university of groningen / faculty of science and engineering

Finding Balance: A Series of U-Net Models for Image Segmentation of Overlapping Organoids

Bachelor's Project Thesis

Stefania Radu, s3919609, m.s.radu@student.rug.nl,
Daily supervisor: Asmaa Haja*
Supervisor: Prof. Dr. Lambert Schomaker

Abstract: The interest in automatically analyzing biomedical images increased in the past years, as an accurate localization and segmentation of organoids can help with the early detection of malignancies and predict diseases, such as cancer. The morphometric appearances of these images and the high level of overlapping in the organoids make the segmentation task challenging. This paper studied a simple U-Net and also proposes a Dual U-Net model with a shared encoder and two decoders, one for binary segmentation of the mask and one for the multi-class segmentation of overlaps. A significant addition to the U-net are the residual-atrous skip connections which reduce the semantic gap between the encoder and the decoder. The issue of high imbalance between the classes is addressed using a combination between the Focal Loss and the Focal Tversky Loss, which significantly improved the performance of the model. Ten networks were trained on more than 17,000 images with overlapping and non-overlapping organoids and obtained promising results. When tested on 88 new images, the final models achieved an F1 score of 0.83 for the mask channel and 0.43 for the overlapping channel. The Jaccard Index was 0.72 for the mask and 0.34 for the overlap.

1 Introduction

Organoids are 3D tissue structures derived from adult or embryonic stem cells that can replicate the micro-anatomy of any organ (Kaushik et al., 2018). Unlike traditional 2D cell lines, organoids contain several types of cells and facilitate the study of tissue physiology or the development of diseases. In the study of cancer, for instance, cells are taken directly from the tumor and used to create organoids that will be an in-vitro equivalent of the in-vivo tissue (Schutgens & Clevers, 2020). They are used to study the development of infectious diseases, such as the one caused by the Zika virus (Garcez et al., 2016) or genetic disorders, especially in organs with no regenerative capacity such as the brain (Freedman et al., 2015). As they are harvested directly from the patient's tissue, organoids make personalized and more effective treatments easier to obtain. In order to reach the treatment phase as early as possible, automatic techniques are used to accelerate the analysis process.

Deep learning (DL) is one of the most popular methods for computer-aided diagnosis and examination of biomedical data, such as cells or organoids. Due to the heterogeneous morphometric appearance and the high level of overlapping (Kaushik et al., 2018), the study of organoids is challenging. As a consequence, many DL models are focusing on hematologic images consisting of cells on a background, which differs from the bright field images of organoids. They are successfully used to diagnose diseases, such as acute leukemia (Rastogi et al., 2022), cervical cancer (Lu et al., 2016) or brain tumors (Chang et al., 2018). A popular architecture is the encoder-decoder U-Net model (Ronneberger et al., 2019), which is generally applied to biomedical data and serves as a starting point for this paper. However, the U-Net focuses mostly on the binary discrimination task and very few models

*Supervision by A. Haja was supported by EU grant ITN PERICO (GA ID: 812968)

had been applied to organoid datasets.

Different and enhanced versions of the U-Net can be used to solve the problem of multi-class semantic segmentation in overlapping organoids. Semantic segmentation differentiates between the classes in an image by assigning a class label to each pixel. Binary segmentation is used for the mask channel and it is significantly more facile than multi-class segmentation, where the model has to choose between multiple candidate classes, as it is the case with overlaps. When these organoids overlap, some are partially or completely obstructed by others, which might result in them not being analysed and relevant information being lost along the way. Moreover, clusters of cells can form masses, which are an indicator of tumors, as in the case of breast cancer (Grys et al., 2017). This is why the detection of the overlaps is a difficult problem, especially when there is a high imbalance between classes: background, object, overlap.

Class imbalance had been shown to have a detrimental effect in many real-life classification tasks (Buda et al., 2018) and the current dataset encounters the same issue. Class imbalance involves having one class in the training set with much more examples than the rest of the classes. Several methods are used to address this problem, for example, under-sampling or over-sampling, which remove or add data, such that a balance is achieved. Other techniques that operate at the model level include the introduction of weights based on the frequency of classes (Zhou & Liu, 2005). Loss functions with factors that modulate the number of false positives (FPs) or false negatives (FNs) (Lin et al., 2017; Salehi et al., 2017; M. Li et al., 2021) represent an effective solution, which does not change the inner distribution of the data.

The current study attempts to solve the problem of overlapping organoids in a highly imbalanced dataset. To accomplish this, multiple additions had been brought to the classical U-Net, including a second decoder for the multi-class segmentation, residual-atrous skip connections, and a learning rate scheduler. Moreover, to address the class imbalance issue, the models use a combination of Focal loss (Lin et al., 2017) and Focal Tversky loss (Salehi et al., 2017), which down-weight the most frequent class.

This paper is organized into 6 sections and the structure is as follows: section 2 presents a literature review on numerous image segmentation methods, focusing on the biomedical field; section 3 contains information about the data, the structure of the main model and a description of the different experimental designs; section 4 discusses the results of the experiments and section 5 compares different configurations. The paper ends with section 6, which includes conclusions and future research.

2 Literature review

Semantic segmentation has a wide range of applications. Several datasets include images of common everyday objects (Lin et al., 2014) or traffic scenarios (Geiger et al., 2013), used to train autonomous driving agents. In essence, semantic segmentation is a classification task, where each pixel is assigned a class label. Binary segmentation is the most elementary case, as it only involves 2 classes, usually the background and the foreground.

To differentiate between the 2 classes, early methods use edge detection matrices that perform convolution over an input image to detect boundaries. Some examples are the Sobel filter or the Laplace kernel, which detects both horizontal and vertical edges (Dhankhar & Sahu, 2013). Thresholding segmentation methods like the Otsu algorithm (Xue & Titterington, 2011) can classify the background of an image using a global threshold. In region-based segmentation, the goal is to detect the immediate boundaries of pixels, and one popular approach is watersheding (Yang et al., 2006), where the image is decomposed into catchment basins that are enclosed by watersheds, that act as boundaries. These methods do not, however, raise to the expectation set by the complex multi-class segmentation tasks.

DL models started to gain more popularity due to their better performance in the detection of multiple classes (Liu et al., 2017). The power of convolutional neural networks, or short CNNs, lies in the convolutional layers that extract features, the non-linear layers that apply an activation function and the pooling layers that reduce spatial resolution (Minaee et al., 2021). One architecture for object recognition that achieved impressive results on large datasets is the VGGNet (Simonyan & Zisserman, 2014). Rastogi et al. (2022) used LeuFeatx − a variation on the VGG16 − to differentiate be-

tween multiple leukemia sub-types. However, in order to make the input reconstruction task possible, the encoding path of the VGGNet is insufficient. This led to a new category of DL models, those with an encoder-decoder architecture. The segmentation task involves pixel-wise classification and to achieve this, the information in the input needs not only to be harvested, but also rearranged in such a way that the final output displays relevant knowledge. Networks such as the SegNet (Badrinarayanan et al., 2017) use the encoding features of the VGG16 but add a decoder with max-pooling layers that perform non-linear up-sampling. Therefore, these networks become more powerful and can currently address the object overlapping problem in more specific scenarios such as microscopic data.

In the field of biomedical image segmentation, many of the segmentation models are inspired by the U-Net, proposed by Ronneberger et al. (2019). It works by using a constricting path to extract features, followed by a symmetrical decoding path that places the features into context and recreates the image. The classical U-Net was applied on neuronal structures from microscopic stacks, as well as light microscopic images, and solved the segmentation task successfully with Intersection Over Union (IoU) scores of 0.93 and 0.77, respectively. Several variations on the U-Net were designed, including the DenseRes-Unet (Kiran et al., 2022), which segments clustered nuclei from histopathology images and uses residual skip connections instead of traditional ones. Xie et al. (2018) focused on cell counting and compared two Fully Convolutional Regression Networks based on the encoder-decoder architecture. The study demonstrates that results achieved on synthetic data generalize properly on real microscopy images. Naylor et al. (2019) uses CNNs and regression to address the problem of touching nuclei in stained images. In all these cases, one recurrent problem is the class imbalance, in addition to the multi-class classification problem.

In order to reduce the frequency of one or more of the classes, studies use different approaches. One variation of the U-Net uses a shared encoder and two decoders (X. Li et al., 2019) to detect clustered nuclei in glioma images. The first decoder is used for boundary segmentation, the second for the distance map prediction of the interior, and a third forward convolution network is trained as a fusion layer. Sun et al. (2018) uses stacked U-Nets and a hybrid loss function to address the class imbalance in a multi-output road extraction task. As a substitute for changing the structure of the U-Net, the focus is on the loss functions. There are a number of loss functions that can successfully minimise the disproportion between classes.

Focal loss (Lin et al., 2017) is a popular function based on Cross Entropy which tries to reduce the class imbalance. Tian et al. (2019) uses it together with the IoU loss to solve a segmentation task involving 80 different classes. Ryan et al. (2021) segments kidney organoids using Focal loss and Albuquerque et al. (2021) uses it as part of a model that counts cancer cells from zebrafish organoids. Other loss functions which look at the entire image include the Focal Tversky loss (Salehi et al., 2017). While this loss has not yet been applied in organoids datasets, it obtained a decent performance in the segmentation of brain tumors (Ahuja et al., 2021) or biomarkers in cases of bladder cancer (Lakshmi et al., 2019).

The current study adopts the idea of double U-Nets to solve the overlapping problem in organoids. It presents a comparison between two network architectures with two outputs: the binary mask channel and a multi-class overlapping channel. As the class imbalance was a significant issue in the dataset, different combinations of loss functions are used, with the final goal being achieving a balance in the frequency of classes for segmenting overlapping organoids.

3 Methods

3.1 Data

The data used in this project was collected by the University Medical Centre Groningen (UMCG). Digital microscopes are used to take high-resolution CZI images of three different overlapping organoids cultures. These cultures contain organoids that are derived from the same group of cells. Each 3D culture consists of 14 stacks, and every stack represents a horizontal slice in the culture. One example of such a stack is given in Figure 3.1. As the majority of the organoids, especially overlapping ones, are intelligible in the middle stacks, stacks number 6, 7 and 8 were selected to generate the training dataset. Two other stacks from the remaining

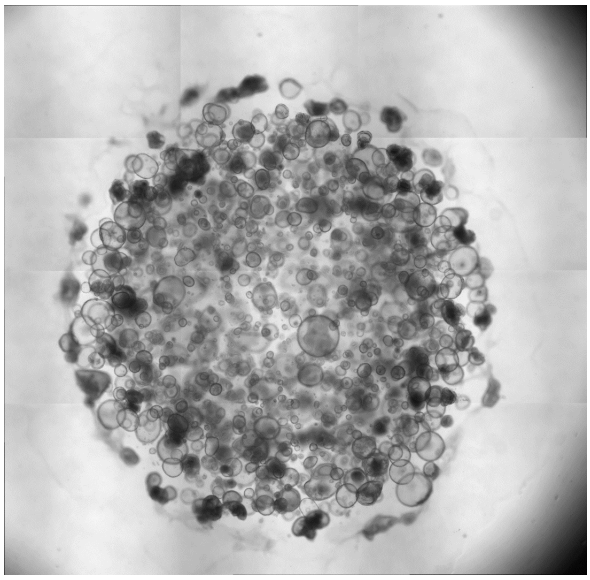

Figure 3.1: Stack 6 of the culture of the organoids culture in the training set. The image has a resolution of 3830 by 2900 pixels.

organoids cultures were set aside and used for testing.

The ground truth mask channel is generated using the Mask-RCNN (He et al., 2017), which localizes and segments the organoids. As parts of automatic segmentation were not accurate, manual work was needed to fix some of the boundaries coordinates. However, this is a tedious process and due to the blurriness and the high level of overlapping, a significant number of organoids, especially large ones, were missed by the Mask-RCNN. This represents one of the main issues encountered in the training process.

To create the ground-truth overlapping channel, a histogram-like method was used to count the number of touching organoids. All overlaps are treated as the same class, regardless of the number of organoids that are overlapping. As a result, there are 3 classes: $0-$ background, $1-$ organoids and $2-$ overlaps. The limitations of the Mask-RCNN resulted in many missed overlaps.

Table 3.1: The distribution of classes in the train and test sets.

	background	mask	overlap
training	65%	34%	1%
testing	44%	48%	8%

All the previously discussed issues, together with the nature of the data created a serious imbalance in the frequency of classes. Table 3.1 shows the distribution of the different classes in the train and test sets. In the training dataset, overlaps account, on average, for only 1% in an image, while the frequency of the background class is 65 higher. The test set consists of images where the sum of mask and overlap pixels account for at least half of the total. Consequently, there are 8 times more overlaps than in the train set and the mask is also more recurrent, as it account for almost 50%, on average. The class imbalance is a common issue in segmentation tasks and solutions including down-sampling the data to match the lowest frequency class can lead to a lack of diversity in the dataset (Moen et al., 2019). This is why, the current study proposes a combination of different losses, which includes class weights.

The initial culture images have a high-resolution, with a size of 3830 by 2900 pixels. As this is incredibly large for a DL network, a sliding window with a step size of 60 pixels is used over the image to create smaller crops of 320 by 320 pixels. The edges of the image are ignored, as they do not contain any organoids. During training, the images are shrunk further with a scaling factor of 0.5. The reason for this is that images with a lower resolution reduce the training time considerably.

To introduce diversity in the dataset, image augmentation techniques are used to generate additional examples. The first augmentation method used is rotation. The ground truth images are rotated with $90°$, $180°$ and $270°$. The second method is affine transformations, including a combination of positive and negative shearing with factors of -0.6 and 0.6, as well as translation on the x and y axes with a value of 1000. An affine transformation preserves parallel lines and creates images that continue to look realistic.

After the data augmentation, the train set contains 21,773 gray-scale crops. One random crop is shown in Figure 3.2. The leftmost crop is the ground truth image, followed by the true mask channel and the true overlapping channel. The mask channel is binary, so it contains two classes, the background, encoded as 0 and the organoids, encoded as 1. In the overlapping channel, there are

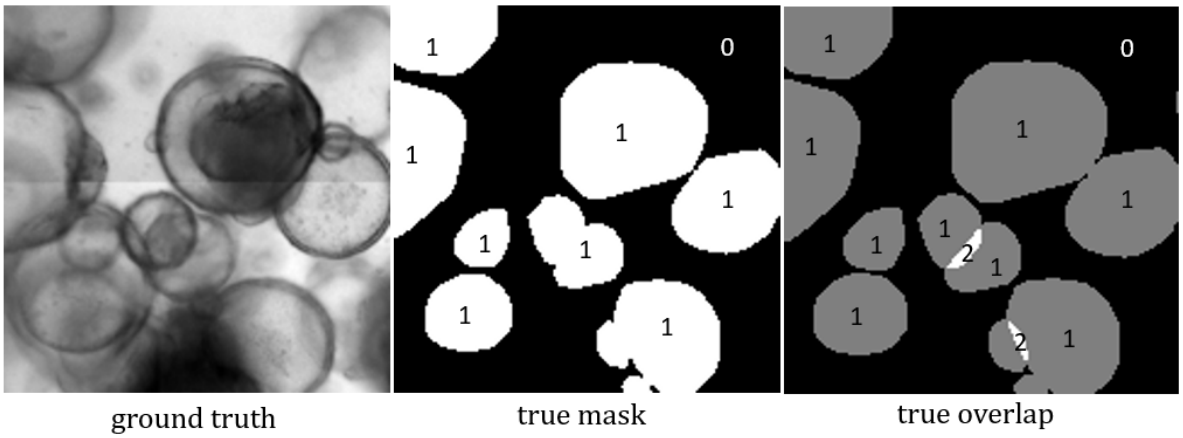

Figure 3.2: A random crop consisting of the ground truth image, the true mask and the true overlap with the corresponding encoding values: 0 - background, 1 - organoid, 2 - overlap.

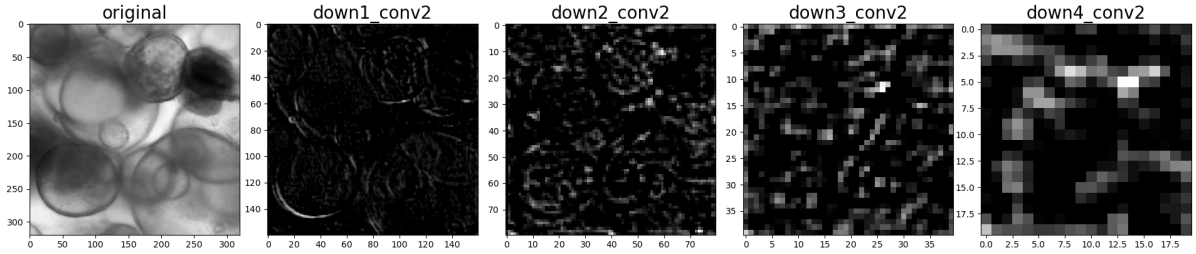

Figure 3.3: Feature maps from each of the four down-convolution blocks of the encoder.

three classes. Apart from the background and the foreground, any kind of overlap in the organoids is encoded using value 2. While the data does contain overlaps between different numbers of organoids (higher than two), the issue of class imbalance discussed previously would have only been aggravated if more classes had been added.

3.2 Model structure

The U-Net model proposed by Ronneberger et al. (2019) serves as a starting point for the architectures used in this paper. The current study focuses on the overlapping channel of the segmentation process, which is a significantly more intricate task, due to the low frequency of examples. The encoder-decoder architecture is preserved, but two main additions are included: the second decoder in the case of the double network and the residual-atrous skip connections. Depictions of the two architectures − the simple network with one decoder and the double network with two decoders − can be seen in Figures 3.7 and 3.8, respectively.

The encoder follows the structure of a convolutional neural network. It contains four down-convolution blocks, where each block is a series of two 3×3 padded convolution layers, each followed by a 2D normalization layer and a ReLU activation function. The role of the convolution layers is to increase the number of channels in the image. The input is a gray-scale 320×320 pixels image (160×160 after scaling) with only one channel and the number of channels doubles with each block of the encoder. The normalization layer stabilizes the gradients in the learning process and the activation function reduces the exponential growth of the weights. At the end of the down block, a 2×2 max-pooling layer is applied, which reduces the size of the image. The role of the encoder is to squeeze the relevant information into a bottleneck and create a representation of the input. The input image that had one channel originally is encoded to a fea-

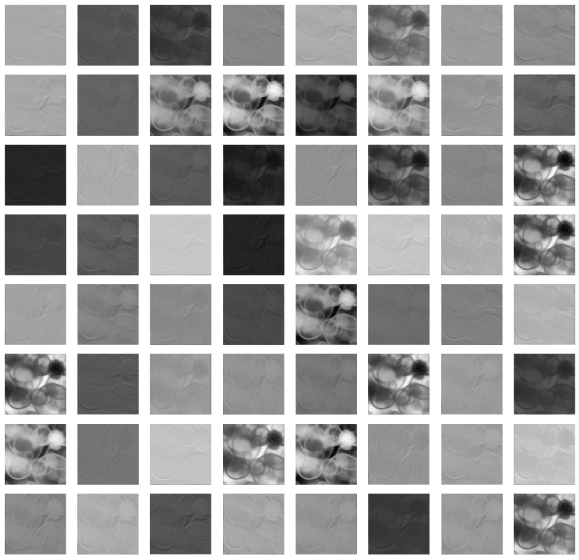

Figure 3.4: The first 64 feature maps of the first convolution layer in the encoder.

ture vector of 1024 values at end of the encoder. Figure 3.3 shows how the contracting path encodes information, starting with low-level features at the beginning of the encoder and finishing with higher-level features. Similarly, Figure 3.4 shows the 64 feature maps, as they are encoded by the first convolution layer of the model.

The decoder is an expanding path that places the extracted features into context and groups the pixels under classes. In the case of the simple U-Net architecture, as the input travels through the decoder, the image is up-sampled using 2×2 up-sampling layers, series of double convolution layers and ReLU activation functions, in a symmetrical manner to the encoder. The number of feature maps is decreased in the final step, and a 1×1 convolution is applied to map the features to the final classes. The output of the network constitutes of pairs of images, the binary mask channel and the three-class overlapping channel. Figure 3.7 depicts the simple architecture and all the steps explained previously. For consistency, the architecture of the double U-Net will be explained next, followed by the skip connections.

The double U-Net has one encoder, but two decoders, one responsible for each channel. As the features of both channels share similarities − the mask part in the two outputs is almost identical − it is reasonable to believe that the branches can be encoded together. While the simple architecture assumes that they can also be decoded together, the double version separates the ways in which the channels are reconstructed. The motivation for this is that the overlaps are areas that are very alike to the organoids themselves, and by separating the features, the aim is to reduce the number of overlaps falsely labelled as organoids. Figure 3.8 shows the two decoders, which are equivalent in terms of structure. The only difference is in the final number of classes of the output: two for the binary mask and three for the overlapping mask. To further highlight the idea that the two decoders work with different information, Figure 3.5 shows the first 64 kernels from the last up-convolution layer in each of the two decoders.

The role of skip connections is to take information from each of the levels of the encoder and copy them directly to the decoder by skipping the bottleneck. This way, the loss of features is reduced. While the conventional U-Net concatenated the feature maps directly, Kiran et al. (2022) suggests that this does not account for the semantic gap between the encoder and the decoder. By following the authors' advice, the current model integrates residual-atrous blocks, which are applied to the feature maps of the encoder. As shown in Figure 3.6, 3×3 and 1×1 convolution layers are applied to the features of the encoder to reduce information loss. The weights are summed together element-wise and an atrous block with two parallel dilated convolution layers is then applied. The convolution layers are 3 by 3 and use dilation rates of 2 and 4. The dilation rate ensures that more spatial information is encoded and that the blurriness of the image is minimised (Kiran et al., 2022). The resulting features are finally concatenated to the corresponding level of the decoder. In the case of the double architecture, the same feature maps are copied in the two decoders.

3.3 Experimental design

This study looked at four different additions to the traditional U-Net and compared a total of 10 models. These four contributions are the residual-atrous skip connections, the architecture of the model, the combination of loss functions and the scheduler.

The 21,773 input images are divided into a train-

Figure 3.5: The first 64 kernels of the mask decoder (left) and the overlap decoder (right) from the final up-convolution blocks, in the case of a double architecture.

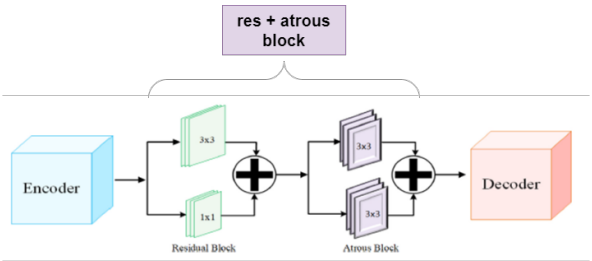

Figure 3.6: The residual-atrous skip connections, which apply dilated layers to the feature maps of the encoder, before the concatenation step, as depicted by Kiran et al. (2022).

Table 3.2: Examples of parameters used during training.

Property	**Value**
Batch size	1
Epochs	3
Validation %	10
Optimizer	RMSprop with momentum
Learning rate	10^{-5}
Loss	Focal Loss + Focal Tversky Loss
Scheduler	ReduceOnPlateau

ing and a validation set. 90% of the data is used for training and 10% for validation. During the implementation, other attempts were also made, such as using 20% of the data for validation, but due to the complexity of the network, this option was too time-consuming and so the 90-10 split was more feasible. The Dice score was used as an evaluation metric for the validation set (see subsection 4.1). The batch size is set to 1, indicating that only one random crop is presented to the network at a time, followed by an update of weights based on the loss. A wide variety of losses, such as Cross Entropy, Focal Loss, Dice Loss, or Focal Tversky Loss are used throughout the experiments. The optimizer used during training was a first-order optimization algorithm − RMSprop with a momentum factor of 0.9. After the gradients are computed based on the loss, the optimizer updates the weights. RMSprop is a technique for mini-batch learning, which deals with the vanishing or explosion of gradients by normalizing the gradients using a squared moving average. The goal of the normalization step is to balance the momentum, by decreasing the step size for large gradients and increasing the step for small gradients. The majority of the models are trained for 2 epochs, but in order to assess the performance of a longer training time, two models are trained for 3 epochs. The starting learning rate was 1×10^{-5} for all the models, but in some, a scheduler was used, while in others the decreasing step was performed manually. All models are tested on 88 unseen images from two different organoids cultures. An example of a training configuration

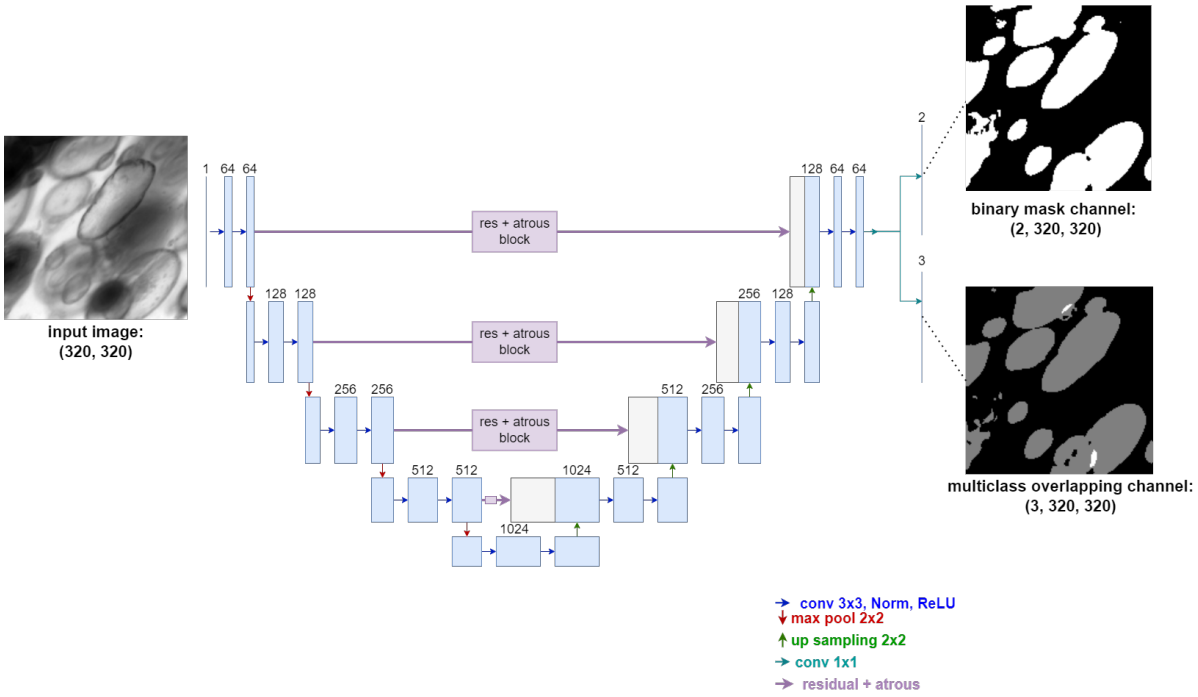

Figure 3.7: The simple U-net architecture with one encoder and one decoder. The two outputs – the mask channel and the overlapping channel – are generated in the end.

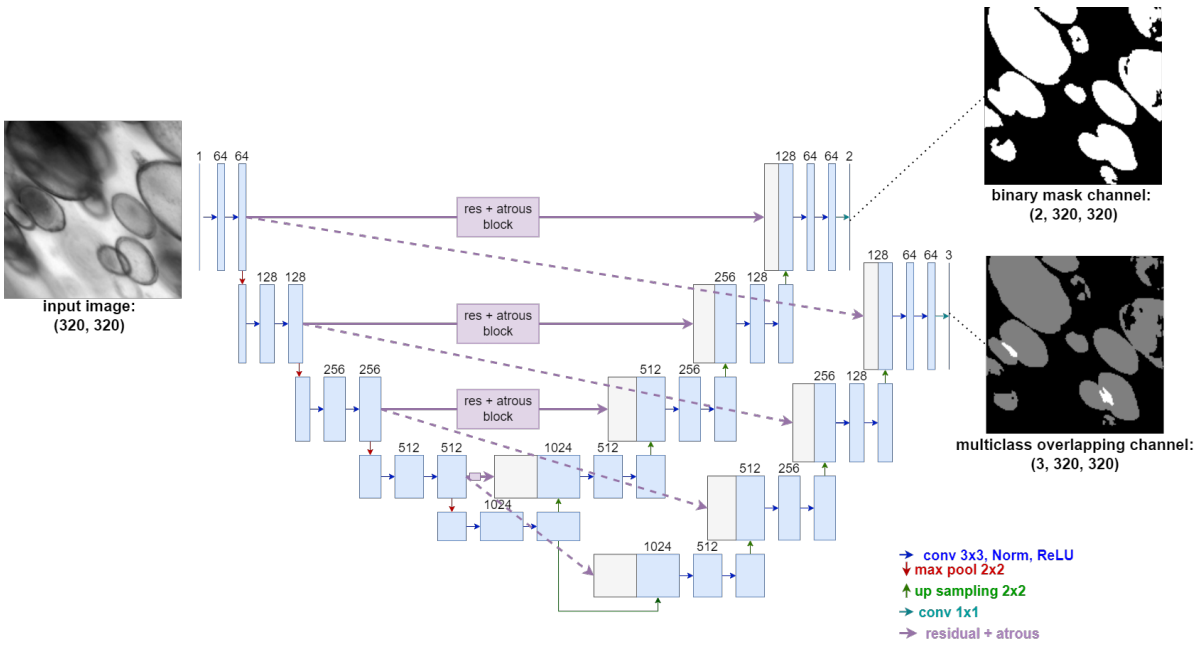

Figure 3.8: The double U-net architecture with a shared encoder and two decoders, one responsible for each channel.

for one of the models is given in Table 3.2. Additionally, transfer learning is used to initialize the weights of the model with pre-trained weights from a U-Net trained on 11 grayscale images of the Carvana dataset *.

3.3.1 *Residual - Atrous skip connections*

The starting model of this study is a simple U-Net, as described in Section 3.2. The skip connections used in this model are trivial. Feature maps from the encoder are copied directly to the decoder and concatenated with the result of the up-sampling layer. While the result of the concatenation is a block with twice as many features, the role of the convolution layers is to decrease this number. In this first experiment, the aim is to compare the performance of the model − especially the F1 score − when it uses traditional skip connections versus enhanced residual-atrous skip connections (Figure 3.6). Two comparisons are presented, involving 4 models that were trained for 2 epochs, using the ReduceOnPlateau learning rate scheduler. The first comparison looks at the effect of the residual-atrous skip connections on a simple architecture. In this case, the weights are updated based on a combination of the Cross Entropy Loss and the Dice Loss. Later on, the same experiment is performed on 2 double U-Nets, which use the Focal Loss and the Focal Tversky Loss. These loss combinations are selected based on multiple experiments with different model configurations. As the results during training (Section 4.2.1) indicate that applying the residual-atrous block in the skip connection improves the validation Dice score and decreases the loss in both scenarios, this addition was kept in the future experiments.

3.3.2 *Loss functions*

Another set of experiments is carried out to test the effect of the loss function. Three different combinations of losses are used on a simple U-Net model with residual-atrous skip connections. The first combination is the classical Cross Entropy (CE) + Dice loss (DL), which is very popular in semantic segmentation tasks, followed by Focal loss (FL) + Dice loss and Focal Loss + Focal Tversky Loss (FTL). Several studies (Badrinarayanan et al., 2017; Rastogi et al., 2022) use Cross Entropy or Categorical Cross Entropy as a low-level loss function. Cross Entropy is a binary loss function that measures the performance of a classification model, by computing the pixel-wise difference, as shown in Equation 3.1, where $p \in [0, 1]$ is the probability for the true class. Similarly, Categorial Cross Entropy is used when more than two classes are involved, as it is the case in the overlapping channel. However, for very imbalanced datasets, the loss tends to be dominated by the most frequent class. This is why, a more viable solution to solve the overlapping problem is Focal Loss (Lin et al., 2017). This function uses the extra $-\alpha(1-p)^{\gamma}$ (Equation 3.2) balanced modulating factor, which reduces the loss contribution of frequent examples, so that the network can focus on the more difficult classes. The parameter γ adjusts the rate at which loss is reduced for well-classified examples and the factor α balances the importance of positive and negative examples. In the current experiments, α is set to 1 and the function is initialized with the weights of the training class (Table 3.1) to achieve the desired balancing effect. In this work, the γ factor is set to 2.

$$CE(p) = -log(p) \tag{3.1}$$

$$FL(p) = -\alpha(1-p)^{\gamma} log(p) \tag{3.2}$$

The Dice Loss (Equation 3.3) is widely used in biomedical segmentation tasks as a second loss function. It is defined as $1 - DS$, where the Dice score (DS) indicates the amount of overlap between the ground truth image and the predicted image. A perfect overlap means a Dice score equal to 1 and so a loss that equals 0. This loss, however, does not address the class imbalance problem. To address this issue, the Focal Tversky Loss (Salehi et al., 2017) is used instead of the Dice loss. It is a generalization of the Dice loss, but uses two extra hyperparameters which modulate the number of false positives and false negatives (Equation 3.4). In order to penalize the false negatives for the less frequent classes (mask and overlap), the β factor is set to 0.7 and α to 0.3.

$$DL = 1 - \frac{2 \times TP}{TP + FP + TP + FN} \tag{3.3}$$

*The data can be found at `https://www.kaggle.com/c/carvana-image-masking-challenge`

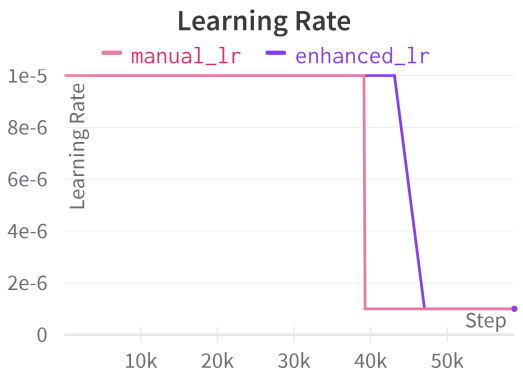

Figure 3.9: Two ways to decrease the learning rate: with the scheduler (*enhanced_lr*) and manually (*manual_lr*).

$$FTL = 1 - \frac{TP}{TP + \alpha FP + \beta FN} \tag{3.4}$$

The experiments conducted in this subsection look at the effect of loss functions on each architecture type. Six models are trained in total, three with the simple architecture and three with the double architecture.

3.3.3 *Model architecture*

The third set of experiments analyses the effect of model architectures. Two models were trained for 2 epochs, a double U-Net with a shared encoder and two decoders (Figure 3.8) and a simple U-Net with a single encoder and decoder (Figure 3.7). Both models use the Focal Loss and the Focal Tversky loss, introduced in the previous section. By using only one decoder, the simple architecture decodes the features concurrently for the mask and overlapping channels and builds the outputs in parallel. On the other hand, the double architecture involves two branches, one responsible for each channel.

3.3.4 *Learning rate and scheduler*

The last experiment studies the effect of the scheduler. While the ReduceOnPlateau scheduler is used in all previous setups, the goal of this current analysis is to show that it does indeed improve the evaluation Dice score, compared to a manual approach. The role of the learning rate is to modulate the step size when moving towards the minimum of the loss function. In order to achieve a balance between exploration and exploitation, the initial learning rate starts high, but decreases with time. The starting learning rate used throughout this study is 10^{-5}. Two methods to adjust the learning rate are compared, one is by manually decreasing its value after each epoch with a factor of 10^{-1} and an automatic one using a scheduler that decreases the learning rate with the same 10^{-1} factor when the evaluation score stops improving. In other words, when the Dice score function stops increasing monotonically, the learning rate is reduced. No plateau movements are allowed. Moreover, to allow the model to explore more the state space at the beginning of the training, the scheduler is frozen for the first two epochs. To preserve the equivalence, the manual decrease was also performed only at the beginning of the second epoch. Figure 3.9 depicts the behaviour of the learning rate in the two scenarios.

4 Results

4.1 Evaluation metrics

Several metrics are used to evaluate the performance of the model, both at the pixel and image levels. All the metrics are computed per each class, not including the background and then averaged. The most trivial metric which looks at the individual pixels is the accuracy, defined in Equation 4.1, followed by precision (4.2) and recall (4.3). A perfect precision score is an indicator that there are no false positives (FPs), so no background pixels miss-classified as mask, for example. On the other hand, a perfect recall score shows that there are no false negatives (FNs), so no missing organoids. Due to the imperfect nature of the true masks, the focus is on the minimisation of FNs.

$$ACC = \frac{1}{cls - 1} \sum_{i=1}^{cls} \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

$$Precision = \frac{1}{cls - 1} \sum_{i=1}^{cls} \frac{TP}{TP + FP} \tag{4.2}$$

$$Recall = \frac{1}{cls - 1} \sum_{i=1}^{cls} \frac{TP}{TP + FN} \tag{4.3}$$

The F1 score (also called Dice Score) is a commonly used metric in classification tasks and it is applied during the evaluation step in training, as well as in testing. The average F1 score is computed based on the number of classes *cls* (Equation 4.4). It is important to note that class 0 (the background) is purposely left out of the computation due to its high frequency, which would have altered the results. Therefore, in the mask channel, the score is computed only between the mask pixels in the true image and the mask pixels in the predicted image. The same applies to the overlapping channel, but in this case, the average is computed between the mask and the overlap parts.

The Jaccard Index (JI) is also used as a metric (Equation 4.5), equivalently to the F1 score. While the JI measures image similarity comparable to the F1 score, when averaged on multiple examples, the values differ and generally, the JI scores are lower than the corresponding F1 score, although they remain positively correlated.

$$F1 = \frac{1}{cls-1}\sum_{i=1}^{cls} 2\frac{precision * recall}{precision + recall} \tag{4.4}$$

$$JI = \frac{1}{cls-1}\sum_{i=1}^{cls} \frac{TP}{TP + FN + FP} \tag{4.5}$$

4.2 Experimental results

There are 10 models trained in total, as illustrated in Table 4.1, which presents an overview. The highest F1-score during training was achieved by the most complex model called *enhanced_lr*. It obtained an F1 score of 0.63 in the overlapping channel and 0.85 in the mask channel. This model uses a double architecture, residual-atrous skip connections, a combination of Focal loss and Focal Tversky loss, as well as the learning rate scheduler. Additionally, it is trained for 3 epochs. However, this model did not deliver good results on the test set. While there might be several reasons for this, the complexity of the training configuration plays an important role.

This Section presents the results of all sets of experiments discussed in Section 3, both during training and testing. The first subsection discusses the residual-atrous skip connections. Subsection 4.2.2 presents a comparison of different combinations of losses, followed by an experiment involving the two architectures (Subsection 4.2.3) and finally a description of the scheduler in subsection 4.2.4.

4.2.1 *Residual - Atrous skip connections*

The first experiment studies the effect of the skip connections type on performance. In both studied scenarios (a simple architecture with CE + DL and a double architecture with FL + FTL) adding the residual-atrous skip connections decreases the loss (Figure 4.1) and increases the training final F1 score (Figure 4.2). While these results concern the overlapping channel, the mask channel has very similar behaviour. However, this result does not generalize in testing. Regardless of the architecture, adding the residual-atrous block to the features of the encoder before concatenation always decreases the testing F1 score both in the mask (Table 4.3) and overlap (Table 4.2). The score decreases from 0.42 to 0.35 in the overlapping channel for a simple U-Net and from 0.41 to 0.39 in a double U-Net. In the case of the mask channel, the decrease is from 0.82 to 0.72 in a simple architecture and from 0.79 to 0.77 in a double architecture. For this comparison, the last value of the F1 score during testing is used, together with the average F1 score obtained on the 88 images that comprise the test set.

4.2.2 *Loss functions*

The second set of experiments investigates three different combinations of losses in the case of a simple U-Net. While the initial combination is the classic Cross Entropy and Dice loss, using the enhanced abilities of the Focal loss and the Focal Tversky loss improves the performance of the models significantly. The "focal" nature of the functions is given by the hyperparameter $\gamma = 2$, which reduces the loss considerably for the overlapping channel (Figure 4.3). The loss value converges to 0.45 for CE + DL, 0.30 for FL + DL and 0.18 for FL + FTL. The flexibility of the modulating factors minimises the class imbalance, which results in higher training Dice scores. Figure 4.4 shows that using increasingly complex combinations of losses has a beneficial effect on the Dice score in the case of a simple U-Net model.

Table 4.1: An overview of the 10 models with respect to the skip connections, model architecture type, loss functions, scheduler and the number of training epochs. The *res-atrous* entry indicates the use of residual-atrous skip connections, as opposed to regular connections. The model architecture has two types: simple U-Net or double U-Net. The loss function includes a combination of the following: Cross Entropy (CE), Focal loss (FL), Dice loss (DL) and Focal Tversky loss (FTL).

Name	Skip connections	Architecture	Loss	Scheduler	Epochs
simple_with_atrous	res-atrous	simple	CE + DL	ReduceOnPlateau	2
simple_no_atrous	regular	simple	CE + DL	ReduceOnPlateau	2
focal_dice	res-atrous	simple	FL + DL	ReduceOnPlateau	2
focal_tversky	res-atrous	simple	FL + FTL	ReduceOnPlateau	2
double_focal_tversky	res-atrous	double	FL + FTL	ReduceOnPlateau	2
manual_lr	res-atrous	double	FL + FTL	no	3
enhanced_lr	res-atrous	double	FL + FTL	ReduceOnPlateau	3
double_focal_dice	res-atrous	double	FL + DL	ReduceOnPlateau	2
double_ce_dice	res-atrous	simple	CE + DL	ReduceOnPlateau	2
double_no_atrous	regular	simple	FL + FTL	ReduceOnPlateau	2

Table 4.2: Testing results of the 10 models for the overlapping channel based on several evaluation metrics.

Name	Accuracy	Precision	Recall	F1 score	Jaccard Index
simple_with_atrous	0.82	0.51	0.32	0.35	0.27
simple_no_atrous	**0.85**	**0.54**	0.42	0.42	0.34
focal_dice	0.83	0.50	0.37	0.38	0.30
focal_tversky	0.84	0.50	**0.45**	**0.43**	**0.34**
double_focal_tversky	0.83	0.50	0.38	0.39	30
manual_lr	0.84	0.51	0.37	0.38	0.30
enhanced_lr	0.83	0.50	0.35	0.36	0.28
double_focal_dice	0.83	0.51	0.40	0.40	0.31
double_ce_dice	0.84	0.53	0.44	**0.43**	**0.3**4
double_no_atrous	0.84	0.52	0.41	0.41	0.33

Table 4.3: Testing results of the 10 models for the mask channel based on several evaluation metrics.

Name	Accuracy	Precision	Recall	F1 score	Jaccard Index
simple_with_atrous	0.74	**0.88**	0.63	0.72	0.58
simple_no_atrous	**0.81**	0.87	0.79	0.82	0.71
focal_dice	0.76	0.86	0.71	0.76	0.63
focal_tversky	**0.81**	0.83	**0.83**	**0.83**	0.71
double_focal_tversky	0.77	0.86	0.71	0.77	0.64
manual_lr	0.76	0.87	0.69	0.76	0.62
enhanced_lr	0.74	0.86	0.66	0.74	0.59
double_focal_dice	0.78	0.85	0.75	0.79	0.66
double_ce_dice	**0.81**	0.85	0.82	**0.83**	**0.72**
double_no_atrous	0.79	0.87	0.75	0.79	0.67

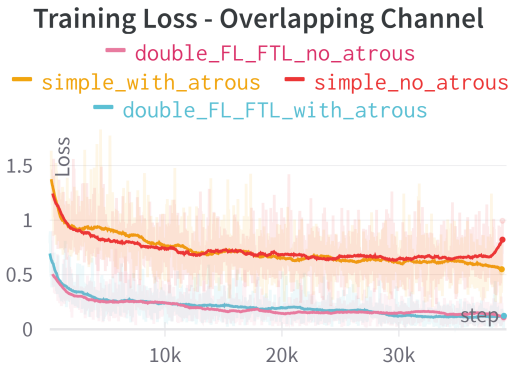

Figure 4.1: A loss comparison between models using the residual-atrous skip connections versus the regular connections. The orange and red models use a simple architecture and the CE + DL loss combination, while the blue and pink models have a double architecture and a FL + FTL loss (Table 4.1).

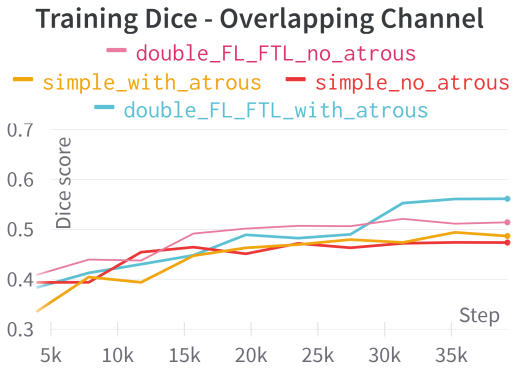

Figure 4.2: A dice comparison between models using the residual-atrous skip connections versus the regular connections. The orange and red models use a simple architecture and the CE + DL loss combination, while the blue and pink models have a double architecture and a FL + FTL loss (Table 4.1).

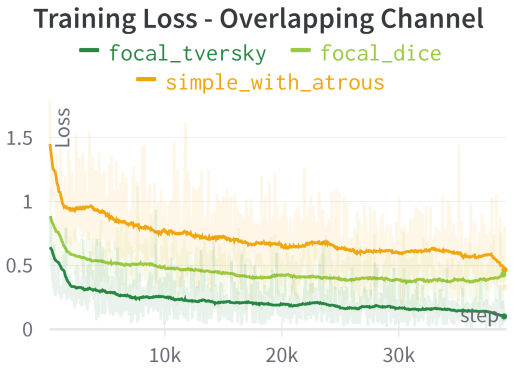

Figure 4.3: A comparison between different combinations of losses in models with a simple architecture (Table 4.1).

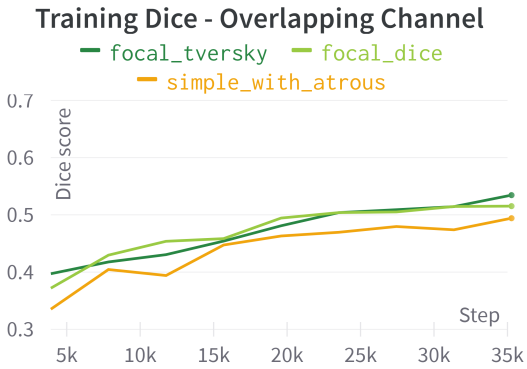

Figure 4.4: A dice comparison between models using different combinations of losses, as shown in Table 4.1.

4.2.3 *Model architecture*

In terms of the model architecture types, the current comparison evaluates a simple and double U-Net, both using the Focal loss and Focal Tversky loss combination. Overall, the double architecture achieves a higher training F1 score in the overlapping channel (Figure 4.6), as well as a lower loss (Figure 4.5) during training. The final training F1 scores are 0.54 for the simple model and 0.56 for the double version.

To further study the effect of different losses on the architecture type, Histograms 4.7 and 4.8 show the final F1 score achieved during training and the average F1 score during testing, respectively. The F1 score increases during training for the double architecture with more complex losses, while the average F1 score on the test set is not high and decreases with complex losses. The F1 score for the simple U-Net also increases during training and this behaviour remains consistent in testing. In summary, the simple U-Net achieves the highest scores

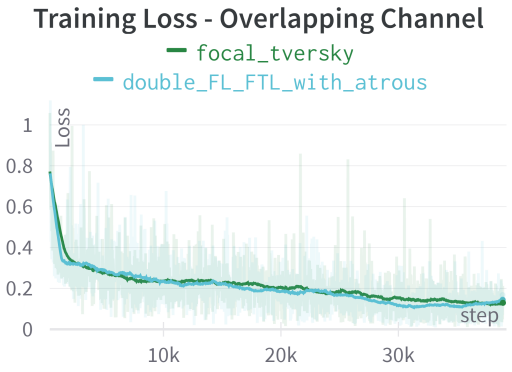

Figure 4.5: The loss during training in the overlapping channel for a simple U-Net compared to a double U-Net.

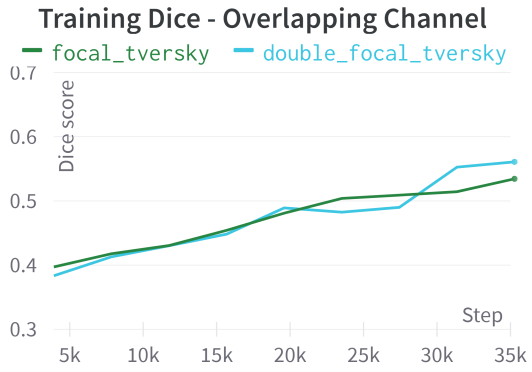

Figure 4.6: The dice score during training in the overlapping channel for a simple U-Net, compared to a double U-Net.

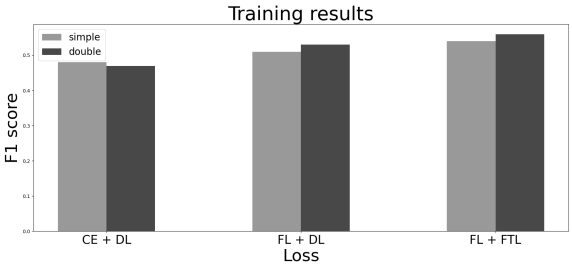

Figure 4.7: The effect of model architecture and loss on the training F1 score.

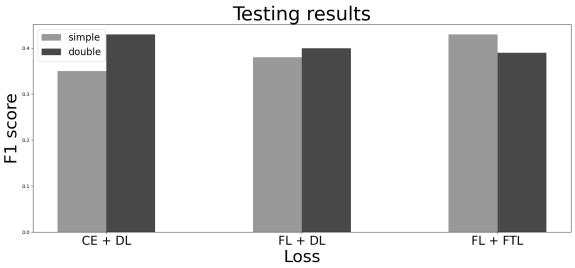

Figure 4.8: The effect of model architecture and loss on the testing F1 score.

with complex losses, while the double U-Net performs best with simple losses.

4.2.4 *Learning rate and scheduler*

The last experiment studies the effect of the learning rate scheduler, as well as experiments with a larger number of training epochs. The initial learning rate value is 10^{-5}. Two models are trained for 3 epochs. One of them uses the ReduceOnPlateau scheduler, which automatically decreases the learning rate (with a factor of 10^{-1}) when the evaluation dice score stops improving. For this experiment, the scheduler is also frozen for the first 2 epochs. The second model uses a manual decrease with the same factor, at the beginning of the second epoch. Figures 4.9, 4.10, 4.11 and 4.12 show that using the scheduler improves performance and that training for additional time increases the dice score significantly. In fact, the model *enhanced_lr* achieves the highest Dice score during training of 0.63 for the overlapping channel (Figure 4.10) and 0.85 for the mask channel (Figure 4.12). The loss function converged to 0.08 for the multi-class channel (Figure 4.9) and 0.01 for the binary channel (Figure 4.11).

5 Discussion

The high level of overlapping in organoids represents a serious challenge in biomedical segmentation tasks. While several deep learning networks have achieved state-of-the-art performance when it comes to binary segmentation, overlapping objects are still an impediment to an accurate diagnosis of many diseases. The encoder-decoder architecture of the U-Net Ronneberger et al. (2019), as well as the skip connections represent useful additions, which resulted in high performance on many datasets.

The current study proposed a variation of the U-

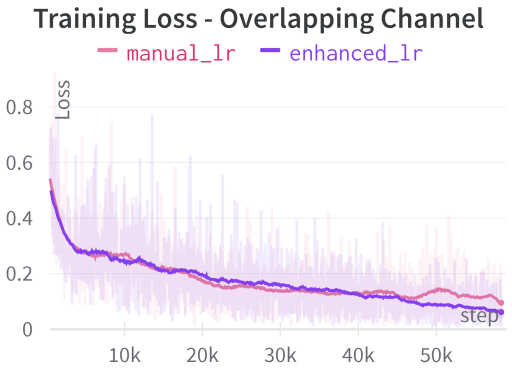

Figure 4.9: The effect of the scheduler on the overlap training loss in the case of a double U-Net (Table 4.1).

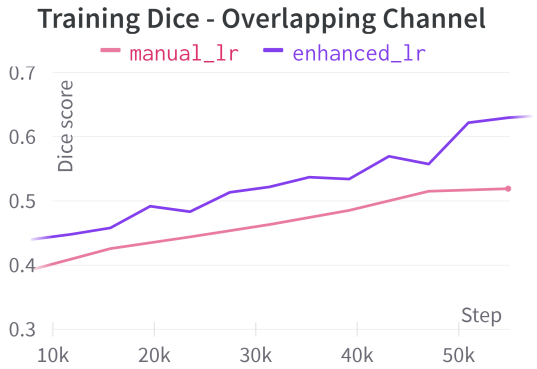

Figure 4.10: The effect of the scheduler on the training dice score of the overlapping channel in the case of a double U-Net (Table 4.1).

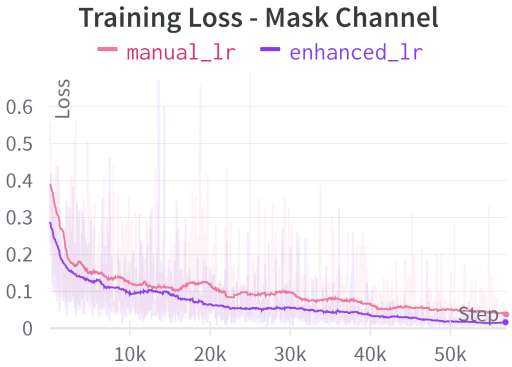

Figure 4.11: The effect of the scheduler on the mask training loss in the case of a double U-Net (Table 4.1).

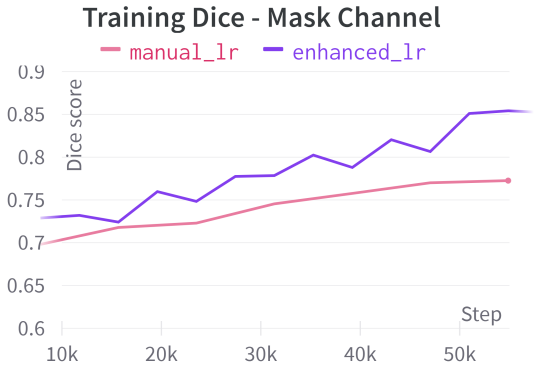

Figure 4.12: The effect of the scheduler on the training dice score of the mask channel in the case of a double U-Net (Table 4.1).

Net that solves the overlapping organoids task in a severely imbalanced dataset. Ten different models were trained and compared with respect to different additions, including the atrous skip-connections, model architecture, loss functions and scheduler.

Starting with the skip connections, the training results show that using residual-atrous skip connections increases the evaluation Dice score (Figure 4.2). However, this finding does not generalize in testing, where Table 4.2 illustrates that regular skip connections achieve an F1 score with 0.07 higher in a simple architecture and 0.02 higher in a double architecture. This does not match with the results of Kiran et al. (2022) on the DenseRes-Unet. An explanation for this can be that, while the role of the skip connections is to copy information from the encoder to the decoder by skipping the bottleneck, the extra convolution layers in the residual-atrous block might create a different bottleneck which results in information loss. Kiran et al. (2022) used a U-Net with dense layers, instead of convolution layers, which might have influenced the results.

Experimenting with different losses and architectures shows that, while more complex losses and a double architecture generally increase the F1 score during training (Figure 4.7), only part of the pattern generalizes in testing. With regards to the double architecture, the results show a decrease in the F1 score with more complex losses during testing

(Figure 4.8). The reason for this might be that the frequency of classes in the testing set does not match the frequency in the training set. The images in the testing dataset were selected such that at least 50% of the overlapping channel contains mask or overlap. The testing dataset has 8 times more overlaps than the training dataset (Table 3.1). Losses such as Focal loss and Focal Tversky loss have hyperparameters that modulate the number of FPs and FNs which were tuned for the training dataset. Thus, as the weights do not match, the performance is not the same for the test set. In the case of the simple architecture, the 2 classes are decoded together so the weights mismatch is reduced and the difference is not as significant as when the channels are considered separately.

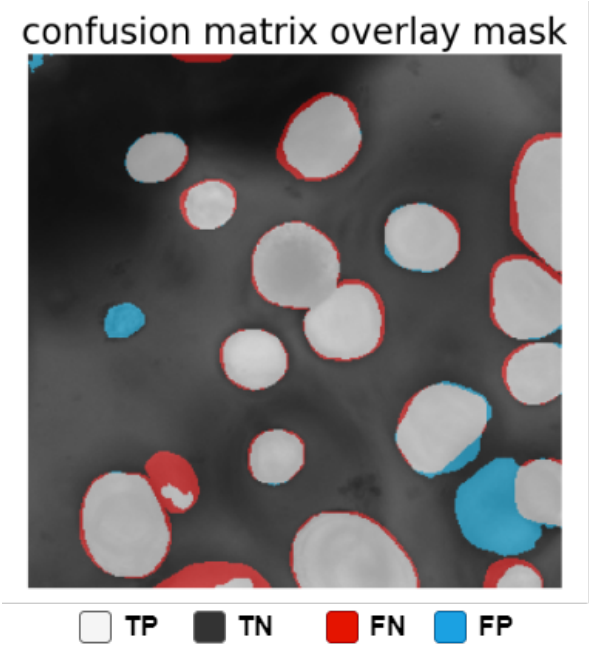

Figure 5.1: Confusion matrix for the mask channel showing the true positives, true negatives, false negatives and false positives.

Training for longer (3 epochs compared to 2) appears to have a positive effect during training (Figures 4.9 - 4.12), which is to be expected. Nevertheless, testing results show that this can introduce overfitting. As the model is going through the test set an extra time, it learns the distribution of the data and losses flexibility when presented with new data. Manually decreasing the learning rate seems to reduce the overfitting effect, but a shorter training time remains the preferable option. Additionally, using the ReduceOnPlateau scheduler improves the F1 score and creates a drop in the overlapping channel loss (Figure 4.9) right around step 46,000, the exact time when the scheduler decides to decrease the learning rate from 10^{-5} to 10^{-6} (Figure 3.9).

Figure 5.2 presents the prediction results of all 10 models for 5 randomly selected images from the test set. Overall, the binary mask has a much more accurate segmentation than the overlapping mask (also seen in Tables 4.2 and 4.3, as well as the confusion matrix in Figure 5.1), as it is uncomplicated for the model to differentiate between 2 classes, compared to 3 in the multi-class task. Many large organoids are missing from the predicted segmentation and the reason for this might be that the model did not have many examples of large organoids in the test set, or they were not segmented in the true mask. Some networks also predict extra organoids that do appear in the original image, but are not segmented in the true mask (for example, the top-left organoid in the first row). This contributed to a decrease in the F1 scores. The difference in class frequency between the train and the test set (Table 3.1) is also an explanation as to why the test results are generally lower than those obtained in training. The best two models in terms of the F1 score, as indicated in Tables 4.2 and 4.3 are *focal_tversky* and *focal_ce_dice*. The first one uses a simple architecture with the complex combination of losses FL + FTL, while the second one has a double architecture and simpler losses CE + DL. One main point that can be accentuated here is that, while complexity can increase performance, too much complexity is detrimental. Thus, combining the best attributes in model *double_focal_tversky* decreases the F1 score, so, once more, balance remains the key.

6 Conclusions

An accurate segmentation of biomedical images is the first step to automatic cancer diagnosis. While cells and nuclei are different structures, when presented as a 2D image, they share many of the morphometric attributes with 2D images of organoids. This study proposed several models based on the

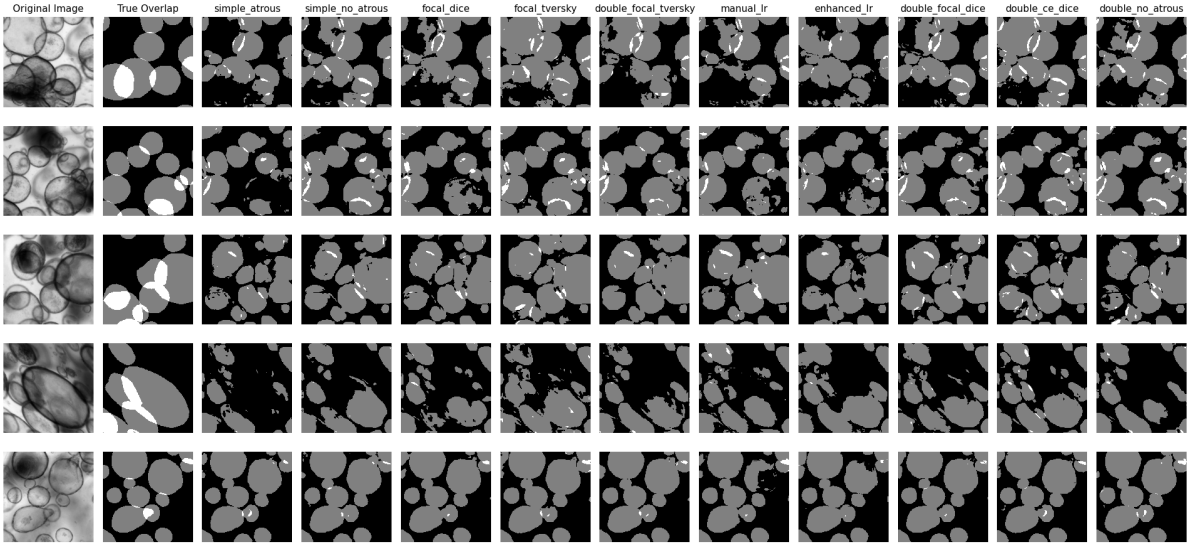

Figure 5.2: The output of the 10 models on 5 random images from the test set.

encoder-decoder architecture and focused on solving the class imbalance issue by using a combination of losses such as the Focal loss and the Focal Tversky loss, which reduces the loss for common examples, so that the model can focus on the infrequent classes. Other additions include the double architecture, as well as the residual-atrous skip connections, which reduce the semantic gap between the encoder and the decoder. The scheduler is also used to automatically decrease the learning rate.

In future work, the focus should be on creating accurate and faithful ground truth masks which include large organoids and overlaps. Correctly annotated data benefits the model, by making it discover the relationships between the background, organoids or overlapping areas. Moreover, with more balanced datasets, the network can be trained to differentiate between different levels of overlaps. In terms of the segmentation process, several studies obtained good results when separating the detection process of the boundaries from that of the center. Xin et al. (2012) used single-path voting and mean-shift clustering for center localization and an interactive model for boundary detection. X. Li et al. (2019) had a double U-Net model with two decoders, for decoding the contours and the interiors separately. A different study (Molnar et al., 2016) attempts to solve the overlapping problem by using active contours and the property that the intensities of touching nuclei are additive.

Finally, the best models proposed in this paper can also be used as part of an online platform, similar to that proposed by Matthews et al. (2022), that can generate binary and overlapping masks for new data, which can be used as training input for improved models.

References

Ahuja, S., Panigrahi, B., & Gandhi, T. K. (2021). Fully automatic brain tumor segmentation using deeplabv3+ with variable loss functions. In *2021 8th international conference on signal processing and integrated networks (spin)* (pp. 522–526).

Albuquerque, C., Vanneschi, L., Henriques, R., Castelli, M., Póvoa, V., Fior, R., & Papanikolaou, N. (2021). Object detection for automatic cancer cell counting in zebrafish xenografts. *Plos one*, *16*(11), e0260609.

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation [Journal Article]. *IEEE transactions on pattern*

analysis and machine intelligence, *39*(12), 2481-2495.

Buda, M., Maki, A., & Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks*, *106*, 249–259.

Chang, K., Bai, H. X., Zhou, H., Su, C., Bi, W. L., Agbodza, E., ... Kalpathy-Cramer, J. (2018). Residual convolutional neural network for the determination of idh status in low- and high-grade gliomas from mr imaging [Journal Article]. *Clinical Cancer Research*, *24*(5), 1073-1081.

Dhankhar, P., & Sahu, N. (2013). A review and research of edge detection techniques for image segmentation. *International Journal of Computer Science and Mobile Computing*, *2*(7), 86–92.

Freedman, B. S., Brooks, C. R., Lam, A. Q., Fu, H., Morizane, R., Agrawal, V., ... others (2015). Modelling kidney disease with crispr-mutant kidney organoids derived from human pluripotent epiblast spheroids. *Nature communications*, *6*(1), 1–13.

Garcez, P. P., Loiola, E. C., Madeiro da Costa, R., Higa, L. M., Trindade, P., Delvecchio, R., ... Rehen, S. K. (2016). Zika virus impairs growth in human neurospheres and brain organoids. *Science*, *352*(6287), 816–818.

Geiger, A., Lenz, P., Stiller, C., & Urtasun, R. (2013). Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, *32*(11), 1231–1237.

Grys, B. T., Lo, D. S., Sahin, N., Kraus, O. Z., Morris, Q., Boone, C., & Andrews, B. J. (2017). Machine learning and computer vision approaches for phenotypic profiling [Journal Article]. *Journal of Cell Biology*, *216*(1), 65-71.

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the ieee international conference on computer vision* (pp. 2961–2969).

Kaushik, G., Ponnusamy, M. P., & Batra, S. K. (2018). Concise review: current status of three-dimensional organoids as preclinical models. *Stem Cells*, *36*(9), 1329–1340.

Kiran, I., Raza, B., Ijaz, A., & Khan, M. A. (2022). Denseres-unet: Segmentation of overlapped/clustered nuclei from multi organ histopathology images [Journal Article]. *Computers in Biology and Medicine*, 105267.

Lakshmi, S., Vijayasenan, D., Sumam, D. S., Sreeram, S., & Suresh, P. K. (2019). An integrated deep learning approach towards automatic evaluation of ki-67 labeling index. In *Tencon 2019-2019 ieee region 10 conference (tencon)* (pp. 2310–2314).

Li, M., Zhang, X., Thrampoulidis, C., Chen, J., & Oymak, S. (2021). Autobalance: Optimized loss functions for imbalanced data [Journal Article]. *Advances in Neural Information Processing Systems*, *34*.

Li, X., Wang, Y., Tang, Q., Fan, Z., & Yu, J. (2019). Dual u-net for the segmentation of overlapping glioma nuclei [Journal Article]. *Ieee Access*, *7*, 84040-84052.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection [Conference Proceedings]. In *Proceedings of the ieee international conference on computer vision* (p. 2980-2988).

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer vision – eccv 2014* (pp. 740–755). Cham: Springer International Publishing.

Liu, J., Chang, W.-C., Wu, Y., & Yang, Y. (2017). Deep learning for extreme multi-label text classification. In *Proceedings of the 40th international acm sigir conference on research and development in information retrieval* (pp. 115–124).

Lu, Z., Carneiro, G., Bradley, A. P., Ushizima, D., Nosrati, M. S., Bianchi, A. G., ... Hamarneh, G. (2016). Evaluation of three algorithms for the segmentation of overlapping cervical cells [Journal Article]. *IEEE journal of biomedical and health informatics*, *21*(2), 441-450.

Matthews, J. M., Schuster, B., Kashaf, S. S., Liu, P., Bilgic, M., Rzhetsky, A., & Tay, S. (2022). Organoid: a versatile deep learning platform

for organoid image analysis [Journal Article]. *bioRxiv*.

Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N., & Terzopoulos, D. (2021). Image segmentation using deep learning: A survey [Journal Article]. *IEEE transactions on pattern analysis and machine intelligence*.

Moen, E., Bannon, D., Kudo, T., Graf, W., Covert, M., & Van Valen, D. (2019). Deep learning for cellular image analysis [Journal Article]. *Nat Methods*, *16*(12), 1233-1246.

Molnar, C., Jermyn, I. H., Kato, Z., Rahkama, V., Östling, P., Mikkonen, P., ... Horvath, P. (2016). Accurate morphology preserving segmentation of overlapping cells based on active contours [Journal Article]. *Scientific reports*, *6*(1), 1-10.

Naylor, P., Laé, M., Reyal, F., & Walter, T. (2019). Segmentation of nuclei in histopathology images by deep regression of the distance map. , *38*(2), 448-459.

Rastogi, P., Khanna, K., & Singh, V. (2022). Leufeatx: Deep learning-based feature extractor for the diagnosis of acute leukemia from microscopic images of peripheral blood smear [Journal Article]. *Comput Biol Med*, *142*, 105236.

Ronneberger, O., Fischer, P., & Brox, T. (2019). U-net: Convolutional networks for biomedical image segmentation [Conference Proceedings]. In *International conference on medical image computing and computer-assisted intervention* (p. 234-241). Springer.

Ryan, A. R., England, A. R., Chaney, C. P., Cowdin, M. A., Hiltabidle, M., Daniel, E., ... Cleaver, O. (2021). Vascular deficiencies in renal organoids and ex vivo kidney organogenesis. *Developmental biology*, *477*, 98–116.

Salehi, S. S. M., Erdogmus, D., & Gholipour, A. (2017). Tversky loss function for image segmentation using 3d fully convolutional deep networks [Book Section]. In (p. 379-387). Springer International Publishing.

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

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sun, T., Chen, Z., Yang, W., & Wang, Y. (2018). Stacked u-nets with multi-output for road extraction [Conference Proceedings]. In *Proceedings of the ieee conference on computer vision and pattern recognition workshops* (p. 202-206).

Tian, Z., Shen, C., Chen, H., & He, T. (2019). Fcos: Fully convolutional one-stage object detection. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 9627–9636).

Xie, W., Noble, J. A., & Zisserman, A. (2018). Microscopy cell counting and detection with fully convolutional regression networks [Journal Article]. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging Visualization*, *6*(3), 283-292.

Xin, Q., Fuyong, X., Foran, D. J., & Lin, Y. (2012). Robust segmentation of overlapping cells in histopathology specimens using parallel seed detection and repulsive level set [Journal Article]. *IEEE Transactions on Biomedical Engineering*, *59*(3), 754-765.

Xue, J.-H., & Titterington, D. M. (2011). t-tests, f-tests and otsu's methods for image thresholding. *IEEE transactions on image processing*, *20*(8), 2392–2396.

Yang, X., Li, H., & Zhou, X. (2006). Nuclei segmentation using marker-controlled watershed, tracking using mean-shift, and kalman filter in time-lapse microscopy. *IEEE Transactions on Circuits and Systems I: Regular Papers*, *53*(11), 2405–2414.

Zhou, Z.-H., & Liu, X.-Y. (2005). Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Transactions on knowledge and data engineering*, *18*(1), 63–77.