A pilot exploratory study of the potentials of deep learning methods in cancer image segmentation and classification

Eric Yi
Richard Montgomery High School
Yanling Liu (liuy5@mail.nih.gov)
NCI at Frederick https://orcid.org/0000-0002-5173-2426

Research article

Keywords: Deep learning, convolutional neural network, digital pathology, tumor image segmentation, stroma, Tumor-Stroma Ratio, transfer learning, tumor image classification

Posted Date: February 8th, 2020

DOI: https://doi.org/10.21203/rs.2.22897/v1

Abstract

Background Tumor classification and feature quantification from H&E histology images are critical tasks for cancer diagnosis, cancer research, and treatment. However, both tasks involve tedious and time-consuming manual examination of histology images. We explored the usage of deep learning methods in segmentation and classification of histology images of cancer tissue for their potential in computer-aided tumor diagnosis and other clinical and research applications. Specifically, we evaluated performance of selected deep learning methods in stroma and glandular objects segmentation in tumor image data and tumor images classification. We automated these tasks to help facilitate downstream tumor image analysis, reduce the labor load of pathologists, and provide them with a second opinion on their analysis.

Methods We modified a patch-based U-Net model and trained it to perform stroma detection and segmentation in cancer tissue. Then the semantic segmentation capabilities of the U-Net model were compared with that of a DeepLabV3+ model. We explored the possible use of transfer learning to train a patch-based model to classify cancer tissue images as carcinoma and sarcoma and to further classify them as carcinoma subtypes.

Results In spite of the limited dataset available for the pilot study, we found that the DeepLabV3+ model performed biomedical image segmentation more effectively than U-Net when k-fold cross-validation was utilized, but U-Net still showed promise as an effective and efficient model when we used a customized validation approach. We believe that the DeepLabV3+ model can perform segmentation with even more accuracy if computation resource constraints are removed or if more data is used to augment the result. In terms of tumor classification, our selected models also consistently achieve test accuracies above 80%, with a model trained using transfer learning with VGG-16 network as the feature extractors, or convolutional base performing best. For multi-class tumor subtype classification, we also observed promising test accuracies from our models, and a customized post-processing method provided even higher prediction accuracy on test set images and this method can be further investigated.

Conclusions This pilot exploratory study provided strong evidence for the powerful potentials of deep learning models for segmentation and classification of tumor image data.

Background

Cancer is the second leading cause of death in the US [1]. Evaluation of histopathological images is an indispensable tool critical for the diagnosis and understanding of cancer and the assessment of therapies in drug discovery. With recent advances in deep learning [2], a subfield of artificial intelligence, the remarkable successes in the application of deep learning methods in tumor image segmentation [3] and tumor classification [4] provided promise to state-of-art computer-aided medical image analysis. The ability of deep learning methods to automatically and quantitatively extract image features from digital pathological images and computationally learn the multilevel representations of data for comprehensive image analysis and classification offers the opportunity for better modeling of disease and potentially

improved prediction of disease and patient outcome [5, 6, 7]. As a result, these deep learning methods may help assist and validate diagnosis results from radiologists and pathologists.

Stromal cells play an important role in the growth and development of cancer [8]. Most anticancer therapies target cancer cells specifically, but the tumor stroma can promote the resistance of cancer cells to such therapies [8]. Therefore, novel treatment strategies should combine anti-cancer and anti-stroma agents. Stroma detection in cancerous tissue would help clinicians and researchers to learn more about the progression of cancer and start early treatments targeting the tumor stroma [9]. It has been postulated that therapy directed against the supporting host tissue rather than the tumor itself will be less prone to resistance because the genetic plasticity of the cancer is not reflected in the stroma [8]. Therefore, by identifying the stroma present in cancerous tissue, scientists would be able to learn more about the progression of cancer and start early treatments targeting the tumor stroma. Interestingly, recent studies show that the tumor-stroma ratio (TSR) derived from image analysis also potentially serves as an independent prognostic factor in many types of cancer [7, 9, 10]. However, much of such work was done manually. For example, human observers were used for visual estimation of TSR in a previous report [10], which is a tedious and time-consuming task and subject to human bias. Such estimation generally involves tumor image segmentation, which is a critical task in medical image processing. However, manual segmentation is still common in clinical settings. Accurate computational segmentation of tumor images by deep learning models to distinguish the stroma and non-stroma tumor areas would largely help facilitate downstream image analysis to accurately derive TSR. This would conceivably expand the TSR-based studies in the field to explore the potentials of TSR as a promising prognosis factor in cancer treatment.

A popular deep learning method involving the convolutional neural network (CNN) called U-Net [11] is already widely used and shows excellent results in biomedical segmentation tasks in the field. Although limited in histology samples in a clinical setting, one way to increase dataset size is by extracting overlapping image patches from the existing images of regions of interest (ROIs). This decrease in size from ROIs to small patches also helps decrease the amount of memory the models will need to run. More importantly, the commonly used patch extraction strategy was shown to be able to help increase the performance of segmentation in previous research [12, 13]. Therefore, we also used the patch extraction strategy in our segmentation of tumor images to annotate the stroma and non-stroma cell areas. To gain insights on the general performance of the popular and conventional U-Net method [11] in tumor image segmentation, we also compared its performance in tumor image segmentation with the recently developed and more unconventional, complex, and computationally expensive DeepLabV3 + model [14].

Tumor types and subtypes assessment from histology images is also a key diagnosis process supporting clinicians' treatment decision and prognosis evaluation. In most cases, tumor classification from histology images is done directly by pathologists [4]. However, computer-aided tumor classification can help pathologists classify cancer tissue images, offering them a second opinion on classification as pre-screening results and helping to reduce their labor load. Deep learning methods were successfully applied to tumor classification in some recent studies [4]. Other studies have shown that a transfer learning

strategy can improve the performance of tumor classification [15]. It is also demonstrated recently that a patch-based CNN can outperform an image-based CNN in tumor classification [16, 17]. Therefore, another goal of our study was to explore and compare the potentials and abilities of various deep learning models with transfer learning and other strategies to improve their performance in tumor type and subtype classification of cancer histology images.

To summarize, in this study, we modified a U-Net model and trained it to perform semantic segmentation of stromal cells in cancer tissue. Also, the performance of the U-Net model was compared with that of a recently developed DeepLabV3 + model to discover which model would segment more accurately. We also evaluated the performances of deep learning models utilizing the transfer learning technique, and we compared the performances of selected models to classify cancer tissue images as carcinoma or sarcoma, and to further classify the images as carcinoma subtypes. Specifically, the VGG-16 [18] and InceptionResNetV2 [19] models were used as the convolutional bases in our transfer learning models.

Methods

Data collection and processing

We first manually segmented stroma on histology whole slide images (WSIs) provided by the FNLCR Molecular Characterization (MoCha) group with results reviewed by a pathologist. The WSIs were cut into patches of 1000 x 1000 pixels to make manual segmentation more manageable. Customized Python scripts were then created, forming a pipeline to facilitate data processing. Briefly, under the guidance of a pathologist, the patches of image data were segmented with GIMP [20], and then the segmented layers were extracted and subsequently converted to black and white binary images. The generated annotations were reviewed by the pathologist. Methods from the NumPy [21] and PIL Image modules [22,23] were utilized to merge the patches back to the original WSIs. Then, methods from the PIL Image module were used to extract smaller patches of 256 x 256 pixels on the original images and ground truth labels (also called masks or segmentation maps) using a sliding window approach, increasing the dataset size. The smaller patches were then converted to NumPy arrays which were fed as input into our deep learning models. Overlapping patches were cropped from the WSIs to help smooth outputted predictions. Ultimately, we ended up with 10,240 patches in our dataset (excludes test set images), 75% of which we used for training and 25% of which we used for validation.

To compare the semantic image segmentation results between the U-Net model [11] and the DeepLabV3+ model [14], the publicly available MICCAI Gland Segmentation (GlaS) Challenge pathology image dataset [24] was downloaded [25]. The dataset consisted of .bmp (bitmap) images split into a training set of 85 images, a test A set of 60 images, and a test B set of 20 images. There were images of six different sizes in the training and test A sets: (574 x 433),

(589 x 533), (775 x 522), (567 x 430), (578 x 433), and (581 x 442). All test B images were of size (775 x 522). All images were downloaded with their corresponding ground truth labels, outlining glandular objects in colon tissue, and they were pre-processed into forms that the deep learning models could use as input. Briefly, customized Python scripts were created to convert the labels to binary black-and-white images and crop the images and ground truth labels into corresponding 256x256 patches via a sliding window approach. The data were then converted into NumPy arrays and fed into the models as an input. We used the patches created from the training and test A sets as our combined training dataset and the test B set as our test set.

For the classification of carcinoma and sarcoma, all images were downloaded from the NCI Patient-Derived Model Repository (PDMR) [26]. As a pilot study, we only selected low-magnification (4x) image data. 244 carcinoma images from 9 different carcinoma subtypes and 180 sarcoma images from 7 different sarcoma subtypes were selected. These images are provided by PDMR as region of interests (ROIs) extracted from the original WSIs. We tried to select a diverse range of images that had vastly different appearances to improve the generalization abilities of our models. After obtaining data from the PDMR database, the data was split into a training, validation, and test set. 10% of the carcinoma and sarcoma images were first randomly set aside to be used as test set images, and then 20% of the remaining images were randomly set aside to be used as the validation set, leaving the other images to be used as the training dataset. After splitting up the images, each set was then converted to overlapping patches via a sliding window approach. Tables 1 and 2 show the details of the dataset split that was used to train the data to optimize the results with patch extraction for our binary classification study between sarcoma and carcinoma and multi-class classification study between subtypes of carcinoma, respectively. In our models, we also applied data augmentation techniques (horizontal flip, dimensional shift, rotation, zoom, etc.) to our image set to increase the generalization capabilities of our classification models, given our small dataset.

Table 1. Dataset split implemented for binary classification of sarcoma and carcinoma

	ROI Images				Patches		
	Total	Train	Val.	Test	Train	Val.	Test
Carcinoma	244	187	33	24	18711	3089	2612
Sarcoma	180	138	24	18	13951	2702	1928

Table 1. Dataset split implemented for binary classification of sarcoma and carcinoma. 10% of the carcinoma and sarcoma images were first randomly set aside to be used as test set images,

and 20% of the remaining images were randomly set aside to be used as the validation set, leaving the other images to be used as the training dataset. After splitting up the ROI images, each set was then converted to patches. Val.: Validation set.

Due to the small number of ROI Images of each tumor type, especially in the case of the dataset split in our multi-class classification study of carcinoma subtypes, patch extraction was again used to increase dataset size [17]. An internal assessment suggested patch extraction be a more effective data preparation technique than other methods such as downsizing (unpublished internal results). Similar to the pipeline we used in segmentation tasks, customized Python scripts were created to add necessary padding and then strategically crop the large images into 256x256 patches with an overlapping shift using a sliding window approach. We also applied a threshold value and discarded patches where more than 90 percent of the pixels in the patch were white and grey background pixels (we define this to be pixels with R, G, and B values all greater than 240).

Table 2. Dataset split implemented for multi-class classification of subtypes of carcinoma

Cancer Type	ROI Images				Patches		
	Total	Training	Validation	Test	Training	Validation	Test
Adenocarcinoma - colon	26	20	4	2	2175	326	147
Adenocarcinoma - pancreas	25	19	4	2	2094	341	490
RCC, clear cell adenocarcinoma	36	27	6	3	2374	406	328
Renal cell carcinoma, NOS	25	19	4	2	1541	222	90
Adenocarcinoma - rectum	25	19	4	2	3174	725	161
Laryngeal squamous cell carcinoma	19	15	3	1	1414	285	91
Pharyngeal Squamous Cell Carcinoma	38	28	7	3	2664	994	411
Lung adenocarcinoma	25	19	4	2	1546	286	142
Squamous cell lung carcinoma	25	19	4	2	1478	313	194
Total	244	185	40	19	18460	3898	2054

Table 2. Dataset split implemented for multi-class classification of subtypes of carcinoma. Patches were converted from each subtype of carcinoma dataset due to a limited number of ROI images in each category with ground truth labels. Dataset split implemented for the multiclass classification task. 10% of the images of each carcinoma subtype (rounded down) were first randomly set aside to be used as test set images, and 20% of the remaining images were randomly set aside to be used as the validation set, leaving the other images to be used as the training dataset. After splitting up the ROI images, each set was then converted to patches.

Segmentation model architectures and performance evaluation

We used the U-Net and DeepLabV3+ models for our segmentation studies. Named after the shape of its architecture, the U-Net convolutional neural network works well for biomedical image segmentation, even with minimal training images [11]. Developed by Google, DeepLabV3+ has achieved state-of-art results on the PASCAL VOC 2012 and Cityscapes datasets (89.0% and 82.1%, respectively), highlighting the model's accurate segmentation abilities [14]. For best performance, the model utilizes an Aligned Xception model backbone modified to support atrous separable convolutions and batch normalization features. A Keras implementation of the DeepLabV3+ model was retrieved from GitHub [27] and integrated into our pipeline, while we recreated the U-Net model from scratch. Both the U-Net and DeepLabV3+ model architectures were built on Python 3 using the Keras [28] and TensorFlow [29] modules and were trained on an Ubuntu workstation with two NVIDIA GTX 1080 GPUs. Multi-GPU parallelism was used to significantly increase computational power and decrease training time [30].

To fairly assess their abilities, both models used the Dice coefficient [31,32] to measure similarity to ground truth labels, and the negative values of the Dice score were used as the loss values that the models learned to minimize in order to boost performance. K-fold cross-validation and Dropout layers were also utilized to help prevent overfitting and improve the models' generalization abilities. Customized Python wrapper functions acted as pipelines to handle the constructions and auxiliary function calls of the model architectures for training, model parameters setting and tuning, as well as performance evaluation. To improve model performance, hyperparameters such as learning rate, batch size, and the number of training epochs were fine-tuned based on previous advice [20]. Depending on the user's available computational resources, the "output stride" variable can also be decreased from 16 to 8 in training the DeepLabV3+ model to achieve slightly better segmentation accuracy at the cost of higher computational complexity [14]. After training, the models outputted test accuracy values as an indicator of model performance. We also predicted masks on test set images that were not seen during training, and these mask predictions were outputted alongside their corresponding images and ground truth labels to visually evaluate performance.

In our work to train a model to automatically detect and segment stroma in WSIs, our U-Net model was also trained with a customized validation scheme. The model was run on a server in a Python Virtual Environment with the following packages: Keras 2.1.2, TensorFlow-GPU 1.4.1, Pillow 4.3.0, NumPy 1.13.3 and Scikit-Image 0.13.1.

We compared the performances of the U-Net and DeepLabV3+ models on the MICCAI GlaS image dataset. In training the models, an initial learning rate of 0.0001 (1e-4) was used, and 4-fold cross-validation was also utilized. Then test set image data were used for prediction by the best-trained models and visually evaluated for performance comparison. For the purpose of comparison, we also trained some models with a customized validation scheme in addition to training others with conventional k-fold cross-validation. In general, the DeepLabV3+ model was trained on batch sizes of 8 (due to memory constraints), while U-Net was trained on batch sizes of 16. We also fine-tuned other hyperparameters. Pipelines for U-Net and DeepLabV3+ with a similar organizational structure were created and run in similar Python Virtual Environments as the stroma image data.

Tumor type and tumor subtype classification model architectures and performance evaluation

A simple convolutional neural network (CNN) was trained from scratch as the reference point [33]. Weight decay and Dropout layers were used at the end of the simple CNN. Additional layers were added if needed, and the number of filters on existing layers were also adjusted to optimize model performance. We used a sigmoid activation at the end of the binary classification models, while we used a softmax activation at the end of the multiclass classification models. The labels assigned to each class (tumor type or subtype) made use of Keras' ImageDataGenerator class and a customized directory structure.

The models trained with transfer learning made use of the VGG-16 [18] and InceptionResNetV2 [19] networks as feature extractors, two models of greatly varying complexity that both come pre-packaged with Keras. We trained the transfer learning models [15,16] with the selected pre-trained models as a convolutional base, and additional fully connected layers (Dense layers) were attached to the end of the model to act as the classifier. Although three different classifiers (Dense layers, global pooling layers, and linear support vector machines) were tested to optimize the models, Dense layers were selected to be used in this study.

We trained the transfer learning models with the techniques of feature extraction and fine-tuning [30]. A customized function was created and used to unfreeze layers and then recompile the model for fine-tuning with a new learning rate. The models were first trained with VGG-16 and InceptionResNetV3 acting as feature extractors, meaning that the model was trained with a completely frozen (untrainable) convolutional base for a certain number of epochs. Then, the

top layers in the model were unfrozen, and the model was trained with fine-tuning for another certain number of epochs.

To evaluate model performance, we used the loss and accuracy metrics pre-packaged with the Keras module. For performance comparison, we also derived AUCs of ROC curves for each trained model with customized R scripts that used the pROC R package [34]. The confidence and accuracy of the binary and multiclass classification models in their test set ROI image predictions were also assessed by Keras' built-in accuracy metric, which we used as test accuracy (patch-wise classification score) for performance evaluation. In addition, to study the details of performance, we also outputted image-wise classification scores and classification results on randomly selected ROIs using test set prediction functions designed for our classification tasks, like the pipeline we built for U-Net and DeepLabV3+.

In outputting predictions for binary classification between carcinoma and sarcoma, each test set image is turned into patches (the ones with too many background pixels are again thrown out) and sent into the model to predict and output class labels. The model outputs a prediction score between 0 and 1 for each patch, where values closer to 0 suggest the model is more confident that the patch should be classified as carcinoma, while values closer to 1 suggest the model is more confident the patch is sarcoma. We then averaged the prediction scores among all inputted patches to compute the image-wise classification score of the model on each overall ROI image. If the ROI ground truth label is carcinoma and the image-wise classification score is less than 0.5, we denoted that the ROI was correctly classified by the model. If the image-wise classification score is greater than 0.5, we denote that the ROI was incorrectly classified. Similarly, ROIs with the ground truth label of sarcoma are correctly predicted if the image-wise classification score is greater than 0.5 and incorrectly predicted if it is less than 0.5. We define this to be the classification result of each test set ROI. We used the image-wise classification scores and classification results of all individual test set ROIs as input for the ROC analysis for performance comparison of all models.

To output test set image predictions of our multiclass classification models for different carcinoma subtypes, we derived a simple customized method to output a final class label prediction by taking advantage of the mode of the patch class labels. First, each ROI is turned into patches, and we obtained the softmax-activated prediction results of the model and assigned a predicted class to each patch according to its largest probability value. The frequency of each predicted class among the patches was then tallied, and the predicted class label for the input image was taken to be the most frequently predicted class among all of its

individual patches. If the predicted class label was the same as the ground truth label, we denoted that we classified the ROI correctly. We used this prediction function on our test set ROIs to output the classification results for our multiclass classification models.

Results

Segmentation of cancer tissue images can distinguish between stroma and non-stroma pixels using a U-Net architecture based deep learning model

Inspired by previous studies showing that tumor-stroma ratio (TSR) serves as an independent prognostic factor in many types of cancer [7,9,10], we sought to examine deep learning methods in the segmentation of tumor images to differentiate the stroma and non-stroma areas as starting points to potentially initialize more deep downstream analysis relevant to researching the role of TSR in cancer prognosis. As a pilot experiment, we chose to use the U-Net method, which is a convolutional neural network that was developed for biomedical image segmentation [11] and has been a conventional method in the field since its development (reviewed in [35, 36]). However, not many studies have applied this technique to differentiate the stroma and non-stroma cells from tumor images of cancer patients. Very few related studies were reported, although only one report found from the literature did show promise of U-Net in segmentation of multi-class segmentation including stroma as one of the classes in prostate cancer images [37]. We set up an image analysis framework that utilized the U-Net model and applied it to image data from cancer patients.

After multiple test runs and tuning hyperparameters, the U-Net model rather consistently obtained a validation Dice score close to 0.80 through these trial runs (Figure 1A). Visual comparison of images of ground truth labels and those of predicted labels indicated rather similar patterns (Figure 1B, 1C,1D).

Comparing the performance of a U-Net architecture based deep learning model and DeepLabV3+ model in semantically segmenting glandular objects in colon tissue

After finding the great capability of the modified U-Net to perform semantic segmentation, we wanted to compare this model's performance in this task with a more complex model, the DeepLabV3+, using the MICCAI Gland Segmentation (GlaS) Challenge dataset. The great success of DeepLabV3+ in the segmentation of other non-medical image datasets [14] inspired us to examine directly the great potential of DeepLabV3+ in the segmentation of tumor images. Also, it would be rather interesting to compare the semantic image segmentation results derived from these two very different neural network architectures to gain some insights on the

potentials of both models. In this study, these models were trained to segment glandular objects in the GlaS dataset rather than the previous stromal cells in cancer tissue due to the higher availability of accessible data from this set.

Table 3. Comparison of segmentation results between U-Net and DeepLabV3+ in multiple exploratory trial runs

Model	Batch Size	Epochs	Ensemble Test Accuracy
DeepLabV3+	4	32	0.847+/-0.002
DeepLabV3+	4	64	0.859+/-0.009
DeepLabV3+	8	16	0.833+/-0.022
DeepLabV3+	8	32	0.850+/-0.013
DeepLabV3+	8	64	0.852+/-0.007
DeepLabV3+	8	64	0.856+/-0.008
U-Net	4	16	0.802+/-0.037
U-Net	4	32	0.840+/-0.022
U-Net	4	64	0.793+/-0.055
U-Net	8	16	0.778+/-0.036
U-Net	8	32	0.794+/-0.055
U-Net	8	64	0.747+/-0.000
U-Net	16	16	0.760+/-0.029
U-Net	16	32	0.761+/-0.032
U-Net	32	32	0.747+/-0.000

Table 3. Comparison of segmentation results between U-Net and DeepLabV3+ in multiple exploratory trial runs. In all of the example trials shown here, we used an initial learning rate of 0.0001 and trained with 4-fold cross-validation. In general, the DeepLabV3+ model was trained on batch sizes of from 4 to 8 (due to memory constraints) and epochs ranged from 8 to 64 (only show epochs above 32 with better performance), whereas U-Net was trained on batch sizes ranging from 4 to 32 and epochs ranged from 16 to 64.

We first trained models of the chosen networks with conventional k-fold cross-validation and evaluated the models' predictions by ensembling the predictions of the k models produced (in our case, k = 4). Interestingly, even after running many trial runs with various conditions, U-Net still showed generally lower ensemble test accuracies compared to DeepLabV3+. The achieved test accuracy scores from U-Net ranged from 0.747 to 0.840. However, the model appears to only achieve scores above 0.8 in very few conditions, while in contrast, the ensemble test accuracies of DeepLabV3+ at various conditions consistently achieved scores above 0.8, with our specific results ranging from 0.833 to 0.859 (Table 3). This implies that DeepLabV3+ seems to more easily achieve a trained model with a relatively higher

performance compared to U-Net, which seems much more unreliable, at least in our results performed on a limited dataset. Visualizing predictions on test set images using the best models trained from both networks based on ensemble test accuracy (Table 3), we also found that DeepLabV3+ had better performance compared to U-Net on gland segmentation on the test dataset (Figure 2), which is consistent with the overall better ensemble test accuracies achieved by DeepLabV3+ (Table 3).

The observation that DeepLabV3+ performed better than U-Net was expected since DeepLabV3+ is a more evolved network than U-Net, although the performance of both DeepLabV3+ and U-Net still appeared to be limited in this setting, despite the promising performance of U-Net on stroma segmentation described earlier.

Notably, due to limitations on our computational resources, we could not increase the batch size inputted into DeepLabV3+ beyond a size of 8, whereas U-Net could use a batch size up to 32 (Table 3). In theory, a larger mini-batch size should help the network converge to a better minimum and therefore better final accuracy [38]. However, we did not see the benefit of larger batch-size in our model training for U-Net, where a model trained with a batch size of 4 achieved the best performance. This is consistent with a recent report about the benefit of a smaller batch size [39].

For the purpose of comparison, we also trained the models using a different experimental customized validation scheme with various conditions shown in Table 4. We were able to reach validation Dice scores consistently above 0.85 with both models, and a validation Dice Score reaching up to 0.98. In addition, the test set image prediction results using the best-trained models derived from both networks showed accurate segmentation results (Figure 3).

Table 4. Comparison of segmentation results between U-Net and DeepLabV3+ in multiple exploratory trial runs with training by customized validation.

Model	Batch Size	Epochs	Validation Dice Scores
DeepLabV3+	8	16	0.8534
DeepLabV3+	8	32	0.8993
DeepLabV3+	8	32	0.8921
DeepLabV3+	8	64	0.9826
DeepLabV3+	8	64	0.9847
U-Net	16	16	0.863
U-Net	16	24	0.9219
U-Net	16	32	0.9426
U-Net	16	32	0.9337
U-Net	16	64	0.9583
U-Net	16	64	0.9847
U-Net	32	16	0.7386

Table 4. Comparison of segmentation results between U-Net and DeepLabV3+ in multiple exploratory trial runs with training by customized validation. In all of the example trials shown here, we used an initial learning rate of 0.0001 and trained by customized validation rather than conventional k-fold cross-validation. In general, the DeepLabV3+ model was trained batch sizes of 8 (due to memory constraints) and epochs ranged from 16 to 64, while U-Net was trained on batch sizes ranged from 16 to 32 and epochs ranged from 16 to 64. Only showed cases with better performance for similar conditions.

To our knowledge, our study would be the first report to show the potentials of DeepLabV3+ in segmentation on tumor images, although it had been successfully used for segmentation on other types of images.

Classification of different cancer types with deep learning models

After testing the capabilities of deep learning models in semantic image segmentation, we turned our attention to exploring the potential of selected deep learning models in cancer type classification. Specifically, the impacts of transfer learning [15,16] and patch extraction [17] strategies, which were initially reported to improve performance in classification, were applied to our publicly available dataset taken from the NCI PDMR database to classify sarcoma and carcinoma. A simple convolutional neural network (CNN) was trained from scratch and compared with models trained using transfer learning, with VGG-16 [18] and InceptionResNetV2 [19] networks as the feature extractors, or convolutional bases. The

performance of the models was assessed on test set ROIs for their patch-wise and image-wise classification results.

While the performance of all models consistently achieved test accuracies (patch-wise classification score) above 80%, the model trained on VGG-16 achieved the highest training, validation, and test accuracies, whereas the simple CNN showed the lowest test accuracy (Table 5). ROC analysis of image-wise classification results showed a similar trend as the patch-wise classification results, with VGG-16 having the best ROC AUC at 0.6875 (Table 5), though the difference is more obvious compared to test accuracy results (Table 5). This may be because the ROC AUCs are assessed at an image level, while the patch-wise classification score directly outputted by the models relies more on aggregated patch-level analysis. On the other hand, the shape of the ROC curves, where all of them are below the diagonal line at the higher specificity side, further suggested the likely adverse impact from the unbalanced dataset used (Figure 4).

Table 5. Best accuracy results achieved with each type of model used for binary classification

Model Type	Training Accuracy	Validation Accuracy	Test Accuracy	ROC AUCs
Simple CNN	0.8695	0.8375	0.8471	0.6163
VGG-16	0.9231	0.8909	0.8673	0.6875
InceptionResNetV2	0.8647	0.8556	0.8573	0.5724

Table 5. Best accuracy results achieved with each type of model used for binary classification. Each model was trained for 20 epochs, with the VGG16 and InceptionResNetV2 models each trained for 10 epochs using feature extraction and 10 epochs for fine-tuning. The transfer learning model trained based on VGG-16 achieved the highest training, validation, test accuracy, and the best AUC of ROC analysis. Accuracy output from the models and AUCs of ROC analysis of the test set results were shown. We consider the patch-wise classification scores as the test accuracy scores. The image-wise classification results are used as input for ROC analysis.

The trends in training and validation loss and accuracy over training time can also be seen in Figure 5. Note that while the InceptionResNetV2 showed a lower test accuracy than VGG-16, its performance may suffer from its high complexity and slow runtime due to constraints of our

server, although it does show promise in being a more accurate model if given more training data or trained for more training epochs.

Table 6. Class labels and prediction scores of all three models on samples 1 to 10 in Figure 6

	Samples	1	2	3	4	5
Class Labels	Ground Truth	Sarcoma	Sarcoma	Sarcoma	Sarcoma	Sarcoma
	Simple CNN	Carcinoma	Sarcoma	Sarcoma	Sarcoma	Carcinoma
	VGG-16	Carcinoma	Sarcoma	Carcinoma	Carcinoma	Carcinoma
	InceptionResNetV2	Carcinoma	Sarcoma	Carcinoma	Sarcoma	Carcinoma
Prediction Scores	Ground Truth	1	1	1	1	1
	Simple CNN	0.1348	0.8468	0.5519	0.6193	0.4386
	VGG-16	0.2665	0.8703	0.1949	0.2806	0.1833
	InceptionResNetV2	0.2357	0.8214	0.4668	0.6912	0.2974

	Samples	6	7	8	9	10
Class Labels	Ground Truth	Carcinoma	Carcinoma	Carcinoma	Carcinoma	Carcinoma
	Simple CNN	Sarcoma	Carcinoma	Carcinoma	Carcinoma	Carcinoma
	VGG-16	Carcinoma	Carcinoma	Carcinoma	Carcinoma	Carcinoma
	InceptionResNetV2	Carcinoma	Carcinoma	Carcinoma	Carcinoma	Carcinoma
Prediction Scores	Ground Truth	0	0	0	0	0
	Simple CNN	0.5548	0.3754	0.1996	0.08708	0.2295
	VGG-16	0.09891	0.1734	0.1715	0.007035	0.02944
	InceptionResNetV2	0.2774	0.1682	0.2391	0.06385	0.09268

Table 6. Detailed class labels and prediction scores of all three models on randomly selected samples 1 to 10 with images shown in Figure 6. Prediction results are shown for the simple convolutional neural network (Simple CNN), VGG-16 transfer learning model, and InceptionResNetV2 transfer learning model with the best test accuracies (patch-wise classification scores). Prediction scores (image-wise classification scores) are the predicted class probability values derived from the corresponding model. For ground truth, the prediction score is defined as 1 for sarcoma and 0 for carcinoma. The class label is the predicted class based on the corresponding prediction score dependent on whether the score is closer to 1 or 0 (image-wise classification result). Note that the designed prediction function for these models breaks a test set into evenly sized patches, outputs prediction scores on each individual patch, and then averages the predictions across all the patches to produce a single predicted class probability value as the prediction score for the whole image, where values closer to 0 show that the model is more confident about classifying the image as carcinoma and values closer to 1 show that the model is more confident classifying the image as sarcoma.

As shown in Table 6, we have also studied the details of the image-wise classification scores and results compared to their ground truth labels on ten randomly selected test set images (shown in Figure 6). While the model based on VGG-16 achieved the lowest number of correct classifications, we noticed it still performed the best on the ROI predictions in two aspects. Every case of carcinoma was accurately classified, with the closest prediction scores to the ground truth score of 0 amongst all three types of models. On the other hand, even though the model failed to correctly classify more sarcoma images than the other two, it managed to classify one sarcoma image with the highest score (0.8703) closest to its ground truth score

(Table 6). The images that this model incorrectly predicted also had prediction scores closest to 0.5, meaning that it was the least confident in its prediction when it predicted incorrectly. Combined with the fact that the VGG-16-based model also achieved the highest overall test accuracy and the highest AUC from ROC analysis (Table 5), the great performance of this model mainly seems to benefit from its high specificity.

Multi-class classification of different subtypes of carcinoma with deep learning models

Having achieved relatively high accuracy in classifying carcinoma and sarcoma samples in our binary classification experiments, we extended our work to classify different carcinoma subtypes within a total of nine subtypes of carcinoma in our dataset. We chose to mainly focus on classifying carcinoma subtypes because we had more carcinoma data and subtypes in our dataset than sarcoma data and subtypes. We slightly modified our binary classification models to work for multiclass classification as described in our Methods section.

We conducted an extensive empirical evaluation with different scenarios of hyperparameter tuning. Despite the limited number of training runs performed, one of our multiclass classification models utilizing the transfer learning scheme based on VGG-16 was able to reach a test accuracy (patch-wise classification score) of just under 59% (Table 7). Considering that we trained each model to learn to classify between nine classes, this result was extremely good, far better than a baseline random classifier with a test accuracy of about 1/9, or 0.11 (in a balanced dataset). This is also consistent with the above results of our binary classification models. Furthermore, after we performed additional processing on the patch-wise classification scores to compute the image-wise classification result (by taking advantage of the mode of the patch class labels as mentioned in the Methods section), we were able to consistently classify 9 out of 10 sample test set ROIs correctly (the test set images were from all nine classes, data not shown), which appears significantly more accurate than the 59% test accuracy (Table 7). These observations will be subjected to further investigation.

Table 7. Best accuracy results achieved with each type of model used in multiclass classification

Model Type	Training Epochs	Training Accuracy	Validation Accuracy	Test Accuracy
Simple CNN	25	0.6373	0.2925	0.3248
VGG16	15 /10	0.8034	0.5351	0.5949
InceptionResNetV2	20/25	0.7604	0.5539	0.5604

Table 7. Best accuracy results achieved with each type of model in multiclass classification. Each model was trained for a different number of epochs. The first value in the epochs column for the VGG-16 and InceptionResNetV2 models refers to the number of epochs trained with feature extraction, while the second value denotes the number of epochs trained with fine-tuning. The transfer learning model trained based on VGG16 achieved the highest test accuracy (patch-wise classification score), while the InceptionResNetV2 model achieved the highest validation accuracy.

Discussion

In this study, we were able to use the U-Net model to rather accurately perform segmentation of the stroma and non-stroma areas in tumor images. Although it is a pilot experiment with a limited amount of data available, we believe our results are already more accurate than what was expected and show great promise in sparking more future in-depth investigations once more data is available. Since manually segmenting the stroma can be difficult, even with the guidance of a professional pathologist, the human error in classifying stroma areas could have caused inconsistencies in the training data which would have affected the model performance. We believe that the modified U-Net model could likely attain higher Dice scores with further hyperparameter tweaking (namely, adjusting the batch size and learning rate), adding batch normalization layers to the model, increasing the number of training patches, and/or using data augmentation techniques to artificially increase the amount of data seen by the model. In summary, our work suggests the great potential of U-Net as a segmentation tool to help automatically and accurately assess the TSR as a prognosis biomarker.

In our comparison of segmentation results between U-Net and DeepLabV3+, we observed performance differences with and without k-fold cross-validation between the two models. DeepLabV3 + generally performed better, which makes sense based on its excellent performance on non-medical image data. However, we note that we were still able to train U-Net models that could output prediction images that appeared almost as accurate as those produced by DeepLabV3 + models when using our customized validation scheme. Overall, we observed a plateau of performance from U-Net and DeepLabV3 + on our dataset around 0.840 to 0.859 respectively when using k-fold cross validation. This leads us to believe that our results could be better if more data was available. Thus, we cannot eliminate the possibility that U-Net may still have viability as an effective model in biomedical image segmentation given a larger dataset.

We note that our results for our customized validation scheme for both models are exceedingly high when trained for 64 epochs (achieving a validation Dice score of greater than 0.98). These values are not fully indicative of the models' performance on test set images, but we believe they are still somewhat exaggerated due to the nature of our custom validation scheme. However, this does not change the fact that we were still able to output very accurate prediction results on test set images.

We believe that further investigation into the capabilities of both models would be beneficial. For example, we may gain further insight into the models by comparing their performances in segmenting more types of tissue, such as necrosis (regions of dead cells), or in segmenting out multiple types of tissue at once.

Due to its complexity, we also observed DeepLabV3 + encounter constraints on our server's capacity, such as the limit on the batch size allowed without causing the server to run out of memory. The current capacity of our server could also not allocate enough memory for the DeepLabV3 + model to train with an output stride of 8, which limited its ability to improve its performance on segmentation as well. The model was more limited by memory constraints.

We did notice that the model also required much more time to train than U-Net. Given the relatively small size of the dataset (around 3000 training patches), the DeepLabV3 + model still required three to four times as long to train even when training on half as many epochs as U-Net (90–120 minutes compared to 20–30 minutes on U-Net). Training the DeepLabV3 + model with a pre-trained set of weights could help decrease training time and increase accuracy.

We conclude that DeepLabV3 + shows better promise in being a useful tool with better performance in biomedical image segmentation, which would perform even better if memory constraints are removed. U-Net may still be an effective and efficient model to utilize if given a larger dataset, however.

For the classification of sarcoma and carcinoma, we have achieved consistent patch-wise classification scores around 85% with all models, with VGG-16 achieving the highest performance. We also believe that we may achieve higher image-wise and patch-wise classification scores by implementing several improvements. For one, we only used a tiny dataset for the experiment, and, given a larger dataset, we may see higher classification scores. We believe that the performance of all of the models could also be improved by fine-tuning hyperparameters. Implementing class weighting and adjusting patch size could be two other options to further improve model performance. Furthermore, in this study, we restricted the number of training epochs that each model trained for to 20. This was done to get a more comparable performance assessment for each of the models, but all the models could very likely improve if this constraint on the number of training epochs was removed or increased and the models could train longer.

Our multiclass classification results on carcinoma subtypes also appear very promising. While the test accuracies of our models appear to be relatively low, they are indeed very good values, considering the baseline probability of random classification results for this dataset with a total of nine classes. Even though our best model does not have a particularly high test accuracy, after we took advantage of the mode of the patch class labels to derive image-wise classification results, we were still able to consistently classify 9 out of 10 sample test set ROI images correctly.

In the future, like our discussion of our binary classification models, we can also implement class weighting in our multiclass classification models to mitigate the effects of class imbalances. We believe that we may also see further increases in the accuracy of model predictions if other types of post-

processing custom functions were tested and compared to the performance of the one that we designed, as mentioned in the Methods section.

Based on our results, we concluded that VGG-16 has the best overall performance on binary classification, consistent with its best overall performance in multiclass classification as well, although more datasets may be needed to validate the results. Compared to the performance of simple CNNs, especially when looking into the AUCs of ROC analysis, networks with transfer learning appeared to have better performance. However, we concede that InceptionResNetV2 could possibly have better performance if given more computational resources, since the extremely long training time of this complex network prevented us from training the network as sufficiently as the other two.

Importantly, our results in sarcoma vs. carcinoma classification helped provide a baseline for low quality and/or resolution images. It shows the high accuracy we may achieve even without using high-quality original resolution images. The use case could be using low-resolution images from clinical reports, manuscripts, or data sources not capable of generating high-quality images such as taking a picture using a cell phone from a microscope.

There are other lessons learned on the technical aspects of this study. For example, potentially beneficial combinations of convolutional, Dense, Dropout, and Regularization layers could be tested (the performances of new architectures trained from scratch could be tested). In utilizing transfer learning, we can also try unfreezing the model from different layers or try adapting other pre-trained models with transfer learning like Xception, ResNet, and MobileNet. We can also perhaps attempt to train our own model from scratch (like U-Net) on another, larger biomedical dataset and then use transfer learning with our trained model to see if that will improve performance on our smaller dataset.

Combining the raw pixel information currently utilized by the classification models with the segmentation maps produced by the semantic segmentation models is another interesting avenue of further research that could potentially lead to more refined tissue type classifications useful for pathologists and pathologic diagnosis.

Conclusions

We have explored many deep learning models for their potentials in tumor image segmentation and classifications in research and clinical settings, and tested transfer learning and other strategies for their impacts on model performance. Overall, we believe the progress made and lessons learned from this pilot exploratory study can help the medical image analysis community in utilizing deep learning models for segmentation and classification of tumor image data in both research and clinical settings.

List Of Abbreviations

CNN: Convolutional Neural Network; TSR: Tumor-Stroma Ratio; FNLCR: Frederick National Laboratory for Cancer Research; GlaS: MICCAI Gland Segmentation; WSIs: Whole Slide Images; ROI: Region of Interest (in

cancer histology images); PDMR: NCI Patient-Derived Model Repository; ROC: Receiver Operating Characteristic Curve; AUC: Area Under the Curve.

Declarations

Acknowledgments

The authors would like to thank Uma Mudunuri, the director of the ABCS infrastructure group, for the support of this internship research work sponsored by SCRTA fellowship. This project has been funded in whole or in part with Federal funds from the National Cancer Institute, National Institutes of Health, under Contract No. HHSN261200800001E. The content of this publication does not necessarily reflect the views or policies of the Department of Health and Human Services, nor does mention of trade names, commercial products, or organizations imply endorsement by the U.S. Government. This research was supported (in part) by the National Institutes of Health.

Authors' contributions

EY designed and implemented customized python pipelines for handling and operating the selected deep learning networks/modules. He performed the data handling and analysis for the study. He also drafted, wrote, and revised the manuscript. YL is the mentor of EY's internship of two consecutive summers, provided funding and proposed the initial studies, and reviewed/revised the manuscript. All authors read and approved the final manuscript.

Funding

This project has been funded in whole or in part with Summer Cancer Research Training Award (SCRTA) fellowship from National Cancer Institute, and Federal funds from the National Cancer Institute, National Institutes of Health, under Contract No. HHSN261200800001E. The content of this publication does not necessarily reflect the views or policies of the Department of Health and Human Services, nor does mention of trade names, commercial products, or organizations imply endorsement by the U.S. Government.

Availability of data materials

Most of the datasets analyzed during the current study are available in the repositories that were described in the Methods section. The MoCha dataset for our included stroma study is currently not available for public access.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

References

1. Rebecca L. Siegel RL,Miller KD,Jemal A. Cancer Statistics. CA CANCER J CLIN 2019;69:7–34

2. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015; 521(7553):436-44.

3. Isin A, Direkoglu C, Sah M. Review of MRI-based Brain Tumor Image Segmentation Using Deep Learning Methods. Procedia Computer Science, 2016; 102:317-324

4. Zeng Z, Mao CS, Vo A, Nugent JO, Khan SA, Clare SE, Luo Y. Deep learning for cancer type classification. BioRxiv, 2017; 055715

5. Janowczyk A, Madabhushi A. Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. J Pathol Inform. 2016; 7:29.

6. Tomita N, Cheung YY, Hassanpour S. Deep neural networks for automatic detection of osteoporotic vertebral fractures on CT scans. Comput Biol Med. 2018; 98:8-15.

7. Geessink OGF, Baidoshvili A, Klaase JM, Ehteshami Bejnordi B, Litjens GJS, van Pelt GW, Mesker WE, Nagtegaal ID, Ciompi F, van der Laak JAWM. Computer aided quantification of intratumoral stroma yields an independent prognosticator in rectal cancer. Cell Oncol, 2019; 42(3):331-341.

8. Valkenburg KC, de Groot AE, Pienta KJ. Targeting the tumour stroma to improve cancer therapy. Nat Rev Clin Oncol. 2018;15(6):366-381.

9. Zhang XL, Jiang C, Zhang ZX, Liu F, Zhang F, Cheng YF. The tumor-stroma ratio is an independent predictor for survival in nasopharyngeal cancer. Oncol Res Treat. 2014;37(9):480-4.

10. Xi KX, Wen YS, Zhu CM, Yu XY, Qin RQ, Zhang XW, Lin YB, Rong TH, Wang WD, Chen YQ, Zhang LJ. Tumor-stroma ratio (TSR) in non-small cell lung cancer (NSCLC) patients after lung resection is a prognostic factor for survival. Thorac Dis. 2017; 9(10):4017-4026.

11. Ronneberger O, Fischer P. U-Net: Convolutional Networks for Biomedical Image Segmentation. 2015, arXiv:1505.04597

12. Brancati, N.; Frucci, M.; Gragnaniello, D.; Riccio, D. Retinal vessels segmentation based on a convolutional neural network. In progress in pattern recognition, image analysis, computer vision, and applications; Springer: Cham, Switzerland, 2017; pp. 119–126.

13. Wang C, Zhao Z, Ren Q, Xu Y, and Yu Y. Dense U-net Based on Patch-Based Learning for Retinal Vessel Segmentation. Entropy 2019; 21(2):168

14. Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. arXiv: 1802.02611. Available at: https://arxiv.org/pdf/1802.02611.pdf Reviewed at: https://medium.com/@sh.tsang/review-deeplabv3-atrous-separable-convolution-semantic-segmentation-a625f6e83b90.

15. Pan, S. J., & Yang, Q. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 2010; 22(10):1345–1359

16. Huynh BQ, Li H, Giger ML. Digital mammographic tumor classification using transfer learning from deep convolutional neural networks. J. Med. Imag. 2016; 3(3):034501.

17. Hou L, Samaras D, Kurc TM, Gao Y, Davis JE, Saltz JH. Patch-based Convolutional Neural Network for Whole Slide Tissue Image Classification. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit. 2016; 2016:2424-2433

18. Simonyan K, Andrew Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2015, arXiv:1409.1556.

19. Szegedy C, Ioffe S, Vanhoucke V, Alemi A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. 2016; arXiv:1602.07261v2.

20. GIMP available at http://gimp.org.

21. Oliphant TE. A guide to NumPy, 2016; USA: Trelgol Publishing.

22. Python Imaging Library. Available at: http://www.pythonware.com/products/pil/

23. Pillow documentation. Available at: https://pillow.readthedocs.io/en/stable/

24. Sirinukunwattana JK, Pluim JPW, Chen H, Qi X, Heng P, Guo Y, Wang L, Matuszewski BJ, Bruni E, Sanchez U, Böhm A, Ronneberger O, Cheikh B, Racoceanu D, Kainz P, Pfeiffer M, Urschler M, Snead DRJ, Rajpoot NM. Gland Segmentation in Colon Histology Images: The GlaS Challenge Contest. http://arxiv.org/abs/1603.00275.

25. MICCAI Gland Segmentation (GlaS) Challenge pathology image dataset. Available at (https://warwick.ac.uk/fac/sci/dcs/research/tia/glascontest/download/

26. NIH Patient-Derived Model Repository (PDMR), available at: https://pdmr.cancer.gov/

27. Keras implementation of the DeepLabV3+ model was retrieved from GitHub, available at https://github.com/MLearing/Keras-Deeplab-v3-plus.

28. Keras module available at: http://kera.io

29. TensorFlow module available at http://tensorflow.org

30. Chollet, F. Deep Learning with Python Shelter Island, 2018; NY: Manning Publications.

31 Dice LR. Measures of the amount of ecologic association between species. Ecology. 1945;26:297–302

32. Zou KH, Warfield SK, Bharatha A, Tempany CM, Kaus MR, Haker SJ, Wells WM 3rd, Jolesz FA, Kikinis R.Statistical validation of image segmentation quality based on a spatial overlap index.Acad Radiol. 2004; 11(2):178-89.

33. A simple convolutional neural network (CNN) was described at: https://towardsdatascience.com/a-simple-cnn-multi-image-classifier-31c463324fa

34. Xavier Robin, Natacha Turck, Alexandre Hainard, et al. pROC: an open-source package for R and S+ to analyze and compare ROC curves. BMC Bioinformatics, 2011; 12:77. DOI:

35. Yao W; Zeng Z, Lian C; Tang H. Pixel-wise regression using U-Net and its application on pansharpening. Neurocomputing 2018; 312: 364–371.

36. Hesamian MH, Jia W, He X, Kennedy P.Deep Learning Techniques for Medical Image Segmentation: Achievements and Challenges.J Digit Imaging. 2019; 32(4):582-596.

37. Li J, Sarma KV, Chung HK, Gertych A, Knudsen BS, Arnold CW.A Multi-scale U-Net for Semantic Segmentation of Histological Images from Radical Prostatectomies.AMIA Annu Symp Proc. 2018; 2017:1140-1148.

38. Peng C, Xiao T, Li Z, Jiang Y, Zhang X, Jia K, Yu G, Sun J. MegDet: A Large Mini-Batch Object Detector. 2018; arXiv:1711.07240

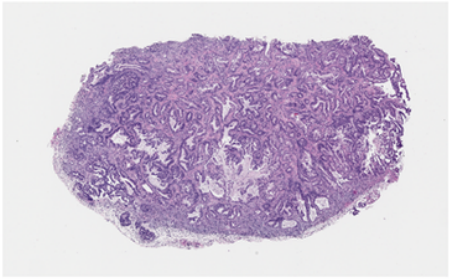
39. Masters D, Luschi C. Revisiting Small Batch Training for Deep Neural Networks. 2018; arXiv:1804.07612

Figures

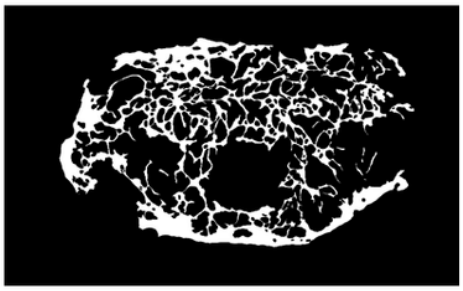
A

Trial	Final Validation Dice Score
1	0.7553
2	0.7448
3	0.7725
4	0.7940
5	0.7838

B



C



D

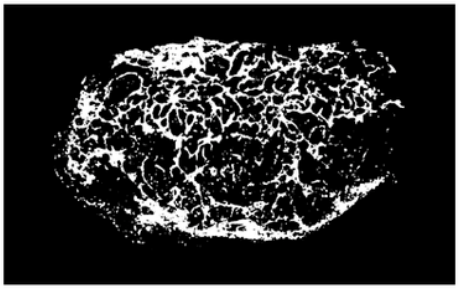


Figure 1

Results of multiple trial runs to test the potentials of the U-Net model in segmenting cancer image data to differentiate stroma and non-stroma areas. A. Table of final validation Dice scores of five trial runs. B. Original image. C. Image of ground truth label. D. Image of the predicted label.

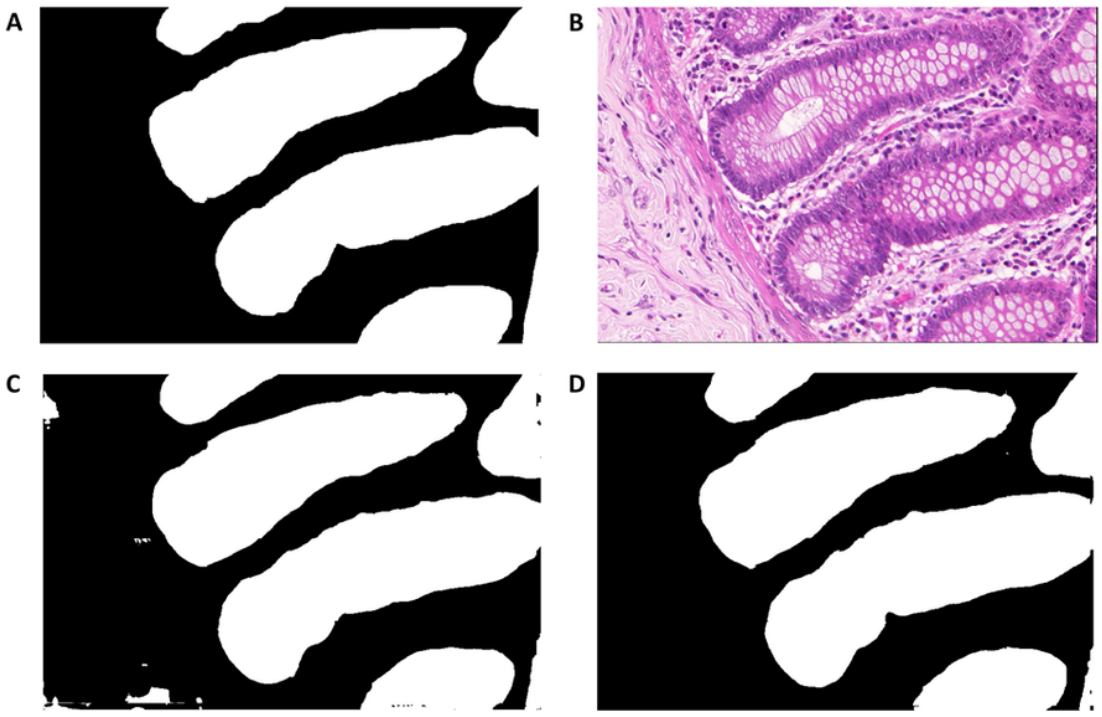


Figure 2

Example segmentation results from ensemble predictions on the test set data using trained models with k-fold cross-validation. A. Ground truth labels. B. Original image of the test set. C. Image of U-Net ensemble prediction. D. Image of DeepLabV3+ ensemble prediction.

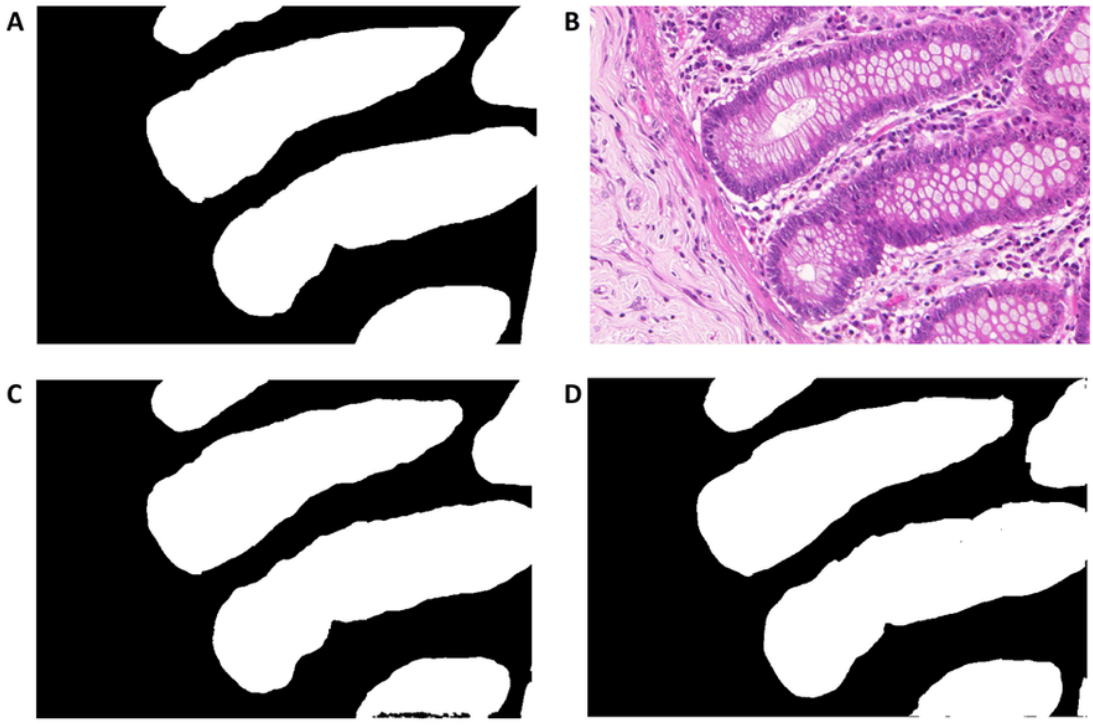


Figure 3

Example segmentation results from predictions on the test set data using trained models with customized validation. A. Ground truth labels. B. Original image of the test set. C. Image of U-Net prediction. D. Image of DeepLabV3+ prediction. Each method used a trained model with the highest ensemble test accuracy achieved.

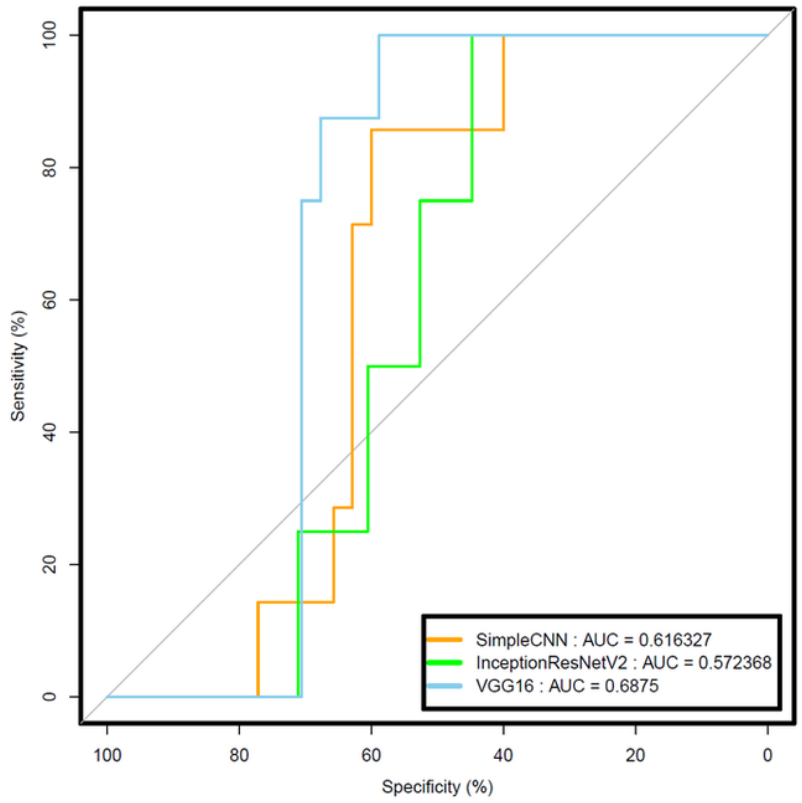


Figure 4

ROC curves of prediction results from trained models of each network. Only the best performed trained models from each network were plotted.

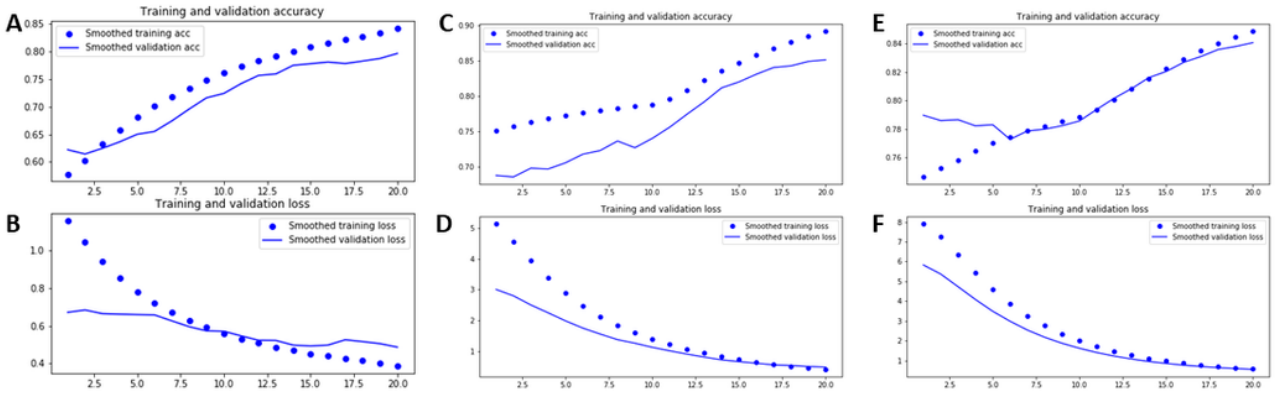


Figure 5

Visualization of the training/validation losses and accuracies over 20 training epochs for the best model of each model type for binary classification. Simple CNN, (A.B). VGG16, (C,D). InceptionResNetV2, (E, F). Top rows(A, C, E): training and validation accuracy. Bottom row (B, D, F): training and validation loss. In all three models, validation accuracy shows an increasing trend at the end of training time, suggesting that model performance can still be improved if trained for more epochs. The first 10 epochs of the VGG-16 and InceptionResNetV2 graphs are trained with feature extraction, and the second half is trained with fine-tuning. The stagnant validation accuracy while training with feature extraction compared with the increasing trend while training with fine-tuning suggests that fewer epochs trained with feature extraction and more epochs with fine-tuning may further increase model performance.

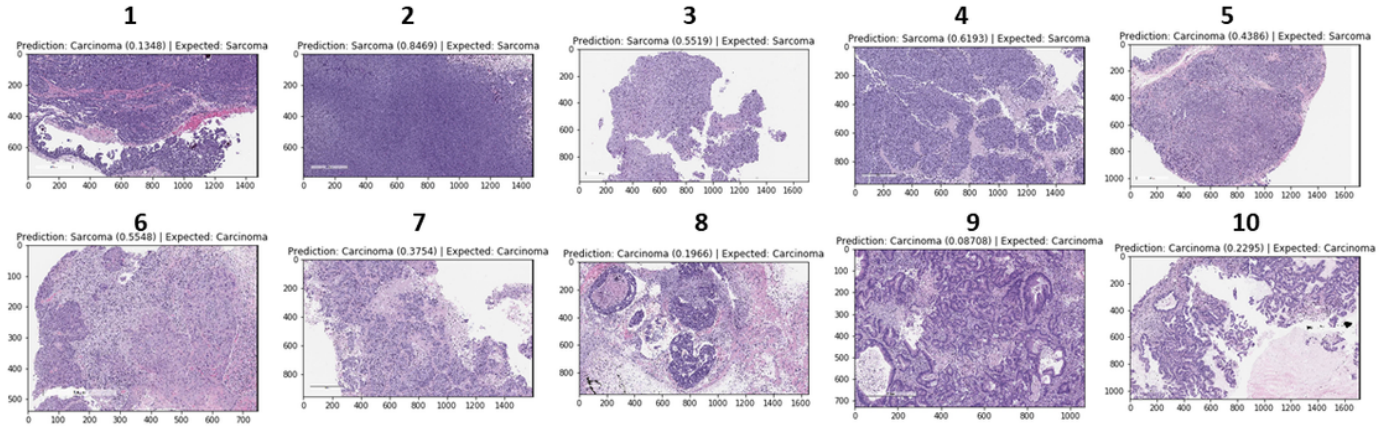


Figure 6

Randomly selected 10 images from test set with detailed individual prediction results compared with the ground truth labels shown in Table 6. Samples 1 to 5 are sarcoma and samples 6 to 10 are carcinoma by ground truth labels.