DEEP LEARNING BASED CANCER METASTASES DETECTION

DAYONG WANG PHD[1]

ADITYA KHOSLA PHD[2]

RISHAB GARGEYA[1]

HUMAYUN IRSHAD PHD[1]

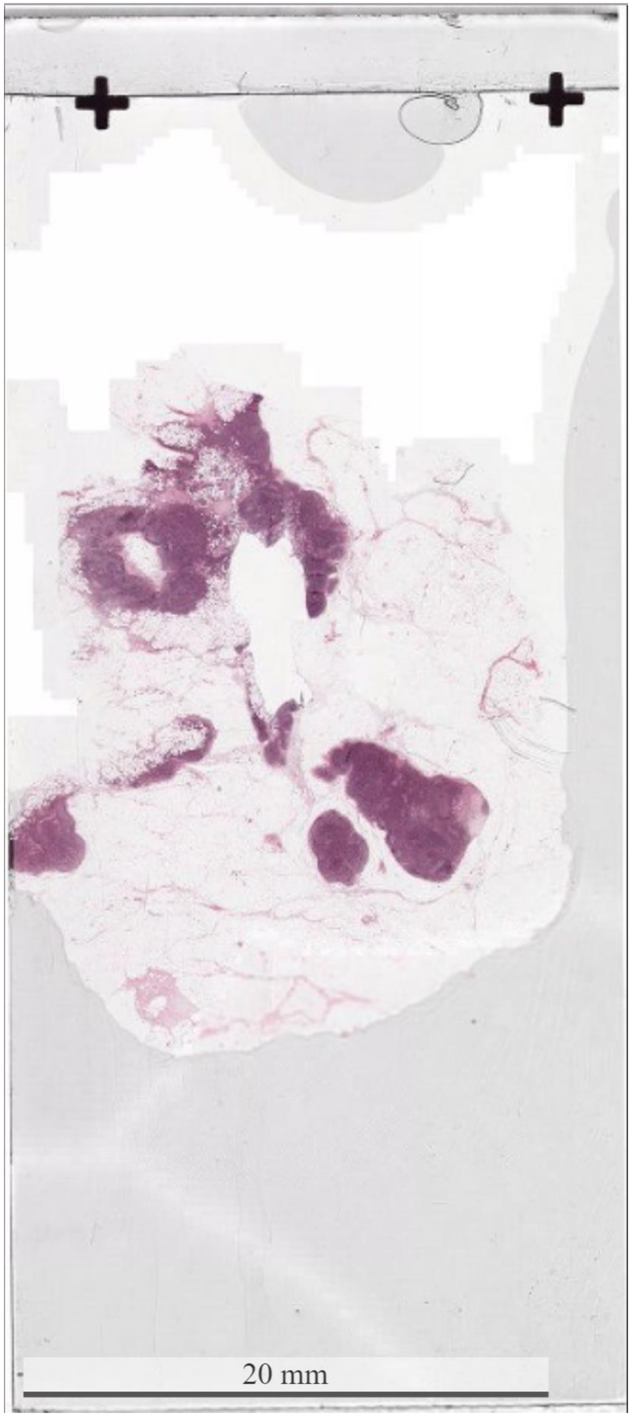
ANDREW H BECK MD PHD[1]

[1] Department of Pathology, Harvard Medical School and Beth Israel Deaconess Medical Center

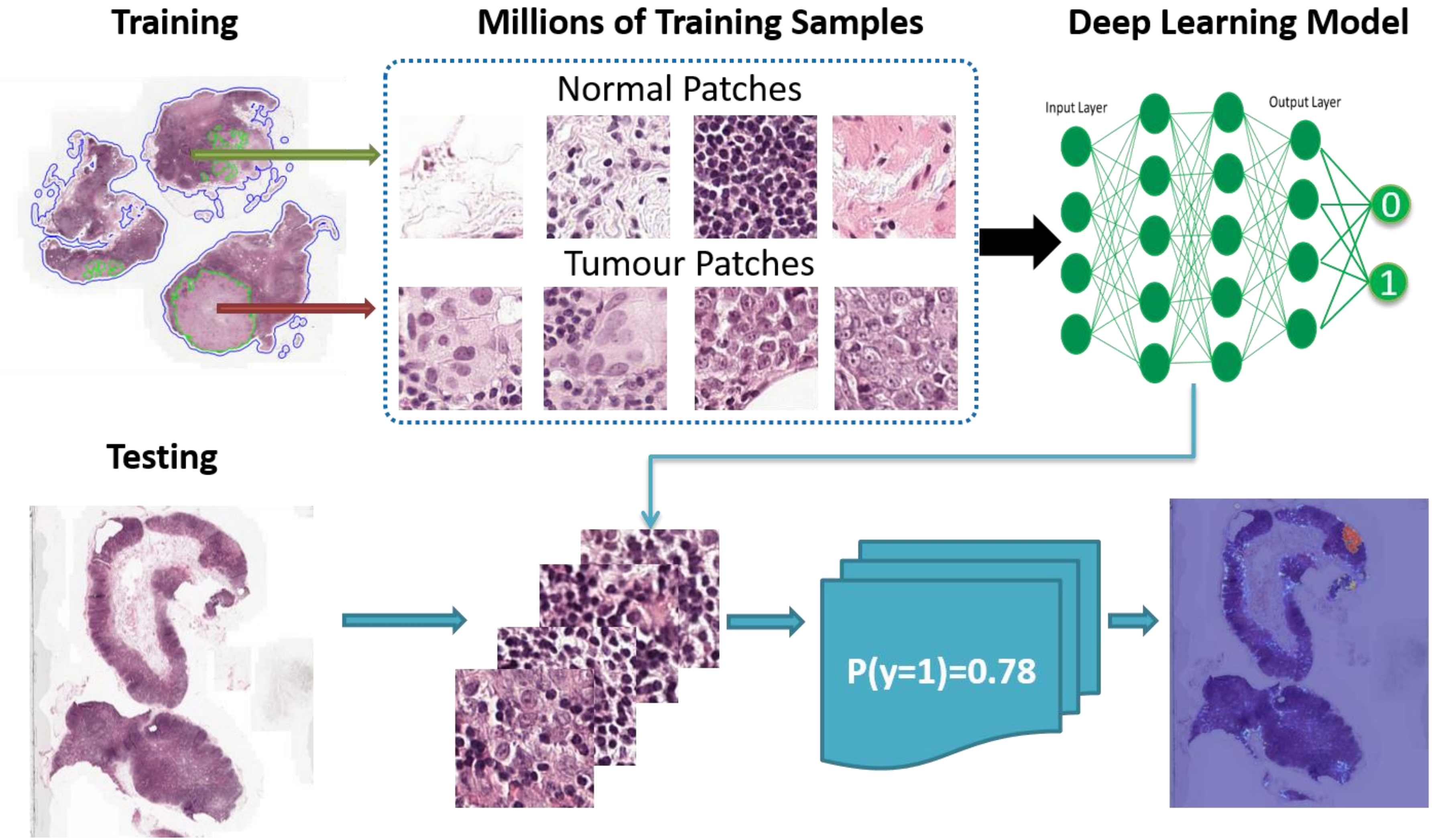
[2] MIT Computer Science and Artificial Intelligence Laboratory

TASKS

- Training and Evaluation
  - 110 tumor slides
  - 160 normal slides
  - 130 evaluation slides
- 1st Task
  - Whole slide level prediction
  - *Binary classification problem*
- 2nd Task
  - Find metastasis location
  - *Segmentation problem*



SYSTEM FRAMEWORK



Training

Millions of Training Samples

Deep Learning Model

Normal Patches

Tumour Patches

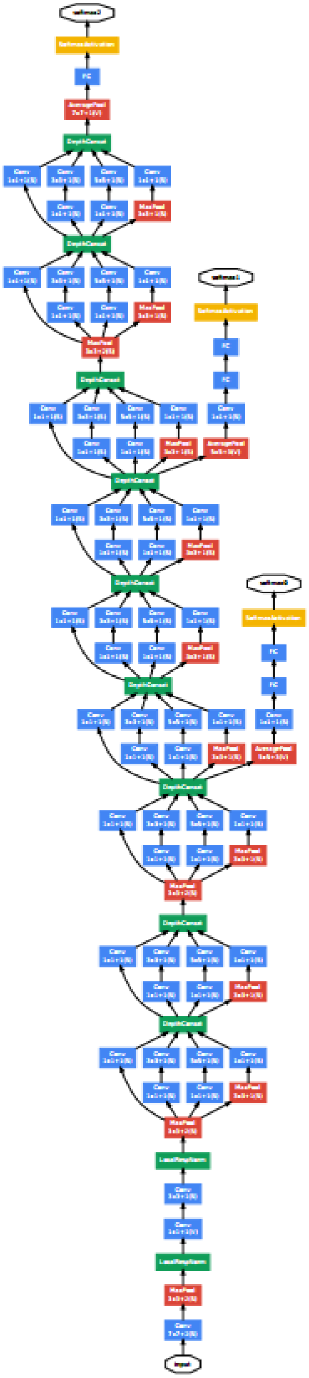
Testing

P(y=1)=0.78

SEVERAL ESSENTIAL COMPONENTS

- Network architecture
- Training set construction
- Computing environment
- Post-processing for classification and segmentation

NETWORK ARCHITECTURE DESIGN

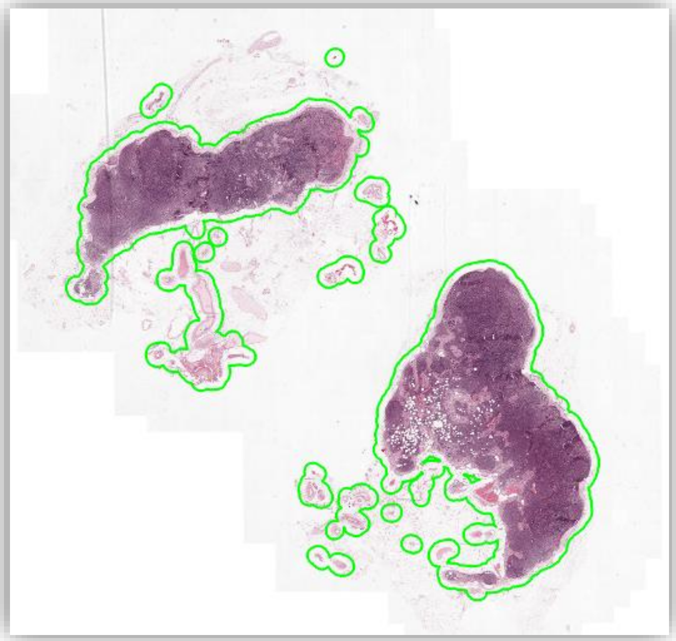


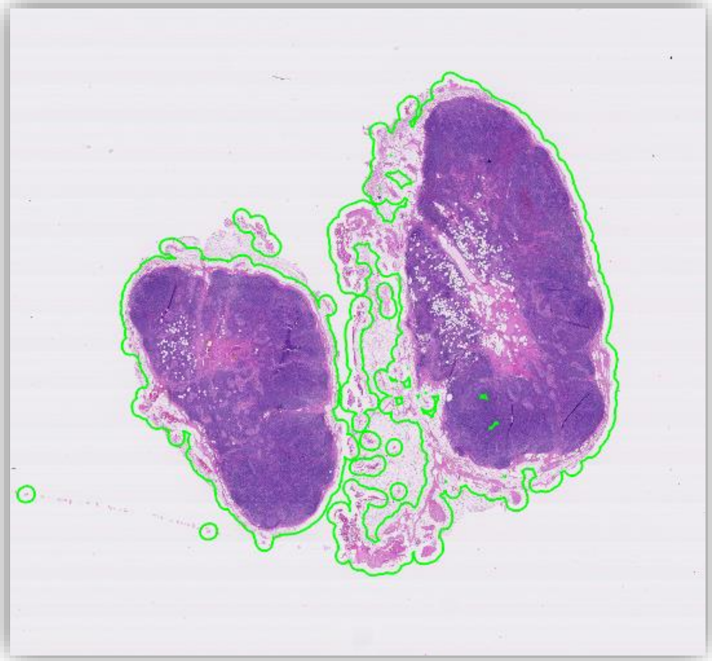
- We compared several networks
  - **GoogLeNet (Szegedy et al. ILSCV 2014): 98.4%** 🙂
  - VGG16 (Simonyan and Zisserman): 97.9%
  - FaceNet (Wang et al. 2015): 96.8%
  - AlexNet (Krizhevsky et al. NIPS 2012): 92.1%

- Details of GoogLeNet
  - 27 layers in total
  - ~6 million parameters
  - three loss layers
  - Christian Szegedy et al. Going Deeper with Convolutions

TRAINING SET CONSTRUCTION

- Preprocessing
  - Tissue region segmentation (Otsu's method of foreground segmentation)
  - Remove 82% of WSI region on average



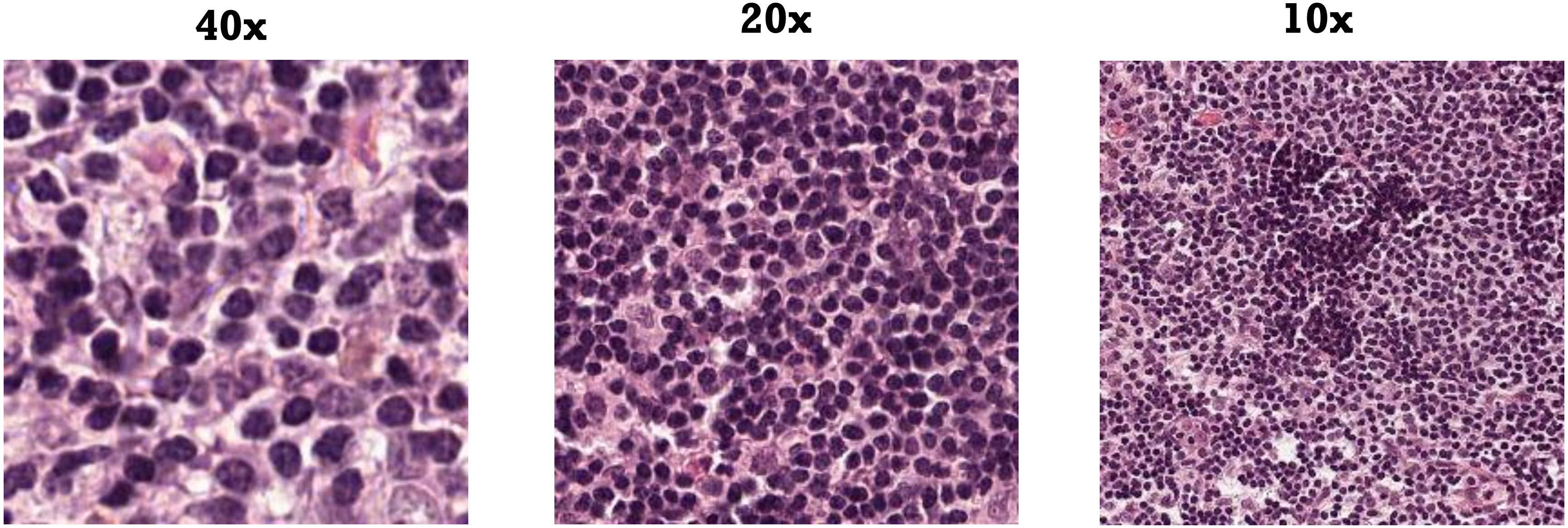


TRAINING SET CONSTRUCTION

- Step 1:
  - Randomly extract patches (256 x 256) on the tissue region
    - Tumor slide : 1K positive and 1K negative from each slide
    - Normal slide: 1K negative from each slide
    - ~290K training patches
- Step 2:
  - Make predictions and construct heatmaps
  - Extract additional ~60K training patches from false positive regions
  - 290K + 60K = 350K training patches in total

PATCH EXTRACTION AT 40X

- We evaluate performance with patch extraction at several magnifications
  - Experimental results indicate that 40x is the best

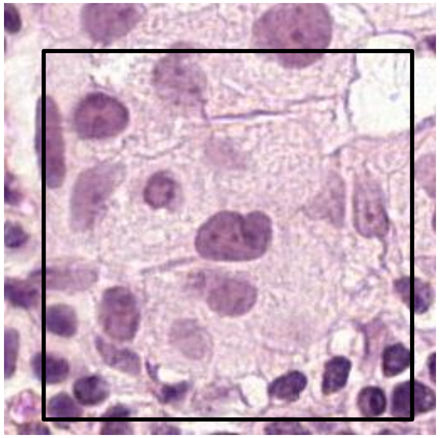


Example patches of size 256 x 256 with 40x, 20x and 10x magnification

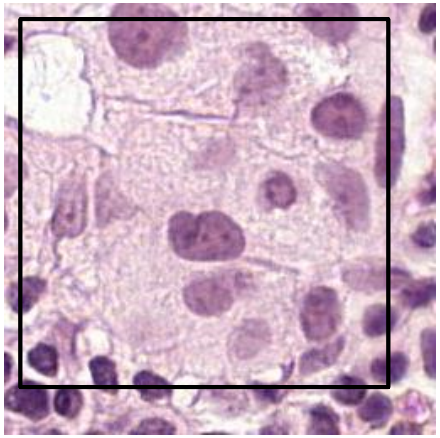
Normal Slide, ID : 001

DATA AUGMENTATION

- Randomly crop a 224 x 224 sub-region and flip patches horizontally



or

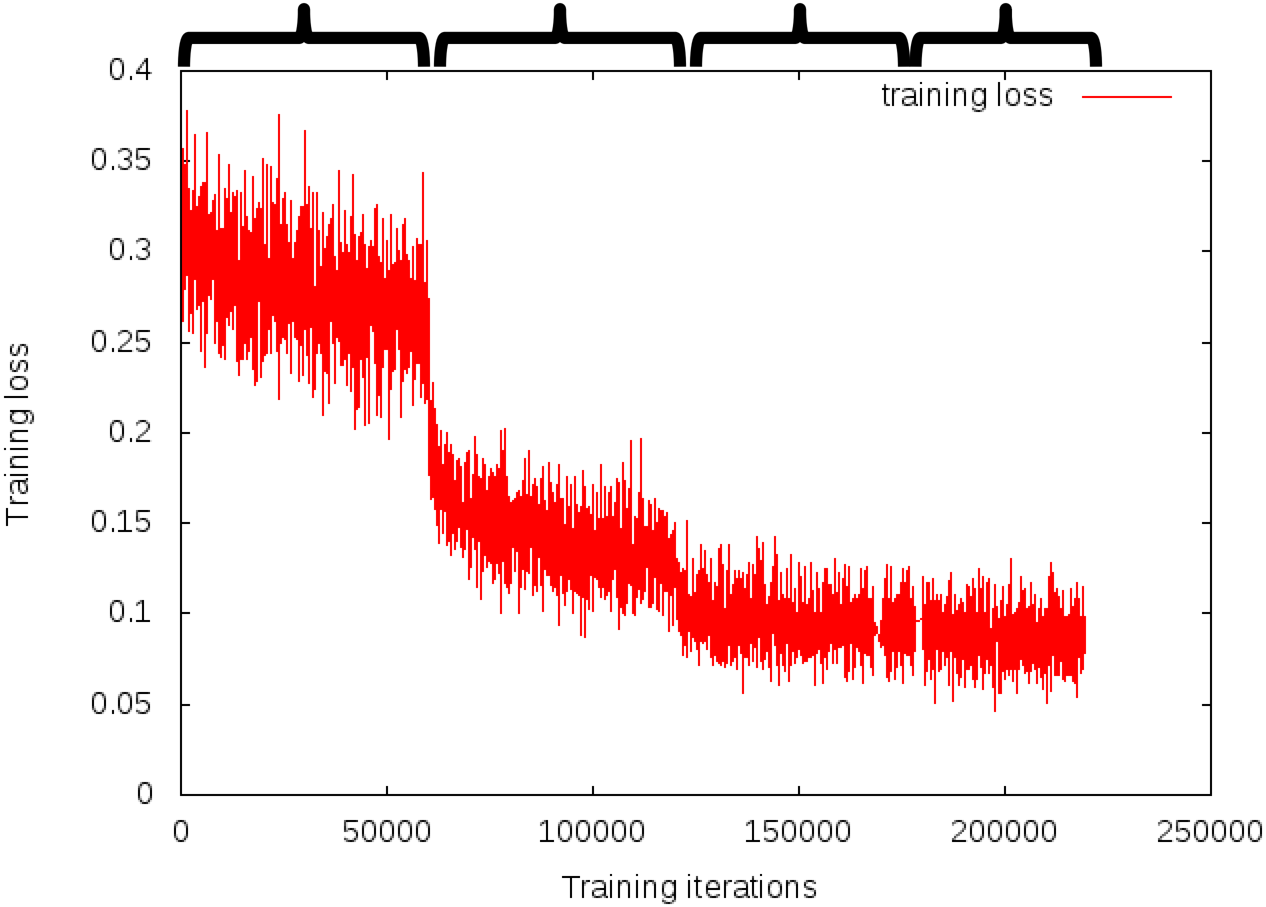


NETWORK TRAINING

- Deep model is trained from scratch using mini-batch SGD

Learning Rate

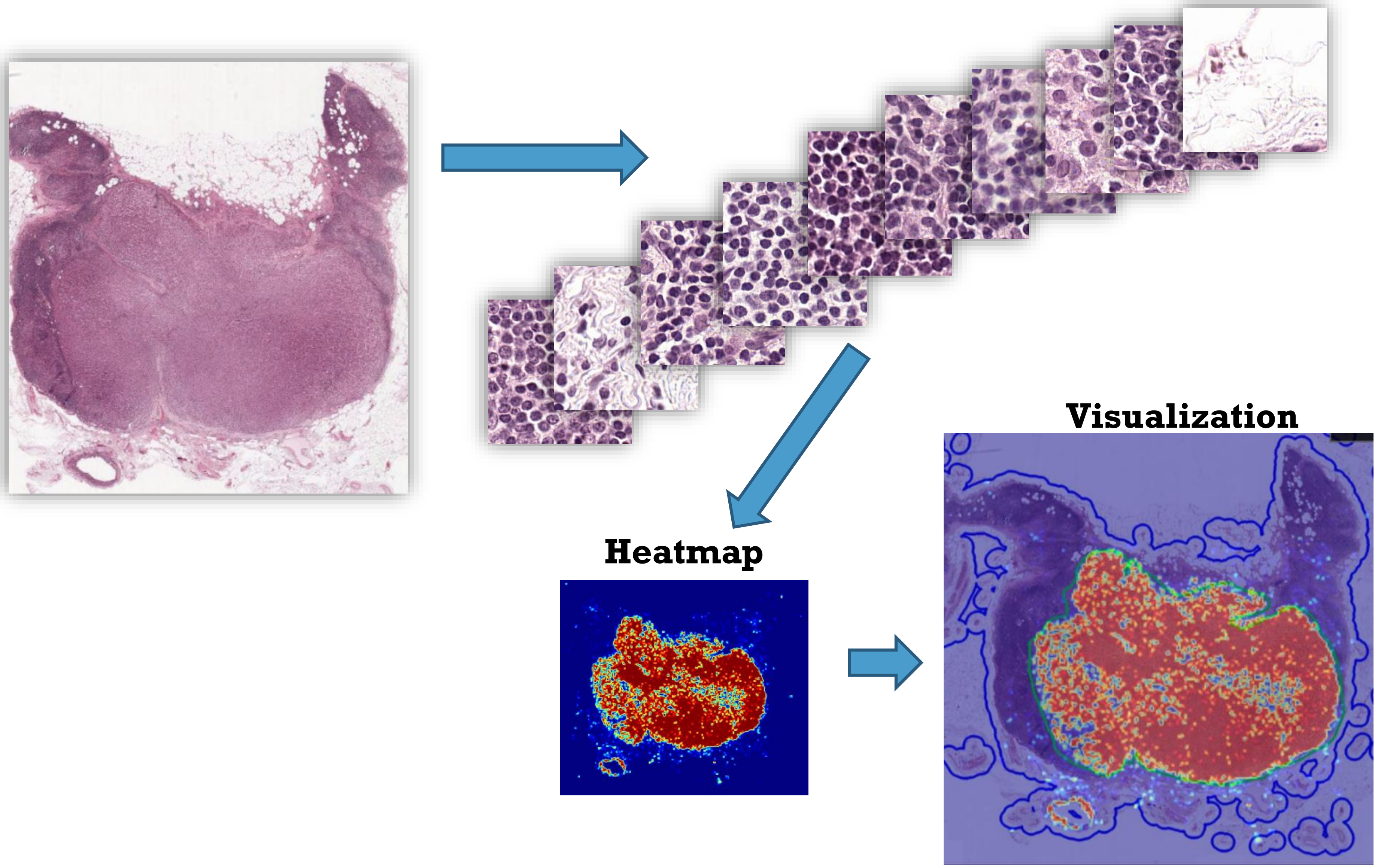
1e-2 **1e-3** **1e-4** **1e-5**



Environment

- GPU:
  - 2 x NVidia Tesla K80 graphics cards
- CPU:
  - Intel ® Xeon® CPU E5-2620 v3 @ 2.40GHz
  - #cores=12
- Hard Disk:
  - 4T SSD
- Memory:
  - 64 GB RAM

TUMOR PROBATILITY HEATMAP GENERATION

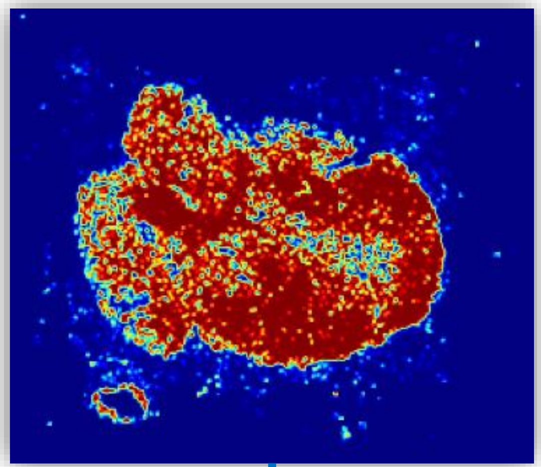


Heatmap

Visualization

POST-PROCESSING FOR SLIDE-BASED TUMOR CLASSIFICATION

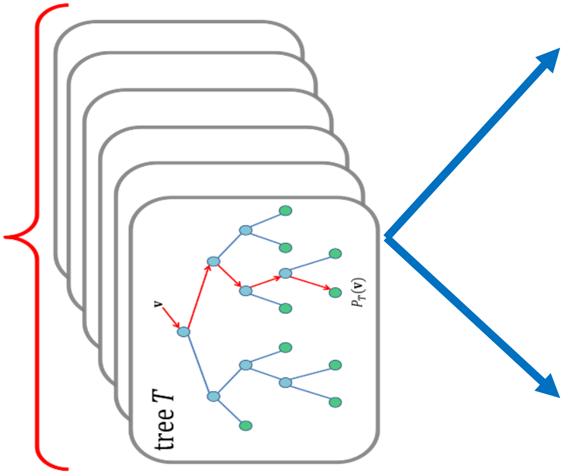
- Extracting higher level features from tumor heatmaps



- the percentage of tumor region over the whole tissue region
- the area ratio between tumor region and the minimum surrounding convex region
- the average prediction values
- the longest axis of the tumor region
- …….

28-dim feature vector

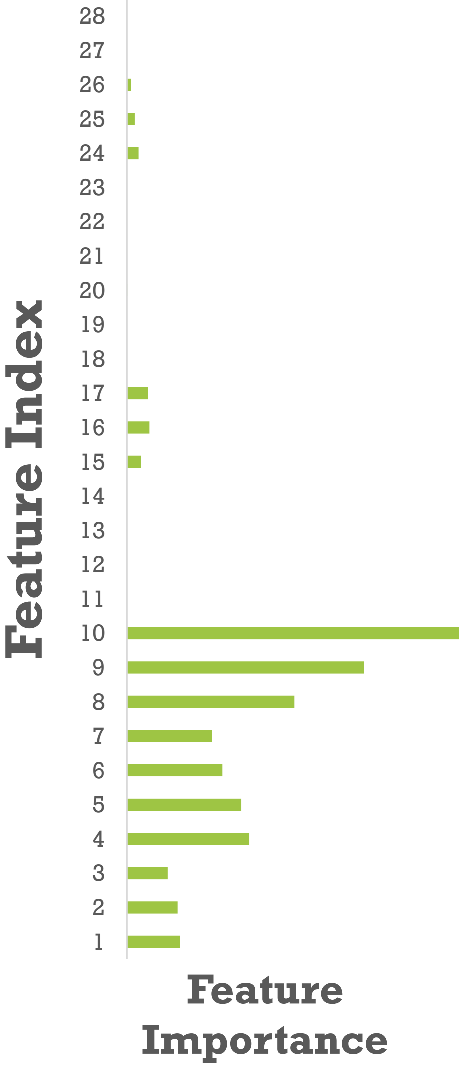
Random Forest Classifier
(#tree = 50)



Tumor

Normal

POST-PROCESSING FOR SLIDE-BASED TUMOR CLASSIFICATION



Top 5 important features, computed using the "regionprops" function in skimage. t is the threshold value

- **Feature 10:** given t=0.5, the longest axis in the largest tumor region
- **Feature 09:** given t=0.5, ratio of pixels in the region to pixels in the total bounding box ("extent")
- **Feature 08:** eccentricity of the ellipse that has the same second–moments as the region. ("eccentricity")
- **Feature 04:** ratio of tumor region when t=0.9 to the tissue region
- **Feature 05:** given t = 0.5, the area of largest tumor region

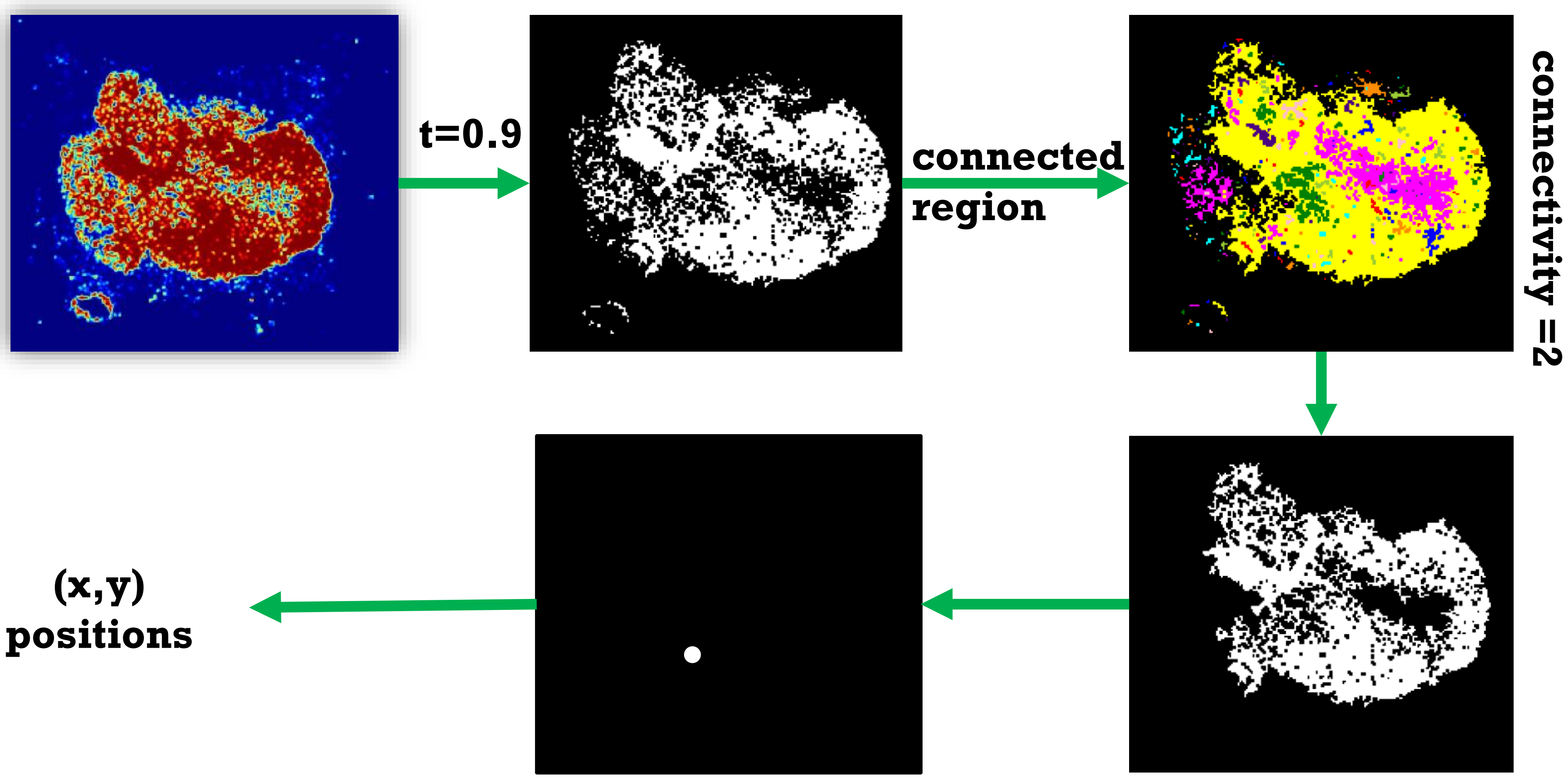
POST-PROCESSING FOR LESION-BASED TUMOR REGION SEGMENTATION

- Train a sensitive model (D-1) for estimation of tumor location (threshold = 0.9)

- Train a more specific model (D-2) for tumor probability estimation
  - ~30K extra training patches extracted from normal area adjacent to tumor region

LESION-BASED TUMOR REGION SEGMENTATION

- Generate the locations using H-1



t=0.9

connected region

connectivity =2

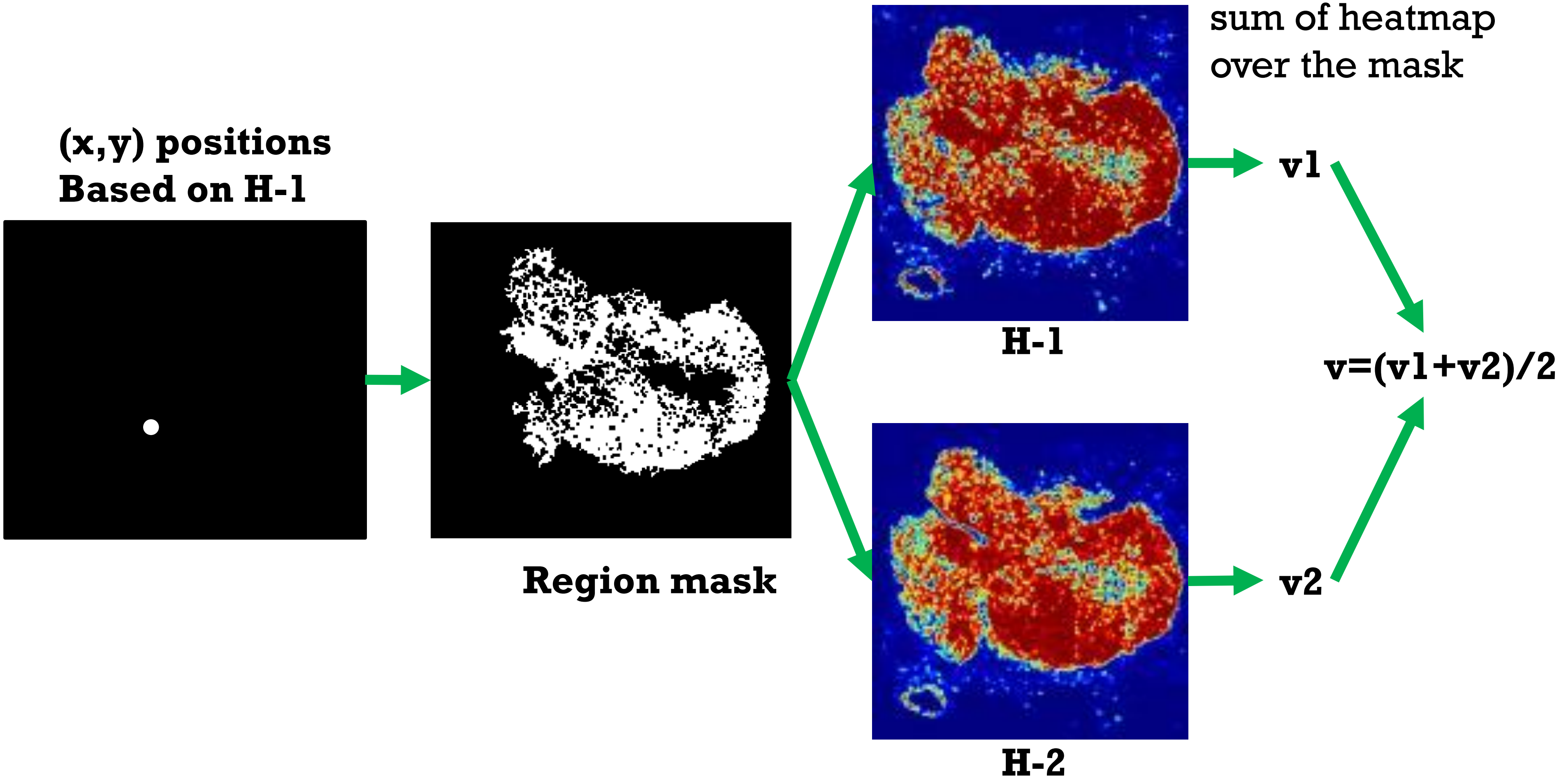
Largest region, label = 74

Use central point as the tumor position

(x,y) positions

LESION-BASED TUMOR REGION SEGMENTATION

- Generate the prediction value using H-1 and H-2



(x,y) positions
Based on H-1

Region mask

H-1

H-2

sum of heatmap
over the mask

v1

v2

v=(v1+v2)/2

CONCLUSIONS

- We developed a deep learning based framework for metastatic cancer detection in lymph nodes
  - **Architecture**: Based on GoogLeNet
  - **Training**: Additional training patches from false positive and tumor adjacent regions
  - **Post-processing**:
    - Random forest classifier on heatmap-based features for classification task
    - Integration of a more sensitive (for tumor location) and more specific (for tumor probability) model for segmentation task