DEEP CONVOLUTIONAL NETWORK FOR TUMOR BUD DETECTION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

By
Soner Koç
April 2019

Deep Convolutional Network for Tumor Bud Detection
By Soner Koç
April 2019

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Çiğdem Gündüz Demir(Advisor)

Selim Aksoy(Co-Advisor)

Ramazan Gökberk Cinbiş

Shervin Rahimzadeh Arashloo

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

DEEP CONVOLUTIONAL NETWORK FOR TUMOR BUD DETECTION

Soner Koç
M.S. in Computer Engineering
Advisor: Çiğdem Gündüz Demir
Co-Advisor: Selim Aksoy
April 2019

The existence of tumor buds is accepted as a promising biomarker for staging colorectal carcinomas. In the current practice of medicine, these tumor buds are detected by the manual examination of a immunohistochemically (IHC) stained tissue sample under a microscope. This manual examination is time-consuming as well as it may lead to inter-observer variability. In order to obtain fast and reproducible examinations, developing computational solutions has been becoming more and more important. With this motivation, this thesis presents a fully convolutional network design for the purpose of automatic tumor bud detection, for the first time. This network design extends the U-net architecture by considering up-to-date learning mechanisms. These mechanisms include using residual connections in the encoder path, employing both ELU and ReLU activation functions in different layers of the network, training the network with a Tversky loss function, and combining outputs of different layers of the decoder path to reconstruct the final segmentation map. Our experiments on 3295 image tiles taken from 23 whole slide images of IHC stained colorectal carcinomatous samples show that this extended version helps alleviate the vanishing gradient problem and those related with having a high class-imbalance dataset. And as a result, this network design yields better segmentation results compared with those of the two state-of-the-art networks.

Keywords: Deep learning, fully convolutional networks, digital pathology, tumor budding, colorectal carcinomas.

ÖZET

TÜMÖR TOMURCUKLANMA TESPİTİ İÇİN DERİN EVRİŞİMSEL AĞ

Soner Koç
Bilgisayar Mühendisliği, Yüksek Lisans
Tez Danışmanı: Çiğdem Gündüz Demir
İkinci Tez Danışmanı: Selim Aksoy
Nisan 2019

Tümör tomurcuklanmasının gözlenmesi, kolorektal karsinomların evrelenmesinde ümit veren bir biyobelirteç olarak kabul edilmektedir. Mevcut tıp uygulamalarında, bu tümör tomurcukları, immünohistokimyasal (IHK) olarak boyanmış doku örneğinin mikroskop altında manuel olarak incelenmesi ile tespit edilmektedir. Öte yandan bu manuel inceleme zaman kaybına ve aynı zamanda gözlemciler arası değişkenliğe yol açabilmektedir. Hızlı ve tekrarlanabilir incelemeler için, bilgisayar destekli çözümler geliştirmek giderek daha fazla önem kazanmaktadır. Bu motivasyon ile, bu tez, otomatik tümör tomurcuklanma tespiti amacıyla ilk defa bir tam evrişimsel ağ tasarımı sunmaktadır. Bu ağ tasarımı, güncel öğrenme mekanizmaları dikkate alınarak U-net mimarisi üzerine geliştirilmiştir. Bu mekanizmalar, kodlayıcı evresinde veri besleme bağlantılarının kullanılmasını, hem ELU hem de ReLU aktivasyon fonksiyonlarının ağın farklı katmanlarında kullanılmasını, ağın Tversky hata hesaplama fonksiyonuyla eğitilmesini ve son bölütleme haritasının geri çatılması için geri kodlayıcının farklı katmanlarında elde edilen çıktıların birleştirilmesini içerir. IHK ile boyanmış kolorektal karsinom örneklerinin 23 tam biyopsi slayt görüntüsünden alınan 3295 görüntü üzerinde yaptığımız deneyler, bu genişletilmiş versiyonun gradyan sıfırlanması problemini ve çok büyük sınıf dengesizliği durumuna sahip veri setindeki olumsuzlukları hafifletmeye yardımcı olduğunu göstermiştir. Sonuç olarak, bu ağ tasarımının, yaygın kullanılan iki başka ağa kıyasla daha iyi bölütleme sonuçları verdiğini göstermiştir.

Anahtar sözcükler: Derin öğrenme, tam evrişimsel ağlar, dijital patoloji, tümör tomurcuklanması, kolorektal karsinomlar.

Acknowledgement

This thesis is the final step of my journey in obtaining my M.S. degree. At the end of this journey, I would like to thank all people who motivate and support me throughout my thesis studies.

First and foremost, it would be difficult to overstate my sincere gratitude to my supervisors, Çiğdem Gündüz Demir and Selim Aksoy, for giving me the opportunity to work with them, their continuous support, patience, and motivation in my master studies. Their guidance assisted me in all the time of research and writing this thesis. Thanks to their valuable guidance, I have had experience in different areas in computer vision, medical image analysis, and deep learning. I consider myself lucky to have an academic experience under their immense technical knowledge.

I am grateful and want to thank Ramazan Gökberk Cinbiş and Shervin Rahimzadeh Arashloo for attending my jury and reading my thesis.

I would like to thank Fraunhofer-Institute for Integrated Circuits IIS, Erlangen-Germany, for providing whole slide images of colorectal carcinoma biopsy samples to be used in my studies.

I would also like to thank Hamdi Dibeklioğlu and Can Fahrettin Koyuncu for their comments in our useful discussions about deep learning and medical image analysis.

I owe my deepest gratitude to my big family, first of all, my parents, Mehmet Koç and Hacer Koç, and my dear wife, Merve Koç, and also my sisters, Yüksel and Göksel, and my brothers Atilla, Bülent, and Serdar. Feeling my family's endless support and love have always encouraged me to persist no matter what and have always been a source of motivation for me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Today, histopathological examination is the gold standard to diagnose many neoplastic diseases including cancer. This routine pathology practice involves manually examining biopsy specimens under a microscope [1]. This use of microscopes results in frequent errors in the process of diagnosis, prevents standardization of analysis, and requires a lot of time and effort due to the huge workload of manually analyzing the biopsies [2]. This workload of pathologists has been increasing significantly with an increase of cancer cases in developed and developing countries. In order to increase the reliability and efficiency of these analyses as well as to decrease the workload of the pathologists, developing computational tools for digital pathology has become more and more important.

The tumor bud (TB) detection is crucial especially for staging colorectal carcinoma (CRC), the third most diagnosed type of cancer worldwide and one of the leading causes of cancer-related deaths [3]. A tumor bud is identified by the presence of a single tumor cell or a cluster of up to five nested tumor cells at an invasive front of colorectal carcinomatous regions or around the center of colorectal carcinomatous tumor mass, which is usually distributed as improper clusters

[4, 5]. The intensity of tumor buds is directly proportional to a lymph node and distant metastasis of a CRC patient [4]. It has been used as a supportive factor that is incorporated into risk staging protocols. For these reasons, the existence of tumor buds is accepted as an independent adverse prognostic indicator for CRC [4, 5].

For detecting tumor buds, immuno-histochemically (IHC) stained samples are used since it is very hard, almost impossible, to detect tumor buds in a tissue stained with the routinely used hematoxylin and eosin (HE) technique. In an IHC stained sample, an antibody called cytokeratin does not stain lymph or stromal cells like HE, and thus, helps distinguish metastatic tumor buds from their nearby non-tumor bud cells. Therefore, CRC patients can directly benefit from the accurate and reliable detection of tumor buds. However, tumor buds are mostly detected by pathologists and this approach brings about a huge inter-observer variance especially if a lymph node is negative and there are tiny metastatic parts in whole colorectal carcinoma slides [4]. Furthermore, it is very time-consuming and painstaking for the pathologists as they need to examine large amounts of data in this manual detection [6]. Although tumor bud detection and evaluation are laborious, there have been only a few computational studies that concentrate on its automated detection until deep learning emerges. In recent decades, there have been research efforts to develop automated computer-assisted image processing solutions to help pathologists analyze tumor buds.

With the improvements in the area of deep learning, and in particular in its subfield called convolutional neural networks (CNNs), a wide range of computer vision problems [7, 8, 9] has been tackled more accurately. There are significant improvements in performance thanks to CNNs in detection, recognition, and segmentation of objects. Similarly, in medical segmentation problems, solutions where a CNN is applied [10, 11, 12] outperform the state-of-the-art methods.

Although CNN-based solutions are quite successful when compared to traditional approaches, they have some limitations related with the training set size and the depth of the required network. The success of the methods typically relies on how big the training set is and how deeper the network can grow [13]. On

the other hand, fully convolutional networks (FCNs) constitute an architecturally variant of CNNs and they deliver better results than the latest CNN approaches even when they are provided with fewer training images [13]. Pixel-wise FCN solutions take an arbitrary-sized input and output an efficient inference in the same size with the input. In recent years, learning models have been proposed for alleviating the limitations of using an FCN for medical image segmentation; namely, a huge class-imbalance in the input data and data scarcity. One of the most popular and state-of-art architectures for semantic medical image segmentation is the U-Net network of Ronneberger *et al.* [14]. It preserves the high resolution feature and context information of the input until the end of the network by combining segmentation information of the feature extraction step with segmentation of following decoder layers of the network.

Likewise, for tumor bud detection, CNNs working on small patches cannot satisfy some of the criteria that show the existence of a tumor bud due to their restricted fields of view since these image patches are cropped out of a whole colorectal carcinoma slide just as to include a tumor bud instance. These criteria are the existence of tumor bud in a tumor mass or its invasive front and the closeness of a tumor bud instance to a group of other tumor buds. The criteria are dictated by the location of a tumor bud instance with respect to tumorous colon glands. These should be taken into account at the same time to get more accurate results for tumor bud detection. Thus, patch-based approaches may need a considerable amount of pre- or post-processing.

On the other hand, fully convolutional networks, which are specialized for semantic medical image segmentation, can learn significant features related to the criteria of being a tumor bud instance. In particular, these networks provide a better solution with the possibility of working with images that have a wider field of view. They are capable of utilizing the locations of tumor buds together with their contexts. These networks give superior performance for segmentation task even when they are trained with fewer training images compared to patch-based CNN solutions [14, 15, 16]. Furthermore, residual learning approaches are adapted in many models [17, 18] to strengthen the learning process by improving the gradient flow since an increase in the depths of neural networks results in a

commonly faced saturation problem.

With this motivation, the aim of this thesis is to employ a fully convolutional network for automatic tumor bud detection in an image of an IHC stained colorectal carcinomatous sample. To this end, a network architecture similar to U-Net has been designed and implemented by also adapting up-to-date robust learning mechanisms.

1.2 Contribution

This thesis proposes to use a fully convolutional network that is enhanced with recent learning mechanisms for tumor bud detection in contrast to the use of a patch-based CNN. These mechanisms include the use of residual connections in the designed network and exponential rectified linear units (ELUs) together with rectified linear units (ReLUs) in order to alleviate saturation problems of the gradients. This also includes the use of the Tversky loss function to learn a more effective network on our high class-imbalance dataset. Additionally, the designed network combines multiple segmentation maps at different scales in its reconstruction path to obtain a better final segmentation output. In particular, this thesis has the following contributions:

- It proposes to use a fully convolutional network, which we call Residual Tumor Bud Net (RTB-net), for the purpose of automatic tumor bud detection in the images of IHC stained colorectal carcinomatous samples. To the best of our knowledge, this is the first fully convolutional network that has been designed for this particular purpose. In this designed network, localization and classification of tumor buds are performed simultaneously in contrast to a patch-based CNN approach, which requires additional pre- and post-processing steps to output pixel-wise localized and classified segmentation maps.

- The designed RTB-net adapts a residual learning mechanism for the feature

extraction steps in its encoder path and selects its activation functions to diminish the vanishing gradient problem in a highly imbalanced CRC tumor bud dataset. It also combines segmentation maps of different scales in its decoder path to obtain better final segmentation. Additionally, it uses the Tversky loss function in the training process to help alleviate the negative effects of working with a high class-imbalance dataset.

1.3 Outline of the Thesis

The rest of the thesis is organized as follows: In Chapter 2, the necessary background regarding tumor budding, deep learning, and convolutional neural networks are given, and related studies are presented. In Chapter 3, the details of the proposed network for automatic tumor bud detection are explained. In Chapter 4, the dataset, evaluation metrics, experiments, and the results are presented in detail. Finally, in Chapter 5, the thesis is concluded with a discussion on its potential future research directions.

Chapter 2

Background

This chapter provides some background on tumor buds and outlines the basics of deep convolutional neural networks. It also presents some related work on the application of fully convolutional networks for medical image segmentation and tumor bud detection.

2.1 Tumor Budding

Cancer staging refers to extracting information about a patient's cancer such as how big a tumor instance is and how a tumor instance has spread. Staging of the cancer is informative for doctors, especially to know how serious cancer is and to help plan the treatment process of cancer. The TNM (tumor, node, metastases) system is the gold standard used as a cancer staging system also for colorectal carcinoma. It helps divide patients into prognostic groups [19].

However, in colorectal carcinoma, a wide diversity of colorectal tumors in similar stages can exhibit significant dissimilarities in their behaviours among different patients. This observation created a need for another prognostic biomarker for colorectal carcinoma. Tumor buds are accepted as promising biomarkers to overcome this issue [5, 19, 20].

Tumor buds correspond to clusters of up to five malignant cells in the stroma of a tumor. Figure 2.1 gives examples of different tumor bud cases. They are typically detected in close affinity to the invasive front of a colorectal tumor mass and this is called peritumoral budding which is the most common scenario. Tumor bud within the tumor mass is called intratumoral budding which is a rare case compared to peritumoral budding [21, 22, 23]. Figure 2.2 gives an illustration for both tumor bud types.

Furthermore, what makes a tumor bud noteworthy is that it is a robust prognostic factor in colorectal cancer. It has a direct relationship with the high tumor grade and dense TNM stages [5, 20, 21, 22, 23]. First, tumor buds are related with lymph node metastases in pT1 colorectal cancer [24]. Second, appearance of a tumor bud in stage two colorectal cancer is a clue for better survival rate compared to rare and no tumor bud cases [25, 26]. However, until the last years, tumor buds have not found a practical use in research due to the lack of a standardized scoring system. International Tumor Budding Consensus Conference (ITBCC) was organized in 2016 to make an agreement on the standardized scoring system for tumor buds to be used in colorectal cancer-related research [4]. Thanks to the consensus, tumor bud related research has gained great momentum.

2.2 Deep Learning

Deep learning, as one of the sub-field of machine learning, gives the capability to computational models to discover features of data with multiple layers of abstractions. It is not task specific and it can be applied in wide domains, which includes speech recognition, object detection, and signal processing. As the main point, the definition of features is not required in deep learning applications in advance. These are learned in an automated way where a sequence of layers in deep learning helps extract features from low-level to more abstract features.

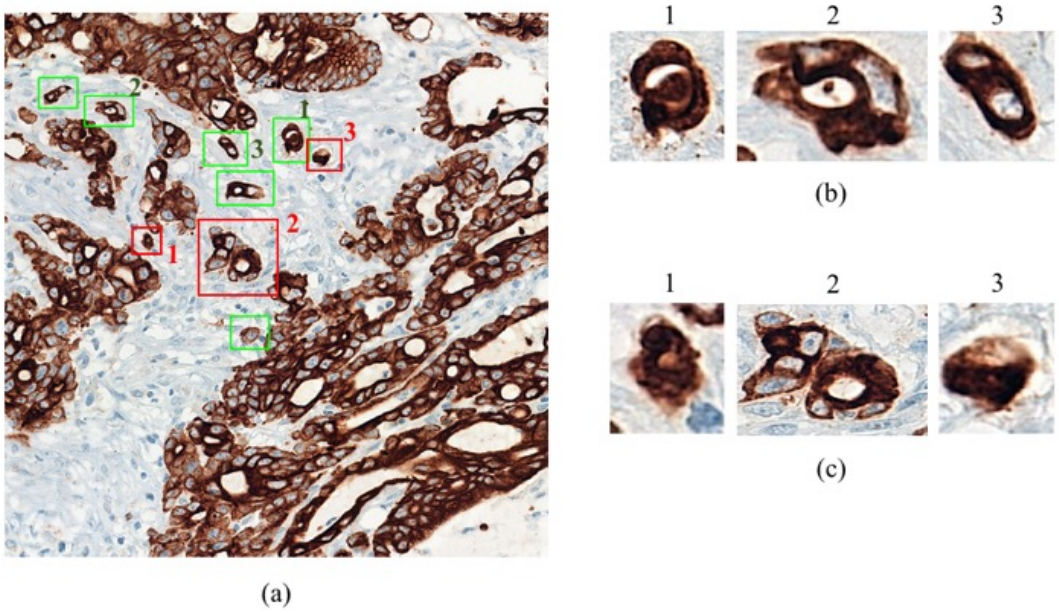
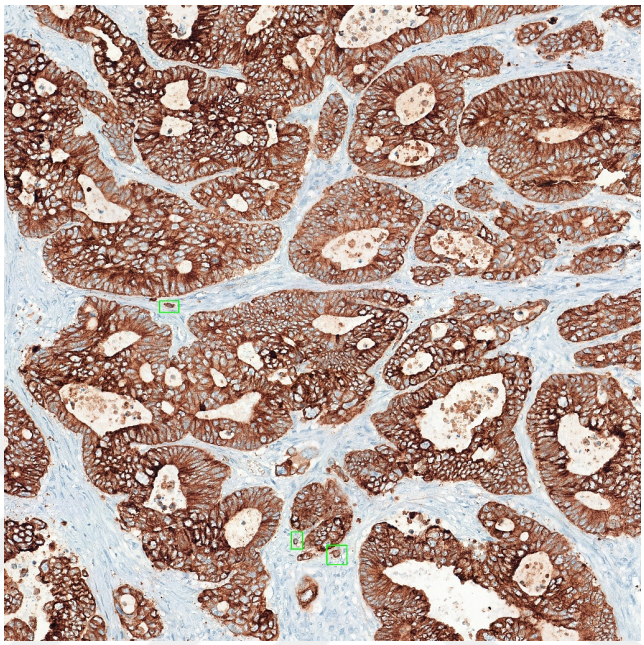
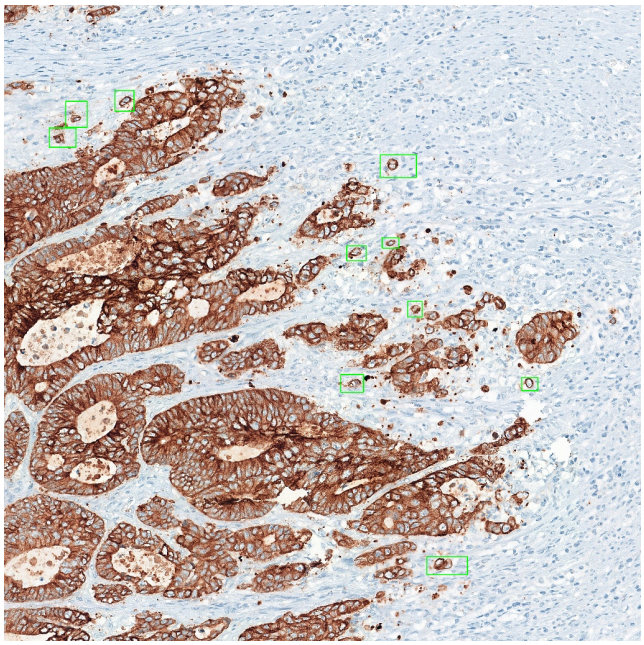


Figure 2.1: Examples of tumor bud cases. (a) Instance of an intratumoral region. Green boxes are ground truth annotations and red boxes are examples of regions that do not satisfy the set of criteria to be named as a true tumor bud. (b) Tumor bud instances with different number of distinguishable cells from the given intratumoral region that satisfy the criteria of being a tumor bud [4]. (c) Possible tumor budding occurrences that do not satisfy the criteria for being a tumor bud with different deficiencies. First one on the left, excluded due to no visible nucleus. Second, in the middle, excluded due to having over five visible cell nuclei. Third, excluded because of not having a clear and sharp cell borders.



(a)



(b)

Figure 2.2: Examples of regions that have tumor buds. (a) Intratumoral budding region sample which is part of a big tumor mass. (b) Peritumoral budding region sample which is located at the invasive front of a big tumor mass.

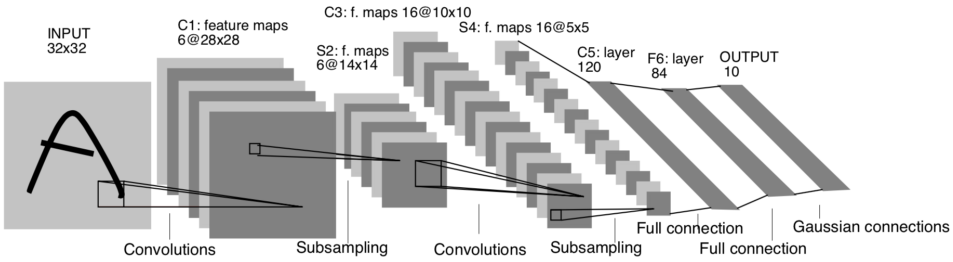


Figure 2.3: Architecture of the CNN called LeNet-5 which is designed for the digit recognition problem [27].

2.3 Convolutional Neural Networks

The convolutional neural network (CNN) is the most preferred deep learning architecture in the computer vision research area. The network accepts an image of raw pixels and outputs classification scores or pixel-level classifications for each class represented in the input. A CNN passes an input through a set of hidden layers which are composed of convolution layers, non-linearity layers, and pooling layers. There are a myriad of different CNN architectures with different architectural approaches and parameters. One of the classical architectures is LeNet of LeCun [27] whose architecture is portrayed in Figure 2.3. Layers that form convolutional neural networks are described below in detail with extra layers that are related to the content of the proposed architecture in the thesis.

Convolution Layer: The convolution operation is an essential component of a CNN. It is based on the dot product of sliding windows called learnable filters (kernel) on local regions of an input matrix. Filters pass over all the local regions in the input with a defined striding value that is a shifting amount to change the current working local region. Generally, volumes of filters have small sizes (like 3, 5, 7, 11) and their depths are always the same as the input. Due to this, a huge number of convolution operations are applied on input matrices if these are not as small as the kernel size. At the end, convolutions generate an activation map corresponding to the filters applied. As a result of consecutive convolutions, the network captures features of the input data thanks to the learned filters and

outputs a convolutional layer in the size that is result of Equation 2.1 where W is the input size, K is the filter size, P is the padding value, and S is the stride value.

$$\text{Output Size} = \frac{W - K - 2P}{S} + 1 \tag{2.1}$$

- Filter (Kernel) Number: The number of filters to be applied with the same depth as the input.
- Filter (Kernel) Size: The size of each filter to be used on a local region of the input.
- Zero Padding: To preserve the input and the output size, zeros are added to the edges of the input.
- Stride: The number of pixels to slide a filter over a local region of the input.

Transposed Convolution Layer: Transposed convolution layer is an inverse operation of the convolution layer. It is known as a learnable upsampling layer. It is popular in fully convolutional networks for upsampling the output of a convolution layer to the original image resolution. Hence, it is advantageous in detection and segmentation problems.

ReLU (Rectified Linear Unit) Layer: The main purpose of the layer is to introduce non-linearity to the network after each convolutional layer. It prevents the output to be a linear combination of the input with adding sparsity by setting negative values to zero as $f(x) = \max(x, 0)$. Previously, tanh and sigmoid activation functions were popular among scientific projects, but it is proven that as a non-linear activation function, ReLU layers are more robust and efficient [28]. Furthermore the network can be trained much faster with computational efficiency coming from the sparsity of ReLU which does not have a considerable negative effect on the accuracy. Besides, the ReLU activation function has a positive effect on the vanishing gradient problem as it only saturates in one direction.

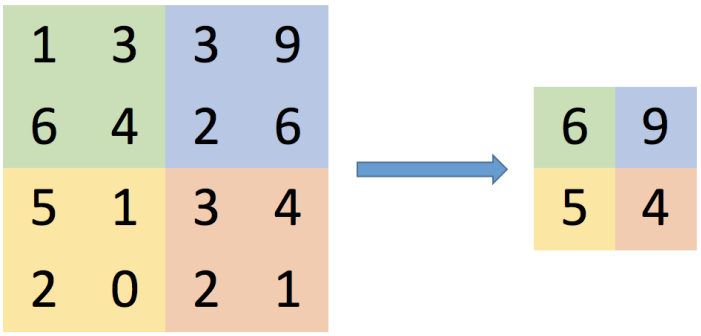


Figure 2.4: Illustration of a max-pooling operation. 2x2 filter is used with a stride value of 2.

ELU (Exponential ReLU) Layer: An exponential rectified linear unit (ELU) is like a rectified linear unit (ReLU) that eases the vanishing gradient problem thanks to having a linearly positive side. As opposed to ReLU, ELU has a slope for negative values as a log curve. It gives tolerance to negative values that help approach mean activations to zero as Equation 2.2. The main aim is to bring gradients much closer to natural gradients. The parameter of an ELU, α, controls the amount of saturation on the negative side. According to [29], networks designed with ELU layers have a faster learning process compared to ReLU adapted networks, in particular, for networks that have over five layers.

$$ELU(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha(\exp(x) - 1), & \text{otherwise} \end{cases} \tag{2.2}$$

Max-Pooling Layer: After convolution and non-linearity operations, commonly a max-pooling layer is applied to extract the highest activation value. Figure 2.4 is an illustration of the max-pooling operation for a single depth slice. This operation reduces the data size and help control overfitting to the training samples.

Dropout: Overfitting is one of the most common problems in deep learning applications where the parameters of the network memorize the training set.

Dropout eliminates a random set of parameters in the corresponding layer [30].

Loss Function: Loss function, also called objective function, is used to update the weights of the network toward the negative direction of the gradient during the gradient descent algorithm.

2.3.1 U-net

Fully convolutional networks (FCN) have gained popularity in the last few years with their outstanding performance in medical segmentation problems. One of the most popular FCNs specialized for medical image segmentation is called U-net that has gained attention thanks to its successful encoder and decoder structure. These two comprise the main part of the U-net architecture. The original U-net network architecture is given in Figure 2.5.

Encoder, downsampling part, aims to extract features for classification. It aggregates semantic information by working like a classic CNN. Two repeated convolution operations are followed by ReLU activation. Then, max-pooling is applied for image size reduction. In each downsampling step in the encoder part, several feature channels are duplicated. To cover the main aim of segmentation, maintaining semantic information along with spatial information is critical for the success of the neural network. Therefore, spatial information has to be preserved, which is handled by the decoder part.

Decoder begins after the last convolution block (bottleneck) in the encoder part of the U-net architecture and moves up feature maps with transposed convolution operations. In the beginning, it receives the semantic information that is extracted through the encoder part. Then, it brings together that information with high-resolution feature maps that come from the encoder path via long skip connections.

Architectures like U-net have various advantages for image segmentation problems. First, it helps learn contextual and global features at once. Second, it

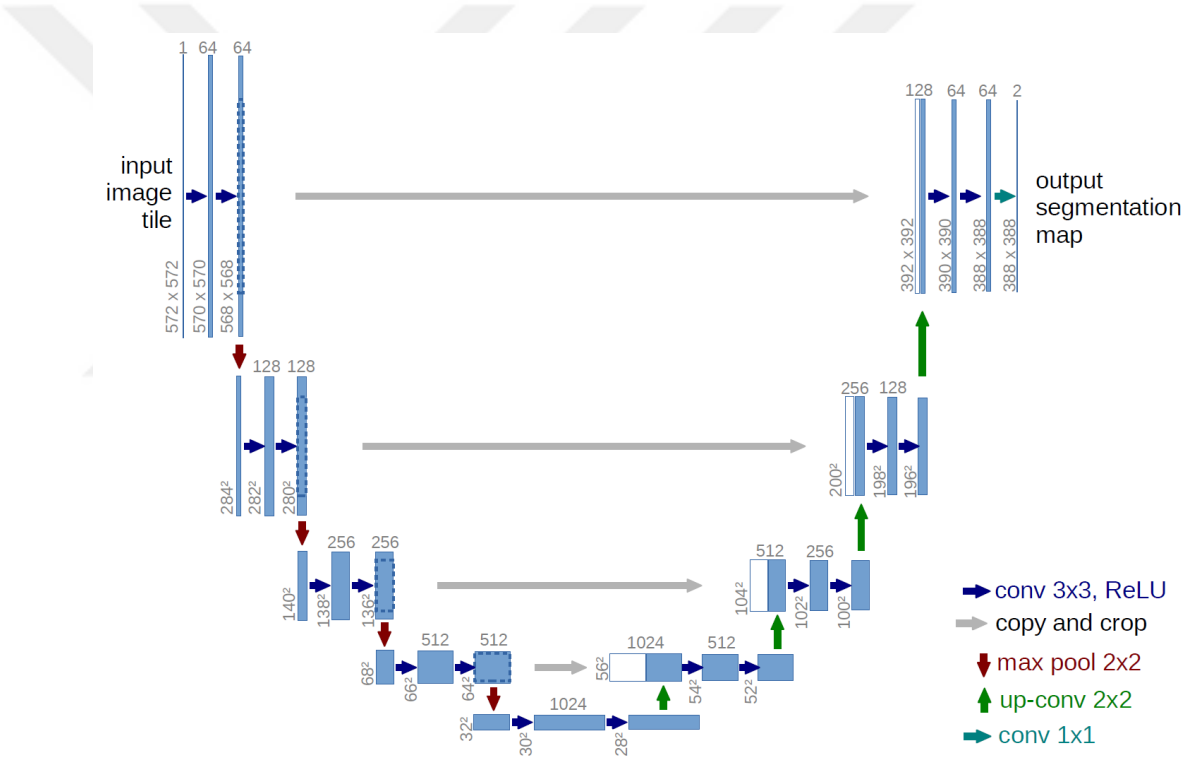


Figure 2.5: The original U-net architecture [14].

gives reasonably better performance in segmentation problems with very limited training samples. Third, the architecture takes an entire image as an input and processes it in convolution blocks as a whole by maintaining the entire context of the input. This is pointed as a major advantage of architectures like U-net compared to patch-based CNN solutions [14, 15].

2.4 Related Work

In the medical image segmentation literature, traditional machine learning approaches aim to classify foreground and background pixels based on handcrafted features, often following some post-processing steps to make the final predictions [31, 32, 33]. After the revolution in the field of artificial intelligence by deep learning, deep convolutional neural networks have driven the progress in medical image segmentation related tasks in the field.

In segmentation, a classification of each pixel is required. A classifier is applied to patches of an input image that are extracted around each pixel in a sliding window fashion. For instance, Ciresan *et al.* [34] applied deep neural networks to medical image segmentation. By using a sliding window, every pixel in each patch of stacks of electron microscopy images is passed through a fully connected CNN classifier for boundary prediction. It has some deficiencies like being slow due to separate runs for each image patch in the network and the need for making a trade-off between smaller and larger sizes of image patches (learning a limited context and a high number of pooling operations, respectively).

However, an adaptation of the fully convolutional approaches in CNN designs brings advantages compared to fully connected approaches [7]. Performance of fully convolutional approaches are improved in solutions such as recurrent neural networks (RNN) [35], DeepLab [36] and SegNet [37] where improvements for fine-tuning in a large dataset, efficient highlighting semantics on images, upsampling low-resolution input features for a better end-to-end solution are introduced. Most of the proposed approaches are designed to solve generic computer

vision tasks. On the other hand, there are some works that have been designed particularly for medical image segmentation tasks.

The U-shaped architecture of Ronneberger *et al.*, called U-net [14], is the fundamental fully convolutional approach adapted in recent years for medical image segmentation tasks which resemble convolutional auto-encoder designs. It has two stages: encoder and decoder. The number of feature maps is increased in the encoder stage with a reduction of the input size. Then, a reverse operation is done in the decoder stage (Figure 2.5). The fundamental feature of this approach is the established connections among these two stages, which are called long skip connections. These connections forward feature maps of early convolutional layers of the network to the later convolution layers in the decoder stage.

The design of Ronneberger *et al.* is also improved with some modifications especially by adding residual blocks to the U-net architecture [15, 18, 38]. Residual blocks are introduced in [17] with additional special skip connections among layers of VGGnet [13] in short intervals to overcome the vanishing gradient issue of deep neural networks. This is due to the depth of the network, which increases the complexity. This situation directly hampers convergence of the network from the beginning of the learning to the end [17, 39].

There are also other CNN based medical image segmentation approaches that are designed in a fully convolutional fashion [40, 41, 42]. In these works, the introduced skip connections in the U-net based architectures are removed. A transposed convolution operation is applied to low-resolution maps of segmentation to obtain the original input resolution. Also, additional segmentation maps are adapted from the initial layers of the network. In some of them, these maps are added to the final segmentation maps, and in some others, loss functions are updated in terms of the weights of these segmentation maps. However, the effects of long and short skip connections are analyzed, and both of the connections are observed as advantageous in deep network models for medical image segmentation tasks [43].

Pixel-wise fully convolutional approaches can be both more efficient and more

effective than patch-based approaches by improving their deficiencies like time-consuming operations due to redundant computations and not accessing global features in the input. However, fully convolutional approaches have to deal with two important limitations of medical image segmentation tasks. These are data scarcity and class-imbalance in medical tasks. Patch-based approaches can increase the number of samples in a dataset by sampling patches from input images to overcome data scarcity, which is not possible in pixel-wise fully convolutional approaches that operate on full image tiles as input. Hence, data augmentation techniques are adapted to increase the number of samples in a dataset with image transformations such as rotation, and translation [14, 15, 44, 45].

Additionally, training of pixel-wise FCN architectures has to take precautions for a class-imbalance in medical images [46]. Although patch-based approaches can balance their dataset by using an equal number of samples from each class, working with large image tiles in pixel-wise approaches is restricted with the original class distribution of the input which is mostly dominated by background pixels. In [15, 44, 38, 47], careful adaptation of a loss function by considering the pixel distribution is done to handle the class-imbalance of medical images.

Each of the models in the aforementioned studies has been trained on different medical image dataset from computer tomography (CT) to magnetic resonance imaging (MRI) to whole slide biopsy samples of diverse cancerous types. In this thesis, whole slide biopsy samples of colorectal carcinoma are the input of the proposed model for tumor bud detection. To the best of our knowledge, only Bokhorst *et al.* [16] studied tumor bud detection in colorectal carcinoma. The model in [16] is based on a patch-based convolutional approach. As mentioned, these models have potential problems such as the use of limited context and the vanishing gradient problem in the network.

In conclusion, several deep learning based approaches are developed to address and solve problems in medical image segmentation tasks. There are three main roadblocks for the developed works in this field. First, the memory demands of working with high-resolution images and storing the associated large number of feature maps in the network, is often solved by dividing input images to a few

patches [40, 41, 34, 42]. Second, data scarcity is handled by sampling patches and data augmentation [14, 15, 44, 45]. Third, class-imbalance is tackled by designing loss functions according to the class distributions in the dataset [47, 15, 44, 38]. In the proposed model, a pixel-wise fully convolutional neural network is designed for tumor bud detection in colorectal carcinoma samples in a fully automatic way with reasonable solutions to common problems in medical image segmentation.

Chapter 3

Methodology

This chapter discusses the details of a fully convolutional network (FCN), which we call RTB-net, designed by this thesis for the purpose of automatically segmenting and localizing tumor buds in the images of IHC stained colorectal carcinomatous (CRC) samples. This designed network is inspired by the two popular learning models, namely, the deep residual model [17] and the U-net model [14]. This network design is explained in the following sections and its schematic overview is given in Figure 3.1.

This chapter is organized as follows: First, the overview of the proposed network is described in Section 3.1. Then, the architecture of the proposed network with its stages and components is explained and architectural similarities and differences with respect to the inspired fully convolutional approaches are discussed in Section 3.2. This section also describes the designed residual blocks of the proposed architecture, which is different from the standard forward convolution blocks; these residual blocks are developed for helping construct a more effective deeper model [14, 48]. At the end, in Section 3.3, training of the network is explained with its loss function (Tversky loss), which is used to obtain better improvements over the basic Dice coefficient loss. This loss definition gives flexibility in changing the weights of false positives and false negatives.

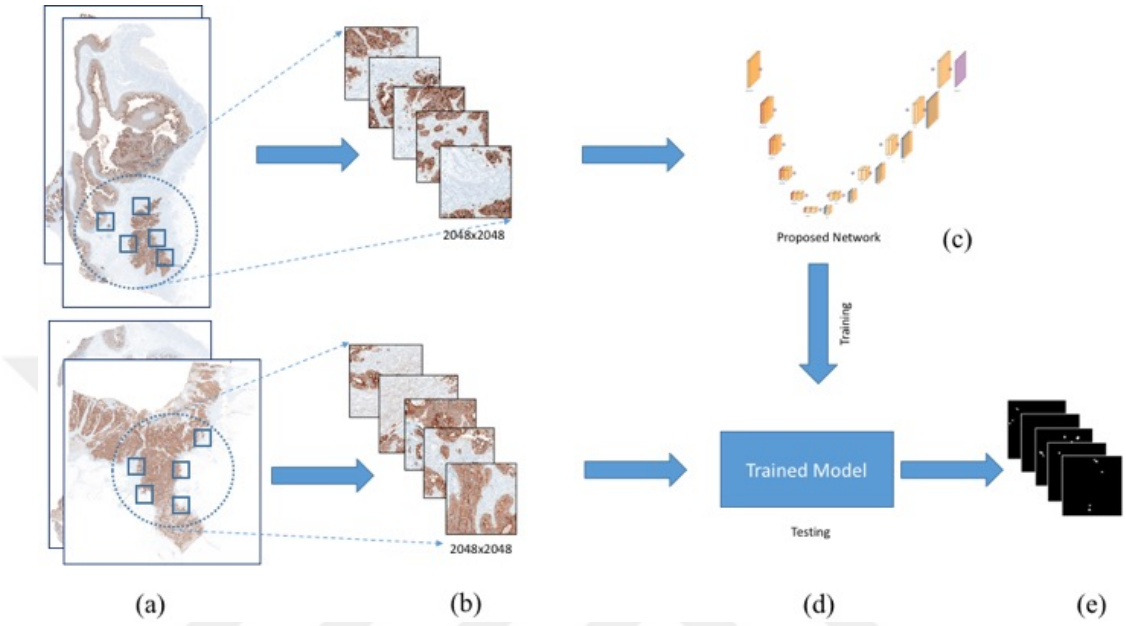
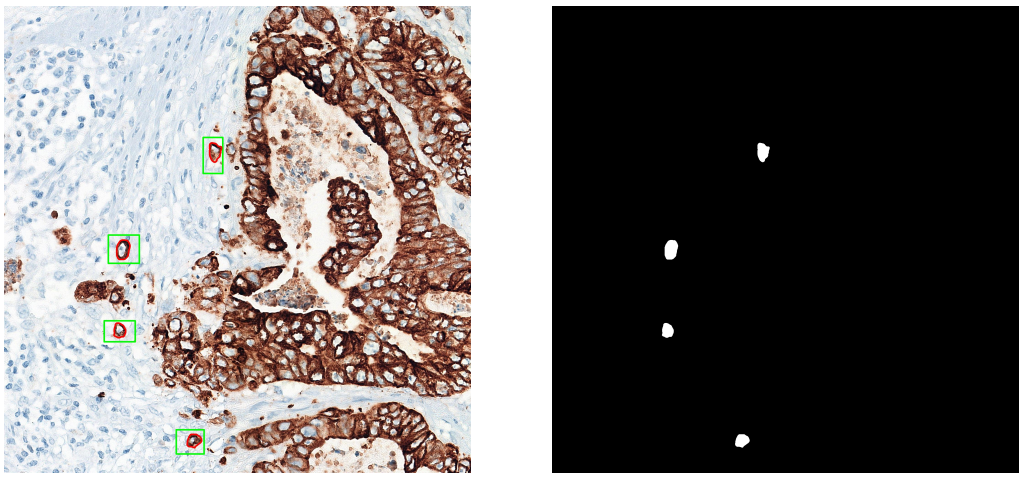


Figure 3.1: Schematic overview of the proposed approach. The training and test sets are formed by cropping out 2048 × 2048 image tiles (b) from whole slide images of colorectal carcinoma samples (a). After learning the model on the training set (c), the probability maps (e) are generated by the trained model for the test set samples (d).



(a) (b)

Figure 3.2: The input and output for an example image tile. (a) 2048 × 2048 RGB image tile cropped from an whole slide image where tumor bud bounding box annotations are drawn on the image tile. (b) 2048 × 2048 output map for the given RGB image tile. Pixels for the annotated tumor buds are extracted by the mask extractor explained in Section 3.1 and illustrated in Figure 3.3.

3.1 Proposed Model

The proposed RTB-net model takes an RGB image tile and outputs a segmentation map. The input and output for an example image tile are shown in Figure 3.2. The baseline of the proposed architecture is similar to that of the U-net model (Figure 2.5), which is a fully convolutional network design of Ronneberger *et al.* [14], and to other similar networks [15, 44, 45] where feature maps of the encoding path are joined with feature maps of the decoding path via long skip connections.

The input of RTB-net is an original RGB image tile cropped out of a whole slide image of an IHC stained colorectal carcinomatous sample. It has a size of 2048×2048 with no rescaling operation. When compared with other fully convolutional networks, this size of the input is relatively large. The use of a high-resolution image tile as an input can be problematic due to memory insufficiency. However, working with a larger image size is supported in many articles with improved results [34, 49]. This also enhances tumor bud detection in our method since it is easier to learn the internal and external morphological structure of a tumor bud with larger image tiles. Particularly, it has an outstanding impact on learning a nested cell count in a tumor bud instance, which is accepted as a strong evidence of the tumor bud existence [4]. Note that an input image is scaled between 0 and 1 by dividing its pixel values by 255, which is the maximum value in the RGB color space. This scaling makes the training step much faster and also helps alleviate the vanishing gradient problem.

The output of RTB-net is a segmentation map whose size is the same as the input image tile. The given annotations include a bounding box for each tumor bud instance. This bounding box is drawn small for some instances and larger for some others. Additionally, it does not indicate the exact morphology of the instance. Thus, to obtain more effective output maps, we develop a simple tumor bud extractor that approximately distinguishes the pixels of tumor bud instances from those of the others. For each given bounding box, this extractor first calculates a threshold on its pixels. It first takes the average of all pixels and

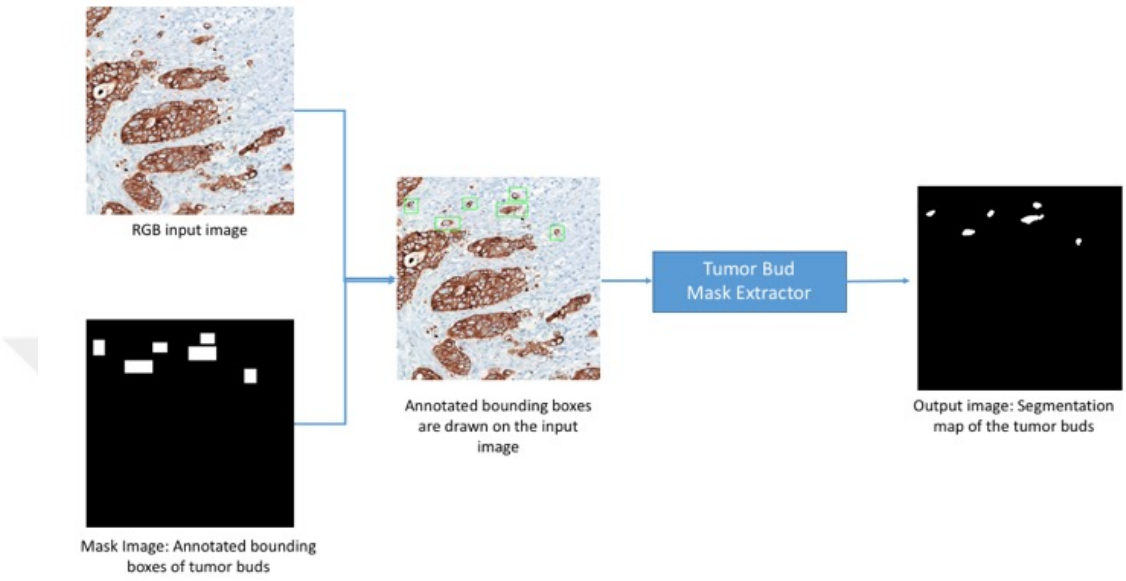


Figure 3.3: Illustration of mask extraction from the given bounding box annotations.

then lowers this average to its 85 percent to take relatively darker pixels. After filling the holes in the taken pixels and opens them by a morphological operator, it applies majority filtering to smooth the subregions. For both morphological opening and majority filtering, it uses a square structuring element with an edge size of 9. At the end, it identifies all connected component whose size is greater than 1500 pixels as tumor buds. If no connected component with this size is identified in a given bounding box, this extractor takes the one with the largest size even though this size is less than 1500 pixels. Note that this extractor (with its selected parameters) is only used for preparing the output for a training image tile. It is not used for the test image tiles. This mask extractor is illustrated in Figure 3.3.

The architecture of the proposed RTB-net is explained in the following section.

3.2 Network Architecture

The architecture of the proposed RTB-net model is based on that of U-net [14]. This architecture of RTB-net is given in Figure 3.4. As seen in this figure, the depth of the network is set to six to reflect the average tumor bud area in the input image tiles. The receptive field of the deepest block is consistent with this area. This network has three stages: 1) The encoder stage, which extracts features from the input until the bottom part of the U-shaped structure. 2) The bottleneck stage, which is the bottom-most part of the network that extracts last features from the input. 3) The decoder stage, which reconstructs the map from the features via transposed convolution (t-convolution) operations. Table 3.1 gives the details of the operations used in each of these stages.

3.2.1 Encoder Stage

The input image is passed through a set of convolutional blocks, called residual learning blocks (RLB). These residual learning blocks are designed for our tumor bud detection application. The comparison of these blocks and those of U-net and ResNet is given in Section 3.2.4.

The residual learning blocks in the encoder stage include two convolutional operations using 3×3 filters. The inputs of these convolutional operations are first fed to an ELU (exponential linear unit) activation unit in the top three RLBs of the encoder stage and to a ReLU (rectified linear unit) in the last two RLBs of the encoder stage (except for the initial encoding operation, which applies a convolution operation to the input image tile without any activation unit). This joint use of ELU and ReLU activation units is adapted from Chen *et al.* [50]. The definitions of the ELU and ReLU activation functions are given in the following equations; these functions are also plotted in Figure 3.5.

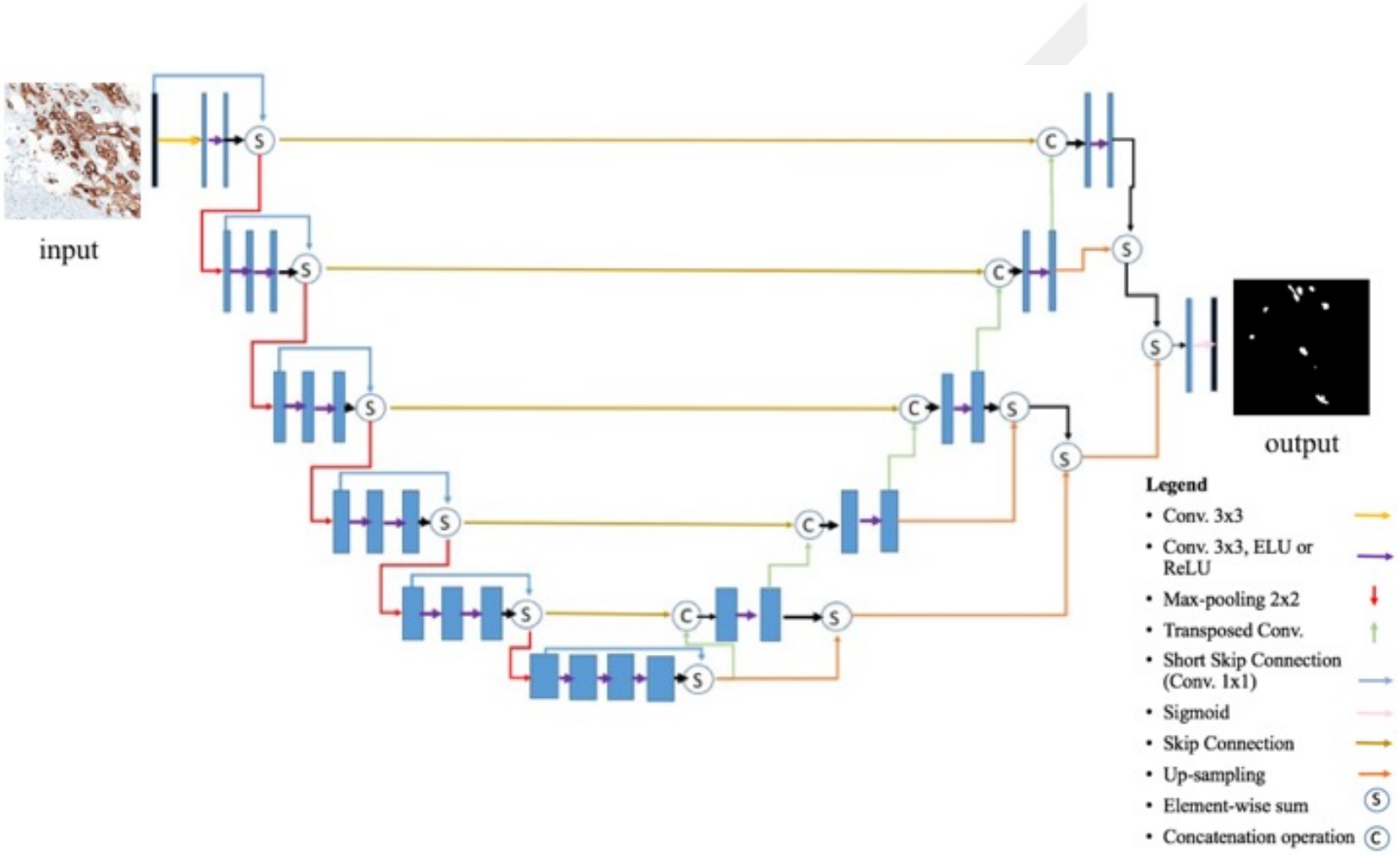


Figure 3.4: The proposed network architecture with its components being illustrated in different colors.

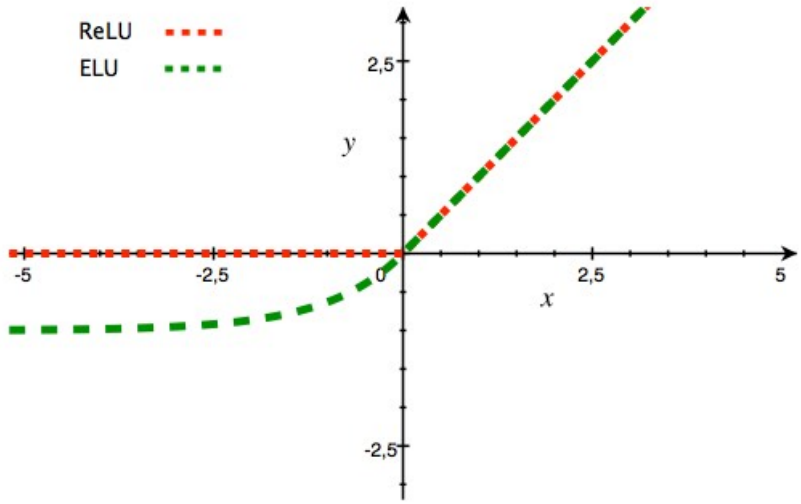


Figure 3.5: ELU and ReLU activation functions. Here ELU is plotted taking $\alpha = 1$.

$$\text{ReLU}(x) = \max(x, 0) \tag{3.1}$$

$$\text{ELU}(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha(\exp(x) - 1), & \text{otherwise} \end{cases} \tag{3.2}$$

where α should be selected in between 0 and 1. It determines the minimum (negative) value that ELU gives. In our work, we set its value to 1.

The motivation behind jointly using the ELU and ReLU activation functions is as follows: ReLU is one of the most commonly used non-linear activation functions for a hidden unit, which has shown to work quite efficiently [28]. However, it always gives zero for the negative axis, which "kills" the weights for the negative inputs. On the other hand, the output of ELU is exactly the same with that of ReLU for the positive axis, but unlike ReLU it also gives non-zero outputs for the negative axis. Thus, the use of ELU might be more effective especially for the top RLBs of the encoder stage. For its bottom (deeper) layers, ReLU is still used since killing the weights by making some of the inputs (negative inputs) zero is effective to decrease a possible saturation of ELU in the negative axis due to the network's increasing complexity [50].

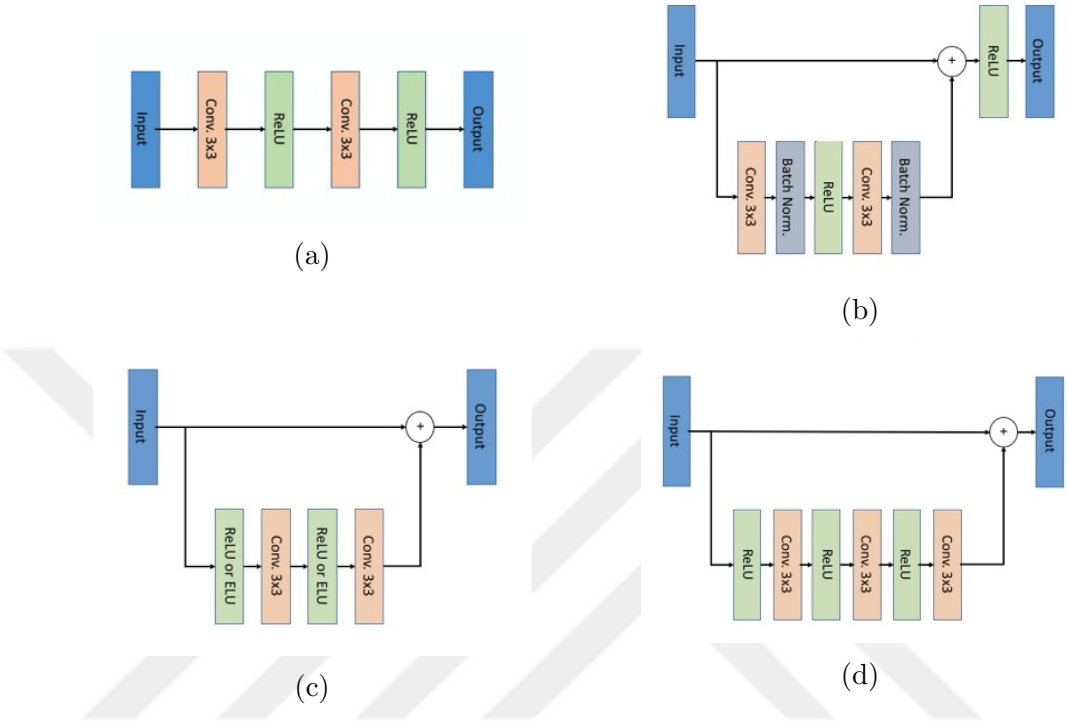


Figure 3.6: Learning blocks used in the inspired models and the proposed RTB-net model. (a) The learning block used by U-net [14]. (b) The residual learning block used by ResNet [17]. (c) The residual learning block used in the encoder stage of the proposed RTB-net. (d) The residual learning block used in the bottleneck stage of the proposed RTB-net.

In both of the convolution operations, padding and striding values are set to one to preserve the input resolution. To improve the gradient flow, residual learning connections (short skip connections) are added to the RLBs. For that, the input of each RLB is summed (in element-wise) to the output of its last convolution [17] and this sum will be the final output of the corresponding RLB. The RLB design used in the encoder stage is illustrated in Figure 3.6(c). Afterwards, the output of an RLB is given to the max-pooling layer that uses a 2×2 filter to reduce an image size for the deeper layers of the encoder stage.

3.2.2 Bottleneck Stage

The RLB of the bottleneck stage differs from those of the encoder stage. This block has one more convolution layer. Likewise, the input for this additional convolution operation is fed to the ReLU activation function beforehand. The main reason behind this is to strengthen the outputs of the decoder layers for better reconstruction. The RLB design for the bottleneck stage is illustrated in Figure 3.6(d). Upon completion of the bottleneck stage, the output of its last convolution is summed with the input of this stage (via short ship connection) and passed to the deepest layer of the decoder stage.

3.2.3 Decoder Stage

Each decoder learning block performs the following three operations: 1) The transposed convolution (t-convolution) operation is applied to feature maps of the previous layer using 2×2 filters. 2) The feature maps and their matching feature maps in the encoder stage are concatenated via long skip connections. 3) The convolution operation with a 3×3 filter is applied on the concatenated maps. Here, the ELU activation unit is used on the maps before the convolution.

To obtain a better final segmentation at the end of the network, segmentation maps from different layers of the decoder stage are combined via element-wise summation. The between-layers summations used in our network are illustrated in Figure 3.4. This approach was introduced in the design of Long *et al.* [7] and adapted by many recent works [40, 42, 45]. Segmentation maps produced at the previous layers of the decoder stage are upsampled to make element-wise summation possible among different layers.

Furthermore, to mitigate the overfitting problem, which causes the weights of the network to be very tuned to the training samples, the dropout regularization is added in the third and fifth layers of the decoder stage. The dropout factor is selected as 0.5 [51].

At the end, the output maps of the last decoder are given to the sigmoid function to obtain the posterior of a pixel belonging to the tumor bud class.

3.2.4 Residual Learning Blocks

The importance of short and long skip connections for medical image segmentation problems have been experimentally shown with the U-net and deep residual networks [43]. Our proposed RTB-net model also uses both long and short skip connections. The use of long skip connections are generally the same among different models [45, 15, 18]; this use is based on the skip connection use of U-net [14]. It helps forward computed feature maps in the encoding stage to the decoding stage. On the other hand, short skip connections, which are also called residual connections, are used in different ways among different models [45, 38]. A use is selected according to the difficulty of the features to be extracted for the given problem and the depth of the network. Hence, the performance of models may change with each designed residual block structure (which uses different residual connections).

The residual connections were initially introduced in [17]. It opens a new channel for an information flow in the network. It gives the capability to subsequent layers of the network for accessing the feature maps of their previous layers in short intervals. These residual connections are not taken into consideration in the U-net model. In our proposed RTB-net model, the information flow comes from two parts: 1) The unvarying input of the residual learning block and 2) the output generated after applying convolutional and non-linearity layers on the input. These two information flows are concatenated at the end of the block with an element-wise sum operation.

Figure 3.6 demonstrates the learning blocks used in the inspired models and the proposed model. The basic learning block, which does not use residual connections, is given in Figure 3.6(a). The one with a residual connection is given in Figure 3.6(b). This is the block used by [17], which introduced the use of residual connections. This block is slightly different than ours in the sense that

it applies a non-linear transformation on the sum to obtain the output and then max-pools this obtained output. Our model applies a nonlinear transformation on the max-pooled input before feeding it to a convolution layer. Moreover, it defines a residual connection from an unvarying input (on which such a non-linear transformation is not applied) to the output of the last convolution layer of the corresponding RLB. Our experiments show that this slightly different version is more effective for our tumor bud detection application. The RLBs used in the encoder and bottleneck stages are given in Figure 3.6(c) and 3.6(d), respectively. The learning blocks in the decoder stage do not use residual connections since each block uses only a single convolution and since it has long skip connections.

3.3 Network Training

One of the crucial issues in medical image segmentation is the huge class-imbalance problem. Dealing with this problem, the loss function used in training can have a great impact. Particularly, a loss function based on overlap measures seem more robust against this problem [52].

The Dice loss (DL) is one the most effective and popular loss functions widely used in medical image segmentation tasks [15]. This loss is based on an overlap measure and quantifies how similar two maps (ground truth and predicted maps) are. It uses the Dice similarity coefficient (DSC) that calculates overlaps between the foreground pixels (in our application, tumor bud pixels) in the ground truth and predicted maps. The definitions of these two measures for a single image tile I are given in the following equations

$$\mathrm{DSC}(I) = \frac{2\sum_n p_n g_n + \epsilon}{\sum_n p_n + \sum_n g_n + \epsilon} \tag{3.3}$$

$$\mathrm{DL}(I) = 1 - \mathrm{DSC}(I) \tag{3.4}$$

where g_n is the class label of a particular pixel n (1 for the tumor bud class and 0 for the background class) and p_n is the predicted probability of this pixel

belonging to the tumor bud class. Thus, $p_n g_n$ is greater than 0 only for the true positive pixels. Here, the ϵ term is used to avoid the division-by-zero error.

Although the Dice loss is a commonly used function for image segmentation tasks with the high class-imbalance problem, it has an important deficiency of giving the same importance to false negatives and false positives. This becomes a more important issue when the class-imbalance becomes larger (when an image contains only a few foreground pixels) [53]. This may result in detecting almost no foreground pixels. We have encountered this problem also in our tumor bud detection application since the image tiles we use contain only a few tumor bud pixels (Section 4.1). In such cases, giving more importance (weight) to minimizing false negatives may alleviate this problem. With this motivation, we use the Tversky loss (TL) in the training of our RTB-net model. This loss is based on the Tversky similarity index (TSI) [54] and proposed to be used for training of deep neural networks by [55]. The definitions of these two measures for an image tile I are given as follows

$$\mathrm{TSI}(I) = \frac{\sum_n p_n g_n + \epsilon}{\sum_n p_n g_n + \alpha \sum_n (1 - p_n) g_n + \beta \sum_n p_n (1 - g_n) + \epsilon} \tag{3.5}$$

$$\mathrm{TL}(I) = 1 - \mathrm{TSI}(I) \tag{3.6}$$

where α and β are the coefficients determining penalties given to false negatives and false positives, respectively. The sum of these two coefficients should be 1. Note that when $\alpha = \beta = 0.5$, the Tversky similarity index and Dice similarity coefficient are the same. When $\alpha > 0.5$, the loss function gives more importance to minimizing false negatives. Considering this, in our experiments, we select $\alpha = 0.6$ and $\beta = 0.4$.

The settings used in our training process are listed in Table 3.2. The loss calculated on validation images is used for early stopping. Additionally, the Adam optimizer [56] is used to adaptively select the learning rates; its parameters are also listed in Table 3.2. The weights of the networks are initialized by the approach of He *et al.* [57] through the training process.

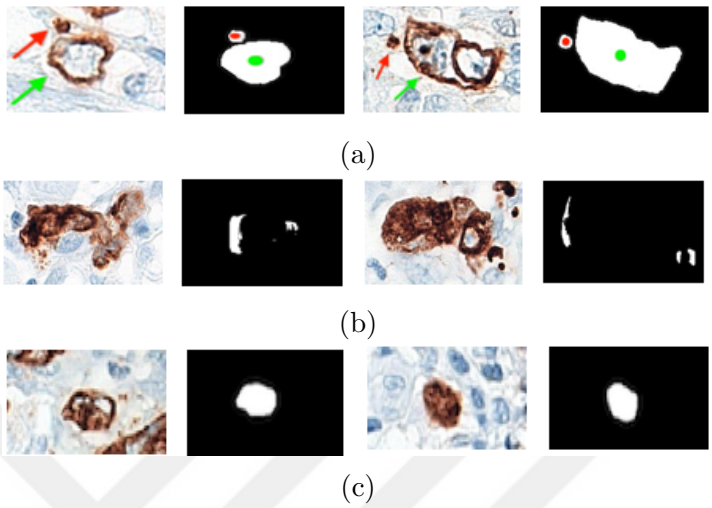


Figure 3.7: Small area elimination is effective for removing (a)-(b) noisy components around actual tumor buds as well as (c) tiny brown regions.

3.4 Postprocessing

After obtaining the class label map from the RTB-net model, we identify the large enough connected components of the tumor bud pixels as tumor bud instances. For that, the connected components smaller than the area threshold of 1000 pixels are eliminated. This area threshold is selected considering the smallest tumor bud in the training set. This small area elimination helps remove noisy components around actual tumor buds as well as eliminate tiny brown regions that are not annotated as tumor buds (Figure 3.7).

Table 3.1: Layers and operations of the proposed RTB-net model.

	Activation unit	Input (resolution, feature no)	Output (resolution, feature no)	Filter size	Receptive field size
Encoder Stage					
Convolution		$2048^2, 3$	$2048^2, 16$	3×3	3×3
Convolution	ELU	$2048^2, 16$	$2048^2, 16$	3×3	5×5
Max-pooling		$2048^2, 16$	$1024^2, 16$	2×2	6×6
Convolution		$1024^2, 16$	$1024^2, 32$	3×3	10×10
Convolution	ELU	$1024^2, 32$	$1024^2, 32$	3×3	14×14
Max-pooling		$1024^2, 32$	$512^2, 32$	2×2	16×16
Convolution		$512^2, 32$	$512^2, 64$	3×3	24×24
Convolution	ELU	$512^2, 64$	$512^2, 64$	3×3	32×32
Max-pooling		$512^2, 64$	$256^2, 64$	2×2	36×36
Convolution		$256^2, 64$	$256^2, 128$	3×3	52×52
Convolution	ReLU	$256^2, 128$	$256^2, 128$	3×3	68×68
Max-pooling		$256^2, 128$	$128^2, 128$	2×2	76×76
Convolution		$128^2, 128$	$128^2, 256$	3×3	108×108
Convolution	ReLU	$128^2, 256$	$128^2, 256$	3×3	140×140
Max-pooling		$128^2, 256$	$64^2, 256$	2×2	156×156
Bottleneck Stage					
Convolution		$64^2, 256$	$64^2, 512$	3×3	220×220
Convolution	ReLU	$64^2, 512$	$64^2, 512$	3×3	284×284
Convolution		$64^2, 512$	$64^2, 512$	3×3	348×348
Decoder Stage					
T-convolution		$64^2, 512$	$128^2, 512$	2×2	
Concatenation	ELU	$128^2, 512$	$128^2, 768$		
Convolution		$128^2, 768$	$128^2, 256$	3×3	
T-convolution		$128^2, 256$	$256^2, 256$	2×2	
Concatenation	ELU	$256^2, 256$	$256^2, 384$		
Convolution		$256^2, 384$	$256^2, 128$	3×3	
T-convolution		$256^2, 128$	$512^2, 128$	2×2	
Concatenation	ELU	$512^2, 128$	$512^2, 192$		
Convolution		$512^2, 192$	$512^2, 64$	3×3	
T-convolution		$512^2, 64$	$1024^2, 64$	2×2	
Concatenation	ELU	$1024^2, 64$	$1024^2, 96$		
Convolution		$1024^2, 96$	$1024^2, 32$	3×3	
T-convolution		$1024^2, 32$	$2048^2, 32$	2×2	
Concatenation	ELU	$2048^2, 32$	$2048^2, 48$		
Convolution		$2048^2, 48$	$2048^2, 16$	3×3	

Table 3.2: Settings used in the training of the proposed RTB-net model.

Parameter	Value
Input image resolution	2048×2048
Number of epochs	100
Number of epochs for early stopping	40
Dropout factor (decoder stage)	0.5
Batch size	1
Learning rate (Adam optimizer)	$5{\times}10^5$
β_1 (Adam optimizer)	0.9
β_2 (Adam optimizer)	0.999

Chapter 4

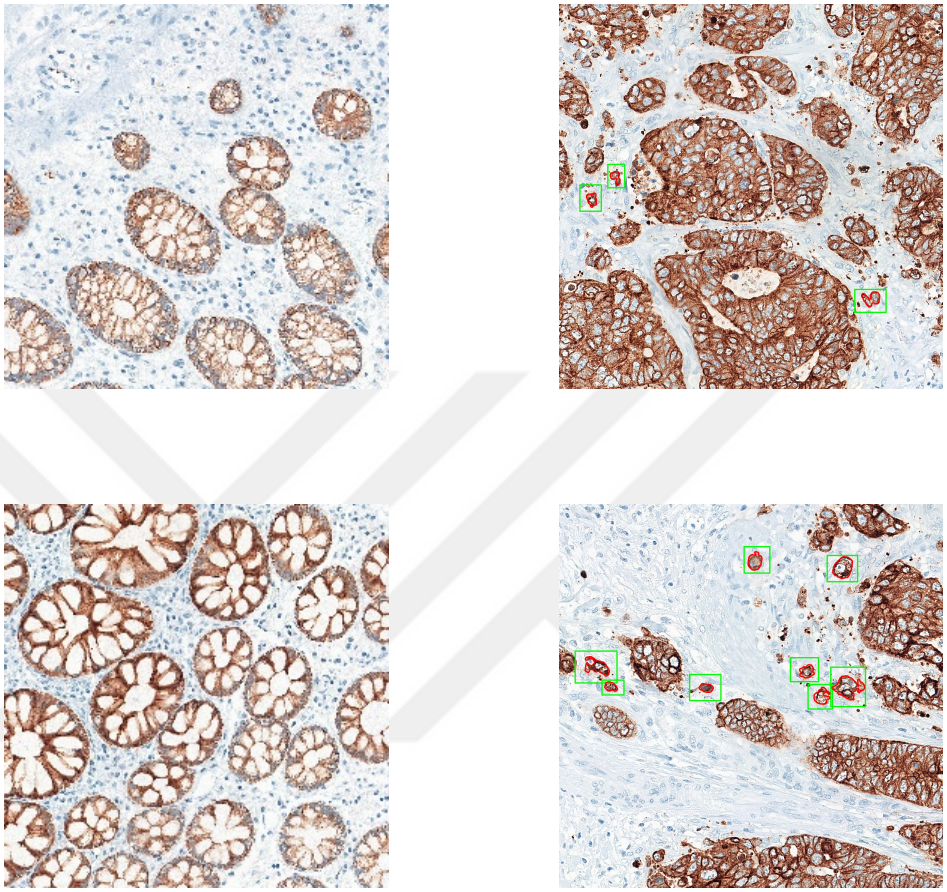
Experiments

This chapter gives the details of the experiments conducted throughout this thesis. First, it explains the dataset used in the experiments in detail. After that, it gives the configuration of the setup of the computer system used in the experiments, discusses the evaluation metrics, and lists the selected values of model parameters. Finally, it presents quantitative and visual results together with the comparison of U-net [14] and ResUnet [18].

4.1 Dataset

Experiments are conducted on 23 digital whole slide images (WSIs) of colorectal carcinomatous samples provided by University Hospital Erlangen. The IHC stained colorectal carcinomatous samples were scanned at $40\times$ magnification in the average size of $100,000 \times 200,000$ pixels captured by a 4M resolution high-speed camera. Digital stained images include both carcinomatous and normal tissue regions. Each whole slide image was divided into 2048×2048 pixel image tiles to work more efficiently with the proposed network architecture.

Each tumor bud in these images was manually annotated by providing a bounding box containing this tumor bud. These manual bounding box annotations were



(a) (b)

Figure 4.1: Examples of normal and carcinomatous tissue samples taken from the dataset. (a) Normal tissue samples. (b) Carcinomatous tissue samples including tumor buds.

provided by the pathologists of University Hospital Erlangen. Figure 4.1 shows examples of normal and carcinomatous tissue samples taken from the dataset. In the given carcinomatous samples, tumor bud annotations of pathologists are also drawn. However, throughout the training, only image tiles with at least one tumor bud are used as training instances to increase the stability of the network.

The dataset is divided into training, validation, and test sets. All images in these datasets have 2048×2048 pixel resolution and they are not resized. The training set has 2416 images taken from 14 whole slide images (WSI) of colorectal carcinoma samples. These images correspond to intratumoral or peritumoral regions within WSIs and include a total of 9892 annotated tumor buds. The

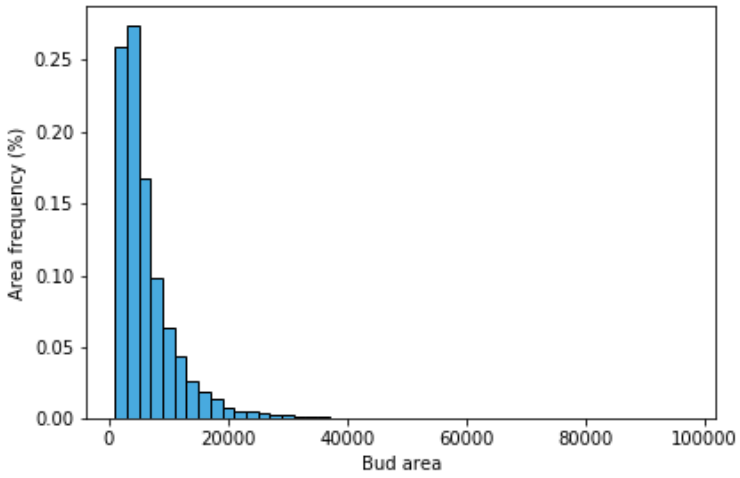


Figure 4.2: Area distribution of tumor buds in the training set is given. Each histogram bin corresponds to $2,000$ pixels. In the training set, the minimum and the maximum tumor buds are nearly $1,000$ and 98000 pixels

validation set has 257 images taken from two WSIs and includes a total of 801 tumor buds. It is used for early stopping and for searching the network's hyper-parameters during the training. The test set comprises 622 images from seven WSIs. They are also taken from intratumoral and peritumoral regions and they have a total of 1888 annotated tumor buds. The details of these datasets are provided in Table 4.1. The details of the area distribution of tumor buds in the training set is given in Figure 4.2; this distribution is used to select the area threshold parameter used in the postprocessing step.

Note that these datasets also include tumor bud annotations on which the pathologists are not certain. These "unsure" tumor buds are not used in the training of RTB-net and also not included in the numbers given in Table 4.1. However, the proposed network is tested on certain tumor bud annotations (Table 4.4) as well as all of them including the unsure ones (Table 4.5). Figure 4.3 gives the distribution of certain and unsure tumor bud annotations in the datasets. The training, validation, and test datasets have 14.91, 22.22, and 27.06 percents unsure tumor bud annotations, respectively.

The existence of imbalance distribution among classes is a very common and

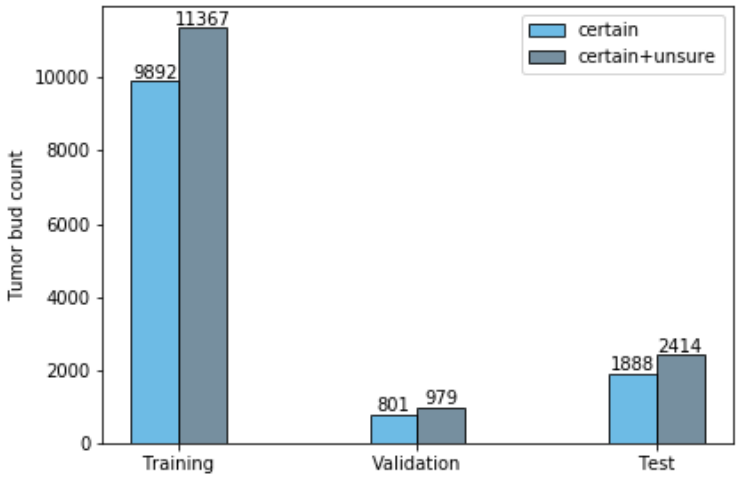


Figure 4.3: Tumor bud counts of the datasets with and without including the unsure tumor bud annotations.

critical problem in medical image segmentation tasks. Our datasets also suffer from this problem. As seen in Table 4.2, our datasets include a very huge number of background pixels (∼99.5 percent on the average), but only a few number of tumor bud pixels (∼0.5 percent on the average).

Table 4.1: Datasets used in our experiments.

Dataset	WSI count	Image count	Tumor bud count	Avg tumor bud per image
Train	14	2416	9892	4.10
Validation	2	257	801	3.12
Test	7	622	1888	3.04
Total	23	3295	12581	3.82

4.2 Experimental Setup

For the implementation, the Python programming language is used with the Keras framework [58]. Keras is an open source neural network library at the top of the Tensorflow framework [59]. It is chosen since it is very user-friendly and modular. Additionally, experiments are done on a machine that has a 62 GB

Table 4.2: Class percentages in the datasets used in our experiments.

		Background pixels			Tumor bud pixels		
Dataset	Image count	Mean	Min	Max	Mean	Min	Max
Train	2416	99.52	93.80	99.99	0.48	0.01	6.20
Validation	257	99.60	95.96	99.96	0.40	0.04	4.04
Test	622	99.58	97.02	99.96	0.42	0.04	2.98

RAM and NVIDIA GeForce RTX-2080 Ti GPU with a memory size of 11 GB and a memory speed of 14 Gbps.

4.3 Evaluation Metrics

For obtaining the quantitative results, we use three metrics: precision, recall, and F-score.

In order to calculate these metrics, true positive (TP), false positive (FP), and false negative (FN) tumor buds are identified. The definitions of these metrics are given below.

Let P be a set of predicted tumor buds and T be a set of tumor buds annotations.

True Positive (TP): is the number of objects $p \in P$ that have at least 10 % overlap with an object $t \in T$.

False Positive (FP): is the number of objects $p \in P$ that do not have at least 10 % overlap with an object $t \in T$.

False Negative (FN): is the number of objects $t \in T$ that do not have at least 10 % overlap with an object $p \in P$.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.2}$$

$$\text{F1 Score} = \frac{2 \cdot \text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}} \tag{4.3}$$

4.4 Parameter Selection

In the experiments, different combinations of selectable parameters were tested to obtain the most efficient result. The best result is achieved with the following parameters:

- For brown intensity and area elimination thresholds of the mask extraction from bounding box annotated data are set to 0.85 and 1500 pixels, respectively.
- For α and β values of the Tversky loss function, 0.6 and 0.4 are used, respectively.
- For an overlapping (hit) calculation, the overlapping ratio is set to 10%.
- For an elimination in the postprocessing step, minimum tumor bud area in the training dataset is set same as the area threshold.

4.5 Experiments and Results

This section covers the experiments in detail with quantitative and visual results of the proposed convolutional model which works on segmentation and detection of CRC tumor buds in a fully automatic way.

There are two additional architectures that are tested in the experiments for comparison with the proposed model. To the best of our knowledge, there is

no specific model for automatic tumor bud detection in a fully convolutional way. Therefore, recent models that have high performance in medical image segmentation tasks are selected to better observe the performance gain of the proposed model: U-net and ResUnet.

U-net is used as the first model to make evaluations and comparisons with the proposed model. As mentioned before, it has a great performance in medical image segmentation, which contains low-level details and maintains high-level information at the same time. In contrast to the primary version of the U-net [14], feature concatenation operations and forward convolutions are added instead of the crop and copy methods. These modifications enhance the performance of the model. Figure 4.4 is the used U-net model throughout the experiments and loss function and the optimizer algorithm of the model are the same as the proposed model.

ResUnet is currently one of the most efficient deep network approaches. It combines the advantages of both residual neural network [17] and U-net [14] which are respectively good at the smooth gradient flow of residual approaches to overcome the degradation problem due to the complexity that comes from the increasing depth of the networks and the effectiveness of long skip connections to feed deeper layers with feature maps of initial layers. ResUnet architecture that is used in the evaluation is like the architecture in [18] with the same loss function and optimizer algorithm with the proposed model. The architecture of the model is pictured in Figure 4.5.

The experiments are designed to study the following:

- Effect of image magnification
- Effect of using residual learning
- Comparison with two state-of-the-art models

First of all, the performance of the proposed model is evaluated with downsampled input images instead of original size image tiles to show how downsampling

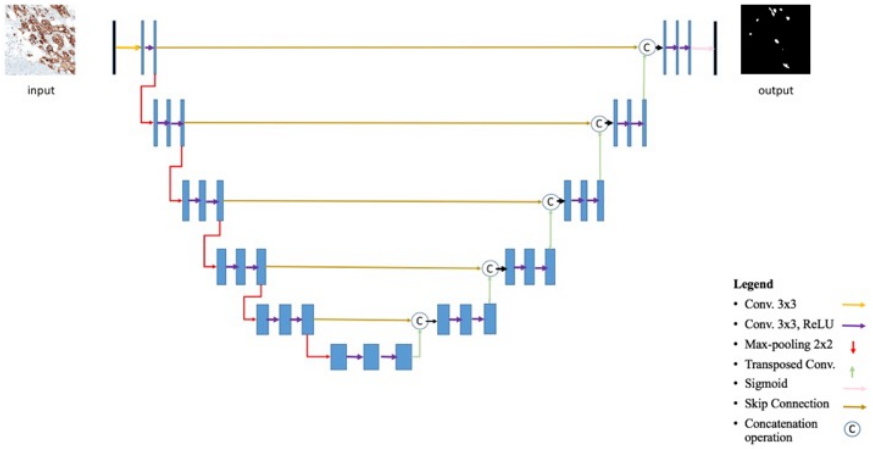


Figure 4.4: U-net model used in the experiments.

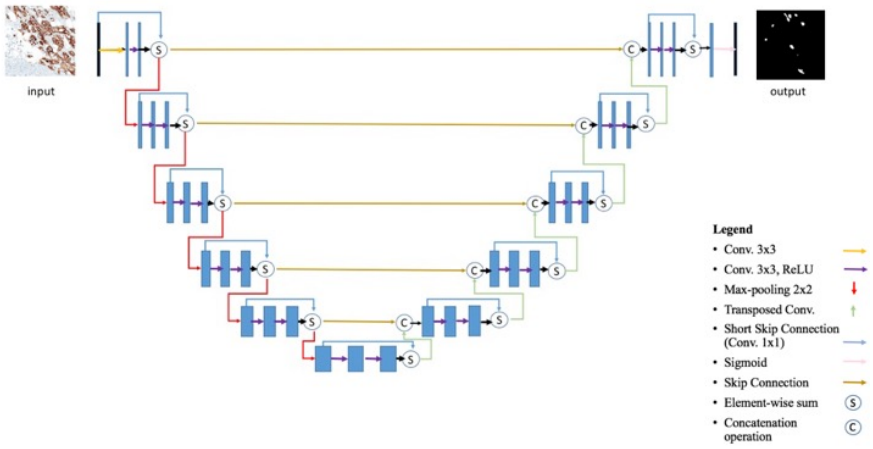


Figure 4.5: ResUnet model used in the experiments.

of the input affects training. Table 4.3 provides the quantitative results of the proposed model when it is trained with different input image sizes: four-to-one (to 512×512) and two-to-one (to 1024×1024) downsampled input images.

The result shows that downsampling operation has a negative effect on the detection of CRC tumor buds. The result of the original resolution case has around 3% and 1% higher F1 score than the downsampled cases as a result of the predictions with the test set, respectively. Although recall values of the downsampled cases are considerably higher than the original one, the same is not valid for the precision values which is significantly decreased in comparison to the percentage increase in the recall values. The main reason behind this is the failure to learn significant internal and external morphological features of the CRC tumor bud with the reduced resolution of the input images which causes a high number of negative predictions. The indicator of this is the ratio of the increase in recall values to the decrease in precision values. The test cases with downsampled images have approximately 2.3% and 1% increase in recall values and 7% and 3% decrease in precision values respectively. In addition to the quantitative results, the comparative visual results of the downsampled cases are also given in Figure 4.6.

Table 4.3: Evaluation results of the proposed model with different downsampled input images.

Method	Resolution	Recall	Precision	F1-score
RTB-net	512×512	**83.42**	67.59	74.68
RTB-net	1024×1024	82.21	70.62	75.97
RTB-net	2048×2048	81.57	**72.77**	**76.92**

In order to explain the performance gain from the proposed model, a set of experiments are followed. As an outcome of the experiments, the quantitative results of the proposed model and other state-of-the-art models are given in Table 4.4. The table has several performance metrics of the models like recall, precision, and F1 score to evaluate their predictions with respect to the test set. The given quantitative results in the table are the median results of nearly five performed test runs of each model.

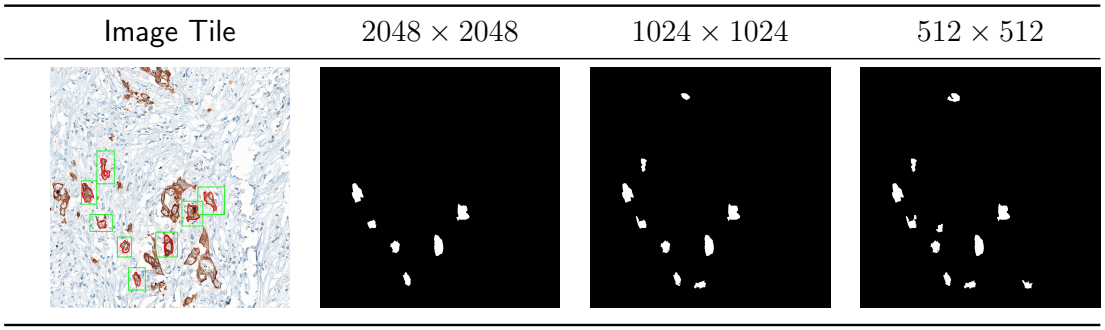


Figure 4.6: Visual comparison of predictions of the downsampled cases.

From Table 4.4, it is clear to see that the performance of RTB-net in terms of all quantitative parameters is better than that of U-net and ResUnet. It has respectively 4.4% and 1.7% better F1 Score than other models which is 76.92. In addition to the quantitative results, the comparative visual results of the models are also given in Figure 4.7.

Table 4.4: Evaluation results of the proposed network and the state-of-the-art networks.

Method	Year	Recall	Precision	F1-score
U-net [14]	2016	77.37	70.30	73.67
ResUnet [18]	2018	80.10	71.64	75.63
RTB-net	2019	**81.57**	**72.77**	**76.92**

The quantitative results in Table 4.4 are calculated as a result of the test of the models with the test set, which does not include the unsure annotated tumor buds. In addition to that analysis, the same experiment is repeated with the test set that has also unsure annotated tumor buds. There are nearly 30% more annotated tumor buds in addition to the precisely annotated ones, as the distribution emphasized in Section 4.1. The experiment informs about how false positive predictions of the models relate to being tumor bud. Table 4.5 presents the quantitative results of the models with the addition of the unsure labeled tumor buds to the test set during test. From the table, it is clearly understood that an important part of the false positive predictions is eliminated with counting them as a tumor bud after the addition of the unsure labeled tumor buds to the set. When compared with Table 4.4, recall values are decreased due to the false

negatives. On the contrary, precision values are significantly increased for each model by counting some of the false positives as a tumor bud with the unsure annotations. RTB-net has the best performance among the models in terms of F1 score with 77.65 when counting the unsure annotations.

Table 4.5: Evaluation results of the proposed network and the state-of-the-art networks with counting unsure labeled tumor buds.

Method	Year	Recall	Precision	F1-score
U-net [14]	2016	72.00	77.47	74.63
ResUnet [18]	2018	75.06	78.63	76.80
RTB-net	2019	**75.55**	**79.81**	**77.65**

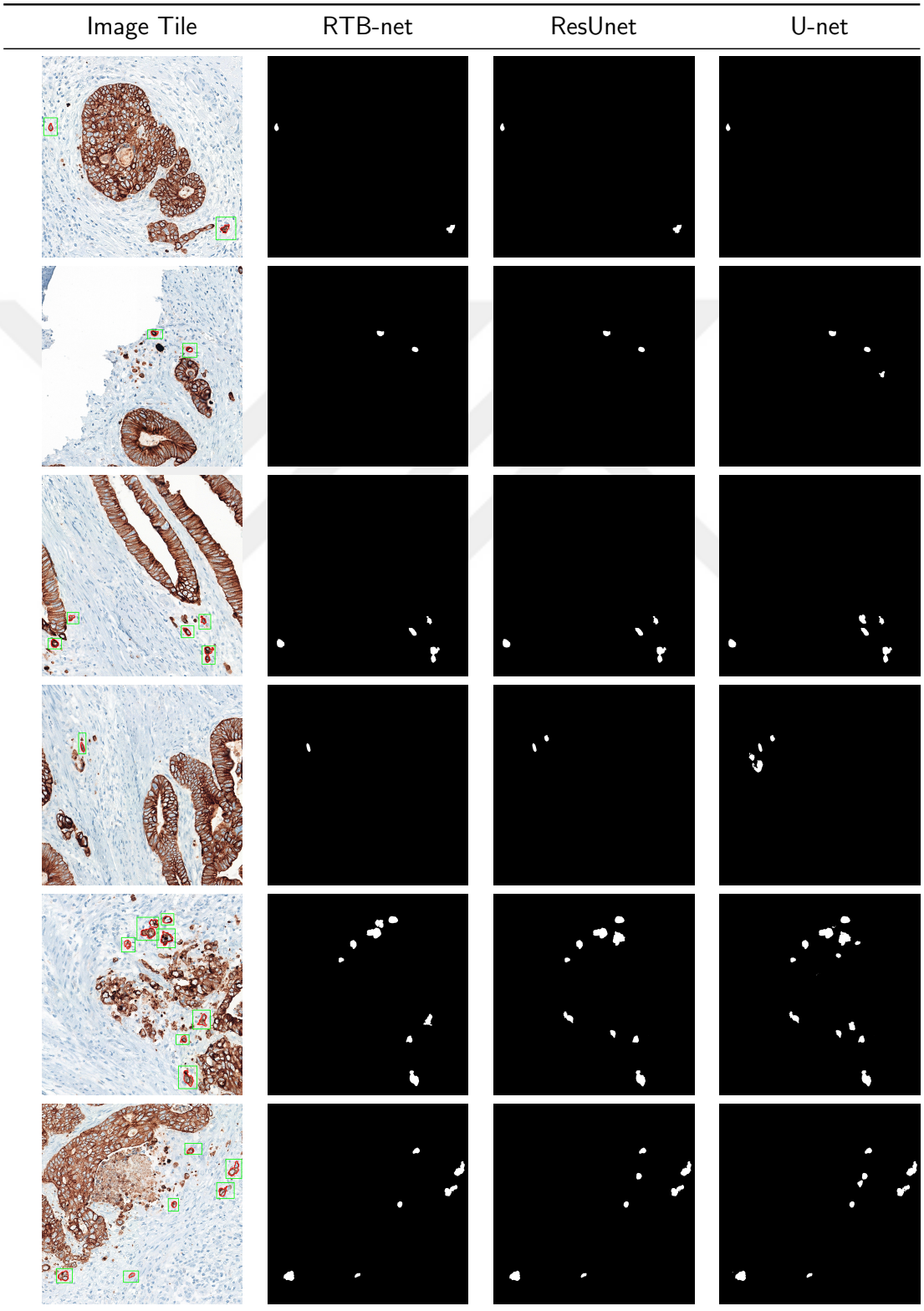


Figure 4.7: Visual comparison of predictions of the evaluated models.

Chapter 5

Conclusion

This thesis presents a fully convolutional network design for the purpose of tumor bud detection. The design relies on the U-net architecture but extends it by also considering up-to-date learning mechanism. These mechanisms include 1) using residual connections 2) jointly using the ELU and ReLU activation units 3) using a Tversky loss function in training, and 4) combining segmentation maps from different layers of the decoder path. All these mechanisms are used to help alleviate the vanishing gradient problem and the problems related with having high class-imbalance dataset. Our experiments on 3295 image tiles taken from 23 whole slide images show that extending the architecture with these mechanism improves the tumor bud detection results.

5.1 Future Work

The performance of the proposed FCN can further be increased with the following improvements. Investigating these improvements is considered as future work.

- Multiple classes can be defined for tumor buds according to their sizes and the problem can be considered as multi-class segmentation.

- The network can be trained to output more false positives. In this way, the number of false negatives can be reduced. After that, false positives can be eliminated by training another network (for example a CNN working on the patches, each of which is cropped for containing a single false negative bud).

- Postprocessing can be performed to eliminate false positives. This elimination can use handcrafted features from each bud candidate. These features may include the distance from a detected bud to the tumor mass boundary, the tumor bud area, and the tumor bud shape.

This proposed network works on finding tumor buds on a given image tile. It can be extended to find all tumor buds in a whole slide image. For that, a sliding window approach can be used. This extension is considered as another future work of this thesis.

Bibliography

[1] E. H. Ackerknecht, "Rudolf Virchow: Doctor, Statesman, Anthropologist," *Rudolf Virchow: Doctor, Statesman, Anthropologist*, 1953.

[2] S. S. Raab, D. M. Grzybicki, J. E. Janosky, R. J. Zarbo, F. A. Meier, C. Jensen, and S. J. Geyer, "Clinical Impact and Frequency of Anatomic Pathology Errors in Cancer Diagnoses," *Cancer*, vol. 104, no. 10, pp. 2205–2213, 2005.

[3] F. Grizzi, G. Celesti, G. Basso, and L. Laghi, "Tumor Budding as a Potential Histopathological Biomarker in Colorectal Cancer: Hype or Hope," *World Journal of Gastroenterology: WJG*, vol. 18, no. 45, p. 6532, 2012.

[4] A. Lugli, R. Kirsch, Y. Ajioka, F. Bosman, G. Cathomas, H. Dawson, H. El Zimaity, J.-F. Fléjou, T. P. Hansen, A. Hartmann, S. Kakar, C. Langner, I. Nagtegaal, G. Puppa, R. Riddell, A. Ristimäki, K. Sheahan, T. Smyrk, K. Sugihara, B. Terris, H. Ueno, M. Vieth, I. Zlobec, and P. Quirke, "Recommendations for Reporting Tumor Budding in Colorectal Cancer Based on the International Tumor Budding Consensus Conference (ITBCC)," *Modern Pathology*, vol. 30, p. 1299, 2017.

[5] A. Rogers, D. Winter, A. Heeney, D. Gibbons, A. Lugli, G. Puppa, and K. Sheahan, "Systematic Review and Meta-Analysis of the Impact of Tumour Budding in Colorectal Cancer," *British Journal of Cancer*, vol. 115, no. 7, p. 831, 2016.

[6] B. J. Czerniecki, A. M. Scheff, L. S. Callans, F. R. Spitz, I. Bedrosian, E. F. Conant, S. G. Orel, J. Berlin, C. Helsabeck, and D. L. Fraker, "Immunohistochemistry with Pancytokeratins Improves the Sensitivity of Sentinel Lymph Node Biopsy in Patients with Breast Carcinoma," *Cancer*, vol. 85, no. 5, pp. 1098–1103, 1999.

[7] J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, 2015.

[8] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, 2014.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.

[10] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, R. Kim, R. Raman, P. C. Nelson, J. L. Mega, and W. D. R., "Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs," *JAMA*, vol. 316, no. 22, pp. 2402–2410, 2016.

[11] B. E. Bejnordi, J. Lin, B. Glass, M. Mullooly, G. L. Gierach, M. E. Sherman, N. Karssemeijer, J. Van Der Laak, and A. H. Beck, "Deep Learning-Based Assessment of Tumor-Associated Stroma for Diagnosing Breast Cancer in Histopathology Images," in *International Symposium on Biomedical Imaging*, pp. 929–932, 2017.

[12] D. Wang, A. Khosla, R. Gargeya, H. Irshad, and A. H. Beck, "Deep Learning for Identifying Metastatic Breast Cancer," *arXiv preprint arXiv:1606.05718*, 2016.

[13] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[14] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional Networks for Biomedical Image Segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, Springer, 2015.

[15] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation," in *International Conference on 3D Vision*, pp. 565–571, IEEE, 2016.

[16] J.-M. Bokhorst, L. Rijstenberg, D. Goudkade, I. Nagtegaal, J. van der Laak, and F. Ciompi, "Automatic Detection of Tumor Budding in Colorectal Carcinoma with Deep Learning," in *Computational Pathology and Ophthalmic Medical Image Analysis*, pp. 130–138, Springer, 2018.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[18] Z. Zhang, Q. Liu, and Y. Wang, "Road Extraction by Deep Residual U-net," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 5, pp. 749–753, 2018.

[19] A. Lugli, T. Vlajnic, O. Giger, E. Karamitopoulou, E. S. Patsouris, G. Peros, L. M. Terracciano, and I. Zlobec, "Intratumoral Budding as a Potential Parameter of Tumor Progression in Mismatch Repair–Proficient and Mismatch Repair–Deficient Colorectal Cancer Patients," *Human Pathology*, vol. 42, no. 12, pp. 1833–1840, 2011.

[20] L. De Smedt, S. Palmans, and X. Sagaert, "Tumour Budding in Colorectal Cancer: What Do We Know and What Can We Do," *Virchows Archiv*, vol. 468, no. 4, pp. 397–408, 2016.

[21] B. Mitrovic, D. F. Schaeffer, R. H. Riddell, and R. Kirsch, "Tumor Budding in Colorectal Carcinoma: Time to Take Notice," *Modern Pathology*, vol. 25, no. 10, p. 1315, 2012.

[22] A. Lugli, E. Karamitopoulou, and I. Zlobec, "Tumour Budding: A Promising Parameter in Colorectal Cancer," *British Journal of Cancer*, vol. 106, no. 11, p. 1713, 2012.

[23] H. Van Wyk, J. Park, C. Roxburgh, P. Horgan, A. Foulis, and D. C. McMillan, "The Role of Tumour Budding in Predicting Survival in Patients with Primary Operable Colorectal Cancer: A Systematic Review," *Cancer Treatment Reviews*, vol. 41, no. 2, pp. 151–159, 2015.

[24] S. L. Bosch, S. Teerenstra, J. H. de Wilt, C. Cunningham, and I. D. Nagtegaal, "Predicting Lymph Node Metastasis in pT1 Colorectal Cancer: A Systematic Review of Risk Factors Providing Rationale for Therapy Decisions," *Endoscopy*, vol. 45, no. 10, pp. 827–841, 2013.

[25] H. Ueno, H. Mochizuki, Y. Hashiguchi, H. Shimazaki, S. Aida, K. Hase, S. Matsukuma, T. Kanai, H. Kurihara, and K. Ozawa, "Risk Factors for an Adverse Outcome in Early Invasive Colorectal Carcinoma," *Gastroenterology*, vol. 127, no. 2, pp. 385–394, 2004.

[26] V. H. Koelzer, I. Zlobec, and A. Lugli, "Tumor Budding in Colorectal Cancer—Ready for Diagnostic Practice," *Human Pathology*, vol. 47, no. 1, pp. 4–19, 2016.

[27] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[28] X. Glorot, A. Bordes, and Y. Bengio, "Deep Sparse Rectifier Neural Networks," in *International Conference on Artificial Intelligence and Statistics*, pp. 315–323, 2011.

[29] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)," *arXiv preprint arXiv:1511.07289*, 2015.

[30] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting,"

The Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929–1958, 2014.

[31] N. Ramesh, J.-H. Yoo, and I. Sethi, "Thresholding Based on Histogram Approximation," *IEE Vision, Image and Signal Processing*, vol. 142, no. 5, pp. 271–279, 1995.

[32] N. Sharma and A. K. Ray, "Computer Aided Segmentation of Medical Images Based on Hybridized Approach of Edge and Region Based Techniques," in *International Conference on Mathematical Biology*, pp. 150–5, 2006.

[33] Y. Y. Boykov and M.-P. Jolly, "Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in ND Images," in *IEEE International Conference on Computer Vision*, vol. 1, pp. 105–112, IEEE, 2001.

[34] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images," in *Advances in Neural Information Processing Systems*, pp. 2843–2851, 2012.

[35] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional Random Fields as Recurrent Neural Networks," in *IEEE International Conference on Computer Vision*, pp. 1529–1537, 2015.

[36] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs," *arXiv preprint arXiv:1412.7062*, 2014.

[37] V. Badrinarayanan, A. Handa, and R. Cipolla, "Segnet: A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling," *arXiv preprint arXiv:1505.07293*, 2015.

[38] M. Z. Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, "Recurrent Residual Convolutional Neural Network Based on U-net (R2U-net) for Medical Image Segmentation," *arXiv preprint arXiv:1802.06955*, 2018.

[39] K. He, X. Zhang, S. Ren, and J. Sun, "Identity Mappings in Deep Residual Networks," in *European Conference on Computer Vision*, pp. 630–645, Springer, 2016.

[40] H. Chen, Q. Dou, L. Yu, J. Qin, and P.-A. Heng, "VoxResNet: Deep Voxelwise Residual Networks for Brain Segmentation from 3D MR Images," *NeuroImage*, vol. 170, pp. 446–455, 2018.

[41] H. Chen, X. Qi, L. Yu, and P.-A. Heng, "DCAN: Deep Contour-Aware Networks for Accurate Gland Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2487–2496, 2016.

[42] Q. Dou, H. Chen, Y. Jin, L. Yu, J. Qin, and P.-A. Heng, "3D Deeply Supervised Network for Automatic Liver Segmentation from CT Volumes," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 149–157, Springer, 2016.

[43] M. Drozdzal, E. Vorontsov, G. Chartrand, S. Kadoury, and C. Pal, "The Importance of Skip Connections in Biomedical Image Segmentation," in *Deep Learning and Data Labeling for Medical Applications*, pp. 179–187, Springer, 2016.

[44] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-net: Learning Dense Volumetric Segmentation from Sparse Annotation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 424–432, Springer, 2016.

[45] B. Kayalibay, G. Jensen, and P. van der Smagt, "CNN-Based Segmentation of Medical Imaging Data," *arXiv preprint arXiv:1701.03056*, 2017.

[46] M. Buda, A. Maki, and M. A. Mazurowski, "A Systematic Study of the Class Imbalance Problem in Convolutional Neural Networks," *Neural Networks*, vol. 106, pp. 249–259, 2018.

[47] F. Isensee, J. Petersen, A. Klein, D. Zimmerer, P. F. Jaeger, S. Kohl, J. Wasserthal, G. Koehler, T. Norajitra, and S. Wirkert, "NNU-net: Self-Adapting Framework for U-net-Based Medical Image Segmentation," *arXiv preprint arXiv:1809.10486*, 2018.

[48] M. Z. Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, "Improved Inception-Residual Convolutional Neural Network for Object Recognition," *Neural Computing and Applications*, pp. 1–15, 2017.

[49] H. Li, R. Zhao, and X. Wang, "Highly Efficient Forward and Backward Propagation of Convolutional Neural Networks for Pixelwise Classification," *arXiv preprint arXiv:1412.4526*, 2014.

[50] W. Chen, Y. Zhang, J. He, Y. Qiao, Y. Chen, H. Shi, and X. Tang, "W-net: Bridged U-net for 2D Medical Image Segmentation," *arXiv preprint arXiv:1807.04459*, 2018.

[51] P. Baldi and P. J. Sadowski, "Understanding Dropout," in *Advances in Neural Information Processing Systems*, pp. 2814–2822, 2013.

[52] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. J. Cardoso, "Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pp. 240–248, Springer, 2017.

[53] N. Abraham and N. M. Khan, "A Novel Focal Tversky Loss Function with Improved Attention U-net for Lesion Segmentation," *arXiv preprint arXiv:1810.07842*, 2018.

[54] A. Tversky, "Features of Similarity.," *Psychological Review*, vol. 84, no. 4, p. 327, 1977.

[55] S. R. Hashemi, S. S. M. Salehi, D. Erdogmus, S. P. Prabhu, S. K. Warfield, and A. Gholipour, "Tversky as a Loss Function for Highly Unbalanced Image Segmentation Using 3D Fully Convolutional Deep Networks," *arXiv preprint arXiv:1803.11078*, 2018.

[56] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[57] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep Into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," in *IEEE International Conference on Computer Vision*, pp. 1026–1034, 2015.

[58] F. Chollet, "Keras." https://keras.io, 2015.

[59] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, and M. Isard, "Tensorflow: A System for Large-Scale Machine Learning," in *12th Symposium on Operating Systems Design and Implementation*, pp. 265–283, 2016.