-infected monolayers to quantify the virus titer. The images were classified using deep transfer learning (DTL) that fine-tunes the last layers of a pre-trained Resnet18 (ImageNet). To exclude toxic concentrations of potential drugs, the network was expanded to include a toxic score (TOX) that detected cell death (CPETOXnet). With this analytic tool, the inhibitory effects of chloroquine, hydroxychloroquine, remdesivir, and emetine were validated. We thus developed a simple method and provided open access implementation to quantify SARS-CoV-2 titers and drug toxicity in experimental settings, which may be applied to assays involving other viruses in future. The quantification of virus titers from brightfield images could accelerate the experimental approach for antiviral testing (Werner et al., 2021).

Personal Contribution

Formulated sentences

Raphael Marvin Kronberg (R.M.K.) performed computational experiments and data analysis, e.g. he calculated the metrics for the different Deep Neural Networks. He discussed the data and rewrote the deep learning part of the paper. The implementation of the Deep Neural Networks and the pipeline in Python using Pytorch as framework was carried out by R.M.K..

Bullet points (CRediT version)

*Conceptualization, J.W. and P.A.L.; methodology, J.W., **R.M.K.**, P.S., P.N.O., L.M., H.S., J.N.K., A.B., A.A.P., K.S.L., and P.A.L.; software, **R.M.K.** and P.A.L.; validation, J.W., **R.M.K.**, P.S., P.N.O., L.M., H.S., J.N.K., A.B., A.A.P., K.S.L., and P.A.L.; formal analysis, J.W., **R.M.K.**, P.N.O., L.M., and P.A.L.; investigation, J.W., **R.M.K.**, P.S., P.N.O., L.M., and P.A.L.; resources, H.S., S.B., and P.A.L.; data curation, J.W., **R.M.K.**, P.N.O., L.M., and P.A.L.; writing—original draft preparation, P.A.L.; writing—review and editing, J.W., **R.M.K.**, P.S., P.N.O., L.M., H.S., S.B., J.N.K., A.B., A.A.P., and K.S.L.; visualization, J.W., **R.M.K.**, P.S., and P.A.L.; supervision, P.A.L.; project administration, P.A.L.; funding acquisition, P.A.L. All authors have read and agreed to the published version of the manuscript (Werner et al., 2021).*

Importance of the Research and Contribution to this Thesis

The automated Deep Learning-based classification of cytopathic effects in brightfield images serves as an example of how artificial intelligence can improve the approach for the development of novel drugs. It thus answers the first research question: How could researchers automatically evaluate drug screenings against viruses with a Deep Learning approach?

viruses

MDPI

Article

Deep Transfer Learning Approach for Automatic Recognition of Drug Toxicity and Inhibition of SARS-CoV-2

Julia Werner [1,†], Raphael M. Kronberg [1,2,†], Pawel Stachura [1], Philipp N. Ostermann [3], Lisa Müller [3], Heiner Schaal [3], Sanil Bhatia [4], Jakob N. Kather [5], Arndt Borkhardt [4], Aleksandra A. Pandyra [4], Karl S. Lang [6] and Philipp A. Lang [1,*]

1 Department of Molecular Medicine II, Medical Faculty, Heinrich-Heine-University, 40225 Düsseldorf, Germany; Julia.Werner2@med.uni-duesseldorf.de (J.W.); Raphael.Kronberg@hhu.de (R.M.K.); Pawel.Stachura@med.uni-duesseldorf.de (P.S.)
2 Mathematical Modelling of Biological Systems, Heinrich-Heine-University, 40225 Düsseldorf, Germany
3 Institute of Virology, Medical Faculty, Heinrich-Heine-University, 40225 Düsseldorf, Germany; Philipp.Ostermann@uni-duesseldorf.de (P.N.O.); Lisa.Mueller@uni-duesseldorf.de (L.M.); Schaal@uni-duesseldorf.de (H.S.)
4 Department of Pediatric Oncology, Hematology and Clinical Immunology, Medical Faculty, Center of Child and Adolescent Health, Heinrich-Heine-University, 40225 Düsseldorf, Germany; Sanil.Bhatia@med.uni-duesseldorf.de (S.B.); Arndt.Borkhardt@med.uni-duesseldorf.de (A.B.); AleksandraAnna.Pandyra@med.uni-duesseldorf.de (A.A.P.)
5 Department of Medicine III, University Hospital RWTH Aachen, 52074 Aachen, Germany; Jkather@ukaachen.de
6 Institute of Immunology, Medical Faculty, University of Duisburg-Essen, 45147 Essen, Germany; KarlSebastian.Lang@uk-essen.de
* Correspondence: Langp@uni-duesseldorf.de
† These authors are equal parts first authors.

Abstract: Severe acute respiratory syndrome coronavirus-2 (SARS-CoV-2) causes COVID-19 and is responsible for the ongoing pandemic. Screening of potential antiviral drugs against SARS-CoV-2 depend on in vitro experiments, which are based on the quantification of the virus titer. Here, we used virus-induced cytopathic effects (CPE) in brightfield microscopy of SARS-CoV-2-infected monolayers to quantify the virus titer. Images were classified using deep transfer learning (DTL) that fine-tune the last layers of a pre-trained Resnet18 (ImageNet). To exclude toxic concentrations of potential drugs, the network was expanded to include a toxic score (TOX) that detected cell death (CPETOXnet). With this analytic tool, the inhibitory effects of chloroquine, hydroxychloroquine, remdesivir, and emetine were validated. Taken together we developed a simple method and provided open access implementation to quantify SARS-CoV-2 titers and drug toxicity in experimental settings, which may be adaptable to assays with other viruses. The quantification of virus titers from brightfield images could accelerate the experimental approach for antiviral testing.

Keywords: SARS-CoV-2; deep transfer learning; deep learning; drug screening; emetine; chloroquine; remdesivir; hydroxychloroquine

check for updates

Citation: Werner, J.; Kronberg, R.M.; Stachura, P.; Ostermann, P.N.; Müller, L.; Schaal, H.; Bhatia, S.; Kather, J.N.; Borkhardt, A.; Pandyra, A.A.; et al. Deep Transfer Learning Approach for Automatic Recognition of Drug Toxicity and Inhibition of SARS-CoV-2. *Viruses* **2021**, *13*, 610. https://doi.org/10.3390/v13040610

Academic Editor: Meehyein Kim

Received: 27 January 2021
Accepted: 30 March 2021
Published: 2 April 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Severe acute respiratory syndrome coronavirus-2 (SARS-CoV-2) emerged in 2019 as a pathogen responsible for the coronavirus disease 2019 (COVID-19), which in a proportion of cases causes severe symptoms such as shortage of breath and lung failure [1]. SARS-CoV-2 binds to the entry receptor ACE2, which triggers uptake and cleavage by the proteases Cathepsin B and TMPRSS2 [2]. If viruses cause no or low cytopathic effects (CPE), immunostaining is used to determine virus titers [3]. In contrast, viruses with strong CPE can be visualized by staining of residual cells, resulting in plaque forming units [4]. For coronaviruses, plaque assays have been established, but effects depend on

Viruses **2021**, *13*, 610

the used cell line and the virus strain [5]. SARS-CoV-2 can be quantified using PCR, which is widely used as a specific and effective diagnostic tool [2]. Furthermore, immunostaining of viral proteins including the nucleocapsid of SARS-CoV-2 have been established to detect infection with SARS-CoV-2 in tissue cultures [6]. All of these protocols involve additional procedures such as fixation and staining to quantify SARS-CoV-2 in tissue culture. These assays have been used to screen for antiviral compounds against SARS-CoV-2 infection. Specifically, hydroxychloroquine and remdesivir were found to reduce SARS-CoV-2 propagation in vitro [6,7]. Both compounds have been tested clinically to treat patients infected with SARS-CoV-2. While hydroxychloroquine was shown to reduce SARS-CoV-2 viral load in a small patient cohort [8], there was no beneficial use in post exposure prophylaxis or as a treatment for mild COVID-19, especially when considering severe side effects [9–11]. Remdesivir was able to reduce recovery time compared to a placebo group in hospitalized COVID-19 patients but, when applied as a monotherapy, did not decrease the high mortality rate [12]. Taken together, in vitro assays involving SARS-CoV-2 propagation have been successfully used to identify potential novel antiviral compounds.

Machine learning is rapidly advancing in different areas of life sciences. Deep neural networks have been used for image classification. Specifically, convolutional neural networks are multilayered trained with a back-propagation algorithm to classify shapes [13]. In various tasks in biomedical research, pretrained neural networks have been retrained and successfully used for specific tasks. Pre-trained network models are usually trained on a large number of images in the ImageNet database, allowing them to classify these images into many categories. By retraining these networks on a domain-specific task, previously learned out-of-domain features can improve model convergence and accuracy. Images are provided in an input layer and are connected to the consequent layers, resulting in classification of the provided image through a classification and output layer [13]. Previous studies have shown that cancer tissues can be classified and mutations or expression profiles predicted using retraining of the neural network 'Inceptionv3' [14,15]. Furthermore, retraining of the network 'Resnet18'can predict the microsatellite instability from hematoxylin and eosin (H&E) histology samples of patients with gastrointestinal cancer [16]. Moreover, survival of cancer patients can be predicted from histology samples in combination with or without other parameters using convolutional neural networks [17–19]. During the SARS-CoV-2 pandemic, neural networks were used to identify pneumonia caused by SARS-CoV-2 from computed tomography (CT) scans [20,21]. However, deep neural networks have not been used to quantify CPE in experimental assays.

Here, we adapt the pretrained neural network 'Resnet18' to classify and score images obtained from SARS-CoV-2 cultures. 'CPEnet' was able to attribute a higher score to images from SARS-CoV-2 infected Vero cells, while non-infected cells were given a low score. These scores correlated with other readouts tested. Moreover, further training on a 'CPETOXnet' included classification of potential toxicities during drug testing. 'CPETOXnet' was able to quantify the inhibition of SARS-CoV-2 replication by chloroquine, hydroxychloroquine, remdesivir, and emetine, while simultaneously identifying the toxic in vitro effects of hydroxychloroquine and emetine.

2. Materials and Methods

2.1. Viruses

SARS-CoV-2 was used as described previously (Sequence Accession Number: EPI_ISL_425126) [22,23]. SARS-CoV-2 was propagated in Vero cells by infection at a multiplicity of infection (MOI) of 0.001. After 72 h, the supernatant was taken and stored as −80 °C until usage.

2.2. SARS-CoV-2 Infection of Cells

Vero cells were cultured as previously described [2]. Cells were cultured in Dulbecco's modified eagle's medium (DMEM) with the addition of 10% foetal calf serum (FCS), minimal essential amino acids, and Penicillin/Streptomycin at 37 °C and 5% CO_2.

Viruses **2021**, *13*, 610

3×10^4 cells were seeded per well in a 96 well plate one day before infection. On the next day, the medium was changed to the cell culture medium containing different concentrations of remdesivir, PUH71, AUY922 (Luminespib), NVP-HSP990, EC144, PF-0429113, BIIB021, Tanespimycin (MedChemExpress, Monmouth Junction, NJ, USA), emetine, chloroquine, or hydroxychloroquine (Sigma-Aldrich, St. Louis, MO, USA) dissolved in DMSO. Moreover, 12 serial 3.16-fold dilutions with each second equivalent to a 10-fold dilution were used. The cells were infected 20 min later with different MOIs. An overlay composed of equal proportions 2× DMEM and 2% methylcellulose was added 2 h post infection. EC_{50} and CC_{50} were measured using GraphPad Prism.

2.3. Immunofluorescence Staining

Two days after infection, the supernatant was discarded and 4% Formalin was added for 30 min. Hank's buffer containing Triton-X was applied to the cells for 20 min followed by 10% FCS in PBS for 1 h to block unspecific binding sites. The cells were stained with a SARS-CoV-2 Nucleocapsid antibody (2019 nCoV) (Sino Biology Inc., Eschborn, Germany) for 1 h. Following washing, Fluorescein (FITC) conjugated AffiniPure Goat Anti-Rabbit IgG (H+L) (Jackson Immuno Research, Cambridgeshire, UK) was added for 1 h. The cells were washed again and analyzed with the Nikon Eclipse TS100 fluorescence microscope. Pictures were taken with the software NIS-Elements F4.30.01.

2.4. Immunofluorescence Staining Data Analysis

Fluorescent images were analyzed with the ImageJ software. Images were changed to 8-bit and the threshold value was adjusted uniformly for each experiment. The particles were analyzed, whereby the percentage of fluorescence was determined.

2.5. Deep Transfer Learning

2.5.1. Architecture

ResNet18 (see Figure S1) was chosen for retraining due to the balance between high accuracy and low prediction time [24,25]. This network has been trained on more than a million images and can classify images into 1000 object categories [26]. For each of the classification tasks, the last two layers (classification and output) were retrained using parameters as previously described [16]. To classify CPE in in SARS-CoV-2 cell cultures into a binary classification (CPE or No CPE) the 'CPEnet' was trained and score was calculated by summing up the class for each sub image divided by the total subimages

$$\text{ScoreCPE} = \Sigma\text{CPE-Tiles}/\Sigma\text{Total-Tiles} \quad (1)$$

The binary 'IFnet' classifies immunofluorescence (IF Signal and No Signal) for each input to quantify immunofluorescence on the whole image is as follows:

$$\text{ScoreIF} = \Sigma\text{IF-Tiles}/\Sigma\text{Total-Tiles} \quad (2)$$

In addition, a 'CPETOXnet' was trained to recognizing cell death in these cultures, which could identify possible toxic effect of compounds being tested (CPE, TOX and No CPE).

$$\text{ScoreTox} = \Sigma\text{Tox-Tiles}/\Sigma\text{Total-Tiles} \quad (3)$$

2.5.2. Data and Training

The images used all had a resolution of 2560 × 1920 pixels and were divided into 224 × 224 pixel sub-images, as this was the input shape for the ResNet18. The labels of the sub-images were inherited from the images.

'CPEnet' was trained on images obtained from 30 negative controls and 32 SARS-CoV 2 infected tissue cultures (MOI: 0.01) from 5 independent experiments using 21/23 for training, 5 for validation, and 4 for testing. 'IFnet' was trained on 40 images each taken from negative controls and SARS-CoV-2 infected cells (MOI: 0.03) after immunostaining for the

Viruses **2021**, *13*, 610

nucleocapsid of SARS-CoV-2 from 4 independent plates using 28 for training, 6 for validation, and 6 for testing. 'CPETOXnet' was trained on 72 images each taken from control cells, staurosporine treated cells (5 µM), and SARS-CoV-2 infected cells (MOI: 0.03) from 3 independent plates using 51 for training, 11 for validation, and 10 for testing. Calculations were performed using Matlab R2020a (Mathworks, Natick, MA, USA) on a desktop computer (i5-6500 CPU @ 3.2GHz (Intel), 8GB RAM or a single GPU, Nvidia Quadro P4000). Furthermore, calculations were performed on a high performance computing cluster of the HHU using Python. The source code is available at: https://github.com/MolecularMedicine2/PyQoVi (Available from: 2 April 2021) (Quantification of Virus in images (e.g., SARS-CoV-2)).

2.5.3. Statistical Analyses

Data are expressed as mean ± SEM. Linear regression was calculated with GraphPad Prism. The networks 'CPEnet', 'IFnet', and 'CPETOXnet' were evaluated on a test dataset and the accuracy and the F-score were determined via calculating the confusion matrix. Statistically significant differences between groups in experiments involving more than one time point were determined using two-way ANOVA.

3. Results

3.1. Retraining of a Convolutional Neural Network to 'CPEnet' Predicts a CPE Score for Images

The infection of Vero cells with SARS-CoV-2 at an MOI of 0.01 resulted in visible detection of CPE in tissue culture after 72 h (Figure 1a). Images were acquired on live tissue cultures with closed plates. Only one image was taken per well. Accordingly, images showed artefacts originating from condensation and shadows (Figure 1a). To retrain 'Resnet18′ to detect CPE, we dissected images (2560 × 1920) exhibiting SARS-CoV-2 mediated CPE and negative controls from several experiments into the required input image size (224 × 224) for 'Resnet18′ (Figure 1b, Figure S1). In total, 30 images of negative controls and 32 images showing SARS-CoV-2 mediated CPE were split into 21/23 images for training, 5 images for validation, and 4 images for testing. Accordingly, 1848/2024 training tiles, 440 validation tiles, and 352 testing tiles were used. Notably, at a MOI of 0.01, CPE was detected in most, but not all image tiles. Hence, we did not expect that all tiles of images from SARS-CoV-2 infected cells would classify as positive. In turn, we expected residual cell death after 72 h of tissue culture in healthy controls resulting in image tiles exhibiting similar features as CPE. Based on these assumptions, image tiles used for training was not used for validation or testing and generated from separate images. In our examples, the averaged F-score to classify CPE on the test dataset was 0.8997, with an achieved accuracy of 0.9063 (Figure 1c,d, Figure S2a). Consistently, when a sample image was classified by 'CPEnet' a score of 0.0628 for a negative control and 0.8636 for a positive control was determined (Figure 1e–g). These data indicated that 'Resnet18′ could be retrained to detect and attribute a number to CPE images in SARS-CoV-2 cultures. Notably, we speculated that the attributed score might reflect the true appearance of image tiles exhibiting CPE regardless of SARS-CoV-2 infection. Hence, further validation is required to investigate whether neural networks can quantify SARS-CoV-2 mediated CPE.

3.2. 'CPEnet' Generated Quantification of SARS-CoV-2 Cultures Correlates with Immunostainings for the Nucleocapsid of SARS-CoV-2

To determine whether the CPE can reflect data on propagation of SARS-CoV-2, we analyzed plates of titrated SARS-CoV-2 infected cells using other methods of quantification. Notably, CPE is visible 72 h after infection, while we only observed modest CPE in tissue culture 48 h after infection (Figure S3). However, expression of the nucleocapsid SARS-CoV-2 protein can be detected after 48 h. To test the accuracy of 'CPEnet', we made 12 serial 3,16-fold dilutions for every second dilution to be 10-fold of SARS-CoV-2 cultures starting with MOIs of 1 and 0.001. As expected, after 72 h, CPE were visible but diminished with increasing dilutions (Figure 2a). The neural network could detect CPE in SARS-CoV-2 infected tissue cultures as well as in immunofluorescence staining of SARS-CoV-2

Viruses **2021**, *13*, 610

nucleocapsid protein (Figure 2b,c). While the positive control (MOI 0.03) was classified approximately to a CPE score close to 1, the negative control was attributed a score close to 0 (Figure 2d). Serial dilutions indicated that the input of SARS-CoV-2/1000 appeared 6 dilutions later, indicating that virus titrations could be detected by 'CPEnet' (Figure 2d). Notably, we observed a similar pattern with immunostaining for the nucleoprotein of SARS-CoV-2 (Figure 2e). When we correlated the obtained CPE score with the quantification of the immunofluorescence, we found a significant correlation with the R square = 0.92 (Figure 2f).

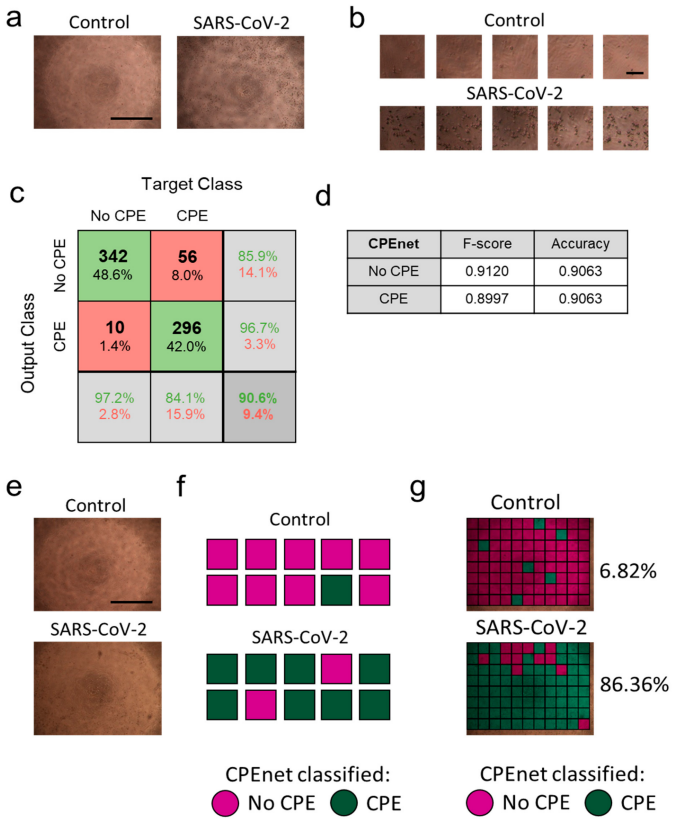


Figure 1. Retraining of 'Resnet18' can identify severe acute respiratory syndrome coronavirus-2 (SARS-CoV-2) mediated cytopathic effects (CPE) in live tissue culture brightfield images. Vero cells were infected with SARS-CoV-2 at an multiplicity of infection (MOI) of 0.01. (**a**) Images were taken 3 days after infection (n = 32) or without (n = 30) infection out of 5 independent experiments (One representative set of images is shown, scale bar = 1 mm). (**b**) Images as in (**a**) were dissected into 224 × 224 image tiles matching the input size of 'resnet18' resulting in 1848 training, 440 validation, and 352 testing control image tiles and 2024 training, 440 validation, and 352 testing image tiles from SARS-CoV-2 infected cells (scale bar = 100µm). (**c**) Confusion matrix for the 'CPEnet' on the test data set. (**d**) F-score and accuracy on the test dataset. (**e**,**f**) Images from control (upper panel) and SARS-CoV-2 infected (lower panel) tissue cultures were classified by 'CPEnet'. (**e**) Original images are shown (scale bar = 1 mm). (**f**) Schematic of individual scoring through analysis of image tiles is illustrated. (**g**) Image tiles are stained in red (CPE) or green (no CPE) on the original images as classified by 'CPEnet'.

Viruses **2021**, *13*, 610

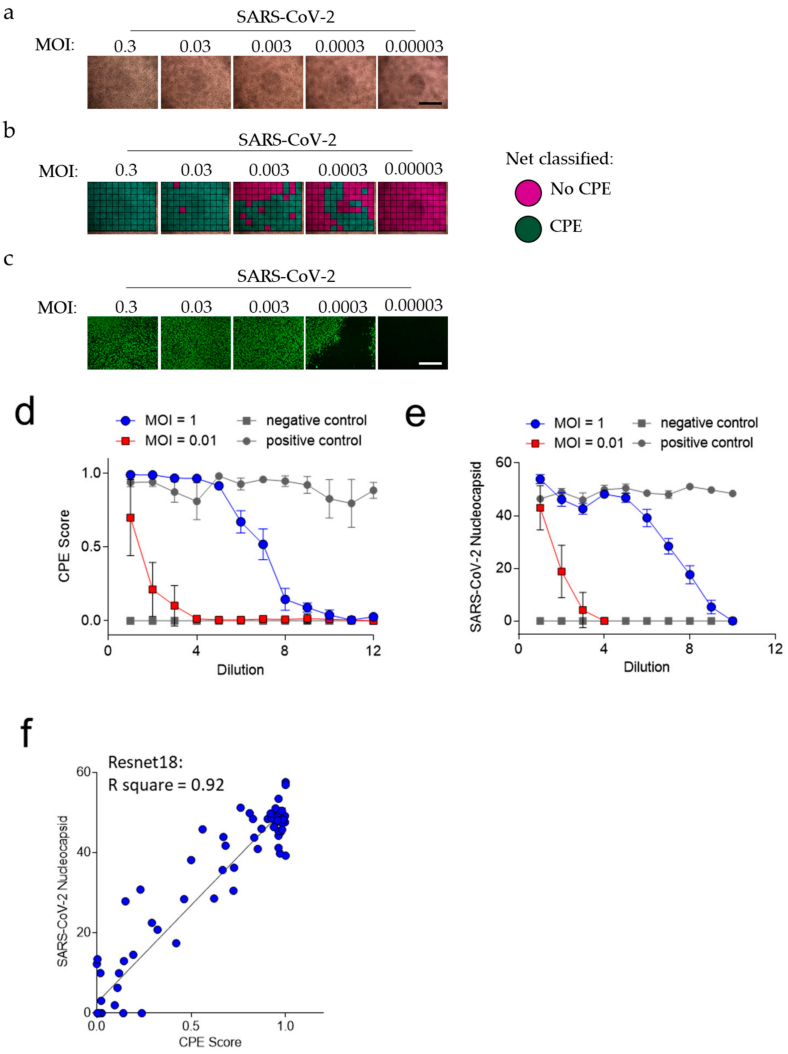


Figure 2. The CPE score correlates with immunofluorescence staining of the SARS-CoV 2 nucleocapsid protein at multiple SARS-CoV-2 titrations. Vero cells were infected with either MOI of 1 or 0.001 followed by serial 3,16-fold dilutions for every second dilution to be 10-fold. (**a**) Brightfield images were taken on day 3 post infection (one MOI = 0.3 representative of $n = 12$ is shown, scale bar = 1 mm). (**b**) Images were classified as indicated by retrained Resnet18. (**c**) 2 days post infection cells were stained with anti-SARS-CoV-2 nucleocapsid antibody (one MOI = 0.3 representative of $n = 12$ is shown, scale bar = 1 mm). (**d**) CPE Score was determined from bright field images on SARS-CoV-2 serial dilutions starting with MOI = 1 (blue line) or MOI = 0.001 (red line). Grey closed circles indicate positive control (MOI = 0.03), closed grey squares indicate negative control ($n = 4$ per well (control); $n = 12$ per well (dilution)). (**e**) Immunofluorescence of nucleocapsid staining of serial dilutions as indicated was quantified using ImageJ ($n = 4$ per well (control); $n = 12$ per well (dilution)). (**f**) Means of the quantification of immunofluorescence from each of 4 repeated experiments as in Figure 2e is shown as a dependence of means of 'CPEnet' ($n = 76$).

Next, we dissected images obtained from the nucleocapsid staining of positive controls (MOI 0.03) and negative controls into training tiles (Figure 3a,b). In total, 28 training images, 6 validation images, and 6 testing images were split into 2464 training tiles, 528 validation

Viruses **2021**, *13*, 610

tiles, and 528 testing tiles, respectively. Validation and test images were not used for training. We retrained a neural network 'IFnet' to detect the proportion of immunofluorescent image tiles with an achieved accuracy of 100% (Figure 3c,d, Figure S2b). In our examples, the averaged F-score to classify IF on the test dataset was 1 (Figure 3e). As expected, the 'IFnet' could detect the SARS-CoV-2 titrations (Figure 3f). We found a significant correlation between the 'IFnet' score equation (2) and the values obtained from the quantification of the immunofluorescence (Figure 3g) with R square = 0.97. Notably, the images quantified by 'IFnet' were the same images used for the quantification of the immunofluorescence, while the quantification of the CPE score was obtained on different plates one day later. Taken together, these data show that neural networks can be used to quantify SARS-CoV-2 cultures with or without immunostaining of viral proteins.

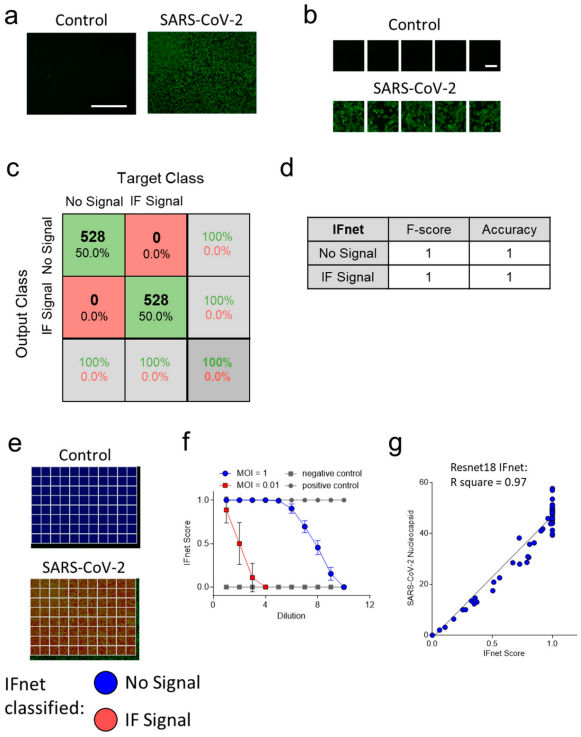


Figure 3. 'IFnet' can distinguish between infected and noninfected cells with SARS-CoV-2 in images taken after immunofluorescence staining. Vero cells were infected with either MOI of 1 or 0.001 followed by serial 3-fold dilutions. (**a**) Representative fluorescence images are shown as indicated 2 days post infection or without infection, after the cells were stained with anti-SARS-CoV-2 nucleocapsid antibody (n = 40, MOI = 0.03, scale bar = 1 mm). (**b**) Images were dissected into 2464 training, 528 validation, and 528 testing 224 × 224 image tiles for each group (scale bar = 100µm). (**c**) Confusion matrix for the 'IFnet' on the test dataset. (**d**) F-score and accuracy on the test dataset. (**e**) Fluorescence images from control (upper panel) and SARS-CoV-2 infected (lower panel) tissue cultures were classified by 'IFnet'. Image tiles are stained in blue (no signal) or red (IF Signal) on the original images. (**f**) IF Score was determined from immunofluorescence images of nucleocapsid staining on SARS-CoV-2 serial dilutions starting with MOI = 1 (blue line) or MOI = 0.001 (red line). Grey closed circles indicate positive control (MOI = 0.03), closed grey squares indicate negative control (n = 4 per well (control); n = 12 per well(dilution)). (**g**) Means of the quantification of immunofluorescence from each of 4 repeated experiments as in Figure 3e is shown in dependence of means of 'IFnet' (Resnet18) predicted IF Score the same experiments (n = 76).

Viruses **2021**, *13*, 610

3.3. 'CPETOXnet' Can Detect Inhibition of SARS-CoV-2 Replication and Identify Toxic Effects In Vitro

During screening compounds for antiviral effects against SARS-CoV-2, cell toxicity is an important parameter for drug screens. Treatment with staurosporine induces rapid cell death, which can be observed in tissue culture plates (Figure 4a). To confirm the toxicity of staurosporine, we carried out an apoptosis assay on Vero cells with significant differences in comparison to the control group (Figure S4). Accordingly, we dissected 72 images (51 for training, 11 for validation, and 10 for testing) from SARS-CoV-2 infected, staurosporine treated, and control cells into 4488 training, 968 validation, and 880 testing tiles, and retrained a 'CPETOXnet', which could predict cell toxicity and CPE (Figure 4b). An overall accuracy of 99.8% on the test data was achieved (Figure 4c, Figure S2c). In our examples, the averaged F-score to classify TOX on the test dataset was 0.9989 (Figure 4d). When we analyzed images taken from tissue cultures at day 2 after infection, 'CPETOXnet' could detect CPE in a proportion of image sections. However, since staurosporine already induced severe cell death 2 days after exposure, we observed a high TOX Score (Equation (3)) in these cultures (Figure 4e; Figure S4). Furthermore, when we analyzed images taken at day 3 post infection, 'CPETOXnet' reported a high CPE Score only for SARS-CoV-2 infected cells, while toxicity was only attributed to staurosporine treated cells (Figure 4e). These data indicate that 'CPETOXnet' can distinguish between late toxicity observed after staurosporine effects and SARS-CoV-2 mediated CPE.

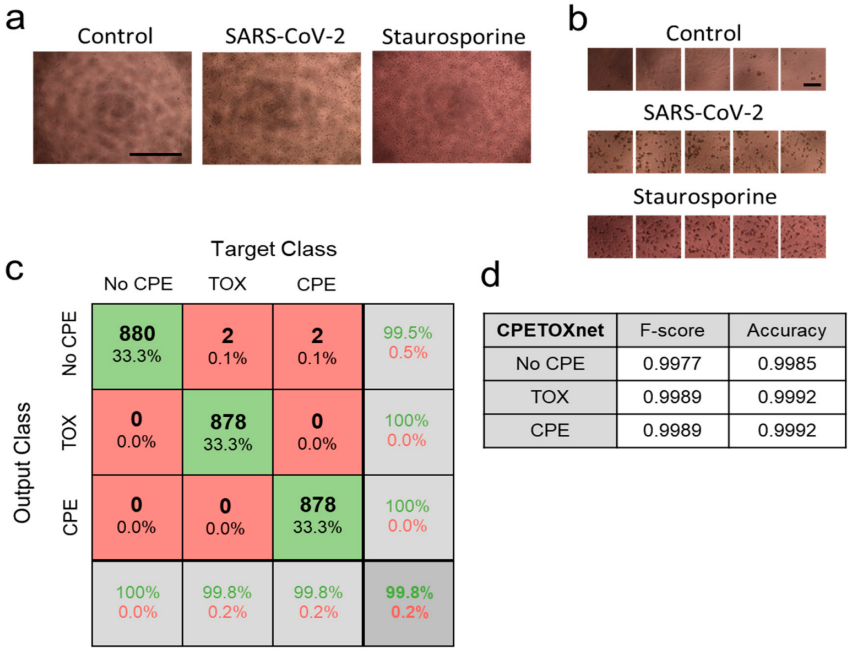


Figure 4. *Cont.*

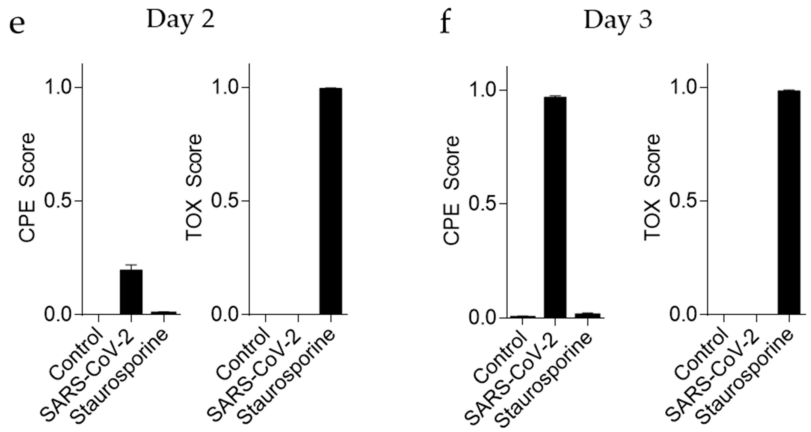


Figure 4. 'CPETOXnet' can distinguish between CPE and toxicity effects in images from SARS-CoV-2 infected or staurosporine treated cells. Images were taken of uninfected Vero cells, infected Vero cells (MOI of 0.03 with SARS-CoV-2), or treated with staurosporine (5 μM). (**a**) Representative images are shown as indicated 3 days after infection or treatment (n = 72, scale bar = 1 mm). (**b**) Images were dissected into 4488 training, 968 validation, and 880 testing 224 × 224 image tiles for each group (scale bar = 100 μm). (**c**) Confusionmatrix for the 'CPETOXnet' on the test data set. (**d**) F-score and accuracy on the test dataset. (**e**,**f**) CPE Score (left panels) and TOX Score (right panels) was attributed by 'CPETOXnet' to images obtained 2 days (**e**) and 3 days (**f**) after infection (MOI = 0.03) with SARS-CoV-2 or incubation with staurosporine (n = 60).

Next, we wondered whether 'CPETOXnet' could be used to identify drugs inhibiting SARS-CoV-2 replication. Accordingly, we treated cells with different concentrations of chloroquine, hydroxychloroquine, remdesivir, and emetine, which have been shown to reduce SARS-CoV-2 replication [6,7,27] and DMSO as a control. We monitored cultured cells for 48 and 72 h post infection. As expected, chloroquine was able to reduce the observed CPE at both time points after infection with no observable in vitro toxicity (Figure 5a, EC_{50} = 9.49 μM). Consistently, hydroxychloroquine was also able to reduce SARS-CoV-2 mediated CPE but lead to cell death at the highest concentration (Figure 5b, EC_{50} = 5.27 μM, CC_{50}~33.37 μM (out of tested range)). Remdesivir also had antiviral effects against SARS-CoV-2 without toxicity in vitro (EC_{50} = 1.12 μM) while emetine induced toxicity at higher concentrations, which likely also contributed to an increased CPE score in this setting (Figure 5c,d, Figure S5a, CC_{50} = 20.27 μM). Notably, since the observed CPE in these concentrations might reflect cell toxicity, the CC_{50} might be even lower. At lower concentrations emetine was able to limit SARS-CoV-2 replication (Figure 5d, Figure S5a–c, EC_{50} = 0.016 μM). In addition, we screened a library consisting of eight different inhibitors of heat shock protein 90 (HSP90), which has been identified as a protein relevant to SARS-CoV-2 infection [28]. Two days after infection an inhibition with PUH71, AUY922 (Luminespib), NVP-HSP990, EC144, PF-0429113, BIIB021, and Tanespimycin could be visualized, although the CPE could not be detected in all samples at this time point (Figure S6). However, three days after infection the toxicity and CPE score was increased. NVP-HSP990 (EC_{50} = 50.93 μM (out of tested range), CC_{50} = 94.64 μM (out of tested range)), EC144 (EC_{50} = 30.43 μM (out of tested range), CC_{50} = 34.62 μM (out of tested range)), PF-0429113 (EC_{50} = 2.625 μM, CC_{50} = 7.966 μM), BIIB021 (EC_{50} = 9.330 μM, CC_{50} = 10.13 μM (out of tested range)), and Tanespimycin (EC_{50} = 2.086 μM, CC_{50} = 2.940 μM) showed the EC_{50} and CC_{50} to be in close proximity, which suggests a transient effect in this experimental setting and requires further validation and in depth analysis (Figure S6). Taken together, we show that pretrained neural networks can classify SARS-CoV-2 cultures and can assist with quantification during drug screening.

Viruses **2021**, *13*, 610

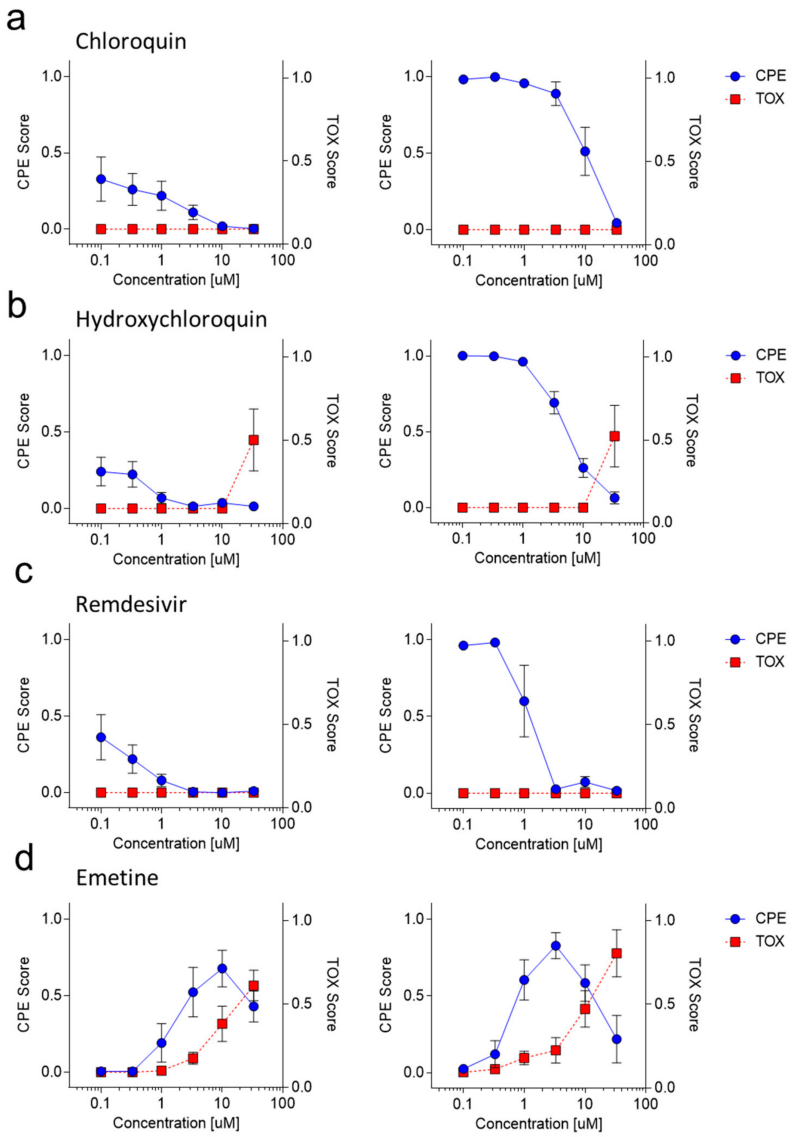


Figure 5. 'CPETOXnet' can detect toxicity and inhibition of SARS-CoV-2 propagation by compounds in images taken from unstained tissue cultures. (**a**–**d**) Vero cells were treated with the indicated concentration of chloroquine (**a**), hydroxychloroquine (**b**), remdesivir (**c**), and emetine (**d**). Images were taken 2 days (left panels) or 3 days (right panels) after infection with SARS-CoV-2 at an MOI of 0.03. CPE scores (blue) and TOX scores (red) were determined by 'CPETOXnet' and are shown in a concentration dependent manner (n = 5).

Next, the program was transferred to the open source machine learning framework PyTorch (Python) to enable a wide availability and a more user-friendly handling. As expected, a reanalysis of Figure 2d and subsequent reanalysis of the correlation of the obtained CPE Score analyzed by Python with the quantification of the immunofluorescence (Figure 2e) leads to a significant correlation with R square = 0.91 (Figure S7). The source code is available at: https://github.com/MolecularMedicine2/PyQoVi (Available from: 2 April 2021).

4. Discussion

In this study, we showed that retraining a deep convolutional neural network can assist in quantifying SARS-CoV-2 infected cell cultures via bright field images. We retrained a pretrained neural network to classify images from SARS-CoV-2 exposed cells. These images were taken on live, fully covered tissue cultures. Moreover, we retrained a 'CPETOXnet' to detect cell toxicity, as well as SARS-CoV-2 mediated CPE. 'CPETOXnet' could show the antiviral activity of chloroquine, hydroxychloroquine, remdesivir, and emetine. Furthermore, we demonstrated that hydroxychloroquine and emetine induced dose-dependent cell toxicity in vitro.

Deep neural networks were already used in medical applications to identify and predict mutations in cancer patients [14,16]. Furthermore, neural networks are used to classify CT scans during diagnosis of COVID-19 [20,21]. Our proposed neural network can identify CPE of SARS-CoV-2 cultures on brightfield images taken from closed tissue culture plates. This experimental setting, while very simple, also causes image artefacts through shadows and/or media. These artefacts are observed in almost all image files. Therefore, the individual image tiles can appear different in shape and color. Interestingly, since neural networks are trained on these image tiles, these artefacts are compensated for. 'CPETOXnet' could detect toxicity in emetine treated cells but also CPE when emetine was further titrated. This is expected, since lower concentrations would result in modest cell death, which might appear as a CPE. Likewise, this would suggest that strong CPE would be classified as toxicity by 'CPETOXnet'. Emetine inhibited SARS-CoV-2 mediated CPE at low concentrations suggesting that the classified CPE in emetine treated cells is likely attributed to the toxicity rather than the SARS-CoV-2 induced effects. Accordingly, the CC_{50} might be lower than attributed through the TOX score. These data suggest that detection of cell toxicity needs to be validated with standard techniques. Furthermore, drugs with absence of a TOX score in this experimental setting need to be further tested for cell toxicity. The CPE Score attributed by the convolutional neural network is not specific for SARS-CoV-2. Accordingly, observed effects from a screen have to be verified with specific methods such as quantitative PCR and/or immunofluorescence approaches.

Moreover, this method relies on CPE in brightfield images. Accordingly, when used in our described assay, it will only show antiviral effects of drugs affecting SARS-CoV-2 induced CPE. Specifically, virucidal drugs, virus neutralizing drugs, or drugs affecting viral entry might show a prominent inhibition by using this assay. However, drugs affecting viral replication will only be detected in this experimental setting if CPE in SARS-CoV-2 cultures is inhibited. To assess the effect of drugs on viral replication in depth, infected cells could be washed shortly after infection with collection of the supernatant over time. The supernatant should be used to infect a fresh set of cells to determine the SARS-CoV-2 titer, which could be also performed with the use of 'CPEnet'.

Although our described approach is simple, it was successfully able to validate compounds that might be useful in early clinical therapy regimens during SARS-CoV-2 infections. Using the described approach, the retrained neural network can be used to detect a variety of effects observed in tissue culture suggesting a broad applicability. Hence, it is tempting to speculate that the described procedures can be used early during an outbreak when there might be a shortage of specific antibodies and/or RNA quantification tools for anti-pathogen testing. However, the data generated is not specific and has to be verified by pathogen specific methods, since contamination with other viruses or bacteria could establish a considerable bias in this setting. Moreover, the use of retrained neural networks in quantifying immunofluorescence images is comparable to other quantification methods. Notably, we also performed our analyses on a single central processing unit (CPU) and with shorter training time on a graphics processing unit (GPU). Accordingly, this approach can be used without major hardware requirements. The biological variance between experiments and other factors such as exposure time, brightfield intensity, and cell density could impact the accuracy of the neural network. Hence, we suggest to collect training images in every specific laboratory setting and from different experiments over

a period of time and a variety of tissue cultures to correct for the variability observed between experiments.

In conclusion, we show the use of deep convolutional neural networks to quantify images during experimental settings of SARS-CoV-2 cultures.

Supplementary Materials: The following are available online at https://www.mdpi.com/article/10.3390/v13040610/s1, Figure S1: ResNet18 Architecture, Figure S2: ROC curve of the trained networks, Figure S3: SARS-CoV-2 infection induces CPE at day 3 post infection, Figure S4: Apoptosis assay of Vero cells stimulated with Staurosporine, Figure S5: Emetine induces toxicity and CPE in infected Vero cells, Figure S6: Drugscreening of HSP90i, Figure S7: Adjusting the Analysis of the CPE Score for Python.

Author Contributions: Conceptualization, J.W. and P.A.L.; methodology, J.W., R.M.K., P.S., P.N.O., L.M., H.S., J.N.K., A.B., A.A.P., K.S.L., and P.A.L.; software, R.M.K. and P.A.L.; validation, J.W., R.M.K., P.S., P.N.O., L.M., H.S., J.N.K., A.B., A.A.P., K.S.L., and P.A.L.; formal analysis, J.W., R.M.K., P.N.O., L.M., and P.A.L.; investigation, J.W., R.M.K., P.S., P.N.O., L.M., and P.A.L.; resources, H.S., S.B., and P.A.L.; data curation, J.W., R.M.K., P.N.O., L.M., and P.A.L.; writing—original draft preparation, P.A.L.; writing—review and editing, J.W., R.M.K., P.S., P.N.O., L.M., H.S., S.B., J.N.K., A.B., A.A.P., and K.S.L.; visualization, J.W., R.M.K., P.S., and P.A.L.; supervision, P.A.L.; project administration, P.A.L.; funding acquisition, P.A.L. All authors have read and agreed to the published version of the manuscript.

Funding: This study was funded by the German Research Council (SFB974, RTG1949), the Jürgen Manchot Foundation and the 'Stiftung für AIDS-Forschung, Düsseldorf'. Raphael M. Kronberg receives funding for his doctorate from the Anton-Betz Foundation of the Rheinische Post.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The source code is available at: https://github.com/MolecularMedicine2/PyQoVi. (Available from: 2 April 2021).

Acknowledgments: We would like to thank the technical assistance of Björn Wefers. Computational infrastructure and support were provided by the Centre for Information and Media Technology at Heinrich Heine University Düsseldorf.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Karagiannidis, C.; Mostert, C.; Hentschker, C.; Voshaar, T.; Malzahn, J.; Schillinger, G.; Klauber, J.; Janssens, U.; Marx, G.; Weber-Carstens, S.; et al. Case characteristics, resource use, and outcomes of 10 021 patients with COVID-19 admitted to 920 German hospitals: An observational study. *Lancet Respir. Med.* **2020**, *8*, 853–862. [CrossRef]
2. Hoffmann, M.; Kleine-Weber, H.; Schroeder, S.; Krüger, N.; Herrler, T.; Erichsen, S.; Schiergens, T.S.; Herrler, G.; Wu, N.-H.; Nitsche, A. SARS-CoV-2 cell entry depends on ACE2 and TMPRSS2 and is blocked by a clinically proven protease inhibitor. *Cell* **2020**, *181*, 271–280.e8. [CrossRef]
3. Battegay, M.; Cooper, S.; Althage, A.; Bänziger, J.; Hengartner, H.; Zinkernagel, R.M. Quantification of lymphocytic choriomeningitis virus with an immunological focus assay in 24- or 96-well plates. *J. Virol. Methods* **1991**, *33*, 191–198. [CrossRef]
4. Dulbecco, R.; Vogt, M. Plaque Formation and Isolation of Pure Lines with Poliomyelitis Viruses. *J. Exp. Med.* **1954**, *99*, 167–182. [CrossRef] [PubMed]
5. Herzog, P.; Drosten, C.; Müller, M.A. Plaque assay for human coronavirus NL63 using human colon carcinoma cells. *Virol. J.* **2008**, *5*, 138. [CrossRef]
6. Wang, M.; Cao, R.; Zhang, L.; Yang, X.; Liu, J.; Xu, M.; Shi, Z.; Hu, Z.; Zhong, W.; Xiao, G. Remdesivir and chloroquine effectively inhibit the recently emerged novel coronavirus (2019-nCoV) in vitro. *Cell Res.* **2020**, *30*, 269–271. [CrossRef] [PubMed]
7. Liu, J.; Cao, R.; Xu, M.; Wang, X.; Zhang, H.; Hu, H.; Li, Y.; Hu, Z.; Zhong, W.; Wang, M. Hydroxychloroquine, a less toxic derivative of chloroquine, is effective in inhibiting SARS-CoV-2 infection in vitro. *Cell Discov.* **2020**, *6*, 1–4. [CrossRef]
8. Gautret, P.; Lagier, J.C.; Parola, P.; Hoang, V.T.; Meddeb, L.; Mailhe, M.; Doudier, B.; Courjon, J.; Giordanengo, V.; Vieira, V.E.; et al. Hydroxychloroquine and azithromycin as a treatment of COVID-19: Results of an open-label non-randomized clinical trial. *Int. J. Antimicrob. Agents* **2020**, *56*, 105949. [CrossRef]

9. Boulware, D.R.; Pullen, M.F.; Bangdiwala, A.S.; Pastick, K.A.; Lofgren, S.M.; Okafor, E.C.; Skipper, C.P.; Nascene, A.A.; Nicol, M.R.; Abassi, M.; et al. A Randomized Trial of Hydroxychloroquine as Postexposure Prophylaxis for Covid-19. *N. Engl. J. Med.* **2020**, *383*, 517–525. [CrossRef]
10. Skipper, C.P.; Pastick, K.A.; Engen, N.W.; Bangdiwala, A.S.; Abassi, M.; Lofgren, S.M.; Williams, D.A.; Okafor, E.C.; Pullen, M.F.; Nicol, M.R.; et al. Hydroxychloroquine in Nonhospitalized Adults with Early COVID-19: A Randomized Trial. *Ann. Intern. Med.* **2020**, *173*, 623–631. [CrossRef]
11. Borba, M.G.S.; Val, F.F.A.; Sampaio, V.S.; Alexandre, M.A.A.; Melo, G.C.; Brito, M.; Mourão, M.P.G.; Brito-Sousa, J.D.; Baía-da-Silva, D.; Guerra, M.V.F. Effect of high vs low doses of chloroquine diphosphate as adjunctive therapy for patients hospitalized with severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) infection: A randomized clinical trial. *JAMA Netw. Open* **2020**, *3*, e208857. [CrossRef]
12. Beigel, J.H.; Tomashek, K.M.; Dodd, L.E.; Mehta, A.K.; Zingman, B.S.; Kalil, A.C.; Hohmann, E.; Chu, H.Y.; Luetkemeyer, A.; Kline, S. Remdesivir for the treatment of Covid-19—Preliminary report. *N. Engl. J. Med.* **2020**. [CrossRef]
13. Le Cun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
14. Coudray, N.; Ocampo, P.S.; Sakellaropoulos, T.; Narula, N.; Snuderl, M.; Fenyö, D.; Moreira, A.L.; Razavian, N.; Tsirigos, A. Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. *Nat. Med.* **2018**, *24*, 1559–1567. [CrossRef]
15. Schmauch, B.; Romagnoni, A.; Pronier, E.; Saillard, C.; Maillé, P.; Calderaro, J.; Kamoun, A.; Sefta, M.; Toldo, S.; Zaslavskiy, M.; et al. A deep learning model to predict RNA-Seq expression of tumours from whole slide images. *Nat. Commun.* **2020**, *11*, 1–15. [CrossRef]
16. Kather, J.N.; Pearson, A.T.; Halama, N.; Jäger, D.; Krause, J.; Loosen, S.H.; Marx, A.; Boor, P.; Tacke, F.; Neumann, U.P.; et al. Deep learning can predict microsatellite instability directly from histology in gastrointestinal cancer. *Nat. Med.* **2019**, *25*, 1054–1056. [CrossRef]
17. Mobadersany, P.; Yousefi, S.; Amgad, M.; Gutman, D.A.; Barnholtz-Sloan, J.S.; Vega, J.E.V.; Brat, D.J.; Cooper, L.A.D. Predicting cancer outcomes from histology and genomics using convolutional networks. *Proc. Natl. Acad. Sci. USA* **2018**, *115*, E2970–E2979. [CrossRef]
18. Saillard, C.; Schmauch, B.; Laifa, O.; Moarii, M.; Toldo, S.; Zaslavskiy, M.; Pronier, E.; Laurent, A.; Amaddeo, G.; Regnault, H. Predicting Survival After Hepatocellular Carcinoma Resection Using Deep Learning on Histological Slides. *Hepatology* **2020**, *72*, 2000–2013. [CrossRef]
19. Kather, J.N.; Krisam, J.; Charoentong, P.; Luedde, T.; Herpel, E.; Weis, C.-A.; Gaiser, T.; Marx, A.; Valous, N.A.; Ferber, D.; et al. Predicting survival from colorectal cancer histology slides using deep learning: A retrospective multicenter study. *PLoS Med.* **2019**, *16*, e1002730. [CrossRef]
20. Zhang, K.; Liu, X.; Shen, J.; Li, Z.; Sang, Y.; Wu, X.; Zha, Y.; Liang, W.; Wang, C.; Wang, K. Clinically applicable AI system for accurate diagnosis, quantitative measurements, and prognosis of covid-19 pneumonia using computed tomography. *Cell* **2020**, *181*, 1423–1433. [CrossRef]
21. Harmon, S.A.; Sanford, T.H.; Xu, S.; Turkbey, E.B.; Roth, H.; Xu, Z.; Yang, D.; Myronenko, A.; Anderson, V.; Amalou, A.; et al. Artificial intelligence for the detection of COVID-19 pneumonia on chest CT using multinational datasets. *Nat. Commun.* **2020**, *11*, 1–7. [CrossRef] [PubMed]
22. Ramani, A.; Müller, L.; Ostermann, P.N.; Gabriel, E.; Abida-Islam, P.; Müller-Schiffmann, A.; Mariappan, A.; Goureau, O.; Gruell, H.; Walker, A. SARS-CoV-2 targets neurons of 3D human brain organoids. *EMBO J.* **2020**, *39*, e106230. [CrossRef] [PubMed]
23. Walker, A.; Houwaart, T.; Wienemann, T.; Vasconcelos, M.K.; Strelow, D.; Senff, T.; Hülse, L.; Adams, O.; Andree, M.; Hauka, S. Genetic structure of SARS-CoV-2 reflects clonal superspreading and multiple independent introduction events, North-Rhine Westphalia, Germany, February and March 2020. *Eurosurveillance* **2020**, *25*, 2000746. [CrossRef]
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. *arXiv* **2015**, arXiv:1512.03385.
25. Canziani, A.; Paszke, A.; Culurciello, E. An Analysis of Deep Neural Network Models for Practical Applications. *arXiv* **2017**, arXiv:1605.07678.
26. Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
27. Choy, K.-T.; Wong, A.Y.-L.; Kaewpreedee, P.; Sia, S.F.; Chen, D.; Hui, K.P.Y.; Chu, D.K.W.; Chan, M.C.W.; Cheung, P.P.-H.; Huang, X. Remdesivir, lopinavir, emetine, and homoharringtonine inhibit SARS-CoV-2 replication in vitro. *Antivir. Res.* **2020**, *178*, 104786. [CrossRef]
28. Emanuel, W.; Kirstin, M.; Vedran, F.; Asija, D.; Theresa, G.L.; Roberto, A.; Filippos, K.; David, K.; Katja, H.; Salah, A. Transcriptomic profiling of SARS-CoV-2 infected human cell lines identifies HSP90 as target for COVID-19 therapy. *Iscience* **2021**, *24*, 102151.

Chapter 5

Improving Treatment Personalization using Deep Learning

In this chapter, we present our use case concerning treatment personalization: Optimal Acquisition Sequence for AI-assisted Brain Tumor segmentation. We first briefly introduce the topic "Improved treatment personalization using Machine Learning" and subsequently present our approach to determining the optimal sequence order for a patient, which undergoing MRI scanning, given on the time budget as shown in Figure 5.1.

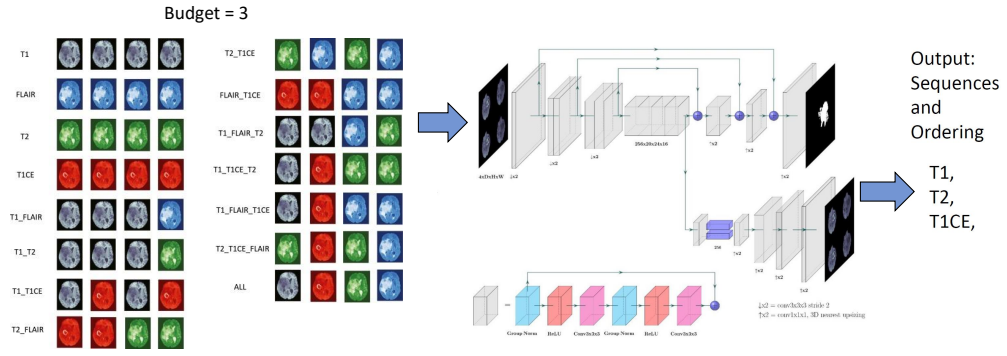


Figure 5.1: Based on this Deep Learning approach to optimize the MRI sequence acquisition order, we can choose the best order for a given acquire time that varies for each patient according to their level of pain, physical condition, and fears. The image is based on Kronberg et al., 2022b.

5.1 Improving Treatment Personalization using Machine Learning

A new patient-oriented healthcare (PH) approach that desires to improve the traditional healthcare system is emerging. By collecting the patient data from patient electronic health records,

Internet of Things sensor devices, wearables, and mobile devices, web-based information and social media is the approach used by PH. For the improvement of patient self-management, disease prediction, disease progression monitoring techniques, and clinical intervention, PH applies Artificial Intelligence techniques to the collected dataset and analyzes the collected data. Especially Machine Learning and Deep Learning techniques are widely used to develop data-driven models. These analytic models are integrated into clinical decision support systems and different healthcare service applications. The Machine Learning models analyze the collected data from sensor devices and a variety of sources to identify relevant patterns and health conditions of the patient. Based on the data patterns, the clinical decision support systems and healthcare apps provide lifestyle advice, care plans, and special treatment for the patient. In contrast, the current clinical practice is that doctors mostly prescribe medicines using trial and error and a one-size-fits-all approach. While most patients may respond to a particular drug in a given dose, a handful of people may either have a minimal effect or suffer severe side effects from the same medication. PH and the use of Machine Learning can improve the quality of care and simultaneously decrease costs. Furthermore, it can help to determine the optimal therapeutic approach with the fewest side effects for individual patients (Ahamed and Farid, 2018).

5.2 Optimal Acquisition Sequence for AI-assisted Brain Tumor Segmentation Under the Constraint of the Largest Information Gain per Additional MRI Sequence

In this section, we provide an overview of the contributions and impact of our paper Kronberg et al., 2022b:

Raphael Kronberg, Dziugas Meskelevicius, Christian Rubbert, Michael Sabel, Markus Kollmann and Igor Fischer

"Optimal acquisition sequence for AI-assisted brain tumor segmentation under the constraint of largest information gain per additional MRI sequence"

In: *Neuroscience Informatics*Volume 2, Issue 4, 2022, 100053.

Main Results in Simple Terms

In the cited manuscript, we dealt with brain tumours, and more specifically, with the MRI images thereof. It is important to note that there are several different types of MRI images and that we examined four of these types of images, called sequences, in more detail. The sequences differ in the addition of contrast agents and the settings of the MRI scanner. In most cases, these images are processed one after the other in a certain order, according to a recommendation guideline.

During MRI scans, a patient must remain absolutely motionless and silent in the MRI machine. This is difficult for many patients because they cannot remain motionless for the entire time due to a general poor physical condition, acute pain, or claustrophobia. When patients move around while a scan is in progress, blurred images called artefacts are created which usually render the MRI images to be unusable. Therefore, the images are often interrupted prematurely in the case of increasing movement or restlessness on the part of the patient. The images are used, for example, to prepare for operations, to follow up operations, or as simple progress checks. Since information concerning the size and localization of a tumour is crucial for treatment and diagnosis, it is important to record this data.

To address this challenge, we set ourselves the task of optimizing the softening of the various sequences in such a way that the maximum amount of information is added with each additional sequence. To measure this information, we used a segmentation algorithm, i.e., the algorithm can enter – in a master pixel – exactly what kind of tumor is found or whether it is non-tumor tissue. This algorithm is a neural network that is trained on the basis of ground truths labeled by experts, so-called segmentation maps, and the MRI images themselves. We calculated all possible combinations of sequence tulips (for one sequence, two sequences, three sequences, and four sequences) while considering the constraints of physical feasibility (e.g.,

some contrast agents have to be administered in a staggered manner) by feeding the neural network only with the appropriate sequences at any given time. Based on the segmentation accuracy, we could determine the optimal sequence for one sequence, for two sequences, and for three sequences. Since with four sequences, all four sequences are recorded, the information content is the same and independent of the recording order. We could show that our proposed order for only three sequences gives significantly better results than the recommended order from the guideline.

Furthermore, we visualized the results of our segmentation.

Remark:We did not compare our segmentation accuracy with other groups because the segmentation algorithm in our optimization method is interchangeable.

Summary/Abstract

To distinguish healthy from highly vital tumor tissue in an automatic segmentation approach, three sequences suffice and the information in T2 or FLAIR imaging is highly redundant. Our experiments show that particularly the T1CE sequence is very important for a good segmentation accuracy, even for tumor edema. We therefore propose to obtain imaging in the order [T1, T2, T1CE, FLAIR] to maximize information gain in a prematurely terminated MRI examination (Kronberg et al., 2022b).

Personal Contribution

Formulated sentences

Raphael Marvin Kronberg (R.M.K.) performed computational experiments and data analysis, e.g. he calculated the metrics and statistics for the different Deep Neural Networks. He discussed the data and wrote the draft of the paper, including the figures. The implementation of the Deep Neural Networks and the pipeline in Python using as framework was carried out by R.M.K..

Bullet points (CRediT version)

Conceptualization: Raphael M. Kronberg, Christian Rubbert, Igor Fischer; Data curation: Christian Rubbert; Formal Analysis: Raphael M. Kronberg, Markus Kollmann, Igor Fischer; Funding acquisition: Michael Sabel, Igor Fischer; Investigation: Raphael Kronberg; Methodology: Raphael Kronberg, Markus Kollmann, Christian Rubbert, Igor Fischer; Project administration: Igor Fischer; Resources: Michael Sabel, Igor Fischer; Software: Raphael M. Kronberg; Supervision: Igor Fischer ; Validation: Dziugas Meskelevicius, Christian Rubbert, Igor Fischer; Visualization: Raphael Kronberg, Igor Fischer; Writing – original draft: Raphael Kro-

nberg; Writing – review & editing: Dziugas Meskelevicius, Michael Sabel, Markus Kollmann, Christian Rubbert, Igor Fischer (Kronberg et al., 2022b).

Importance of the Research and Contribution to this Thesis

The automated Deep Learning-based segmentation of brain tumors on MRI images serves as an exemplary use case of how artificial intelligence can support doctors in everyday clinical practice. The segmentations can be used for diagnosis and surgery planning. By segmenting in 3D, doctors can get a good overview of the size and position of a tumour. The segmentation in our paper shows the different components of the tumour in sufficient detail, this subdivision is preferred, to separate Edema of the surrounding brain tissue from the metabolically active Tumor. Therefore, it answers our third research question: Which MRI Sequence should researchers acquire under a fixed time budget (depending on the patient's condition) for a good segmentation result for brain tumors?

Neuroscience Informatics 2 (2022) 100053

Contents lists available at ScienceDirect

Neuroscience Informatics

www.elsevier.com/locate/neuri

Artificial Intelligence in Brain Informatics

Optimal acquisition sequence for AI-assisted brain tumor segmentation under the constraint of largest information gain per additional MRI sequence

Raphael M. Kronberg [a,b,c], Dziugas Meskelevicius [c], Michael Sabel [c], Markus Kollmann [a], Christian Rubbert [d,*], Igor Fischer [c]

[a] Mathematical Modelling of Biological Systems, Heinrich-Heine University, Universitätsstr. 1, Düsseldorf, 40225, NRW, Germany
[b] Medical Faculty, Molecular Medicine II, Heinrich-Heine University, Universitätsstr. 1, Düsseldorf, 40225, NRW, Germany
[c] Medical Faculty, Department of Neurosurgery, Heinrich-Heine University, Moorenstr. 5, Düsseldorf, 40225, NRW, Germany
[d] University Dusseldorf, Medical Faculty, Department of Diagnostic and Interventional Radiology, Dusseldorf, D-40225, Germany

ARTICLE INFO

Article history:
Received 21 November 2021
Received in revised form 4 February 2022
Accepted 7 February 2022

Dataset link: https://www.med.upenn.edu/cbica/brats2020/data.html

Keywords:
Deep learning
Tumor segmentation
Glioma
Magnetic resonance imaging
Imaging protocol

ABSTRACT

Purpose: Different imaging sequences (T1 etc.) depict different aspects of a brain tumor. As clinical MRI examinations of the brain might be terminated prematurely, not all sequences may be acquired, decreasing the performance of automated tumor segmentation. We attempt to optimize the order of sequences, to maximize information gain in case of incomplete examination.
Methods: For segmentation we used the winner algorithm of the Brain Tumor Segmentation challenge 2018, trained on the BraTS 2020 dataset, with the objective to segment necrotic core, peritumoral edema, and enhancing tumor. We compared the segmentation performance for all combinations of sequences, using the Dice score (DS) as the primary metric. We compare the results with those which would be obtained by attempting to follow the consensus recommendations for brain tumor imaging [T1, FLAIR, T2, T1CE].
Results: The average segmentation accuracy varies between 0.476 for T1 only and 0.751 for the full set of sequences. T1CE has a high information content, even regarding peritumoral edema and information of T2 and FLAIR were highly redundant. The optimal order of sequences appears to be [T1, T2, T1CE, FLAIR]. Comparing segmentation accuracy after each fully acquired sequence, the first sequence (T1) is the same for both, DS for [T1, T2] (proposed) is 6.2% higher than [T1, FLAIR] (aborted recommendations), and [T1, T2, T1CE] (proposed) is 34.8% higher than [T1, FLAIR, T2] (aborted recommendations).
Conclusion: For the purpose of optimal deep-learning-based segmentation purposes in potentially incomplete MRI examinations, the T1CE sequence should be acquired as early as possible.

1. Introduction

Automated MRI based brain tumor segmentation is evolving to become a critical step in precision patient care. For further diagnosis, prognosis and treatment, accurate delineation of tumorous tissue is crucial. In a clinical setting reproducibility, robustness, and quality of segmentations are critical [7,28,15,22,32,33].

In the United States, 32% of the 368,117 brain tumor cases registered between 2009 and 2013 were classified as malignant. About 25.8% of these tumors were high grade gliomas, which include Glioblastoma Multiforme (GBM) [27]. GBM is infamously the most common malignant primary brain tumor and constitutes 45% of all the malignant central nervous system (CNS) tumors, or 80% of the malignant primary CNS tumors [26]. With current treatment options, the median overall survival for patients with newly diagnosed GBM is between 12 and 18 months [16,30].

The gold standard for imaging of gliomas is MRI with and without gadolinium contrast. Accurate localization and segmentation of brain tumors in MR images is essential for diagnosis, growth rate prediction, and treatment planning and may include identifying active vascularized tumor mass in high-grade or non-vascularized in

* Corresponding author.
E-mail addresses: Raphael.Kronberg@hhu.de (R.M. Kronberg), Dziugas.Meskelevicius@med.uni-duesseldorf.de (D. Meskelevicius), Michael.Sabel@med.uni-duesseldorf.de (M. Sabel), Markus.Kollmann@hhu.de (M. Kollmann), Christian.Rubbert@med.uni-duesseldorf.de (C. Rubbert), Igor.Fischer@med.uni-duesseldorf.de (I. Fischer).

https://doi.org/10.1016/j.neuri.2022.100053

low-grade gliomas, tumor necrosis, and the surrounding edema. Manual segmentation of brain tumor and related abnormal tissue from healthy brain tissue is a tedious task, requiring expert knowledge of brain anatomy, neuro-oncology and radiology, and may suffer from inter-rater reliability problems [28,34,2]. Computer-supported segmentation could alleviate some of these problems, especially given the improvements in segmentation accuracy due to better availability of computing resources and the advances in algorithms, which has allowed current state-of-the-art deep learning approaches to transition from using single-slice 2D information to integrating 3D information from adjacent slices. Some tumors, such as WHO Grade I meningiomas, may be easily segmented due to their extra-axial, non-invasive growth pattern with sharply demarcated borders [25,3]. Segmentation of gliomas is more challenging. Gliomas have a distinct infiltrating growth pattern and, together with their surrounding edema, often form diffuse margins and show no distinct border. This phenotype renders them difficult to segment. Therefore, information from more than one MRI sequence is usually combined. The consensus recommendations for standardized brain tumor imaging [12] recommend the acquisition of 1) T1, 2) fluid attenuation inversion recovery (FLAIR), 3) diffusion-weighted imaging (DWI), 4) T2 and 5) contrast enhanced T1 (T1CE).

However, in radiological clinical practice, it is not always possible to acquire all sequences with a sufficient quality. Reasons may include contrast agent intolerance, motion artifacts or even premature termination of a scan in agitated patients or due to clinical deterioration of the patient. It is therefore preferable to acquire the most informative sequences as early as possible in order to obtain the highest achievable automatic segmentation accuracy despite incomplete data.

The Multimodal Brain Tumor Segmentation (BraTS) challenge was established in 2012. Each year a public dataset comprised of T1, T2, FLAIR and T1CE imaging data (369 patients in 2020 Training data) is made available. For a large set of patients, expert segmentations of different aspects of the tumor, such as the contrast-enhancing tumor, are made available as a training dataset. For a smaller subset of patients (the test dataset), the expert labels are not made openly available. Participants may use the training data to develop and optimize segmentation algorithms, apply them to the test data and upload the results of these segmentations to the challenge's website, where they are compared to the expert label only available to the BraTS challenge organizers. Different metrics of segmentation accuracy may then be used to rank the participant's entry.

Our study aims to optimize the brain tumor segmentation accuracy in patients, for whom it may not be possible to acquire all sequences. In order to identify the most informative order of sequences and to better understand the limitations of incomplete data for automatic segmentation, the information content of each sequence and their combinations needs to be quantified. A state-of-the-art machine learning segmentation algorithm (an artificial deep neural network) is trained on the BraTS 2020 data, using all possible combinations of sequences. Segmentation accuracy is quantified starting with single-sequence models (e.g. only T1), combinations of sequences such as [T1, T2] (missing T2 and T1CE information) and finally including all sequences as the best possible model. Based on these findings, we propose an optimal order of imaging and compare the information gain after each fully acquired sequence to the order in the consensus recommendations.

2. Methods

The data used in this study are publicly available, distributed through the BraTS challenge. The local ethics committee approved the study. The requirement for written informed consent was waived.

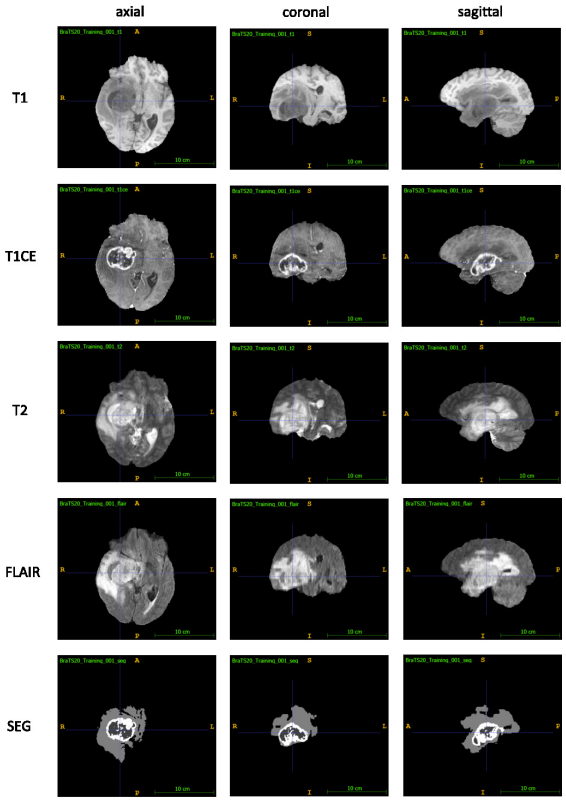


Fig. 1. Exemplary images depicting a WHO Grade IV Glioblastoma Multiforme (GBM) from the Multimodal Brain Tumor Segmentation Challenge (BraTS) 2020 dataset: Each row shows the axial, coronal and a sagittal view of each of the T1, T1CE, T2 and FLAIR sequences as well as the expert segmentation of the tumor (SEG): Necrotic core and non-enhancing tumor (center, dark grey), enhancing tumor (white, surrounding the necrotic core), peritumoral edema (light grey).

2.1. Data

According to consensus recommendations for a standardized brain tumor imaging protocol in clinical trials [12], an optimal MRI protocol should be comprised of a 3D T1, followed by an axial 2D (optionally 3D) FLAIR, axial 2D DWI, axial 2D T2, and, finally, a 3D T1CE.

As a benchmark dataset for brain tumor segmentation, we used the BraTS 2020 dataset, which provides T1, FLAIR, T2 and T1CE images. In the BraTS 2020 data, only the T1CE imaging was originally acquired using an axial 3D MRI acquisition, whereas the other sequences were acquired as 2D MRI acquisitions with variable characteristics [5]. However, the BraTS 2020 as well as previous BraTS challenge data is only made available as uniformly pre-processed 3D image volumes with a voxel size of 1x1x1 mm, which we have used for our analyses. [23,5,4]. The annotations, done by experts, include the contrast-enhancing tumor (ET), the peritumoral edema (ED), the necrotic and non-enhancing tumor core (NCR) and the background (non-tumor voxel) (BG), we don't use the combined labels whole tumor and tumor core. Examples are shown in Fig. 1.

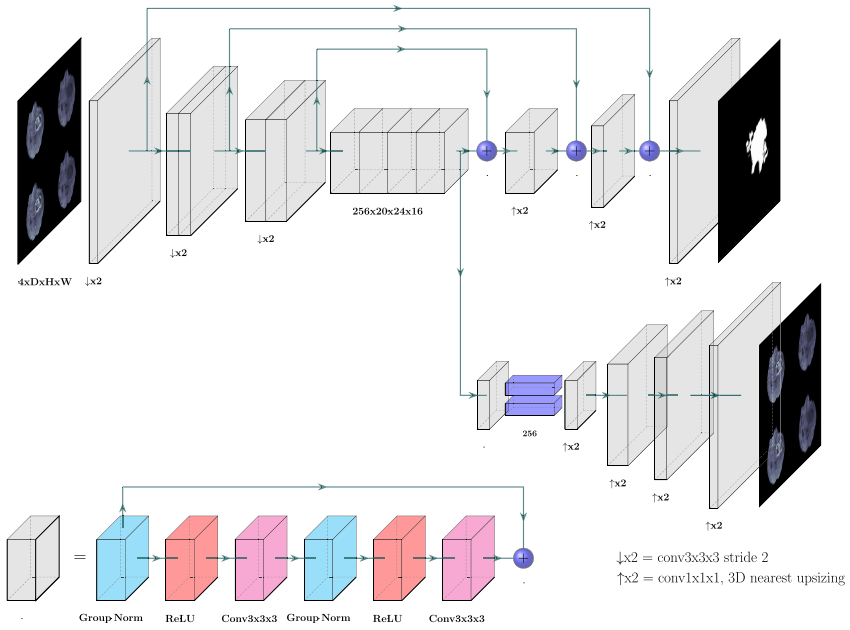


Fig. 2. The network architecture of the VAE-RES-NET, showing the input and the output for a sample from a set containing all four sequences ALL = [T1, T2, FLAIR, T1CE].

2.2. Overview of the experiment

The goal of our analysis is to propose the optimal order of sequences to maximize brain tumor segmentation accuracy using artificial neural networks in case when the MRI examination is terminated prematurely.

The available BraTS 2020 data with expert annotations is split into a training and a test dataset on a patient basis. For the training dataset, 80% of the patients were used (295 of 369 patients, including 20.6% of low-grade and 79.4% of high-grade Gliomas). The test dataset is comprised of the remaining 20% of the patients ($n = 74$, including 18, 9% of low-grade and 81, 1% of high-grade Gliomas).

Due to the use of many non-linear functions in neural networks, it is not possible to simply establish the performance of a model on T1 and T2 sequences separately, and then simply add the accuracies to establish the performance of the model trained on {T1, T2}. To quantify the segmentation accuracy of a state-of-the-art segmentation algorithm when limited by incomplete data, we therefore defined the following sets of sequences Fig. 3. Using each of sequence sets, a state-of-the-art segmentation model is trained on the training dataset. Since the model requires four inputs (T1, FLAIR, T2, T1CE), some sequences may have been used multiple times depending on the sequence set (see subsection 2.4). Each model is then applied on the corresponding sequence sets in the test dataset and segmentation accuracy is calculated using expert annotations. Using these results and based on common clinical imaging constraints, we intend to propose an optimal order of imaging sequences (subsection 2.8).

2.3. Data preprocessing and augmentation

In a first step, the borders of each image volume were cropped to remove empty space, yielding an image volume sized (160; 192; 128) with the brain at the center. Deep Learning is known to perform well on many segmentation tasks and has been applied in the BraTS challenge several times. However, learning in deep neural networks relies on vast quantities of data and is prone to overfitting when applied on small datasets, which means that the model is too well adapted to the limited feature space of the training dataset and may not generalize to previously unseen data. Data augmentation is a solution to the often size-limited datasets in medical imaging [29]. Our augmentation pipeline includes unit noise, normalization, vertical and horizontal flipping (with a probability of 0.5).

2.4. Architecture

We chose the latest available fully published winner of the BraTS challenge available at the beginning of our experiments, a variational auto-encoder residual neural network (VAE-RES-NET, BraTS 2018) [24] and adapted it for our experiments. This segmentation approach follows an encoder-decoder based CNN architecture with an asymmetrically larger encoder to extract image features and a smaller decoder to reconstruct the segmentation mask. In addition, it includes a branch to the encoder endpoint to reconstruct the original image, similar to an autoencoder architecture [24]. The motivation for using the auto-encoder branch is to add guidance and regularization to the encoder part, since the training dataset size is limited [24]. For the output layer we used a logistic sigmoid activation function. The network architecture is shown in Fig. 2.

Using the same model in all experiments is suitable in our case because the number of parameters, channels, and input sequences are kept constant. The network has four input channels, but the number of used sequences varies between k = 1 and k = 4, so we use some input sequences multiple times. We show the configuration in the Fig. 3. Keeping the number of parameters and input channels constant ensures a fair comparison.

2.5. Loss

We used the same loss function as [24]:

$$L = L_D + \lambda L_{\text{VAE}}, \tag{1}$$

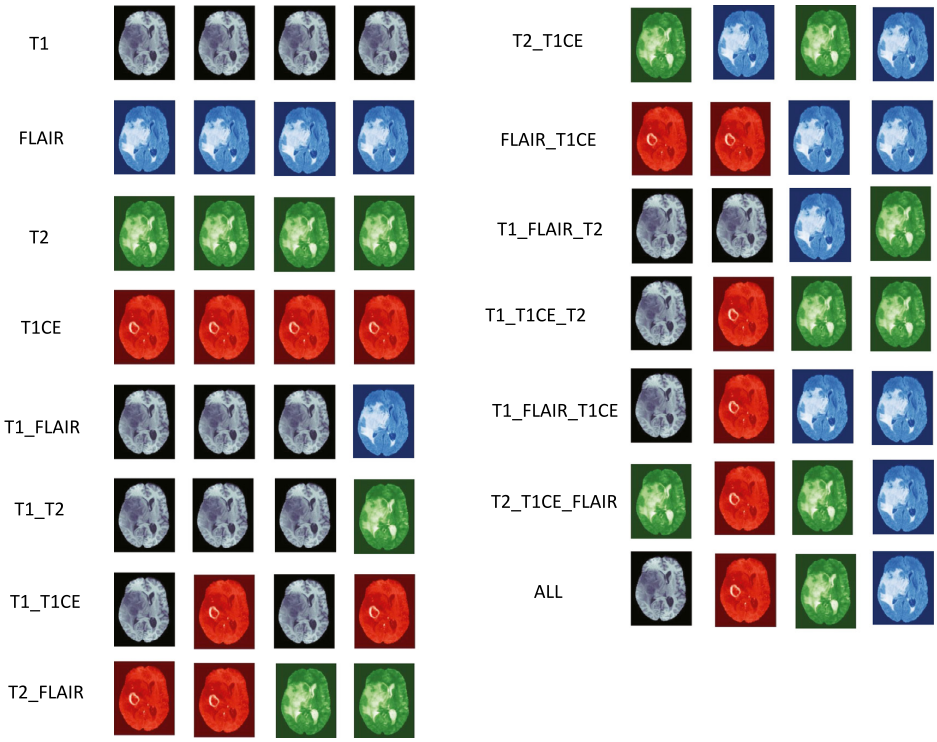


Fig. 3. Input of the VAE-RES-NET per channel. For each subset of sequences. (Channel 1, Channel 2, Channel 3 and Channel 4, from left to right in one row.)

with $\lambda = 10^{-1}$. The two terms are defined as:

$$L_D = 1 - \frac{1}{4}\sum_{i=1}^{4} D_i, \tag{2}$$

where D_i are the Dice scores for each label

$$D_i = \frac{2\sum_{k=1}^{N} p_k \tilde{p}_k}{\sum_{k=1}^{N} p_k^2 + \sum_{k=1}^{N} \tilde{p}_k{}^2 + \epsilon}, \quad i = 1..4, \tag{3}$$

with prediction $\tilde{p} \in [0, 1]$ and ground truth $p \in \{0, 1\}$, and

$$L_{\text{VAE}} = \|X - \tilde{X}\|_2^2 + R(X), \tag{4}$$

where X is the 4D input and $\tilde{X}$ is the decoder prediction of the sequences from the VAE and $R(X)$ of the latent space representation.

2.6. Optimization and regularization

We use the Adam optimizer with initial learning rate of $\alpha_0 = 10^{-4}$ and progressively decreased it according to the formula from [24]:

$$\alpha = \alpha_0 \left(1 - \frac{e}{N_{\text{epoch}}}\right)^{0.9} \tag{5}$$

where $N_{\text{epoch}} = 600$ is the maximum number of epochs and e the current epoch. We use L2 norm regularization on the weights modelled by a weight decay of 10^{-5}.

2.7. Training

We trained the network with the batch size of 8 and 600 epochs on the training dataset using the expert annotations of the BraTS 2020 dataset as ground truth for each set of sequences separately. We choose the paramater setting with the lowest validation loss according to Section 2.5. The predicted probability for each voxel to contain each of the labels (ET, ED, NCR and BG) was used as the objective / loss function in the training. The training was performed on an Nvidia DGX A100 system (Nvidia Corp., Santa Clara, CA, USA) and took less than 18 hours for each set of sequences (Table A1).

2.8. Evaluation

Evaluation was carried out by applying each of the previously trained model to the corresponding set of sequences of the previously unseen test dataset. Segmentation results were then compared with the expert annotations as supplied with the BraTS 2020 dataset. To calculate segmentation accuracy, the voxel-wise most probable classification was used to designate the voxel's label in a binary fashion and then used to compute the Dice score for each label. We considered five key measures: the four labels from the original dataset (BG, NCR, ED and ET) and the averaged Dice score over all four labels. We use equal weighting, with $w = 0.25$ for each label [30]. As another measurement of segmentation accuracy, the Hausdorff distance was calculated using the Python MONAI package. Higher Dice score and, equivalently, lower Hausdorff distance correspond to more accurate segmentation [16].

We used these performance measurements, their statistical comparisons and common clinical imaging constraints, as e.g. mentioned in the consensus recommendations [12], to propose an op-

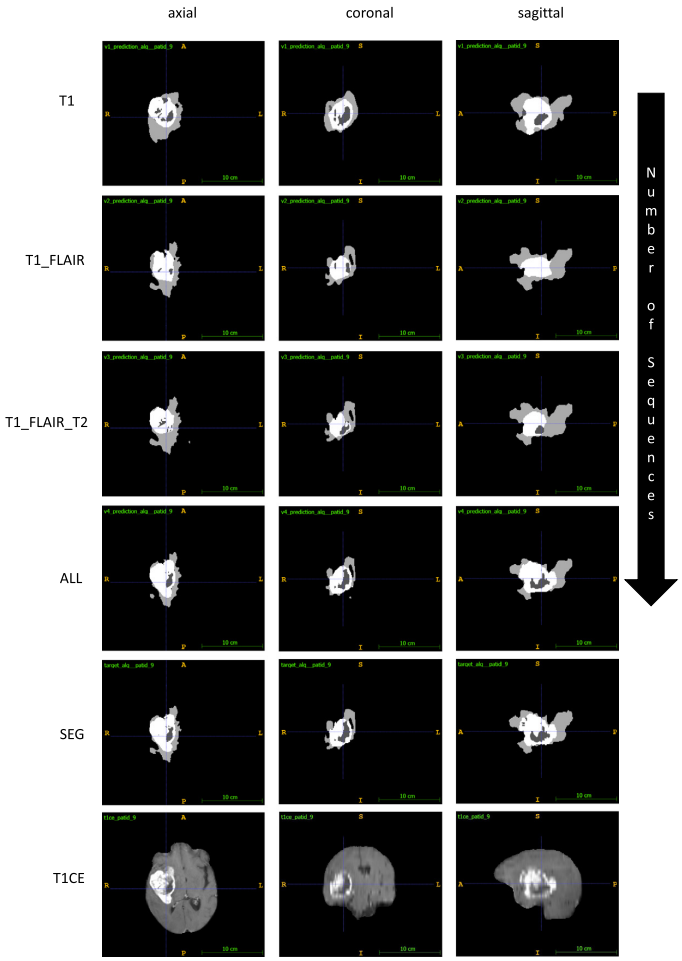


Fig. 4. Segmentation results from different sets of sequences, following the order of sequences as proposed by the consensus recommendations: The rows correspond to the number of sequences (1-4) the fifth row is the ground truths (SEG) and the last row is T1CE (for Orientation). Each column shows one view, the axial, the sagittal and the coronal view. T1 is based on a model trained on T1 sequences only, T1_FLAIR on T1 and FLAIR sequences, T1_FLAIR_T2 on T1, FLAIR and T2 sequences, and ALL on T1, FLAIR, T2 and T1CE sequences.

timal order of sequences. Ideally, we select the single-sequence model yielding the highest segmentation accuracy, then the best two-sequence model containing the previously found sequence until all four available sequences are included in the best possible model. Common clinical constraints include, that T1CE cannot be acquired before T1, 4-8 minutes should pass after contrast injection for an optimal T1CE [1], but the effect of Gadolinium on T2 or FLAIR is negligible or potentially even positive [14,13,21]. Finally, we perform a step-by-step comparison of our proposed imaging sequence order with the order of sequences stipulated by the consensus recommendations in a simulated MRI examination terminated after the first, the second or third sequence.

2.9. Statistics

Variables are expressed as the means ± SD, and are expressed as decimal numbers (percentage). According to whether the samples exhibited a normal distribution and an equal variance, the experimental results were statistically assessed using parametric or nonparametric tests. To compare the segmentation results, i.e. Dice scores, between our proposed order of sequences and the order as stipulated by the consensus recommendations, a Welch t-test, with $\alpha = 0.05$, was used. To account for multiple testing, we controlled the false discovery rate (FDR) using the Benjamini-Hochberg procedure. Analyses were performed using Python Sklearn and R, and diagrams were prepared with Microsoft Excel 2019. We used the ITK-Snap tool for the figures [35].

3. Results

Fig. 4 shows exemplary results obtained by only having the first, the first two, the first three, or all sequences of the consensus imaging recommendations available in comparison with the expert annotations. Using the T1 only, an overall reasonable segmentation accuracy is achieved in comparison to the expert annotations. It has to be noted, that the segmentation of the necrotic core is coarse, the vital (enhancing) tumor is too small, and that the peritumoral edema is too large in comparison with the expert annotations. Adding a FLAIR sequence (T1_FLAIR) improves the segmentation of the peritumoral edema and the vital

Table 1
Dice score for the three labels when comparing the expert annotations (ground truth) with the results from the automatic segmentation: contrast-enhancing tumor (ET), the peritumoral edema (ED) and the necrotic and non-enhancing tumor core (NCR) as well as the average over the three.

Dataset	NCR	ED	ET	Average
T1	0.431 ± 0.257	0.596 ± 0.190	0.399 ± 0.261	0.476 ± 0.179
FLAIR	0.365 ± 0.226	0.709 ± 0.153	0.389 ± 0.230	0.488 ± 0.146
T2	0.468 ± 0.248	0.693 ± 0.169	0.452 ± 0.265	0.538 ± 0.166
T1CE	0.681 ± 0.248	0.659 ± 0.167	0.766 ± 0.262	0.702 ± 0.164
T1_FLAIR	0.407 ± 0.267	0.715 ± 0.159	0.430 ± 0.243	0.517 ± 0.159
T1_T2	0.478 ± 0.272	0.704 ± 0.176	0.464 ± 0.280	0.549 ± 0.184
T1_T1CE	0.692 ± 0.254	0.671 ± 0.170	0.775 ± 0.258	0.713 ± 0.163
T2_FLAIR	0.476 ± 0.263	0.735 ± 0.158	0.430 ± 0.263	0.547 ± 0.159
T2_T1CE	0.673 ± 0.259	0.749 ± 0.168	0.769 ± 0.263	0.730 ± 0.171
FLAIR_T1CE	0.675 ± 0.267	0.793 ± 0.144	0.760 ± 0.279	0.743 ± 0.169
T1_FLAIR_T2	0.447 ± 0.269	0.739 ± 0.149	0.451 ± 0.274	0.546 ± 0.175
T1_T2_T1CE	0.675 ± 0.260	0.757 ± 0.169	0.776 ± 0.259	0.736 ± 0.170
T1_FLAIR_T1CE	0.674 ± 0.261	0.795 ± 0.141	0.749 ± 0.282	0.739 ± 0.171
T2_T1CE_FLAIR	0.681 ± 0.254	0.797 ± 0.155	0.766 ± 0.271	0.748 ± 0.175
ALL	0.683 ± 0.255	0.808 ± 0.138	0.764 ± 0.268	0.751 ± 0.163

Table 2
Hausdorff distance for the three labels when comparing the expert annotations (ground truth) with the results from the automatic segmentation: contrast-enhancing tumor (ET), the peritumoral edema (ED) and the necrotic and non-enhancing tumor core (NCR) as well as the average over the three.

Dataset	NCR	ED	ET	Average
T1	23.84 ± 54.57	11.95 ± 12.25	12.65 ± 13.56	15.04 ± 20.65
FLAIR	15.21 ± 18.29	12.78 ± 17.32	15.40 ± 16.67	14.23 ± 13.53
T2	14.48 ± 32.11	7.62 ± 4.22	10.12 ± 9.48	10.80 ± 12.05
T1CE	7.61 ± 6.59	10.28 ± 7.06	4.83 ± 7.74	7.55 ± 5.15
T1_FLAIR	16.32 ± 32.41	8.96 ± 8.83	11.76 ± 10.16	12.43 ± 12.89
T1_T2	14.18 ± 32.13	8.16 ± 8.47	9.49 ± 9.01	10.28 ± 12.77
T1_T1CE	7.67 ± 6.92	9.72 ± 6.93	4.50 ± 7.80	7.17 ± 5.40
T2_FLAIR	16.44 ± 33.47	8.45 ± 8.07	11.31 ± 10.10	12.23 ± 13.67
T2_T1CE	7.44 ± 5.66	8.82 ± 11.03	4.92 ± 8.02	6.77 ± 6.19
FLAIR_T1CE	7.70 ± 5.50	7.12 ± 7.48	4.78 ± 8.48	6.43 ± 5.46
T1_FLAIR_T2	12.60 ± 7.99	7.33 ± 7.35	11.11 ± 10.16	10.22 ± 6.77
T1_T2_T1CE	6.89 ± 5.30	6.91 ± 5.92	4.75 ± 8.19	5.97 ± 4.72
T1_FLAIR_T1CE	7.73 ± 8.36	7.76 ± 10.64	6.16 ± 13.22	7.15 ± 9.64
T2_T1CE_FLAIR	8.08 ± 7.63	7.14 ± 8.93	4.77 ± 8.06	6.23 ± 5.26
ALL	7.11 ± 6.19	6.45 ± 7.92	4.93 ± 10.40	5.90 ± 6.85

tumor, but decreases segmentation accuracy of the necrotic core, which is again improved by adding a T2 sequence (T1_FLAIR_T2). Finally adding the T1CE (T1_FLAIR_T2_TICE) allows for a better segmentation of necrotic tumor core and contrast enhancing tumor.

The segmentation accuracies for each set of sequences are listed in Table 1 (Dice scores) and Table 2 (Hausdorff distances).

In addition, we plotted the different performance metrics in Fig. 6 (Dice scores) and Fig. 7 (Hausdorff distances).

Incomplete information generally leads to lower segmentation accuracy. Especially missing the T1CE sequence is detrimental, since the information encoded in the contrast enhanced sequence allows for segmentation of vital, enhancing tumor vs. necrotic core and even peritumoral edema. The segmentation accuracy when comparing T1_T2_T1CE and T1_FLAIR_T1CE, i.e., either missing the FLAIR or T2 sequence, are only marginally different, with a larger Dice score and smaller Hausdorff distance for T1_FLAIR_T1CE. However, the standard deviation of the Hausdorff distance was found to be larger in T1_FLAIR_T1CE than in T1_T2_T1CE.

In the single-sequence models, T1CE performs best, but cannot be acquired before T1. T2 and FLAIR result in the next highest average Dice scores, but no statistical significant difference was found between these and the T1 sequence (see Fig. A1). Given, that it takes 4-8 minutes after injection for the contrast agent to reach equilibrium, we propose to start with the T1 sequence, followed by the contrast injection and the acquisition of the T2 sequence. The sequence of contrast injection followed by the T2 acquisition follows the protocol in the consensus recommendation and allows for a standardized minimum delay for the T1CE after contrast injection [12]. This allows for T1CE as the third sequence (T1_T2_T1CE). Finally, the FLAIR sequence may be acquired. Given the common constraints to ordering of MR imaging sequences, the optimal imaging sequence order appears to be [T1, T2, T1CE, FLAIR].

Following the approach in Fig. 4, Fig. 5 provides exemplary segmentation results for our proposed order.

In the final step-by-step comparison of following our proposed order vs. the order stipulated by the consensus recommendations,

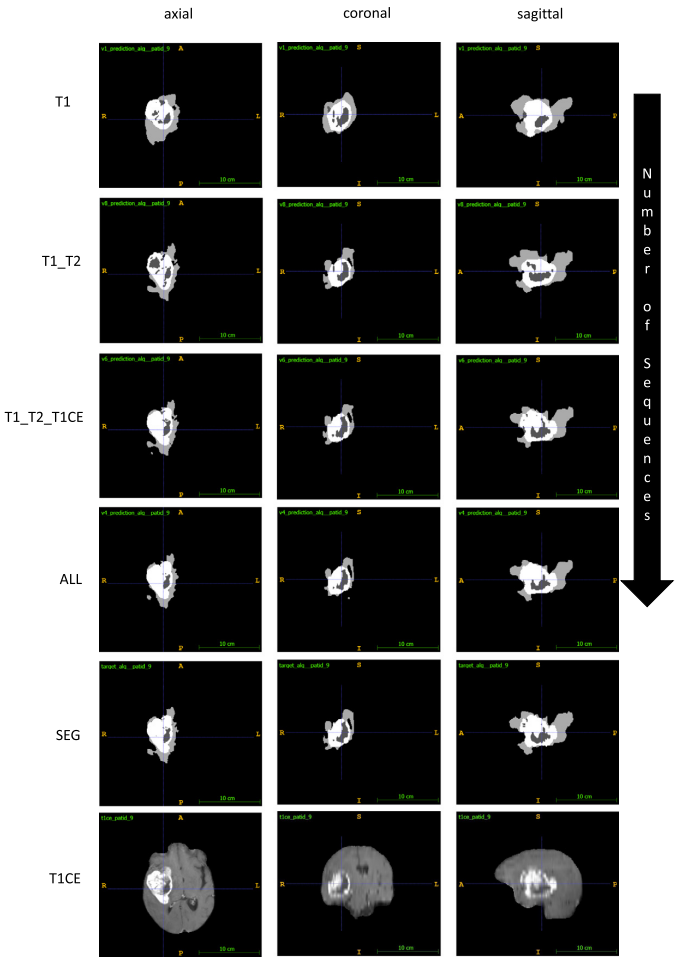


Fig. 5. Sequence by sequence adding by our approach: The rows correspond to the number of sequences (1-4) the fifth row is the ground truths (SEG) and the last row is T1CE. Each column shows one view, the axial, the sagittal and the coronal view.

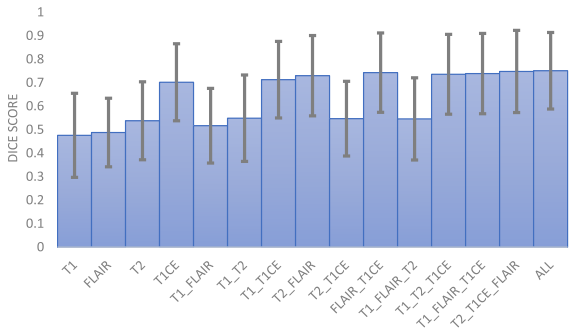


Fig. 6. Averaged Dice score for different sets of sequences and labels.

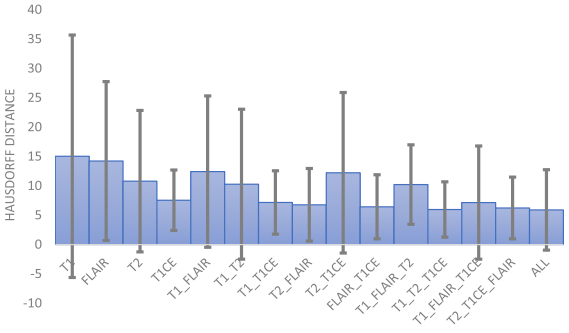


Fig. 7. Averaged Hausdorff for different sets of sequences and labels.

our proposed approach significantly outperforms the sequence order in the consensus recommendations. Comparing segmentation accuracy after each fully acquired sequence, the first sequence (T1) is the same for both, [T2, T1] (proposed) vs. [T1, FLAIR] (aborted recommendations) yields a (ΔAveraged Dice-score $= -0.032$), and [T2, T1, T1CE] (proposed) vs. [T1, FLAIR, T2] (aborted recommendations) a (ΔAveraged Dice-score $= -0.190$) (see Table 3, Fig. 8).

Table 3
Comparison of different order of acquisitions (aborted guideline vs. proposed) using the Welch-test.

Number of Seq.	Aborted guideline Seq. included	Proposed Seq. included	Diff. in AVG DSC
1	T1	T1	-[a]
2	T1, FLAIR	T1, T2	-0.032[b]
3	T1, FLAIR, T2	T1CE, T1, T2	-0.190[c]
4	ALL	ALL	-[d]

[a] There is no difference, because the input sequences are the same.
[b] The difference is not significant.
[c] The difference is significant, 95% Confidence Interval $95\% - CI = (-0.25, -0.13)$, $p < 0.001$.
[d] There is no difference, because the input sequences are the same.

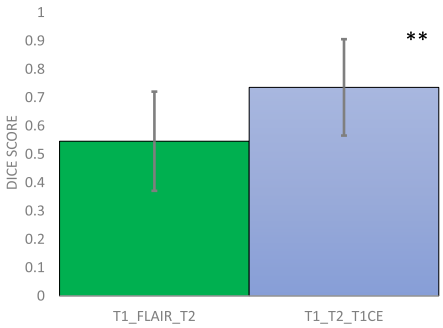


Fig. 8. Comparison of different orders of acquisition (aborted guideline vs. proposed). The bar plot shows a difference in the averaged Dice Score T1_FLAIR_T2 vs T1_T2_T1CE The difference is significant (Welch t-test, with $\alpha = 0.05$).

4. Discussion

The most recent consensus recommendation for imaging in clinical brain tumors trials was published in 2015 [12], but the protocols as well as the order of sequences in a brain tumor imaging protocol still vary across institutions. The current work evaluates the optimal order of imaging sequences for automatic, deep learning-based brain tumor segmentation in potentially prematurely terminated MR examinations. Our results demonstrate the value of the T1CE sequence, as it improves segmentation not only of contrast-enhancing tumor, but also of the necrotic core and peritumoral edema. Missing either T2 or FLAIR information had a very low influence on segmentation accuracy. Therefore, where it is foreseeable that the patient will not be able to finish the whole planned scan due to restlessness, claustrophobia, or neurological instability, we propose [T1, T2, T1CE, FLAIR] as the optimal sequence under common clinical constraints to MR imaging. Our recommended order increases the segmentation accuracy after three sequences by about 34.8% in comparison to the order of the sequences in the 2015 consensus recommendations.

The loss of information in incomplete MR examinations has so far not been systematically studied. The design of our experiment approached the problem in reverse, not by removing sequences and information from the full dataset, but by adding sequence by sequence in order to quantify the information gain per additional sequence. We deliberately chose to keep as many parameters as possible of the experiment constant, and therefore chose the same network architecture regardless of the number of input sequences.

Different MRI sequences visualize different aspects of the tumor. The T1 sequence delivers the anatomical overview of the brain, T1CE sequence enhances the highly vascularized and viable parts of the tumor, while T2 and FLAIR sequences facilitate the evaluation of peritumoral edema and non-contrast-enhancing parts of tumor in gliomas and the extent of the main tumor mass in non-contrast-enhancing low-grade gliomas [6]. Overall, the absence of either T2 or FLAIR information is not as critical for segmentation purposes as intuitively expected, which is likely due to the corresponding hypointense signal of the corresponding areas in the T1 and T1CE. Interestingly, our results show, that, even in the presence of T2 and FLAIR information, the T1CE sequence also carries a substantial amount of information regarding other labels, such as tumor edema. The contrast agent's purpose is to highlight disruptions in the blood-brain barrier and to make viable tumor visible. This, in turn allows for areas which might have been labeled as necrotic core or maybe even peritumoral edema to be correctly identified after adding T1CE information. Given the practical constraints to MR imaging, e.g., that T1CE cannot be acquired before T1 and that some time should pass between contrast injection and T1CE, we propose the above order of sequences. Acquiring the T2 sequence just after contrast injection is common practice, and allows for a standardized minimum amount of time to pass after contrast injection [12]. Deviating from the consensus recommendations, we propose to acquire the FLAIR-sequence after contrast injection. For the VAE-based segmentation algorithm, the T2 and FLAIR only add almost redundant information. Gadolinium contrast agents are known to have an effect on FLAIR imaging, but overall, the effect has been described to be positive [14,13,21] and e.g. raises the conspicuity of lesions. It has to be noted that the order of sequences and contrast injection is not known for the BraTS dataset. Further studies are therefore needed to assess the impact of either post-contrast T2 or post-contrast FLAIR imaging on automatic segmentation. The raised conspicuity of the lesions might even raise the value of a post-contrast FLAIR sequence to future segmentation approaches.

With the advancement of computerized image analysis techniques and artificial intelligence methods, numerous brain tumor segmentation algorithms have been developed. To allow for comparability, benchmark datasets have been provided in the context of the Multimodal Brain Tumor Segmentation (BraTS) challenges since 2012/2013 [4]. Current promising approaches include deep convolutional neural networks with U-Net architecture [9,19,38,31,40] and the variational autoencoder [20]. We chose the architecture combining both because of strength of an U-Net (ResNet) architecture, an established approach in computer vision tasks [17], and the regularization advantages of the variational autoencoder [18]. For the purpose of the study — measuring the Loss of Information due to reduced number of sequences — we consider our model to be suitable, in particular, because the number of parameters, channels and input sequences are kept constant (see Fig. 3).

Our study is limited by the relatively small sample size of the BraTS 2020 dataset, which includes scans of glioblastoma (n = 293) and lower grade glioma (n = 76). However, the BraTS dataset has been used extensively in multiple segmentation challenges, and expert annotations are available. Furthermore, a number of proven segmentation approaches, iteratively improved over the course of the challenges, are available, which may be considered state-of-the-art. Our study is therefore unlikely limited by an imperfect segmentation approach. Furthermore, the BraTS dataset is only available in a heavily pre-processed 3D format to achieve standard resolution, orientation and scaling. Since the consensus recommendations suggest a mixture of 2D and 3D MRI acquisitions, newly acquired data must therefore be processed in the same manner as the BraTS data. As the performance of the approach cannot be evaluated on the online evaluation portal that is provided by the BraTS organizers, we use NCR, ED and ET instead of tumor core, whole tumor and ET. In clinical neurosurgery

this subdivision would be preferred because it is clinically useful to separate edema (ED) of the surrounding brain tissue from the viable tumor (ET, NCR). Patients are likely to undergo additional imaging in clinical routine, such as perfusion imaging or diffusion-weighted imaging. However, these were not available from the BRaTS dataset and could therefore not be included in the current analysis.

As noted above, future work should examine the effect of Gadolinium contrast applied before acquisition of the FLAIR and T2 sequence on the accuracy of segmentation. Depending on the outcome of these studies, T1CE could potentially be included even earlier in a protocol optimized for deep learning-based automatic tumor segmentation.
In addition, evaluation of different segmentation algorithm can be performed, e.g. [36,39,11,37]. Different modification like learning rate variations [8] or increasing the resolution of the images [10] should also be considered. Furthermore, in a multi-center prospective study, additional sequences to those mentioned above should be included for further optimization.

5. Conclusion

To distinguish healthy from different types of tumor tissue in an automatic segmentation approach, three sequences suffice and information in T2 or FLAIR imaging is highly redundant. Our experiments show, that particularly the T1CE sequence is highly important for a good segmentation accuracy, even to refine labels such as tumor edema. We therefore propose to obtain imaging in the order [T1, T2, T1CE, FLAIR], given clinical imaging constraints, to maximize information gain in a potentially prematurely terminated MR examination.

Declaration of competing interest

We have no conflict of interest.

Availability of data and materials

Data is available https://www.med.upenn.edu/cbica/brats2020/data.html.

Acknowledgements

Computational infrastructure and support was provided by the Center for Information and Media Technology (ZIM) at the Heinrich Heine University of Duesseldorf (Germany).

Starting point for our Python implementation was provided by Nikolas Adaloglou.

The first author receives funding for his doctorate from the Anton-Betz Foundation of the Rheinische Post, grant number: 22/2020. The authors state that this work has not received any additional funding.

Appendix A. Additional information

Table A1
A Nvidia DGX A100 system (Nvidia Corp., Santa Clara, CA, USA) linux system with 8 GPUs, provided in the local high-performance computing infrastructure, was used with following software versions: Python version 3.6.5, pyTorch version 1.8.0, CUDA version 11.0, CUDNN version 8004. The table shows the time for training and validation.

Dataset	Walltime
T1	17:39:17
FLAIR	17:32:50
T2	17:34:42
T1CE	17:32:03
T1_FLAIR	17:40:40
T1_T2	17:33:46
T1_T1CE	17:37:05
T2_FLAIR	17:36:04
T2_T1CE	17:36:21
FLAIR_T1CE	17:33:32
T1_FLAIR_T2	17:39:54
T1_T2_T1CE	17:43:55
T1_FLAIR_T1CE	17:40:57
T2_T1CE_FLAIR	17:35:22
ALL	17:41:29

	T1	FLAIR	T2	T1CE
T1	X	> 0.05	> 0.05	<0.001***
FLAIR		X	> 0.05	<0.001***
T2			X	<0.001***
T1CE				X

	T1_FLAIR	T1_T2	T1_T1CE	T2_FLAIR	T2_T1CE	FLAIR_T1CE
T1_FLAIR	X	> 0.05	<0.001***	> 0.05	<0.001***	<0.001***
T1_T2		X	<0.001***	> 0.05	<0.001***	<0.001***
T1_T1CE			X	<0.001***	> 0.05	> 0.05
T2_FLAIR				X	<0.001***	<0.001***
T2_T1CE					X	> 0.05
FLAIR_T1CE						X

	T1_FLAIR_T2	T1_T2_T1CE	T1_FLAIR_T1CE	T2_T1CE_FLAIR
T1_FLAIR_T2	X	<0.001***	<0.001***	<0.001***
T1_T2_T1CE		X	> 0.05	> 0.05
T1_FLAIR_T1CE			X	> 0.05
T2_T1CE_FLAIR				X

Fig. A1. Comparison of different information content. Significant differences, in terms of two-sided Welch-Test with $\alpha = 0.05$, are encoded by a green color and the adjusted p-value. To account for multiple testing, we controlled the false discovery rate (FDR) using the Benjamini-Hochberg procedure. The red colored differences are not significant. The matrices are symmetric.

References

[1] P. Åkeson, C.H. Nordström, S. Holtås, Time-dependency in brain lesion enhancement with gadodiamide injection, Acta Radiol. 38 (1) (1997) 19–24.
[2] E. Alberts, Multi-modal multi-temporal brain tumor segmentation, growth analysis and texture-based classification, PhD thesis, Technische Universität München, 2019.
[3] M.M. Badža, M.Č. Barjaktarović, Classification of brain tumors from MRI images using a convolutional neural network, Appl. Sci. 10 (6) (2020) 1999.
[4] S. Bakas, H. Akbari, A. Sotiras, et al., Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features, Sci. Data 4 (1) (2017) 1–13.
[5] S. Bakas, M. Reyes, A. Jakab, et al., Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge, arXiv preprint, arXiv:1811.02629, 2018.
[6] C. Bergamino, S. Hoey, K. Waller, et al., Comparison of T1wFLAIR and T1wTSE sequences in imaging the brain of small animals using high-field MRI, Ir. Vet. J. 72 (1) (2019) 1–10.
[7] A. Bousselham, O. Bouattane, M. Youssfi, et al., Towards reinforced brain tumor segmentation on MRI images based on temperature changes on pathologic area, Int. J. Biomed. Imaging 2019 (2019).
[8] S.T. Bukhari, H.M. ud Din, A systematic evaluation of learning rate policies in training CNNs for brain tumor segmentation, Phys. Med. Biol. 66 (10) (2021) 105004.
[9] Ö. Çiçek, A. Abdulkadir, S.S. Lienkamp, et al., 3D U-Net: learning dense volumetric segmentation from sparse annotation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2016, pp. 424–432.
[10] A. Deshpande, V.V. Estrela, P. Patavardhan, The DCT-CNN-ResNet50 architecture to classify brain tumors with super-resolution, convolutional neural network, and the ResNet50, Neurosci. Inform. 1 (4) (2021) 100013.
[11] A. Elazab, C. Wang, S.J.S. Gardezi, et al., GP-GAN: brain tumor growth prediction using stacked 3D generative adversarial networks from longitudinal MR images, Neural Netw. 132 (2020) 321–332.
[12] B.M. Ellingson, M. Bendszus, J. Boxerman, et al., Consensus recommendations for a standardized brain tumor imaging protocol in clinical trials, Neuro-Oncol. 17 (9) (2015) 1188–1198.
[13] N. Ercan, S. Gultekin, H. Celik, et al., Diagnostic value of contrast-enhanced fluid-attenuated inversion recovery MR imaging of intracranial metastases, Am. J. Neuroradiol. 25 (5) (2004) 761–765.
[14] H.W. Goo, C.G. Choi, Post-contrast flair MR imaging of the brain in children: normal and abnormal intracranial enhancement, Pediatr. Radiol. 33 (12) (2003) 843–849.
[15] S. Hasan, M. Ahmad, Two-step verification of brain tumor segmentation using watershed-matching algorithm, Brain Inform. 5 (2) (2018) 1–11.
[16] M. Havaei, A. Davy, D. Warde-Farley, et al., Brain tumor segmentation with deep neural networks, Med. Image Anal. 35 (2017) 18–31.
[17] K. He, X. Zhang, S. Ren, et al., Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[18] T. Henry, A. Carre, M. Lerousseau, et al., Brain tumor segmentation with self-ensembled, deeply-supervised 3D U-net neural networks: a BraTS 2020 challenge solution, arXiv preprint, arXiv:2011.01045, 2020.
[19] F. Isensee, P. Kickingereder, W. Wick, et al., No new-net, in: International MICCAI Brainlesion Workshop, Springer, 2018, pp. 234–244.
[20] D.P. Kingma, M. Welling, Auto-encoding variational Bayes, arXiv preprint, arXiv: 1312.6114, 2013.
[21] A. Mahale, S. Choudhary, S. Ullal, et al., Postcontrast fluid-attenuated inversion recovery (FLAIR) sequence MR imaging in detecting intracranial pathology, Radiol. Res. Pract. 2020 (2020).
[22] R. Meier, U. Knecht, T. Loosli, et al., Clinical evaluation of a fully-automatic segmentation method for longitudinal brain tumor volumetry, Sci. Rep. 6 (1) (2016) 1–11.
[23] B.H. Menze, A. Jakab, S. Bauer, et al., The multimodal brain tumor image segmentation benchmark (BRATS), IEEE Trans. Med. Imaging 34 (10) (2014) 1993–2024.
[24] A. Myronenko, 3D MRI brain tumor segmentation using autoencoder regularization, in: International MICCAI Brainlesion Workshop, Springer, 2018, pp. 311–320.
[25] M.W. Nadeem, M.A.A. Ghamdi, M. Hussain, et al., Brain tumor analysis empowered with deep learning: a review, taxonomy, and future challenges, Brain Sci. 10 (2) (2020) 118.
[26] Q.T. Ostrom, H. Gittleman, L. Stetson, et al., Epidemiology of gliomas, in: Epidemiology of Gliomas, Springer International Publishing, 2014, pp. 1–14.
[27] Q.T. Ostrom, H. Gittleman, G. Truitt, et al., CBTRUS statistical report: primary brain and other central nervous system tumors diagnosed in the United States in 2011–2015, Neuro-Oncol. 20 (suppl_4) (2018) iv1–iv86.
[28] N. Porz, S. Bauer, A. Pica, et al., Multi-modal glioblastoma segmentation: man versus machine, PLoS ONE 9 (5) (2014) e96873.
[29] C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning, J. Big Data 6 (1) (2019) 1–48.
[30] R. Stupp, W.P. Mason, M.J. Van Den Bent, et al., Radiotherapy plus concomitant and adjuvant temozolomide for glioblastoma, N. Engl. J. Med. 352 (10) (2005) 987–996.
[31] Z.J. Su, T.C. Chang, Y.L. Tai, et al., Attention U-Net with dimension-hybridized fast data density functional theory for automatic brain tumor image segmentation, in: International MICCAI Brainlesion Workshop, Springer, 2020, pp. 81–92.
[32] L. Sun, S. Zhang, H. Chen, et al., Brain tumor segmentation and survival prediction using multimodal MRI scans with deep learning, Front. Neurosci. 13 (2019) 810.
[33] T. Tarasiewicz, M. Kawulok, J. Nalepa, Lightweight U-Nets for brain tumor segmentation, in: International MICCAI Brainlesion Workshop, Springer, 2020, pp. 3–14.
[34] M. Visser, D. Müller, R. van Duijn, et al., Inter-rater agreement in glioma segmentations on longitudinal MRI, NeuroImage Clin. 22 (2019) 101727.
[35] P.A. Yushkevich, J. Piven, H.C. Hazlett, R.G. Smith, S. Ho, J.C. Gee, G. Gerig, User-guided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability, NeuroImage 31 (3) (2006) 1116–1128.
[36] D. Zhang, G. Huang, Q. Zhang, et al., Cross-modality deep feature learning for brain tumor segmentation, Pattern Recognit. 110 (2021) 107562.
[37] J. Zhang, J. Zeng, P. Qin, et al., Brain tumor segmentation of multi-modality MR images via triple intersecting U-Nets, Neurocomputing 421 (2021) 195–209.
[38] C. Zhao, Z. Zhao, Q. Zeng, et al., MVP U-Net: Multi-view pointwise U-Net for brain tumor segmentation, in: International MICCAI Brainlesion Workshop, Springer, 2020, pp. 93–103.
[39] X. Zhou, X. Li, K. Hu, et al., ERV-Net: an efficient 3D residual neural network for brain tumor segmentation, Expert Syst. Appl. 170 (2021) 114566.
[40] R. Zsamboki, P. Takacs, B. Deák-Karancsi, Glioma segmentation with 3D U-Net backed with energy-based post-processing, in: International MICCAI Brainlesion Workshop, Springer, 2020, pp. 104–117.

Chapter 6

Conclusion

Within the framework of this research, two papers have been accepted for publication: Kronberg et al., 2022b; Werner et al., 2021. In addition, one paper have been submitted for publication: Kronberg et al., 2022a.

Furthermore, two software projects based on the papers (CPENET[1]), (PDACNET[2]), have been published under MIT license or will be after acceptance.

6.1 Main Results

For the detection and classification of pancreatic cancer metastasis in the lymph nodes by using Deep Transfer Learning on scans of stained HE slides, the training of the fine-tuned ResNet18 was carried out on TMA spots and the algorithm was validated using external independent data. While the spots were annotated by pathologists, some spots' patches show tissue such as adipose tissue that do not fit the meta label. Therefore, we introduced the communicators to clean up the labels and added a class called other tissues, which improved the performance on the external data by (Kronberg et al., 2022a). This second step approach shows improvements over only the data analysis step alone.

We implemented a fast Deep Learning-based method for the classification of patched brightfield images of SARS-CoV-2 infected Vero cells with a fine-tuned ResNet18 to determine a cytopathic effect (CPE) score (CPEnet). We thereby added a further class (TOX) of the network parameters to determinate drug toxicity (CPETOXnet) and subsequently analyzed the brightfield images of treated and infected Vero cells with chloroquine, hydroxychloroquine, remdesivir, and ementine and showed the reduction in CPE score and partial drug toxicity at higher drug concentrations, which we used to validate the CPETOXnet (Werner et al., 2021). We thus showed that Deep Learning can be used to improve drug development against SARS-CoV-2.

In clinical routine, various circumstances may prevent the acquisition of complete MRI examinations, which may be detrimental for tasks such as automated brain tumor segmentation. In

[1]Source code: `https://github.com/MolecularMedicine2/PyQoVi`,
[2]Source code: `https://github.com/MolecularMedicine2/pypdac`,

AI-based tumor segmentation, there is an inherent loss of accuracy when certain image types are missing from the data set. The sequence carrying the most information is the contrast-enhanced T1-weighted MRI sequence, while two sequences (T2 and FLAIR) feature almost redundant information for tumor segmentation via VAE-RESNET. Given the common constraints of radiological imaging, we determined an optimal order of MRI imaging sequences to maximize the gain of information in prematurely terminated imaging (Kronberg et al., 2022b).This in future can help to gain more information for surgery preparation or diagnoses.

This work confirms that Deep Transfer Learning is the method of choice, when dealing with few data. In addition, this work shows how import data cleaning for deep learning pipelines is. In summary, with the selected applications, we showed that Machine Learning, especially Deep Learning, is an excellent tool for analyzing medical imaging and can theoretically support medical professionals and researchers in novel drug development, diagnosis, and the personalization of treatments.

6.2 Future Work

For a more accurate validation of the methods and algorithms presented in this thesis, larger, well-annotated, and multi-centered data sets are required. In addition, for the methods (Kronberg et al., 2022a), one could add other tissue and tumor types and test whether they can also be classified. Furthermore, one could try to apply the Deep Learning approach from (Werner et al., 2021) to other viruses. For methods from (Kronberg et al., 2022b), one could apply the same procedure to other MRI images (for example from the torso). The main focus of future research would thus be to clinically validate the results obtained within the framework of this research and, if successful, integrate them into the clinical routine in the areas outlined in this thesis.

Bibliography

Ahamed, Farhad and Farnaz Farid (2018). "Applying internet of things and machine-learning for personalized healthcare: Issues and challenges". In: *2018 International Conference on Machine Learning and Data Engineering (iCMLDE)*. IEEE, pp. 19–21 (cit. on pp. 1, 3, 76).

Ay, Nihat, Paolo Gibilisco, and František Matúš, eds. (2018). *Information Geometry and Its Applications*. Springer International Publishing. DOI: 10.1007/978-3-319-97798-0. URL: https://doi.org/10.1007/978-3-319-97798-0 (cit. on p. 14).

Bergstra, James and Yoshua Bengio (2012). "Random search for hyper-parameter optimization." In: *Journal of machine learning research* 13.2 (cit. on p. 19).

Bishop, Christopher M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag (cit. on p. 5).

Deisenroth, Marc Peter, A Aldo Faisal, and Cheng Soon Ong (2020). *Mathematics for machine learning*. Cambridge University Press (cit. on pp. 5, 7).

Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255 (cit. on p. 21).

Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. (2020). "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (cit. on p. 8).

Echle, Amelie, Niklas Timon Rindtorff, Titus Josef Brinker, Tom Luedde, Alexander Thomas Pearson, and Jakob Nikolas Kather (2021). "Deep learning in cancer pathology: a new generation of clinical biomarkers". In: *British journal of cancer* 124.4, pp. 686–696 (cit. on p. 1).

Ekins, Sean, Ana C Puhl, Kimberley M Zorn, Thomas R Lane, Daniel P Russo, Jennifer J Klein, Anthony J Hickey, and Alex M Clark (2019). "Exploiting machine learning for end-to-end drug discovery and development". In: *Nature materials* 18.5, pp. 435–441 (cit. on pp. 1, 2, 58).

Fatima, Meherwar, Maruf Pasha, et al. (2017). "Survey of machine learning algorithms for disease diagnostic". In: *Journal of Intelligent Learning Systems and Applications* 9.01, p. 1 (cit. on pp. 1, 2, 24).

Gomez-Uribe, Carlos A. and Neil Hunt (Dec. 2016). "The Netflix Recommender System: Algorithms, Business Value, and Innovation". In: *ACM Trans. Manage. Inf. Syst.* 6.4. ISSN: 2158-656X. DOI: 10.1145/2843948 (cit. on p. 1).

Goodfellow, Ian J., Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. Cambridge, MA, USA: MIT Press (cit. on pp. 5, 8).

Granter, Scott R, Andrew H Beck, and David J Papke Jr (2017). "AlphaGo, deep learning, and the future of the human microscopist". In: *Archives of pathology & laboratory medicine* 141.5, pp. 619–621 (cit. on p. 1).

Grinstead, Charles Miller and James Laurie Snell (2012). *Introduction to Probability* -. Heidelberg: American Mathematical Soc. ISBN: 978-0-821-89414-9 (cit. on p. 15).

Hamilton, William (1855). *Discussions on philosophy and literature, education and university reform*. Harper (cit. on p. 5).

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778 (cit. on pp. 8, 9, 20).

Hemdan, Ezz El-Din, Marwa A Shouman, and Mohamed Esmail Karar (2020). "Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images". In: *arXiv preprint arXiv:2003.11055* (cit. on p. 1).

Hsu, Feng-Hsiung (2002). *Behind Deep Blue: Building the computer that defeated the world chess champion*. Princeton University Press (cit. on p. 1).

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger (2017). "Densely connected convolutional networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708 (cit. on pp. 8, 21).

Iandola, Forrest N, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer (2016). "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size". In: *arXiv preprint arXiv:1602.07360* (cit. on p. 21).

Ioffe, Sergey and Christian Szegedy (2015). "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *International conference on machine learning*. PMLR, pp. 448–456 (cit. on p. 10).

Janowczyk, Andrew and Anant Madabhushi (2016). "Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases". In: *Journal of pathology informatics* 7 (cit. on p. 1).

Jumper, John, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Zidek, Anna Potapenko, et al. (2021). "Highly accurate protein structure prediction with AlphaFold". In: *Nature* 596.7873, pp. 583–589 (cit. on p. 1).

Kather, Jakob Nikolas, Johannes Krisam, Pornpimol Charoentong, Tom Luedde, Esther Herpel, Cleo-Aron Weis, Timo Gaiser, Alexander Marx, Nektarios A Valous, Dyke Ferber, et al.

(2019a). "Predicting survival from colorectal cancer histology slides using deep learning: A retrospective multicenter study". In: *PLoS medicine* 16.1, e1002730 (cit. on p. 1).

Kather, Jakob Nikolas, Alexander T Pearson, Niels Halama, Dirk Jäger, Jeremias Krause, Sven H Loosen, Alexander Marx, Peter Boor, Frank Tacke, Ulf Peter Neumann, et al. (2019b). "Deep learning can predict microsatellite instability directly from histology in gastrointestinal cancer". In: *Nature medicine* 25.7, pp. 1054–1056 (cit. on p. 1).

Kilcher, Yannic (2021). "Navigating the Latent Spaces of Deep Neural Networks using Adversarial Techniques". en. PhD thesis. Zurich: ETH Zurich. DOI: 10.3929/ethz-b-000490637 (cit. on pp. 9, 16).

Kim, Yoon (Oct. 2014). "Convolutional Neural Networks for Sentence Classification". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1746–1751. DOI: 10.3115/v1/D14-1181. URL: https://aclanthology.org/D14-1181 (cit. on p. 19).

Kingma, Diederik P and Jimmy Ba (2014). "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (cit. on p. 13).

Kolesnikov, Alexander, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby (2020). "Big transfer (bit): General visual representation learning". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*. Springer, pp. 491–507 (cit. on p. 8).

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25, pp. 1097–1105 (cit. on pp. 8, 9, 21).

Kronberg, Raphael M., Lena Haeberle, Melanie Pfaus, Haifeng C. Xu, Karina S. Krings, Martin Schlensog, Tilman Rau, Aleksandra A. Pandyra, Karl S. Lang, Irene Esposito, and Philipp A. Lang (2022a). "Communicator-Driven Data Preprocessing Improves Deep Transfer Learning of Histopathological Prediction of Pancreatic Ductal Adenocarcinoma". In: *Cancers* 14.8. ISSN: 2072-6694. DOI: 10.3390/cancers14081964. URL: https://www.mdpi.com/2072-6694/14/8/1964 (cit. on pp. 2, 14, 19, 23, 25, 27, 91, 92, 99).

Kronberg, Raphael M., Dziugas Meskelevicius, Michael Sabel, Markus Kollmann, Christian Rubbert, and Igor Fischer (2022b). "Optimal acquisition sequence for AI-assisted brain tumor segmentation under the constraint of largest information gain per additional MRI sequence". In: *Neuroscience Informatics* 2.4. Artificial Intelligence in Brain Informatics, p. 100053. ISSN: 2772-5286. DOI: https://doi.org/10.1016/j.neuri.2022.100053. URL: https://www.sciencedirect.com/science/article/pii/S2772528622000152 (cit. on pp. 3, 10, 15, 75, 77–79, 91, 92, 99).

LeCun, Yann (2019). "1.1 deep learning hardware: past, present, and future". In: *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, pp. 12–19 (cit. on p. 8).

Li, Fei-Fei, Andrej Karpathy, and Justin Johnson (2016). "CS231n: Convolutional Neural Networks for Visual Recognition 2016". In: URL: `http://cs231n.stanford.edu/` (cit. on p. 5).

Martin Abadi, Ashish Agarwal, Paul Barham, and Xiaoqiang Zheng (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. URL: `https://www.tensorflow.org/` (cit. on pp. 1, 16).

Mazurowski, Maciej A, Mateusz Buda, Ashirbani Saha, and Mustafa R Bashir (2019). "Deep learning in radiology: An overview of the concepts and a survey of the state of the art with focus on MRI". In: *Journal of magnetic resonance imaging* 49.4, pp. 939–954 (cit. on p. 1).

Murphy, Robert F (2011). "An active role for machine learning in drug development". In: *Nature chemical biology* 7.6, pp. 327–330 (cit. on pp. 1, 2, 58).

Pan, Sinno Jialin and Qiang Yang (2009). "A survey on transfer learning". In: *IEEE Transactions on knowledge and data engineering* 22.10, pp. 1345–1359 (cit. on p. 20).

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala (2019). "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., pp. 8024–8035. URL: `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf` (cit. on pp. 1, 16).

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay (2011). "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830 (cit. on p. 18).

Pham, Hieu, Zihang Dai, Qizhe Xie, and Quoc V Le (2021). "Meta pseudo labels". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11557–11568 (cit. on p. 8).

Ramakers, Julius, Christopher Frederik Blum, Sabrina König, Stefan Harmeling, and Markus Kollmann (2021). "De Novo Prediction of RNA 3D Structures with Deep Learning". In: *bioRxiv* (cit. on p. 1).

Rambhajani, Madhura, Wyomesh Deepanker, and Neelam Pathak (2015). "A survey on implementation of machine learning techniques for dermatology diseases classification". In: *International Journal of Advances in Engineering & Technology* 8.2, p. 194 (cit. on pp. 1, 2, 24).

Rescigno, Argentina Anna, Eva Vanmassenhove, Johanna Monti, and Andy Way (2020). "A Case Study of Natural Gender Phenomena in Translation. A Comparison of Google Translate, Bing Microsoft Translator and DeepL for English to Italian, French and Spanish." In: *CLiC-it* (cit. on p. 1).

Robbins, Herbert and Sutton Monro (1951). "A stochastic approximation method". In: *The annals of mathematical statistics*, pp. 400–407 (cit. on p. 12).

Rodríguez-Barroso, Nuria, Antonio R. Moya, José A. Fernández, Elena Romero, Eugenio Martínez-Cámara, and Francisco Herrera (2019). "Deep Learning Hyper-parameter Tuning for Sentiment Analysis in Twitter based on Evolutionary Algorithms". In: *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 255–264. DOI: 10.15439/2019F183 (cit. on p. 18).

Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams (Oct. 1986). "Learning representations by back-propagating errors". In: 323.6088, pp. 533–536. DOI: 10.1038/323533a0. URL: https://doi.org/10.1038/323533a0 (cit. on p. 16).

Sajda, Paul (2006). "Machine learning for detection and diagnosis of disease". In: *Annu. Rev. Biomed. Eng.* 8, pp. 537–565 (cit. on pp. 1, 2, 24).

Simonyan, Karen and Andrew Zisserman (2014). "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556* (cit. on pp. 8, 20).

Smith, B. and G. Linden (May 2017). "Two Decades of Recommender Systems at Amazon.com". In: *IEEE Internet Computing* 21.03, pp. 12–18. ISSN: 1941-0131. DOI: 10.1109/MIC.2017.72 (cit. on p. 1).

Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich (2015). "Going deeper with convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9 (cit. on p. 8).

Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna (2016). "Rethinking the inception architecture for computer vision". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826 (cit. on p. 8).

Tan, Chuanqi, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu (2018). "A Survey on Deep Transfer Learning". In: *Artificial Neural Networks and Machine Learning – ICANN 2018*. Ed. by Věra Kůrková, Yannis Manolopoulos, Barbara Hammer, Lazaros Iliadis, and Ilias Maglogiannis. Cham: Springer International Publishing, pp. 270–279. ISBN: 978-3-030-01424-7 (cit. on pp. 19–21).

Tan, Mingxing and Quoc Le (2019). "Efficientnet: Rethinking model scaling for convolutional neural networks". In: *International Conference on Machine Learning*. PMLR, pp. 6105–6114 (cit. on p. 8).

Telgarsky., Matus (2008). "Deep Learning Theory (CS 540)." In: URL: https://www.cs.ubc.ca/~murphyk/Teaching/CS540-Fall08/ (cit. on p. 5).

Tian, Yuchi, Kexin Pei, Suman Jana, and Baishakhi Ray (2018). "Deeptest: Automated testing of deep-neural-network-driven autonomous cars". In: *Proceedings of the 40th international conference on software engineering*, pp. 303–314 (cit. on p. 1).

Tieleman, Tijmen, Geoffrey Hinton, et al. (2012). "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude". In: *COURSERA: Neural networks for machine learning* 4.2, pp. 26–31 (cit. on p. 13).

Vamathevan, Jessica, Dominic Clark, Paul Czodrowski, Ian Dunham, Edgardo Ferran, George Lee, Bin Li, Anant Madabhushi, Parantu Shah, Michaela Spitzer, et al. (2019). "Applications of machine learning in drug discovery and development". In: *Nature Reviews Drug Discovery* 18.6, pp. 463–477 (cit. on pp. 1, 2, 58).

Vinyals, Oriol, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojtek Czarnecki, Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, Timo Ewalds, Dan Horgan, Manuel Kroiss, Ivo Danihelka, John Agapiou, Junhyuk Oh, Valentin Dalibard, David Choi, Laurent Sifre, Yury Sulsky, Sasha Vezhnevets, James Molloy, Trevor Cai, David Budden, Tom Paine, Caglar Gulcehre, Ziyu Wang, Tobias Pfaff, Toby Pohlen, Dani Yogatama, Julia Cohen, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy Lillicrap, Chris Apps, Koray Kavukcuoglu, Demis Hassabis, and David Silver (2019). *AlphaStar: Mastering the Real-Time Strategy Game StarCraft II*. https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/ (cit. on p. 1).

Wang, Shijun and Ronald M Summers (2012). "Machine learning and radiology". In: *Medical image analysis* 16.5, pp. 933–951 (cit. on p. 1).

Werner, Julia, Raphael M. Kronberg, Pawel Stachura, Philipp N. Ostermann, Lisa Müller, Heiner Schaal, Sanil Bhatia, Jakob N. Kather, Arndt Borkhardt, Aleksandra A. Pandyra, Karl S. Lang, and Philipp A. Lang (2021). "Deep Transfer Learning Approach for Automatic Recognition of Drug Toxicity and Inhibition of SARS-CoV-2". In: *Viruses* 13.4. ISSN: 1999-4915. DOI: 10.3390/v13040610. URL: https://www.mdpi.com/1999-4915/13/4/610 (cit. on pp. 2, 14, 57, 59–61, 91, 92, 99).

Wu, Yuxin and Kaiming He (2018). "Group normalization". In: *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19 (cit. on pp. 10, 11).

Xie, Qizhe, Minh-Thang Luong, Eduard Hovy, and Quoc V Le (2020). "Self-training with noisy student improves imagenet classification". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10687–10698 (cit. on p. 8).

Personal Publications

Peer-Reviewed Journal Papers

Julia Werner et al. (2021). "Deep Transfer Learning Approach for Automatic Recognition of Drug Toxicity and Inhibition of SARS-CoV-2". In: *Viruses* 13.4. ISSN: 1999-4915. DOI: 10.3390/v13040610. URL: https://www.mdpi.com/1999-4915/13/4/610

Raphael M. Kronberg et al. (2022b). "Optimal acquisition sequence for AI-assisted brain tumor segmentation under the constraint of largest information gain per additional MRI sequence". In: *Neuroscience Informatics* 2.4. Artificial Intelligence in Brain Informatics, p. 100053. ISSN: 2772-5286. DOI: https://doi.org/10.1016/j.neuri.2022.100053. URL: https://www.sciencedirect.com/science/article/pii/S2772528622000152

Raphael M. Kronberg et al. (2022a). "Communicator-Driven Data Preprocessing Improves Deep Transfer Learning of Histopathological Prediction of Pancreatic Ductal Adenocarcinoma". In: *Cancers* 14.8. ISSN: 2072-6694. DOI: 10.3390/cancers14081964. URL: https://www.mdpi.com/2072-6694/14/8/1964

List of Figures

Eidesstattliche Erklärung
laut §5 der Promotionsordnung vom 15.06.2018

Ich versichere an Eides statt, dass die Dissertation von mir selbständig und ohne unzulässige fremde Hilfe unter Beachtung der „Grundsätze zur Sicherung guter wissenschaftlicher Praxis an der Heinrich-Heine-Universität Düsseldorf“ erstellt worden ist.

Ort, Datum

Raphael Marvin Kronberg