Detecting Cancerous Cells in Gigapixel Images

COMS 4995 - Applied Deep Learning

Akarsh Zingade (auz2000)
Kiran Ramesh (kr2789)
Arjun D’Cunha (ad3545)

Agenda

1. Summary of the Work
2. Motivation
3. Data
4. Method - Source Paper
5. Implementation - Patch Extraction
6. Implementation - Training
7. Implementation - Heatmap Generation
8. Results
9. Future Work

Summary of the work

1) Architectures used: Inceptionv3, VGG19, ResNet50, Custom 10 layer
2) Loss used: Binary Cross entropy loss, Focal Loss
3) Sliding window size used: 299x299, 150x150
4) Slide level used: 0,1,4,5
5) Model type used: Single Scale model, Multi Scale model

Motivation

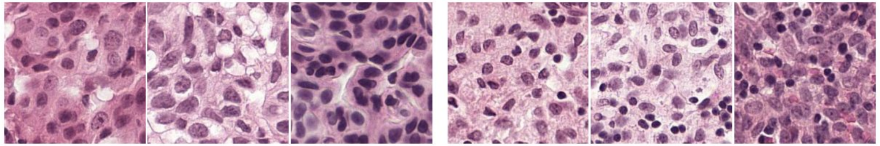
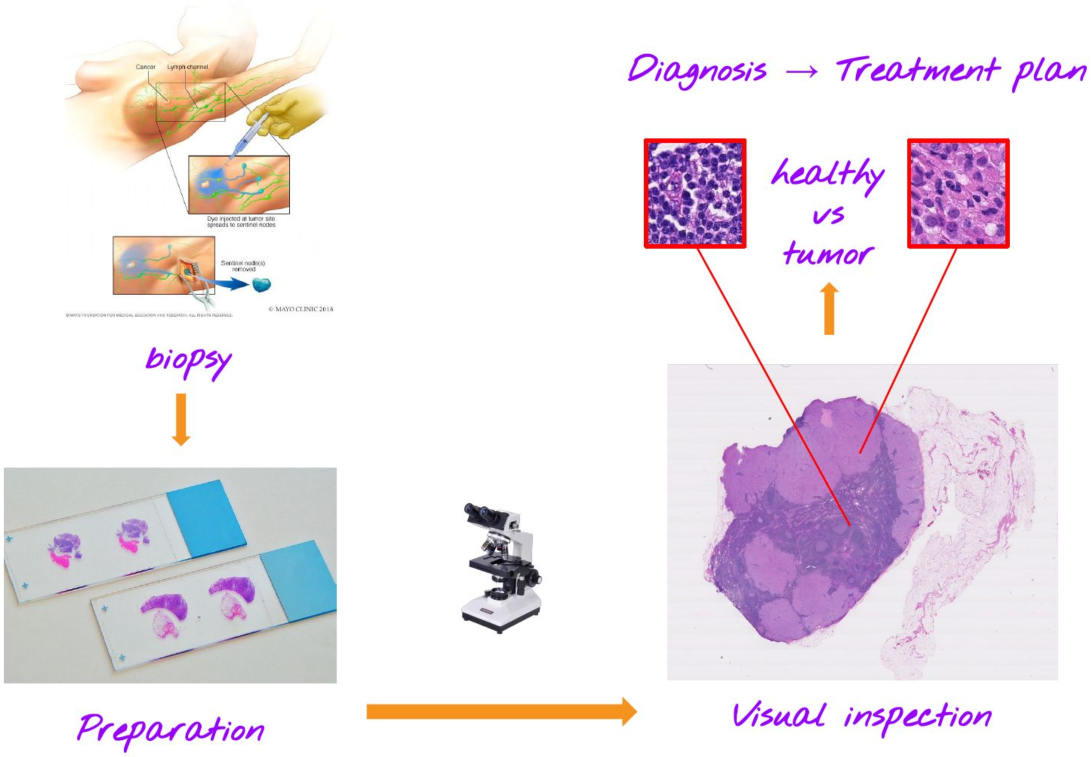


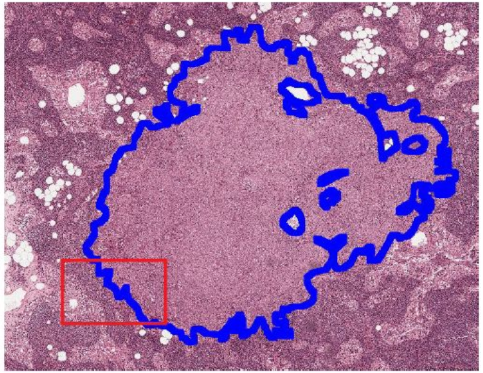
Fig. 1. Left: three tumor patches and **right**: three challenging normal patches.

Goal :- Help reduce misdiagnosis by developing a model which can be inserted into the workflow as an automatic second opinion (if reads differ, flag and ask for second pathologist)



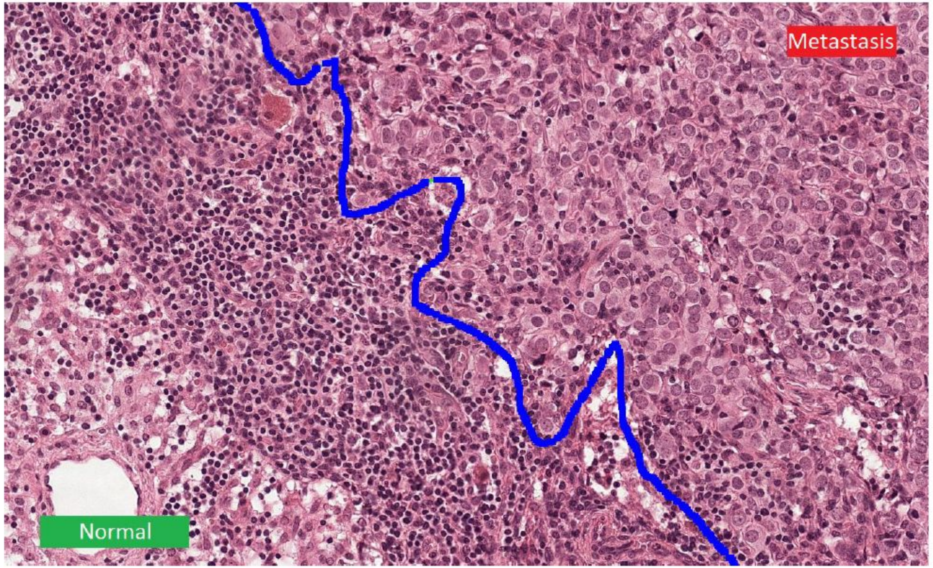
Taken from Joshua Gordon’s Applied Deep Learning Lecture 5, Columbia University

Data - CAMELYON16 Challenge



400 WSI (whole slide images) collected independently from two medical centers in the Netherlands.

- Slide level annotations.
- Importantly, licensed under CC0.
- About 600GB.

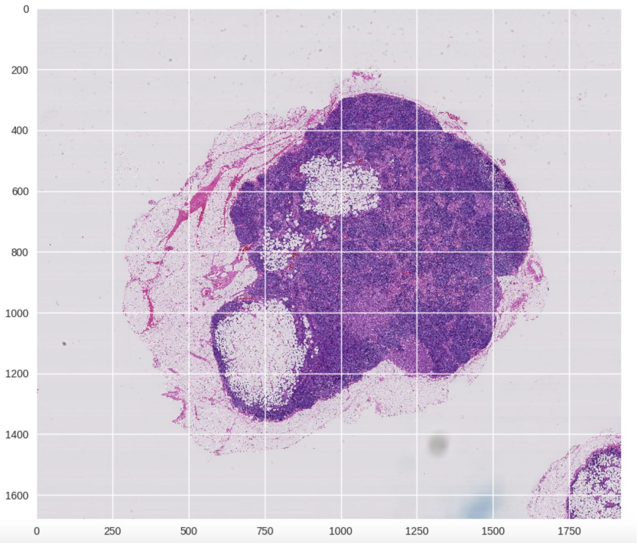


Data provided by Prof. Joshua Gordon via Google Drive

Taken from Joshua Gordon's Applied Deep Learning Lecture 5, Columbia University

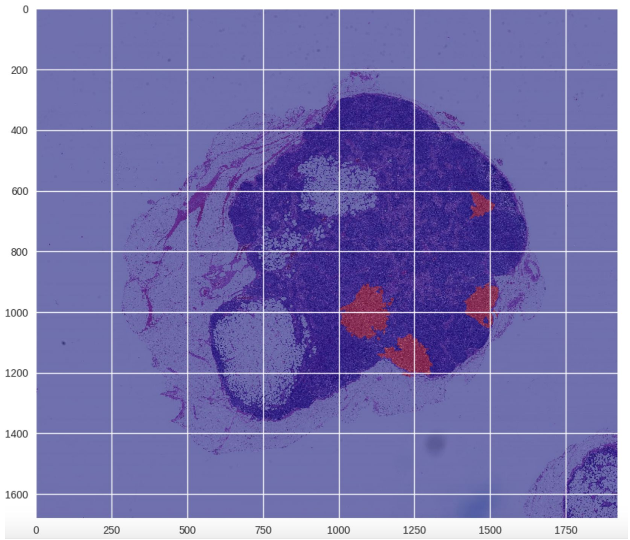
https://camelyon16.grand-challenge.org/

Data - Slide Example



Slide

Mask



Overlay

Method - Source Paper

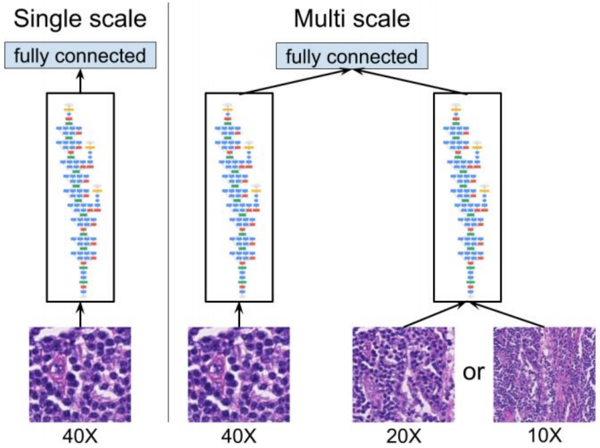


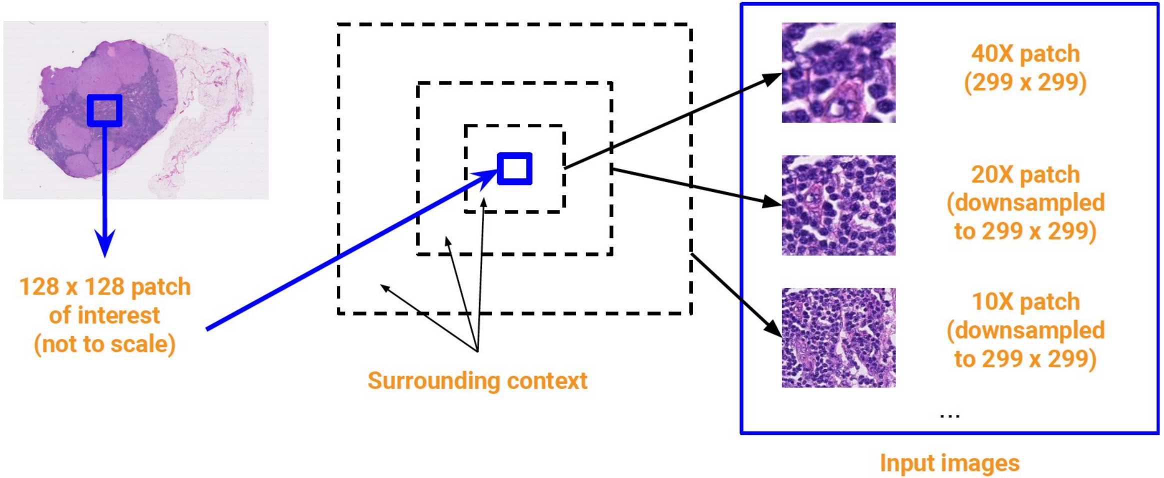
Fig. 3. The three colorful blocks represent Inception (V3) towers up to the second-last layer (PreLogit). *Single scale* utilizes one tower with input images at 40X magnification; *multi-scale* utilizes multiple (*e.g.*,2) input magnifications that are input to separate towers and merged.

Detecting Cancer Metastases on Gigapixel Pathology Images

Yun Liu[1*], Krishna Gadepalli[1], Mohammad Norouzi[1], George E. Dahl[1], Timo Kohlberger[1], Aleksey Boyko[1], Subhashini Venugopalan[2**], Aleksei Timofeev[2], Philip Q. Nelson[2], Greg S. Corrado[1], Jason D. Hipp[3], Lily Peng[1], and Martin C. Stumpe[1]

{liuyun,mnorouzi,gdahl,lhpeng,mstumpe}@google.com

[1]Google Brain, [2]Google Inc, [3]Verily Life Sciences, Mountain View, CA, USA



https://arxiv.org/abs/1703.02442

Patch Extraction

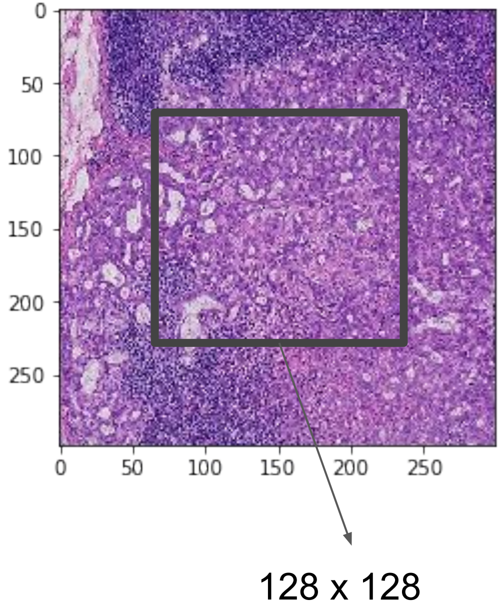
1) Sliding window of size 299x299 swept over the image.

2) The sliding window is strided at 80 pixels

3) Patches with at least 1 cancerous pixel in the center 128x128 of the patch is labelled as cancerous patch

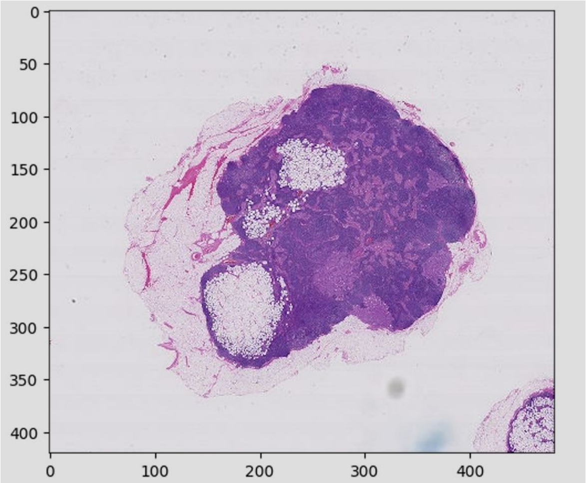
4) Discard the patch if it has less than 50% of the pixels are tissue.

5) To balance the dataset, we randomly sample 200 patches per class per slide.

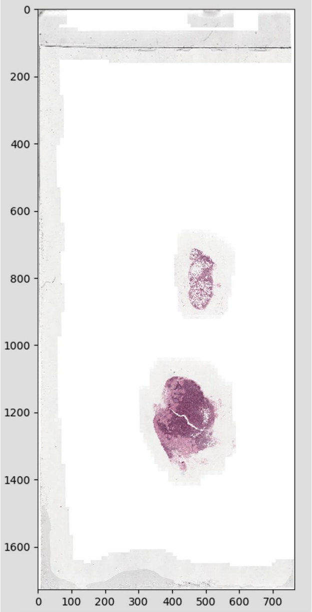


Training and Testing Slides

Slides used for training and validation : 031, 064, 075, 084, 091, 094, 096, 101
Slides used for testing : 016, 078, 110



Slide 091



Slide 031

Training Model - Preprocessing

We use similar preprocessing used for the pretrained models.

For VGG19: Rescale the input image by 1/255.

For InceptionV3: Rescale by 1/255., subtract by 0.5 and multiply by 2

For ResNet50: Rescale by 1/255., subtract by 0.5 and multiply by 2

Training Model - Data Augmentation

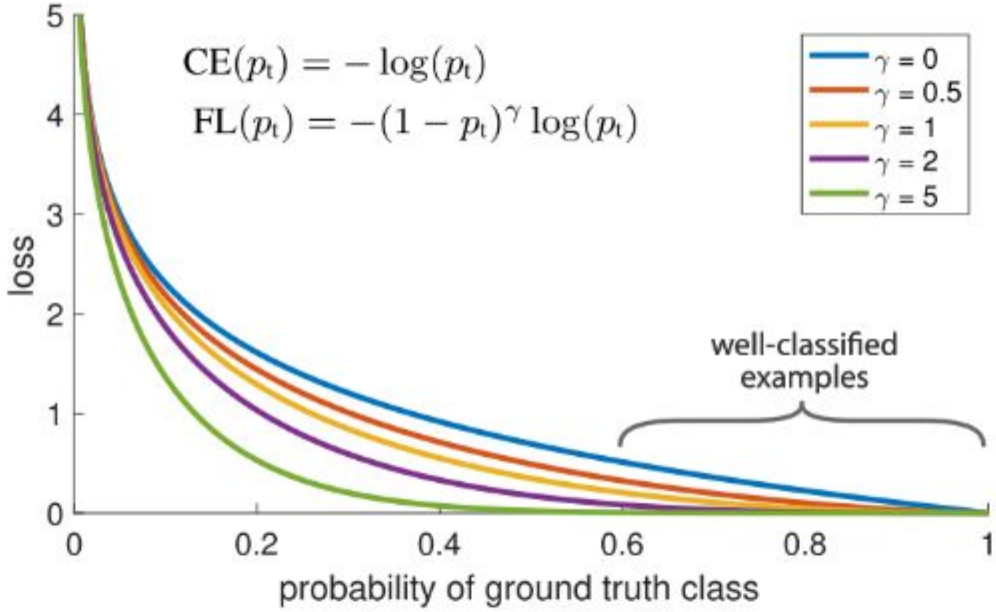
Data Augmentation:

1) Vertical and Horizontal flipping
2) Rotate it by 90 or -90 degrees
3) Change illumination of the image (did not work)

Training Model - Transfer Learning

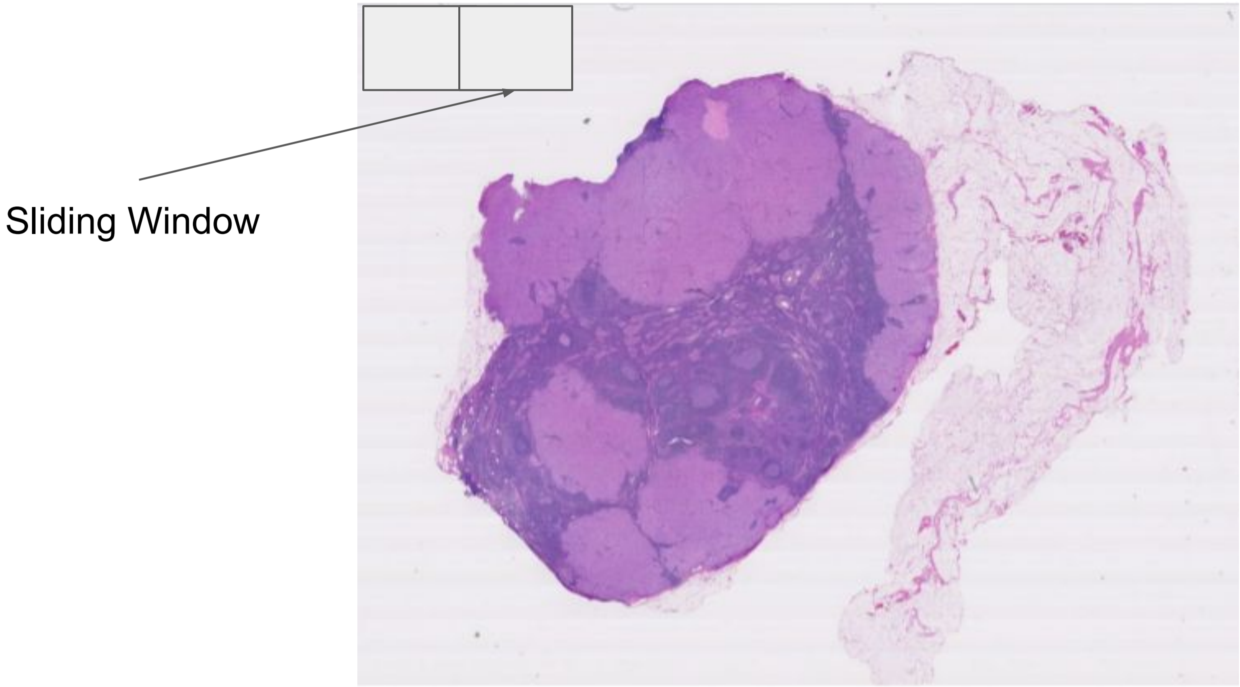
1) Load the ImageNet pretrained weights without the top layers
2) Add Dense and Dropout layers.
3) Freeze the pretrained layers.
4) Train the model with Adam optimizer
5) Unfreeze all or some of the top layers of the ImageNet pretrained layers
6) Train the model with SGD optimizer with a learning rate of 0.0001

Focal Loss



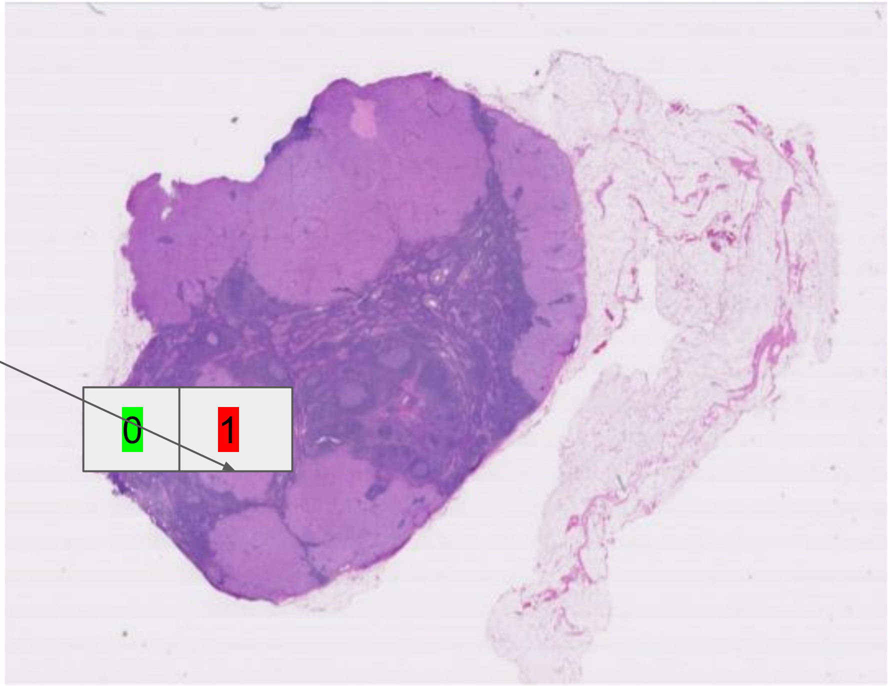
$\mathrm{CE}(p_t) = -\log(p_t)$
$\mathrm{FL}(p_t) = -(1-p_t)^\gamma \log(p_t)$
γ = 0
γ = 0.5
γ = 1
γ = 2
γ = 5
well-classified examples
loss
probability of ground truth class
0
0.2
0.4
0.6
0.8
1
1
2
3
4
5

Heatmap Generation

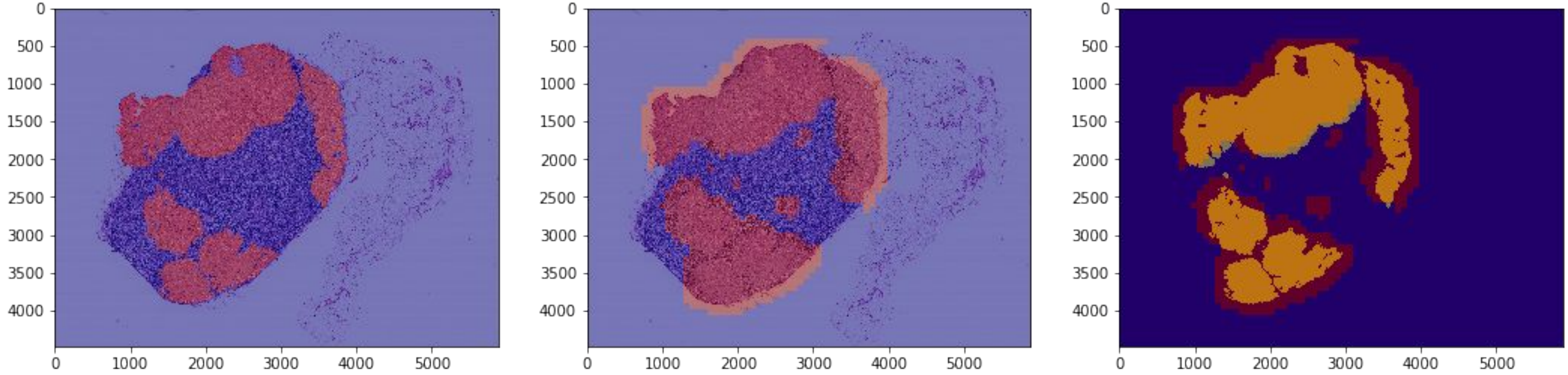


Heatmap Generation

Sliding Window



Example Heatmap



Experimental Results

Model Type	Conv Base	Dense Layer	Optimizer	Loss	Zoom Level	Accuracy	Precision	Recall	F1	AUC
Single	InceptionV 3	256 neurons	Adam	Binary Crossentropy	4	72.5	0.411	0.553	0.469	0.746
	ResNet50	256 neurons	Adam	Binary Crossentropy	4	95.7	0.477	0.955	0.630	0.934
	VGG19	256 neurons	Adam	Binary Crossentropy	4	98.1	0.530	0.938	0.674	0.935
	VGG19	256 neurons	Adam	Focal Loss	4	98.7	0.536	0.950	0.675	0.939
	VGG19	256 neurons	Adam	Binary Crossentropy	0	98.1	0.513	0.943	0.645	0.917
Multi	Inception V3	128 neurons	SGD	Binary Crossentropy	4, 5	99.0	0.638	0.629	0.634	0.781
	Inception V3	128 neurons	SGD	Binary Crossentropy	0, 1	98.2	0.621	0.597	0.614	0.769

Future Work

1) Pretrain each of the model in multi-scale with our dataset and then jointly train them together.
2) Try lower dimension sliding window such as 32x32 or 64x64.
3) Try non-consecutive resolution such as level 4 and level 0.
4) Try focal loss on multi-scale model

Thank you