Journal of Pathology Informatics 13 (2022) 100007

Contents lists available at ScienceDirect

Journal of Pathology Informatics

journal homepage: www.elsevier.com/locate/jpi

Artificial Intelligence-based Tumor Segmentation in Mouse Models of Lung Adenocarcinoma

Alena Arlova [a], Chengcheng Jin [b], Abigail Wong-Rolle [c], Eric S. Chen [b], Curtis Lisle [d], G. Thomas Brown [a], Nathan Lay [a], Peter L. Choyke [a], Baris Turkbey [a], Stephanie Harmon [a,*], Chen Zhao [c]

[a] Artificial Intelligence Resource, Center for Cancer Research, National Cancer Institute, NIH, Bethesda, MD, USA
[b] Department of Cancer Biology, University of Pennsylvania, Philadelphia, PA, USA
[c] Thoracic and GI Malignancies Branch, Center for Cancer Research, National Cancer Institute, NIH, Bethesda, MD, USA
[d] KnowledgeVis, Maitland, FL, USA

ARTICLE INFO

Article history:
Received 9 August 2021
Accepted 14 December 2021
Available online 20 January 2022

ABSTRACT

Background: Mouse models are highly effective for studying the pathophysiology of lung adenocarcinoma and evaluating new treatment strategies. Treatment efficacy is primarily determined by the total tumor burden measured on excised tumor specimens. The measurement process is time-consuming and prone to human errors. To address this issue, we developed a novel deep learning model to segment lung tumor foci on digitally scanned hematoxylin and eosin (H&E) histology slides.
Methods: Digital slides of 239 mice from 9 experimental cohorts were split into training ($n = 137$), validation ($n = 37$), and testing cohorts ($n = 65$). Image patches of 500×500 pixels were extracted from $5\times$ and $10\times$ magnifications, along with binary masks of expert annotations representing ground-truth tumor regions. Deep learning models utilizing DeepLabV3 + and UNet architectures were trained for binary segmentation of tumor foci under varying stain normalization conditions. The performance of algorithm segmentation was assessed by Dice Coefficient, and detection was evaluated by sensitivity and positive-predictive value (PPV).
Results: The best model on patch-based validation was DeepLabV3 + using a Resnet-50 backbone, which achieved Dice 0.890 and 0.873 on validation and testing cohort, respectively. This result corresponded to 91.3 Sensitivity and 51.0 PPV in the validation cohort and 93.7 Sensitivity and 51.4 PPV in the testing cohort. False positives could be reduced 10-fold with thresholding artificial intelligence (AI) predicted output by area, without negative impact on Dice Coefficient. Evaluation at various stain normalization strategies did not demonstrate improvement from the baseline model.
Conclusions: A robust AI-based algorithm for detecting and segmenting lung tumor foci in the pre-clinical mouse models was developed. The output of this algorithm is compatible with open-source software that researchers commonly use.

Introduction

Lung cancer is the leading cause of cancer-related deaths globally, with lung adenocarcinoma (LUAD) being the most common type of non-small lung cancer (NSCLC).[1,2] Genetically engineered mouse models (GEMMs) serve an essential role in pre-clinical studies of cancers, and multiple GEMMs were developed to study the pathophysiology of human lung cancer. GEMMs are inbred mice with precisely controlled genetic modifications, such as point mutations, deletions of chromosomal segments, and inactivations of target genes. GEMMs in pre-clinical studies provide researchers with opportunities to study tumor microenvironment, isolate and control genetic mutations, determine the therapeutic dosage, and observe host immune response.[3] The most common mutations in human LUAD are activating point mutations in *KRAS* and inactivation of *P53*.[4,5] Mice with conditional *KRAS* activation and *P53* loss of function (KP mice) are infected with Cre expressing adenovirus, which activates the transcription of mutant *KRAS* and loss of *P53*. This process induces lung tumors. This GEMM was used to obtain digital whole slide images (WSI) of lung tissue in the present study. It closely resembled human LUAD and was used to study the interactions among tumor cells, the immune system, and the microbiota in the tumor microenvironment.[6]

Histological analyses with hematoxylin and eosin (H&E) staining and immunohistochemistry (IHC) techniques allow researchers to visualize normal and tumor cells. Tumor burden, calculated as the ratio of tumor area to

[*] Corresponding author at: 9000 Rockville Pike NIH, Building 10, Room B3B85, Bethesda, MD 20892, USA.
E-mail address: stephanie.harmon@nih.gov (S. Harmon).

http://dx.doi.org/10.1016/j.jpi.2022.100007

normal tissue area in a sample, is used to judge treatment effects. Therefore, accurate tumor measurement is crucial in determining the outcome of experiments. Manual identification of lesions on WSI by pathologists can be tedious and time-consuming, especially when processing a large dataset. Publicly available open-source tools help researchers detect and segment lesions on WSI, edit annotations, and perform basic analysis.[7,8] However, these semi-automated tools still require extensive and laborious manual annotation, which significantly limits lung cancer research. Hence, an optimized method for tumor measurement in GEMM of lung cancer is urgently needed.

Widespread use of digital pathology and the availability of sufficient computational resources to process large digital image datasets have prompted the development of automated WSI processing methods that aid cancer research. With the help of artificial intelligence (AI), the task of tumor segmentation on digital WSI can be achieved quickly and with accuracy comparable with the experienced pathologist. Deep learning, a branch of AI, has been widely used in digital pathology to detect, segment, and classify cancers across many different diseases. Several recent examples include multiclass classification of breast cancers,[9] classification of epithelial tumors of stomach and colon,[10] lung cancer detection and segmentation.[11–15] In mouse models studying pathologies of other lung diseases, deep learning has been used to assign histological scoring of lung fibrosis and inflammation,[16] quantify injury in lung,[17] and model gene expression from histopathology to predict tuberculosis[18] detect and classify tuberculosis lesions.[19] In this work, we develop an AI system for lung tumor segmentation in mouse models that is easy to use for non-computational cancer researchers and will aid lung cancer research in pre-clinical settings.

Methods

Cohort Description

Our dataset consists of a total of 239 high-resolution WSIs of mouse lung histopathology samples (1 mouse/ image) obtained across 9 different experimental cohorts. All tissue samples were obtained from Kras$^{LSL\text{-}G12D/+}$; P53$^{flox/flox}$ (KP) mice as previously described.[20] To induce lung tumors, KP mice were infected with Sftpc-Cre expressing adenovirus or lentiviral vectors co-expressing Cre and specific sgRNAs. All mice, both male and female, were randomized and used in all experiments. Experimental treatment of mice was conducted as reported previously.[6] Lung lobes with tumors and portions of the spleen were fixed in 4% paraformaldehyde and embedded in paraffin. Each slide contains multiple sections of the lung tissue and one section of the spleen from one mouse. Staining was performed following the standard method for H&E stain. H&E stained slides were scanned using Leica Aperio ScanScope AT2 and Hamamatsu NanoZoomer 2.0-RS at an effective magnification of 40×. Resulting images were saved in .svs (44 images) and .ndpi (195 images) file formats.

Mice from 7 experiments ($n = 184$) were split into 75% training, 20% validation, and 5% testing. The remaining 55 images from 2 experiments were entirely held out for testing to ensure no treatment or batch-related effect. This resulted in the following overall breakdown for the analysis: 137 WSI images for training, 37 WSI images for validation, and 65 WSI images for testing.

Image Annotation and Processing

Within each WSI, lung tissue regions and tumor regions were manually outlined as described previously for tumor burden quantification.[6] Annotations were exported to JSON style format using QuPath software[7] (version 0.2.3). Here, each annotation object is labeled as 'Tumor' for cancer-specific tumor foci within the lungs and 'Lung' representing any lung tissue area (cancer or normal). A representative example is shown in Fig. 1.

For each image in training and validation sets, tiles of size 500×500 pixels were extracted at 5× and 10× magnification, reflecting 2 and 1 μM/pixel resolution, respectively, using OpenSlide.[21] Corresponding Lung and Tumor annotations were mapped to each tile using the python library Shapely (version 1.7.1, https://pypi.org/project/Shapely). Binary masks representing 0 (no Tumor) and 1 (Tumor) were used for the segmentation task.

To evaluate the impact of stain variation on model development, stain normalization was performed on each tile using two methods, Macenko[22] and Vahadane[23] within the Staintools python package (version 2.1.2, https://github.com/Peter554/StainTools). For each method, stain matrices were estimated from all tiles in the training cohort to determine median stain matrix and stain concentration vectors after luminosity standardization (Supplemental Table 1). These features were calculated at 5× and 10× magnification levels separately, further details on each method and visual examples in Fig. 1. For each method, these custom stain matrices were then used to normalize all tiles in training and validation sets as a pre-processing step prior to model development.

Model Development

Training

Two architectures were considered for binary segmentation of tumor regions: UNet[24] and DeepLabV3+.[25] ResNet18, ResNet34, and ResNet50

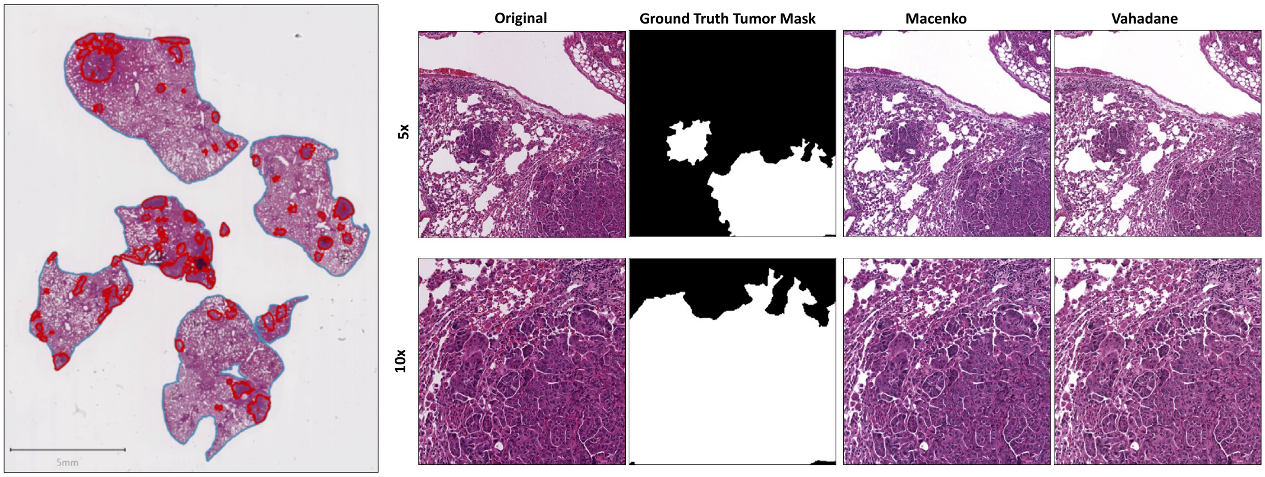

Figure 1. Example WSI and associated image tiles from training set. (Left) WSI with regions of tumor outlined in red and regions of any lung tissue, normal or malignant, outlined in blue. (Right-top) A representative 5× tile extracted from WSI and binary mask converted from expert annotations. The same patch transformed by Macenko and Vahadane are shown. (Right-bottom) A representative 10× tile extracted from WSI, representing the bottom right quadrant of 5× tile, and associated binary masks and normalized features.

backbones were all evaluated.[26] All UNet models were trained using Fast.ai (version 2.2.5).[27] The DeepLabV3 + model was trained using the Semtorch library (version 0.1.1).[28] Additional augmentation to previously described stain normalization included random flipping. Any tile containing Lung tissue with minimum 5% tissue (non-whitespace) area was included in the training. All models were trained using cross-entropy loss and Adam optimization. All models were initialized for one epoch by fine-tuning final layer weights from ImageNet before unfreezing all layers for the remainder of training cycle using discriminative learning rates. Initial learning rate for 5× models was set within 0.00001-0.00005 and for the 10× models was set within 0.00008-0.0001. The selected checkpoint for each model was based on the epoch with the lowest validation loss during training.

Inference

The model inference was performed on WSI input for validation and hold-out testing sets. Here, WSI was loaded using the OpenSlide library, and predictions were obtained on-the-fly for tiles of size 500 × 500 pixels at the specified model magnification. For each tile prediction, the binary segmentation was converted to a polygon structure using the OpenCV python library (version 4.5.2)[29] before being cast back to original pixel coordinates of WSI acquisition and stored as Shapely polygon. To ensure contiguous polygons across neighboring tiles, 20% stride was used during inference (i.e., 100 pixel overlap between adjacent tiles). Following inference of all tiles, a unary union of all polygon predictions was used to create the final structure set of all tumor regions produced from each model. This structure set was saved in JSON format using the geoJSON library (version 2.5.0, https://pypi.org/project/geojson/). Models and code for inference and retraining based on this study are available at https://github.com/NIH-MIP/WSI_LungTumorSeg.

Table 1
Dataset summary.

Split	WSI	Foci (median/img)	5× tiles	10× tiles
Training	137	15,167 (102.5)	23,644	80,402
Validation	37	5,214 (77)	5,792	20,054
Testing	65	3,958 (108)	--	--

Statistical Analysis

The detection performance was measured at the image (mouse) level and individual tumor (foci) level. The segmentation accuracy within each image was measured with the Sørensen–Dice coefficient (Dice), the intersection over union (IoU), and volume similarity (VS) based on standard definitions.[30] The foci-level detection performance was determined by the number of true positives (TP), false positives (FP), and false negatives (FN) compared to the expert ground-truth for calculation of Sensitivity and positive-predictive value (PPV). Here, a true positive is defined as a ground truth tumor region that is correctly identified (i.e., any overlap) with AI-predicted foci. Performance metrics were reported for all models. All results were reported separately for validation and testing datasets. The best model was defined as the model with the highest average Dice score in the validation set.

After selection of the best model, detection performance by foci area (μM^2) was characterized in the training set using receiver operating characteristic (ROC) curve analysis to determine the optimal area cut-point for reduction of false positives in AI-predicted foci using the Youden Index. Detection Sensitivity and number of FPs/image as a function of AI-predicted foci area were analyzed using the free-response operating characteristic (FROC) curve in training and validation sets. Agreement in total tumor area (i.e., burden) between expert annotation and AI was assessed using Bland-Altman analysis (BlandR package, R, version 0.5.3, https://github.com/deepankardatta/blandr).

Results

Summary statistics of the study cohort are shown in Table 1. In total, 29,463 image patches were used for training + validation in models at

Table 2
Model performance metrics.

Mag	Arch	Norm	Validation set						Testing set				
			Tile Dice$^{\pm}$	Dice*	IoU*	Sens	PPV	FP/img	Dice*	IoU*	Sens	PPV	FP/img
5	unet-resnet34	--	0.874	0.874 (0.054)	0.781 (0.078)	0.910	0.232	233	0.846 (0.165)	0.758 (0.177)	0.939	0.254	162
5	unet-resnet18	--	0.872	0.872 (0.057)	0.777 (0.081)	0.911	0.208	315	0.844 (0.162)	0.754 (0.174)	0.941	0.190	267
5	deeplabv3-resnet18	--	0.884	0.875 (0.056	0.781 (0.078	0.908	0.320	174	0.829 (0.164	0.732 (0.177	0.930	0.241	185
5	deeplabv3-resnet34	--	0.883	0.879 (0.052	0.787 (0.073	0.899	0.381	145	0.847 (0.170	0.760 (0.176	0.919	0.402	90.5
5	**deeplabv3-resnet50**	**--**	**0.891**	**0.890 (0.052)**	**0.805 (0.075)**	**0.913**	**0.510**	**75**	**0.873 (0.156)**	**0.797 (0.167)**	**0.937**	**0.514**	**58**
10	unet-resnet18	--	0.879	0.881 (0.051)	0.791 (0.073)	0.929	0.115	598	0.856 (0.151)	0.769 (0.163)	0.940	0.128	427
10	deeplabv3-resnet18	--	0.881	0.881 (0.057	0.791 (0.080	0.908	0.108	509.5	0.854 (0.166	0.770 (0.176	0.943	0.098	471
10	deeplabv3-resnet34	--	0.880	0.881 (0.057)	0.792 (0.082)	0.894	0.255	215	0.864 (0.147)	0.782 (0.161)	0.934	0.237	155
10	deeplabv3-resnet50	--	0.877	0.868 (0.064)	0.771 (0.090)	0.892	0.098	690	0.850 (0.153)	0.760 (0.165)	0.913	0.103	415
5	unet-resnet34	M	0.872	0.862 (0.054)	0.762 (0.077)	0.915	0.216	283	0.849 (0.153)	0.760 (0.170)	0.955	0.196	260
5	unet-resnet18	M	0.871	0.875 (0.052)	0.781 (0.074)	0.911	0.237	275	0.854 (0.153)	0.766 (0.169)	0.949	0.208	263
5	deeplabv3-resnet18	M	0.882	0.876 (0.051	0.783 (0.073	0.939	0.231	289	0.858 (0.149	0.772 (0.165	0.960	0.200	248
5	deeplabv3-resnet34	M	0.881	0.878 (0.064	0.788 (0.088	0.900	0.270	197.5	0.858 (0.153	0.773 (0.164	0.943	0.233	218.5
5	deeplabv3-resnet50	M	0.884	0.883 (0.054)	0.795 (0.078)	0.863	0.525	63	0.852 (0.174)	0.769 (0.182)	0.918	0.442	78
10	unet-resnet18	M	0.875	0.878 (0.054)	0.787 (0.080)	0.921	0.105	697	0.863 (0.155)	0.780 (0.166)	0.953	0.100	535
10	deeplabv3-resnet18	M	0.877	0.875 (0.059	0.782 (0.083	0.899	0.107	539	0.841 (0.175	0.752 (0.180	0.935	0.100	437
10	deeplabv3-resnet34	M	0.869	0.868 (0.064)	0.771 (0.089)	0.872	0.200	251	0.851 (0.174	0.767 (0.179)	0.927	0.171	272
10	deeplabv3-resnet50	M	0.886	0.884 (0.052)	0.795 (0.077)	0.917	0.227	265	0.871 (0.157)	0.794 (0.165)	0.934	0.245	187
5	unet-resnet34	V	0.872	0.870 (0.056)	0.773 (0.079)	0.912	0.217	265	0.854 (0.155)	0.768 (0.169)	0.948	0.212	234
5	unet-resnet18	V	0.871	0.865 (0.051)	0.766 (0.074)	0.925	0.195	321	0.849 (0.154)	0.760 (0.171)	0.959	0.177	308
5	deeplabv3-resnet18	V	0.881	0.878 (0.053	0.786 (0.078	0.931	0.344	161.5	0.845 (0.158	0.754 (0.171	0.950	0.334	131
5	deeplabv3-resnet34	V	0.877	0.879 (0.056	0.788 (0.080	0.906	0.436	105	0.868 (0.142	0.786 (0.158	0.942	0.405	94
5	deeplabv3-resnet50	V	0.885	0.888 (0.054)	0.802 (0.078)	0.905	0.447	100	0.873 (0.147)	0.795 (0.161)	0.936	0.420	90
10	unet-resnet18	V	0.876	0.880 (0.0572	0.789 (0.075)	0.918	0.108	569	0.862 (0.153)	0.778 (0.165)	0.951	0.107	519
10	deeplabv3-resnet18	V	0.880	0.884 (0.054	0.796 (0.079	0.929	0.182	325.5	0.869 (0.149	0.789 (0.163	0.951	0.177	241
10	deeplabv3-resnet34	V	0.881	0.881 (0.057)	0.792 (0.082)	0.894	0.255	215	0.872 (0.140)	0.792 (0.154)	0.934	0.239	195
10	deeplabv3-resnet50	V	0.872	0.881 (0.057)	0.791 (0.082)	0.903	0.278	214	0.869 (0.159)	0.791 (0.163)	0.929	0.251	191

Mag = Magnification. Arch = Architecture. Norm = Stain Normalization Strategy (none, M = Macenko, V = Vahadane). IoU = Intersection over Union. Sens = Sensitivity, PPV = Positive Predictive Value, FP/img = median number of False Positives per image. $^{\pm}$ Tile Dice calculated as mean Dice from all validation tiles. *Dice and IoU reported as mean(stdev) for all WSI.

5× optical equivalent magnification, compared to 100,456 patches in models at 10× optical equivalent magnification. Performance metrics for each of the trained models are presented in Table 2. The best model during patch-based training was found to be the DeepLabV3+ at 5× magnification without the use of stain normalization, achieving patch-based Dice of 0.891 on the validation set.

At WSI inference and conversion, Dice for the 5× DeepLabV3+ remained the best at 0.890 in the validation set, with 91.3% sensitivity and a median 75 false positives/image (Table 2). The reason for the slight discrepancy can be explained due to sliding window (20% stride) during and inclusion of the entire image (i.e., including non-lung structures) for WSI evaluation, reflecting a real-world inference situation. The performance of this model on the unseen test set was found to be 0.873 Dice at 93.7% sensitivity and a median 58 false positives/image (Table 2). Representative examples of best and worst Dice outcomes using the 5× DeepLabV3+ model are shown in Figs. 2 and 3, respectively. No differences in performance were observed between scanners. Only one case in the validation and testing set did not demonstrate any tumor foci on ground truth annotations, with AI producing one false positive in the image (Fig. 4).

Similar to the non-normalized training experiments, the 5× DeepLabV3+ model outperformed UNet at both magnifications and 10× DeepLabV3+ implementation in experiments from each of the stain normalization strategies (Table 2). In general, 10× models performed similarly to 5× counterparts in Dice similarity; however, the 10× models tended to produce a higher number of false positives per image. To evaluate if normalization could boost performance when used during inference (i.e., model was fit from non-normalized images and normalization was applied only at inference), we evaluated the best UNet and DeepLabV3+ non-normalized models with each normalization strategy (Table 3). The result demonstrates increased sensitivity (range 95.7–98.4%) compared to initial models (range 90.0-96.0%); however, this comes at the penalty of increased FP/image and decrease in Dice coefficients in all models.

Qualitative observations of the 5× DeepLabV3+ performance demonstrated the majority of false positives were small in size, as demonstrated in Fig. 4. ROC analysis on AI predictions from the training set determined the optimal threshold to be 12,000 μM^2 for excluding small regions. Figure 5 shows FROC curves for the training and validation sets, using predicted foci size as the risk variable. A reasonable reference comparison would be 400 μM^2 which reflects foci containing <5 tumor cells. Based on the optimal and reference thresholds, the Dice remained unchanged in the validation set (0.890) and increased from 0.873 to 0.887 at the 12,000 μM^2 threshold in the testing set (Table 4). False positives were reduced 10-fold in validation and testing sets; however, this came at the penalty of 6.1% and 4.7% reduction in sensitivity for validation and testing sets, respectively.

Bland-Altman analysis for the error in total tumor burden estimation using the best model is shown in Fig. 6. The bias across validation and testing sets was –0.32 (95% Confidence Interval [CI] –0.95 to 0.30), and the limits-of-agreement lower and upper bounds were –6.53mm^2 (95% CI: –7.60 to –5.46) and 5.88 mm^2 (95% CI: 4.82 to 6.95), respectively.

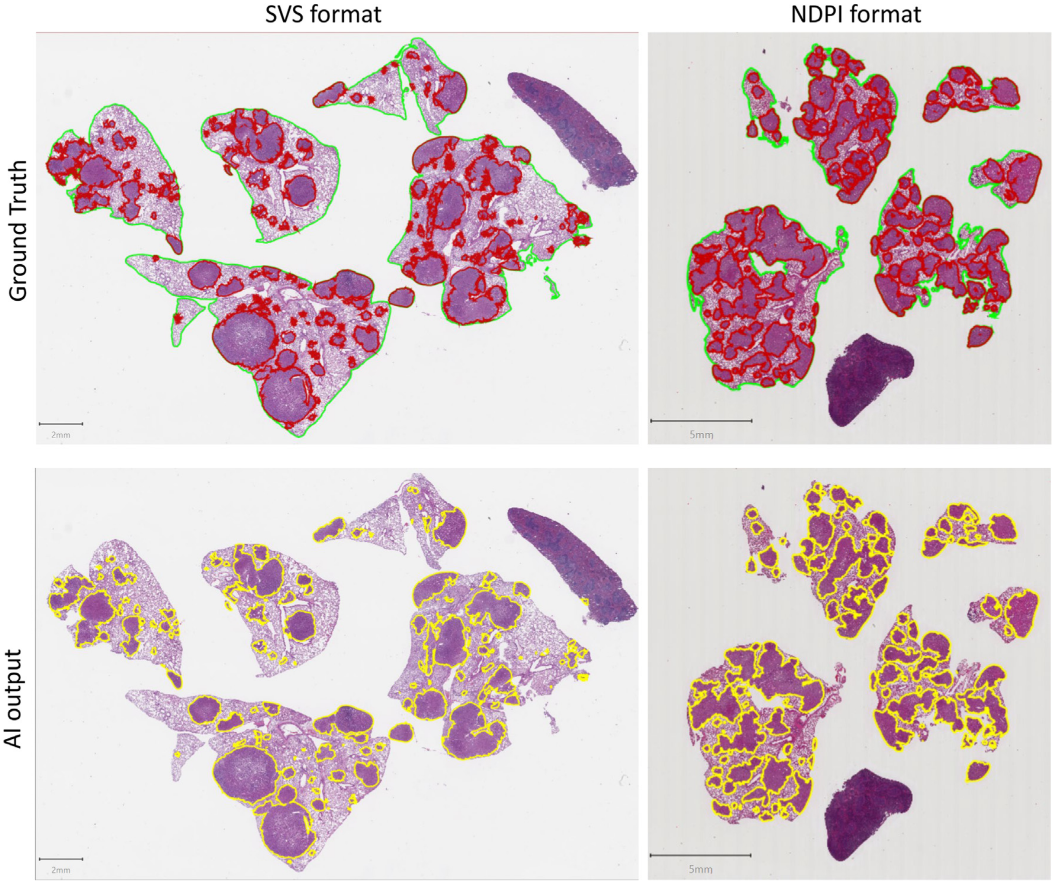

Figure 2. Good Performance Cases for 5× DeepLabV3+ Model. (Left) WSI from Aperio scanner with Dice Coefficient 0.930 from validation set. (Right) WSI from Hamamatsu scanner with Dice Coefficient 0.960 from the test set. For ground-truth annotations, tumor regions are outlined in red and total lung regions are outlined in green. AI outputs are outlined in yellow.

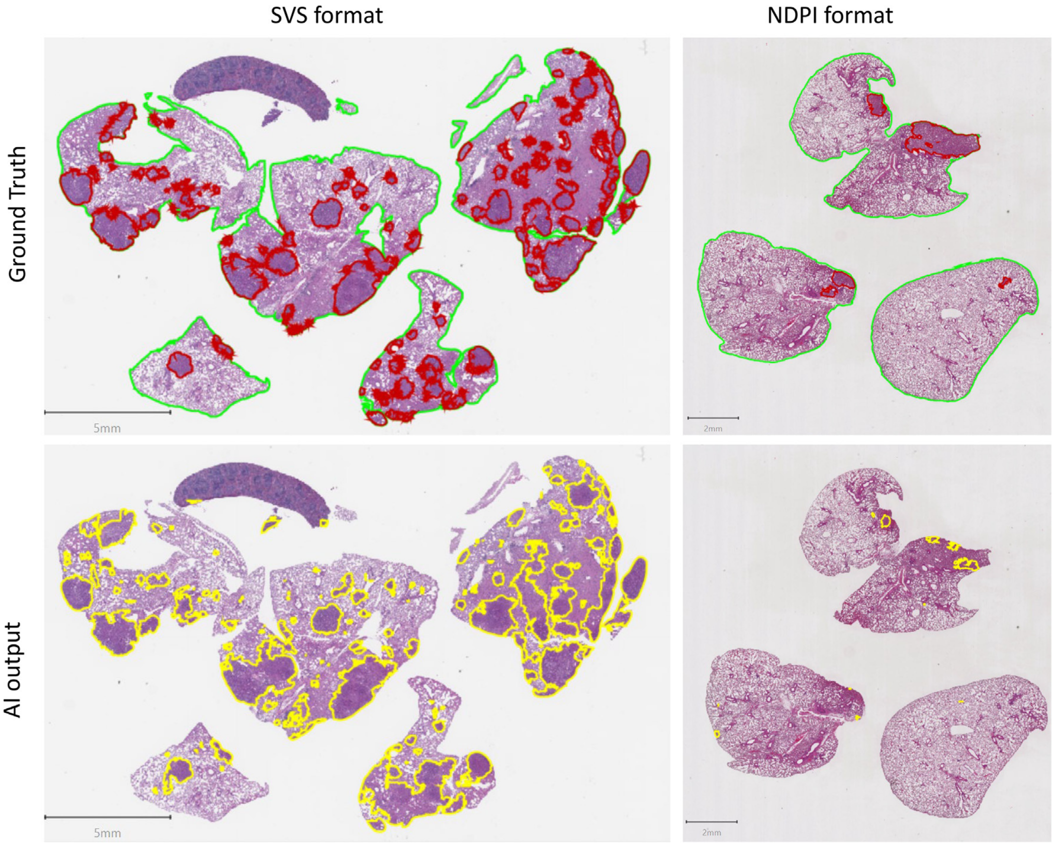

Figure 3. Worst Performance Cases for 5× DeepLabV3+ Model. (Left) WSI from Aperio scanner with Dice Coefficient 0.778 from test set. (Right) WSI from Hamamatsu scanner with Dice Coefficient 0.227 from the test set. For ground truth annotations, tumor regions are outlined in red and total lung regions are outlined in green. AI outputs are outlined in yellow.

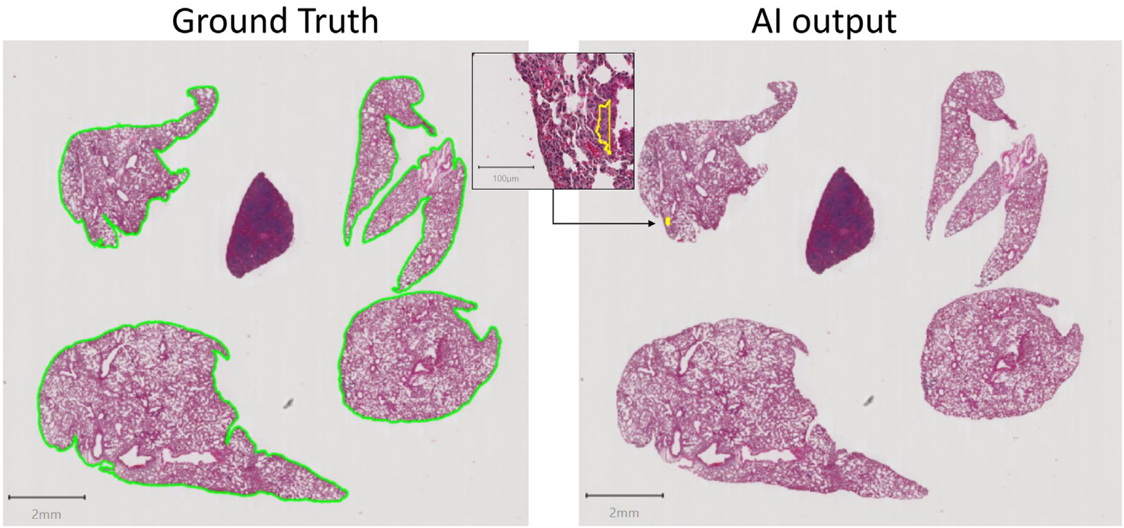

Figure 4. Negative Test Case for 5× DeepLabV3+ Model. (Left) Ground truth annotation demonstrating only lung regions, without the presence of tumor foci. (Right) AI produced a single false positive of approximately 100 μM × 20 μM in size, outlined in yellow.

Table 3
Performance metrics after test-time tile-based stain normalization.

Mag	Arch	Norm	Validation set					Testing set				
			Dice*	IoU*	Sens	PPV	FP/img	Dice*	IoU*	Sens	PPV	FP/img
5	deeplabv3-resnet50	M	0.807 (0.089)	0.685 (0.114)	0.968	0.253	275	0.727 (0.212)	0.606 (0.210)	0.984	0.173	376
10	unet-resnet18	M	0.839 (0.065)	0.727 (0.091)	0.966	0.061	1354	0.779 (0.192)	0.669 (0.198)	0.984	0.041	1878
5	deeplabv3-resnet50	V	0.819 (0.084)	0.702 (0.113)	0.965	0.274	223	0.781 (0.185)	0.669 (0.193)	0.983	0.219	271
10	unet-resnet18	V	0.840 (0.073)	0.73 (0.101)	0.957	0.062	1227.5	0.798 (0.180)	0.691 (0.190)	0.981	0.046	1620

Mag = Magnification. Arch = Architecture. Norm = Stain Normalization Strategy (none, M = Macenko, V = Vahadane). IoU = Intersection over Union. Sens = Sensitivity, PPV = Positive Predictive Value, FP/img = median number of False Positives per image. *Dice and IoU reported as mean (stdev) for all WSI.

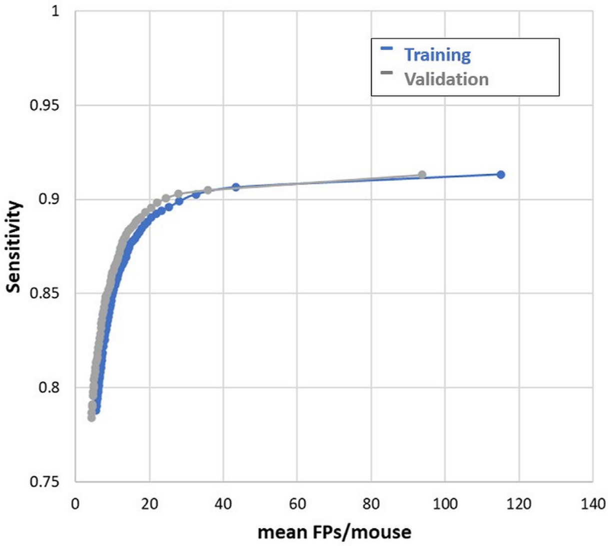

Figure 5. FROC Curve for Training and Validation sets for 5 × DeepLabV3 + Model. Risk is assessed by AI-predicted foci size demonstrating reduction of false positives per image by increasing cut-off threshold (shown in increments of 400 μM^2).

Discussion

Histopathological assessment of tumor burden after experimental treatment conditions is a commonly used endpoint for pre-clinical models.[6] However, accurate measurement of all tumor foci is tedious and error-prone. We have developed an automated AI-based segmentation tool that is able to identify lung adenocarcinoma tumor foci in mouse models with >90% sensitivity in validation and testing cohorts, demonstrating excellent volumetric agreement to ground-truth annotations with 0.890 and 0.873 Dice coefficient, respectively. Furthermore, we have created functionality for this model to output user-friendly file formats that can be read into the publicly available viewer QuPath[7] for further modification or related research analysis.

We evaluated the effect of stain normalization on the quality of AI tumor segmentation. We did not see substantial changes in the performance of any of our models, and the best performing model, 5 × DeepLabV3 + , among all models was observed without the use of stain normalization during training or inference. With the application of the Macenko and Vahadane method on both training and validation/testing data, the mean DC decreased, but only by 1.8% and 0.1%, respectively. Our overall impression was that stain normalization did not improve the results, and the number of false positives increased without meaningful improvement of Dice scores when stain normalization was applied to testing data. A possible explanation is that despite heterogeneity in scanners used during the study, the tissue processing and staining were identical for all animal experiments. Some others reported improved results with Macenko stain normalization (breast cancer classification with EfficientNet,[31] stomach lesions classification with Inception v3[32]), while some reported negative effects on model performance (colon adenocarcinoma segmentation with VGG-19[33]). Validation of our algorithm on an outside dataset could provide a better insight into the effect of stain normalization with the AI models used in this work.

We evaluated model performance at two magnifications, 5 × and 10 × . Within each normalization experiment, all models performed within 2% performance the DeepLabV3 + architecture, but most notably the 10 × models had the highest false-positive rates regardless of the architecture. One possible explanation is 10 × models produce segmentation results at the near-cellular level, leading to a high number of false positives of small sizes. Previous research has shown convolutional neural networks (CNNs) have the ability to learn unique information across various magnifications, resulting in varying magnification selection for different tasks or multimagnification ensemble approaches.[34,35] Within this task, we observed the majority of false positives were substantially smaller in size than ground-truth annotations and could be filtered out using either reasonable expert knowledge or optimal cut-point analysis. These regions were ultimately inconsequential to the focus of this study, i.e., total tumor burden estimation. Downstream analysis, such as the counting of individual tumor cells, may require AI approaches to operate at higher magnification or cascaded approaches in the future.

A major limitation of translation AI research is the development of user-friendly deployment tools or frameworks that can bring AI tools into the hands of users without computational science background.[36] Utilizing the functionality of QuPath to read/write geoJSON files, we have developed a model for which the output can be easily read and modified within the pre-existing software. This enables users to utilize and modify AI-generated output for their research needs. This could additionally serve as an AI-assisted annotation tool for future research evaluating different tasks within adenocarcinoma models, such as classification of disease subtypes or counting of cellular components.

Table 4
Performance metrics after area-based thresholding for 5 × DeepLabV3 + model.

Size threshold (μm^2)	Validation set					Testing set				
	Dice	IoU	Sens	PPV	FP/img	Dice	IoU	Sens	PPV	FP/img
0	0.890 (0.052)	0.805 (0.075)	0.913	0.510	75	0.873 (0.156)	0.797 (0.167)	0.937	0.514	58
400	0.890 (0.052)	0.805 (0.076)	0.905	0.730	34	0.873 (0.156)	0.797 (0.167)	0.933	0.740	22
12000	0.890 (0.053)	0.805 (0.077)	0.852	0.910	7	0.887 (0.111)	0.810 (0.135)	0.890	0.908	5

IoU = Intersection over Union. Sens = Sensitivity, PPV = Positive Predictive Value, FP/img = median number of False Positives per image. *Dice and IoU reported as mean (stdev) for all WSI.

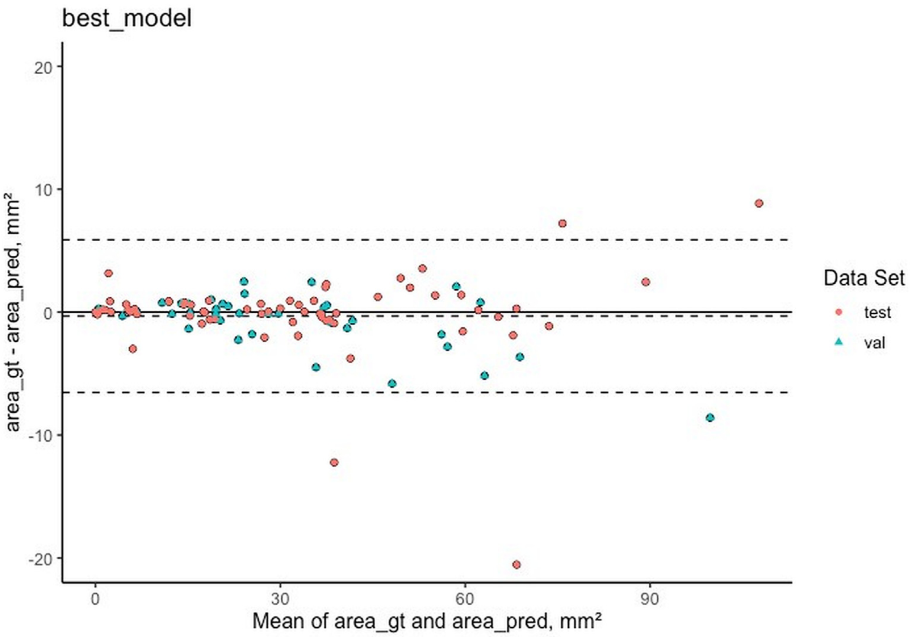

Figure 6. 5× DeepLabV3+ Model Bland-Altman Plot for total tumor burden assessment by Expert vs AI for Validation and Testing datasets.

This work has several limitations. All mice were analyzed under nearly identical experimental and processing conditions, leading to homogeneity in staining profiles across both scanners used in this study. It is well documented that variation in staining conditions or tissue processing artifacts can negatively impact the performance of deep learning models.[37,38] Related, despite controlling for different experimental cohorts of mice, we did not have an external cohort to evaluate the generalizability of these models. Finally, a large number of small false-positive regions may indicate the model will require future fine-tuning for users who wish to capture tumor foci characterized by few-to-several individual cells.

Acknowledgments

This project is in part supported by the Intramural Research Program of the Center for Cancer Research, NCI. CZ is supported in part by SITC-AstraZeneca Immunotherapy in Lung Cancer (Early Stage NSCLC) Clinical Fellowship Award and ASCO Young Investigator Award. CJ and CZ are supported by NIH Bench-to-Bedside and Back Program (BtB). CJ is supported in part by an R00 award from NIH/NCI (CA226400), an Emerson Collective Cancer Research fund, a Lung Cancer Research Foundation (LCRF) pilot grant, and a W.W. Smith Trust Foundation award. CL is supported by Frederick National Labor for Cancer Research/Leidos Grant 17X005. 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 work utilized the computational resources of the NIH HPC Biowulf cluster.

References

1. Sung H, Ferlay J, Siegel RL, et al. Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries [published online ahead of print 2021/02/04]. CA Cancer J Clin 2021;71(3):209–249.
2. Zheng M. Classification and pathology of lung cancer. Surg Oncol Clin N Am 2016;25(3): 447–468.
3. Hynds RE, Frese KK, Pearce DR, Grönroos E, Dive C, Swanton C. Progress towards non-small-cell lung cancer models that represent clinical evolutionary trajectories [published online ahead of print 2021/01/13]. Open Biol 2021;11(1):200247.
4. Jackson EL, Willis N, Mercer K, et al. Analysis of lung tumor initiation and progression using conditional expression of oncogenic K-ras. Genes Dev 2001;15(24):3243–3248.
5. Wang Y, Zhang Z, Lubet RA, You M. A mouse model for tumor progression of lung cancer in ras and p53 transgenic mice. Oncogene 2006;25(8):1277–1280.
6. Jin C, Lagoudas GK, Zhao C, et al. Commensal microbiota promote lung cancer development via γδ T cells [published online ahead of print 2019/01/31]. Cell 2019;176(5): 998-1013.e1016.
7. Bankhead P, Loughrey MB, Fernández JA, et al. QuPath: open source software for digital pathology image analysis [published online ahead of print 2017/12/04]. Sci Rep 2017;7 (1):16878.
8. Della Mea V, Baroni GL, Pilutti D, Di Loreto C. SlideJ: an ImageJ plugin for automated processing of whole slide images [published online ahead of print 2017/07/06]. PLoS One 2017;12(7):e0180540.
9. Mi W, Li J, Guo Y, et al. Deep learning-based multiclass classification of breast digital pathology images [published online ahead of print 2021/06/10]. Cancer Manag Res 2021;13:4605–4617.
10. Iizuka O, Kanavati F, Kato K, Rambeau M, Arihiro K, Tsuneki M. Deep learning models for histopathological classification of gastric and colonic epithelial tumours [published online ahead of print 2020/01/30]. Sci Rep 2020;10(1):1504.
11. Li Z, Zhang J, Tan T, et al. Deep learning methods for lung cancer segmentation in whole-slide histopathology images-the ACDC@LungHP challenge 2019 [published online ahead of print 2021/02/05]. IEEE J Biomed Health Inform 2021;25(2):429–440.
12. Yang H, Chen L, Cheng Z, et al. Deep learning-based six-type classifier for lung cancer and mimics from histopathological whole slide images: a retrospective study [published online ahead of print 2021/03/29]. BMC Med 2021;19(1):80.
13. She Y, Jin Z, Wu J, et al. Development and validation of a deep learning model for non-small cell lung cancer survival [published online ahead of print 2020/06/01]. JAMA Netw Open 2020;3(6):e205842.
14. Coudray N, Ocampo PS, Sakellaropoulos T, et al. Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning [published online ahead of print 2018/09/17]. Nat Med 2018;24(10):1559–1567.
15. Šarić M, Russo M, Stella M, Sikora M. CNN-based method for lung cancer detection in whole slide histopathology images. Paper presented at: 2019 4th International Conference on Smart and Sustainable Technologies (SpliTech); 18-21 June 2019; 2019.
16. Heinemann F, Birk G, Schoenberger T, Stierstorfer B. Deep neural network based histological scoring of lung fibrosis and inflammation in the mouse model system [published online ahead of print 2018/08/23]. PLoS One 2018;13(8):e0202708.
17. Salsabili S, Lithopoulos M, Sreeraman S, et al. Fully automated estimation of the mean linear intercept in histopathology images of mouse lung tissue [published online ahead of print 2021/03/04]. J Med Imaging (Bellingham) 2021;8(2):027501.
18. Tavolara TE, Niazi MKK, Gower AC, Ginese M, Beamer G, Gurcan MN. Deep learning predicts gene expression as an intermediate data modality to identify susceptibility patterns in *Mycobacterium tuberculosis* infected Diversity Outbred mice [published online ahead of print 2021/05/14]. EBioMedicine 2021;67:103388.
19. Asay BC, Edwards BB, Andrews J, et al. Digital image analysis of heterogeneous tuberculosis pulmonary pathology in non-clinical animal models using deep convolutional neural networks [published online ahead of print 2020/04/08]. Sci Rep 2020;10(1):6047.
20. DuPage M, Dooley AL, Jacks T. Conditional mouse lung cancer models using adenoviral or lentiviral delivery of Cre recombinase [published online ahead of print 2009/06/25]. Nat Protoc 2009;4(7):1064–1072.

21. Goode A, Gilbert B, Harkes J, Jukic D, Satyanarayanan M. OpenSlide: a vendor-neutral software foundation for digital pathology [published online ahead of print 2013/09/27]. J Pathol Inform 2013;4:27.
22. Macenko M, Niethammer M, Marron JS, et al. A method for normalizing histology slides for quantitative analysis. Proceedings of the Sixth IEEE international conference on Symposium on Biomedical Imaging: From Nano to Macro; 2009.Boston, Massachusetts, USA.
23. Vahadane A, Peng T, Sethi A, et al. Structure-preserving color normalization and sparse stain separation for histological images [published online ahead of print 2016/04/27]. IEEE Trans Med Imaging 2016;35(8):1962–1971.
24. Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation. Paper presented at: International Conference on Medical image computing and computer-assisted intervention; 2015.
25. Chen L-C, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-decoder with atrous separable convolution for semantic image segmentation. Paper presented at: Proceedings of the European conference on computer vision (ECCV) 2018.
26. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. Paper presented at: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016.
27. Howard J, Gugger S. Fastai: a layered API for deep learning. Information 2020;11(2):108.
28. Lacalle D, Castro-Abril HA, Randelovic T, et al. SpheroidJ: an open-source set of tools for spheroid segmentation. Comput Methods Programs Biomed 2021;200, 105837.
29. Bradski G, Kaehler A. *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc. 2008.
30. Taha AA, Hanbury A. Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool [published online ahead of print 2015/08/12]. BMC Med Imaging 2015;15:29.
31. Munien C, Viriri S. classification of hematoxylin and eosin-stained breast cancer histology microscopy images using transfer learning with EfficientNets [published online ahead of print 2021/04/09]. Comput Intell Neurosci 2021;2021:5580914.
32. Ma B, Guo Y, Hu W, et al. Artificial intelligence-based multiclass classification of benign or malignant mucosal lesions of the stomach [published online ahead of print 2020/10/02]. Front Pharmacol 2020;11:572372.
33. Jiao Y, Li J, Qian C, Fei S. Deep learning-based tumor microenvironment analysis in colon adenocarcinoma histopathological whole-slide images [published online ahead of print 2021/03/12]. Comput Methods Programs Biomed 2021;204:106047.
34. BenTaieb A, Li-Chang H, Huntsman D, Hamarneh G. A structured latent model for ovarian carcinoma subtyping from histopathology slides [published online ahead of print 2017/05/09]. Med Image Anal 2017;39:194–205.
35. Kuklyte J, Fitzgerald J, Nelissen S, et al. Evaluation of the use of single- and multi-magnification convolutional neural networks for the determination and quantitation of lesions in nonclinical pathology studies [published online ahead of print 2021/02/23]. Toxicol Pathol 2021;49(4):815–842.
36. Tizhoosh HR, Pantanowitz L. Artificial intelligence and digital pathology: challenges and opportunities [published online ahead of print 2018/11/14]. J Pathol Inform 2018;9:38.
37. Schömig-Markiefka B, Pryalukhin A, Hulla W, et al. Quality control stress test for deep learning-based diagnostic model in digital pathology. Mod Pathol 2021;34:2098–2108.
38. Ciompi F, Geessink O, Bejnordi BE, et al. The importance of stain normalization in colorectal tissue classification with convolutional networks. Paper presented at: 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017); 2017.