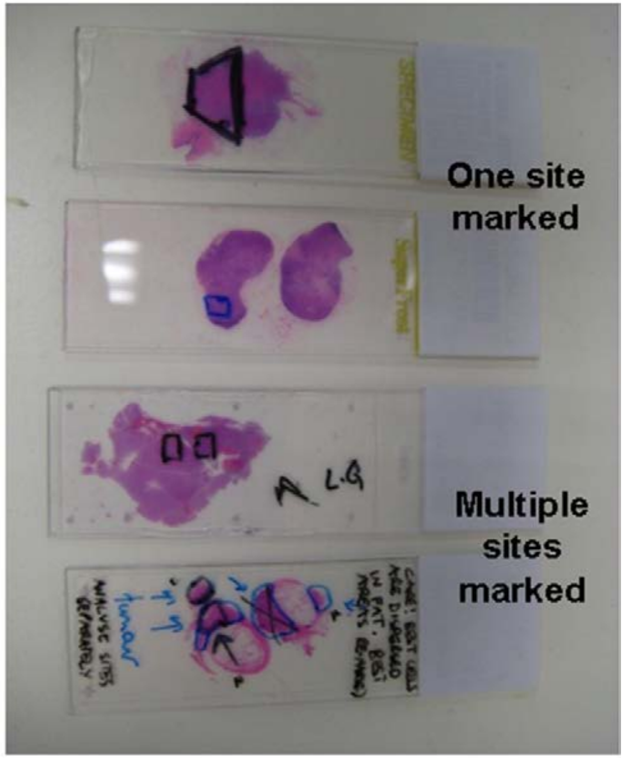
# Fast Mitosis Detection in Histopathological Images using Deep Learning Neural Networks

Yuguang Li
Advisor: Linda Shapiro

# Histopathological Image Analysis – Breast Cancer Diagnosis




Tissue Slide Staining

Image Scanning

**Terminology:**

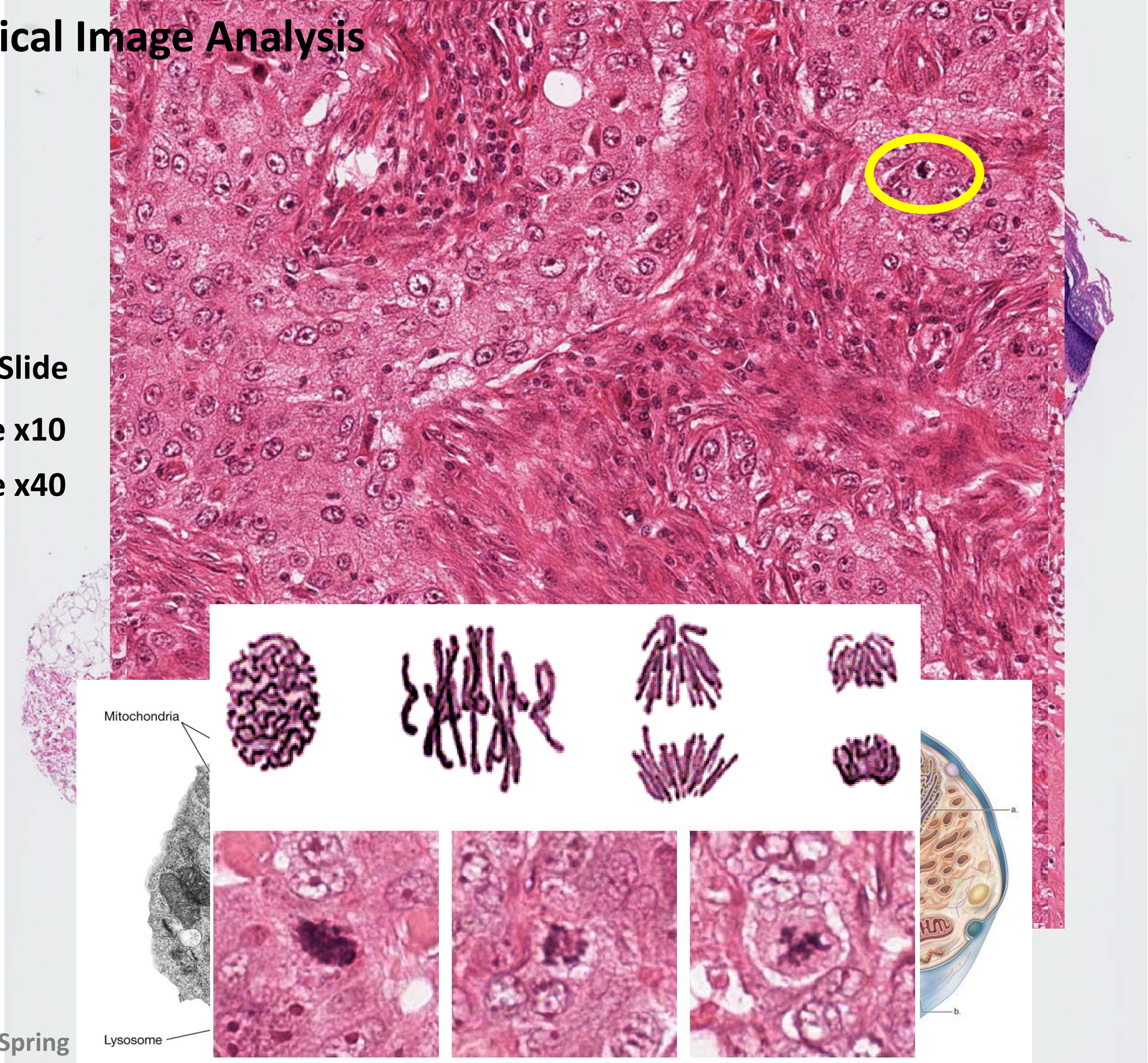
Histopathological Images
Tissue Staining
Whole Image Slides

# Histopathological Image Analysis

Whole Slide

Scale x10

Scale x40




**Terminology:**

Histopathological Images

Tissue Staining

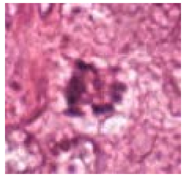
Whole Image Slides

Region of Tumor

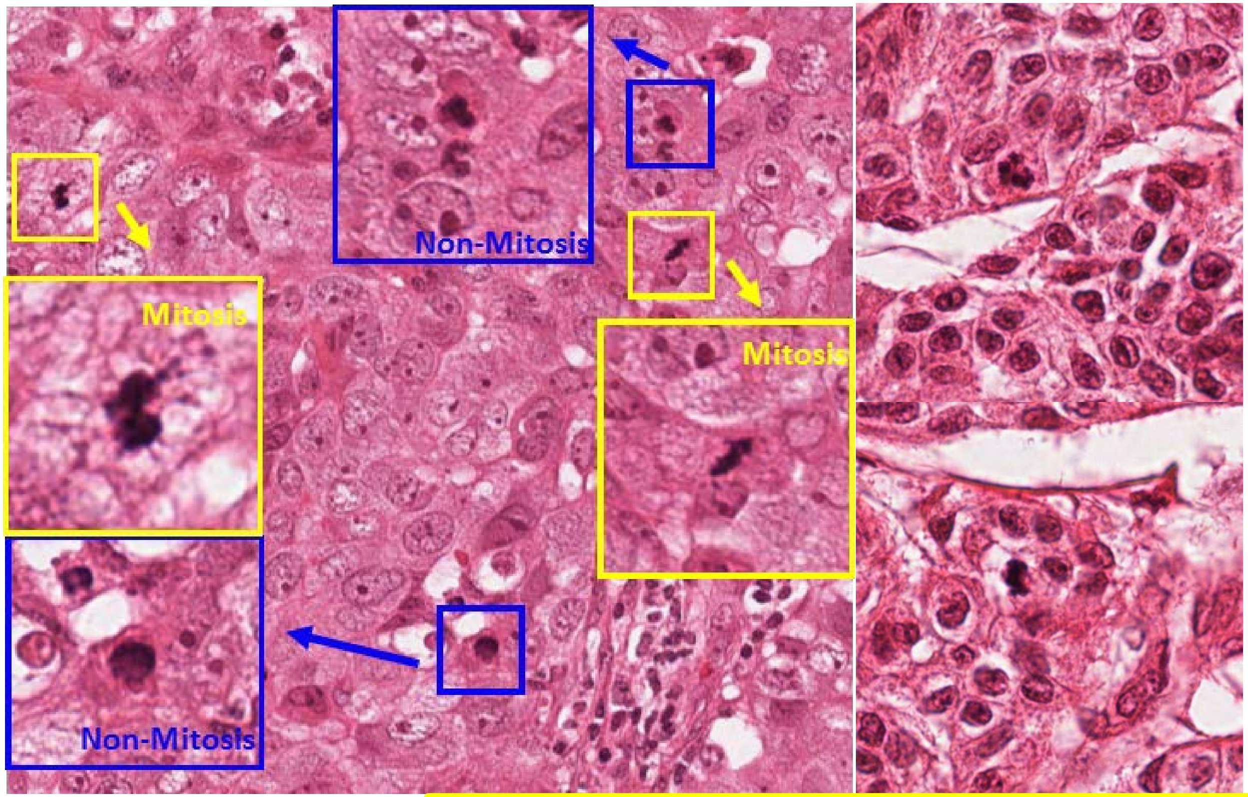
Region of Interest

Cell

Mitosis

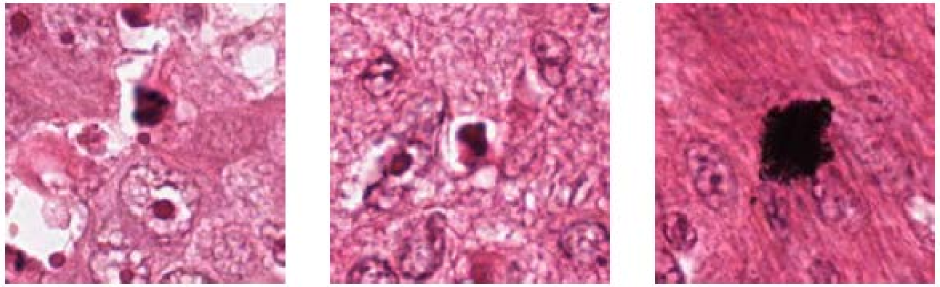


# Objectives: Counting Mitosis




## Challenges in Mitosis Detection:

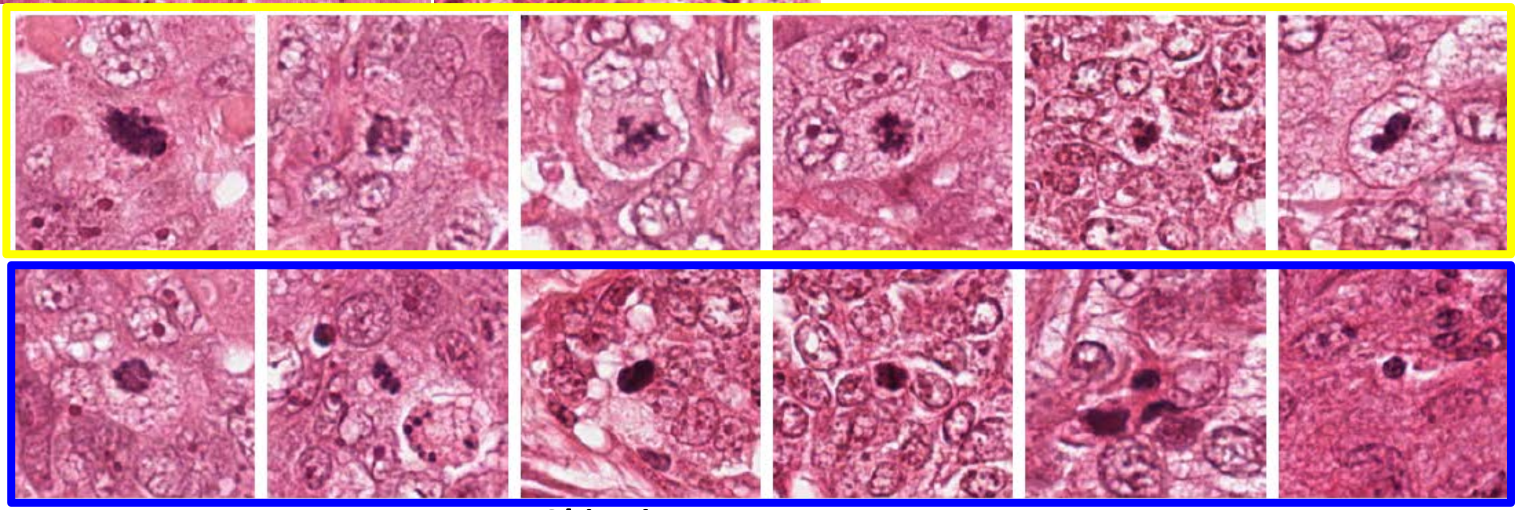
- Small objects with a large variety of shape configurations & texture variation
- Low frequency of appearance.
- Similarity with other types of nuclei



(a) Apoptosis (b) Apoptosis (c) Dust

Mitosis

Non-Mitosis



Objective

**MITOS dataset** – ICPR 2012 Mitosis Detection Challenge

**General Information**

- 50 high-power fields (x40 magnitude) images in 5 different biopsy slides stained with Hematosin.
- Each field has 2048*2048 pixels representing a 512×512μm2 area.
- Expert pathologists manually annotated 300 mitoses. Average Mitosis size 30*30 pixels. Every pixel of each mitosis is labeled.
- The annotated mitoses are what two pathologists agreed on. So there are mitosis-like regions unannotated in these images.

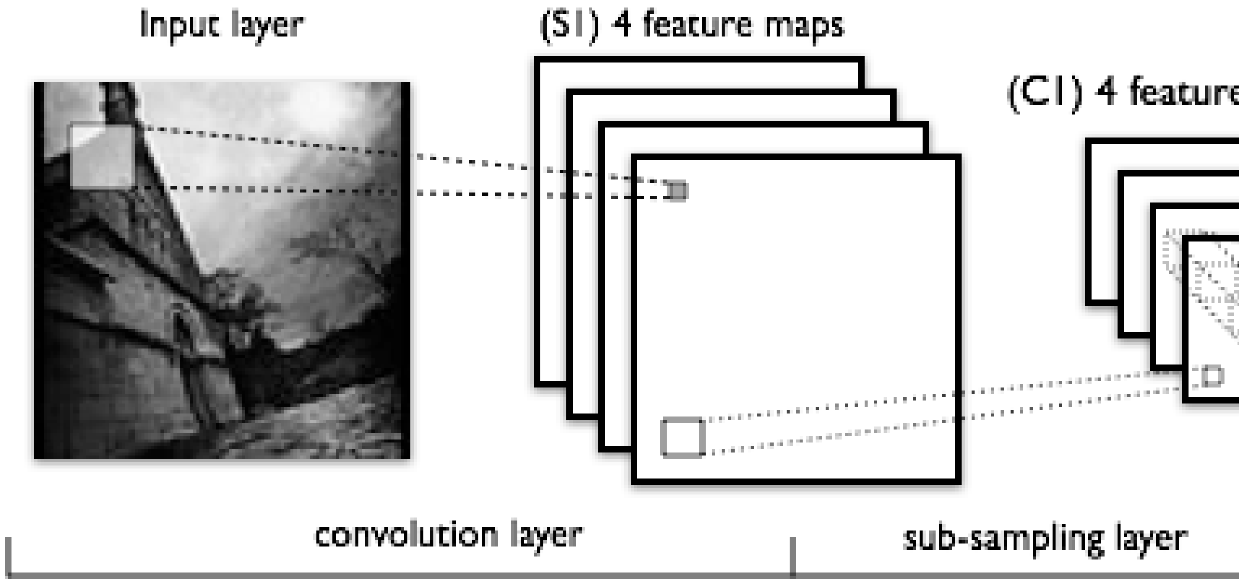
**Training Set**

- 35 images (7 images from each biopsy slides)
- 200 mitoses annotated

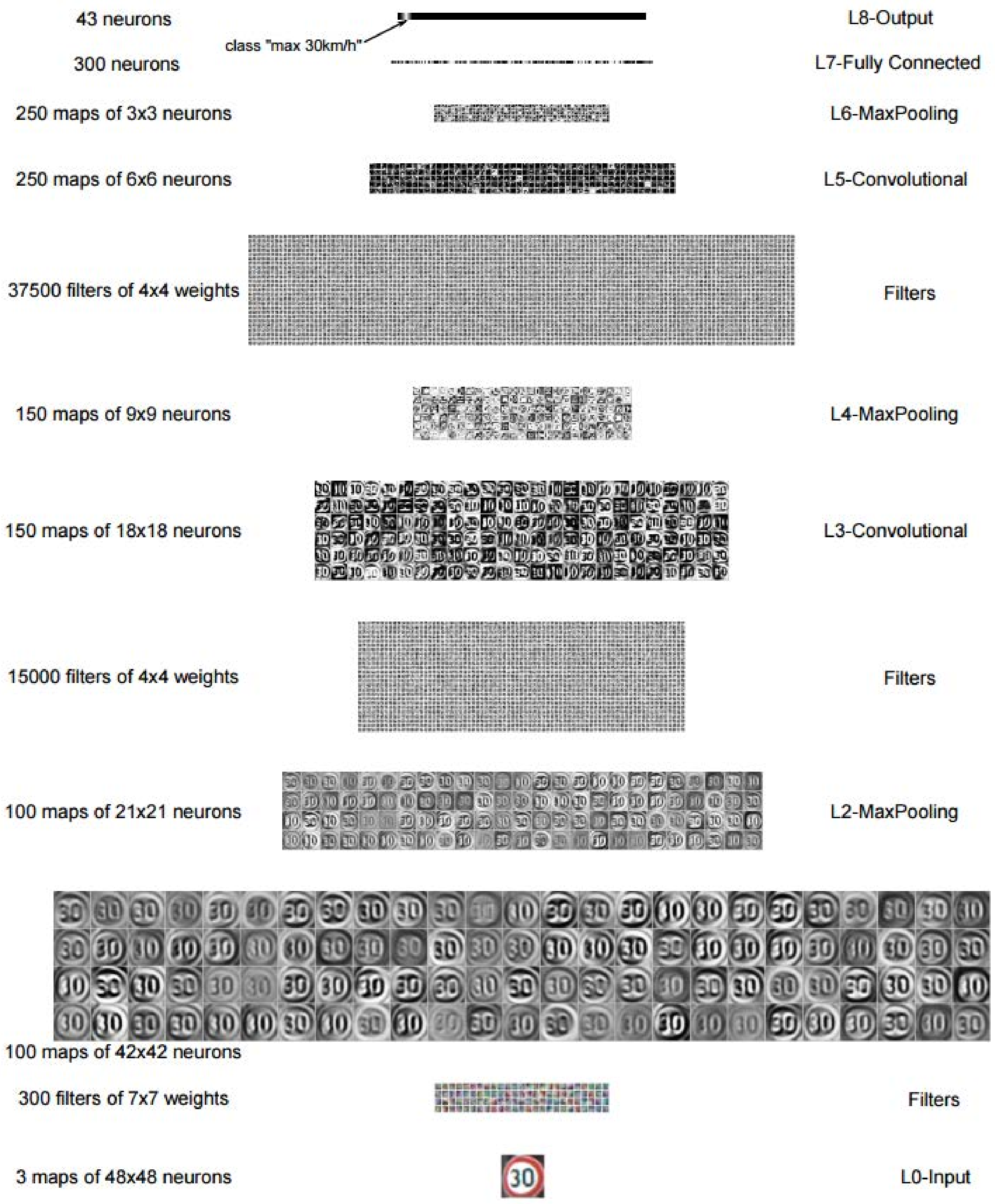
**Testing Set**

- 15 images (3 images from each biopsy slides)
- 100 mitoses annotated

## Convolutional Neural Network




Structure

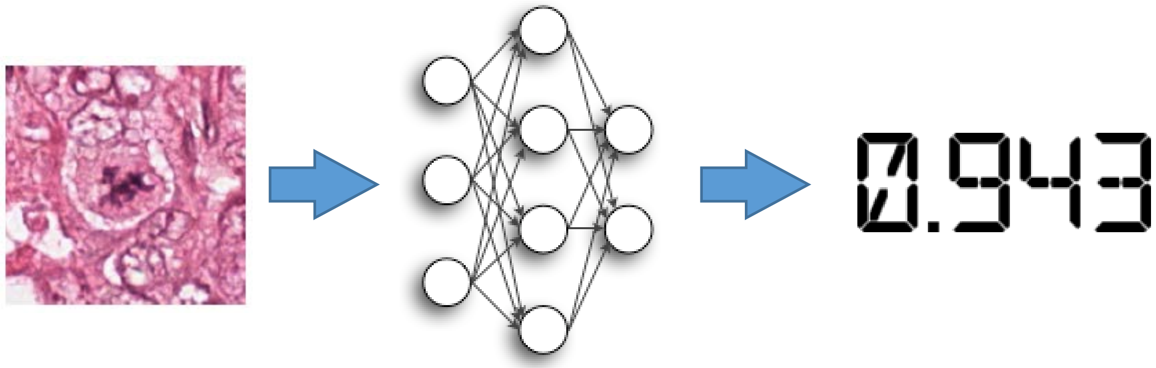




Convolutional N

## Structure of Convolutional Neural Network

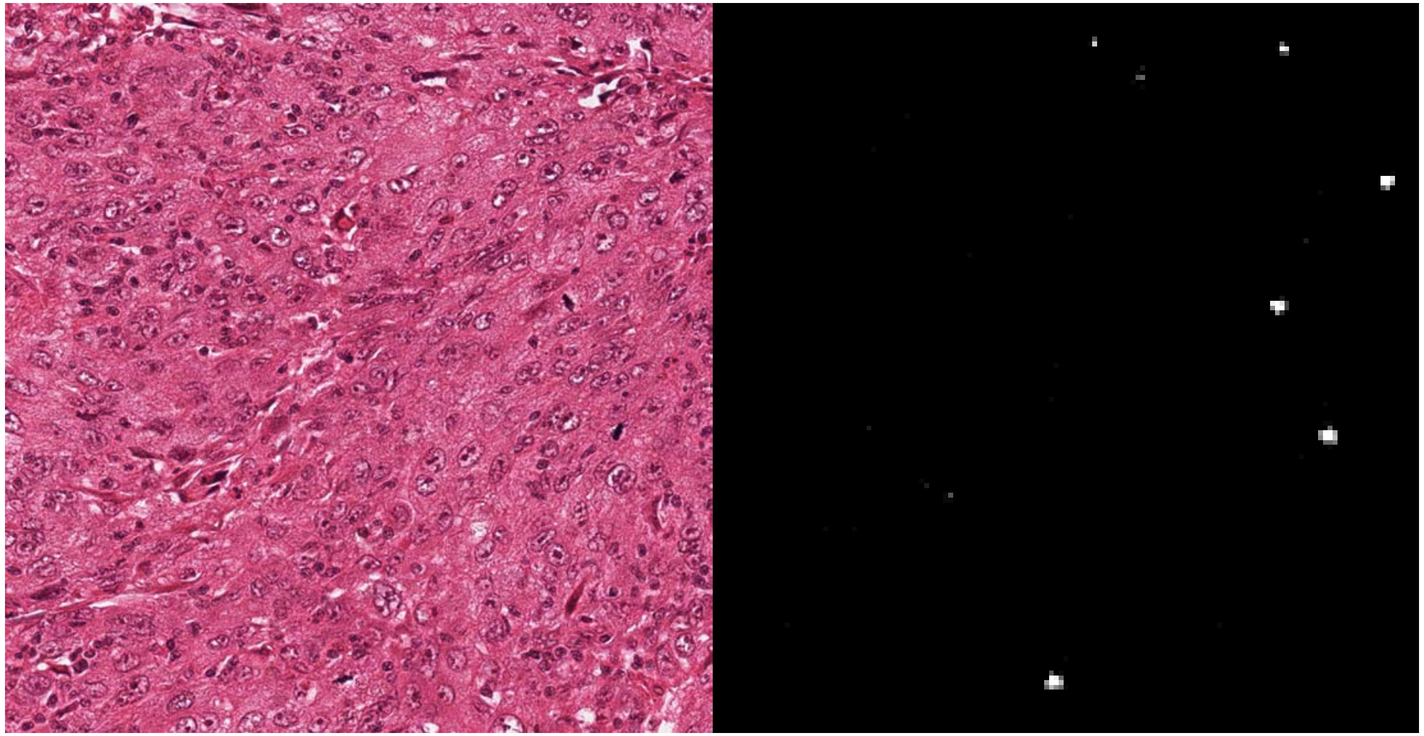
### 12-layer CNN Model

| Layer | Type | Neurons | Filter Size |
|---|---|---|---|
| 0 | Input | 3M x 101 x 101 | -- |
| 1 | Conv | 16M x 100 x 100 | 2 x 2 |
| 2 | MaxPool | 16M x 50 x 50 | 2 x 2 |
| 3 | Conv | 16M x 48 x 48 | 3 x 3 |
| 4 | MaxPool | 16M x 24 x 24 | 2 x 2 |
| 5 | Conv | 16M x 22 x 22 | 3 x 3 |
| 6 | MaxPool | 16M x 11 x 11 | 2 x 2 |
| 7 | Conv | 16M x 10 x 10 | 2 x 2 |
| 8 | MaxPool | 16M x 5 x 5 | 2 x 2 |
| 9 | Conv | 16M x 4 x 4 | 2 x 2 |
| 10 | MaxPool | 16M x 2 x 2 | 2 x 2 |
| 11 | FullyConn | 100 | 1 x 1 |
| 12 | FullyConn | 2 | 1 x 1 |




RGB Patch Input | CNN Classifier | Probability

### Testing method: Sliding Window



Original Image | Probability Map

**Training of Convolutional Neural Network**

## Objective:

Create an image patch classifier react positively when mitosis is found in the center of a patch.

## Challenges:

- Very few positive samples exist in the images. Only 200 Mitosis in training set. All the rest of the areas are non-mitosis

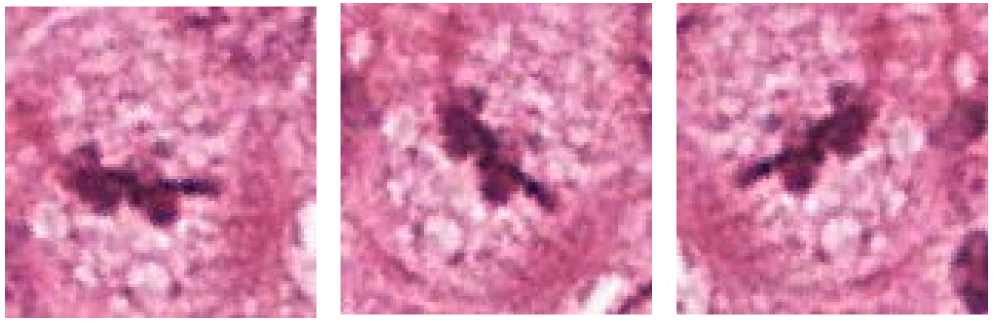
- Trained classifier needs to be rotational and shift invariant.

## Solutions:

- We used random rotation, shift and mirroring to augmented the positive training samples.
- In the first stage, we are using 66000 mitosis and 66000 non-mitosis patches in training samples.
- Non-mitosis samples are randomly selected from the non-mitosis areas.

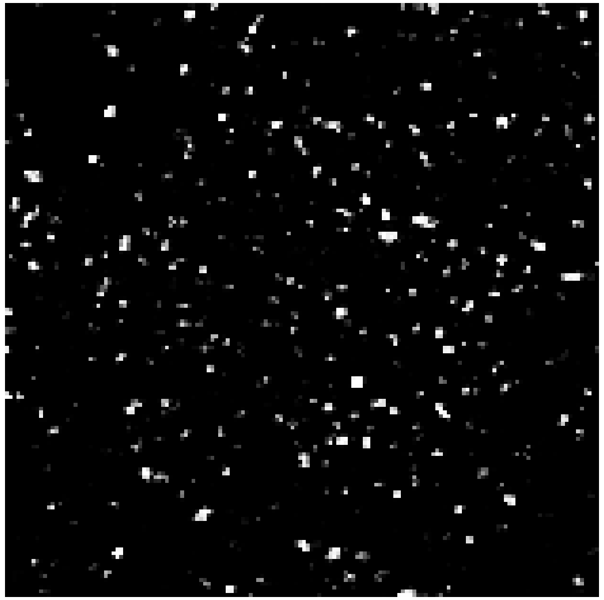
## Augmented Training Samples:

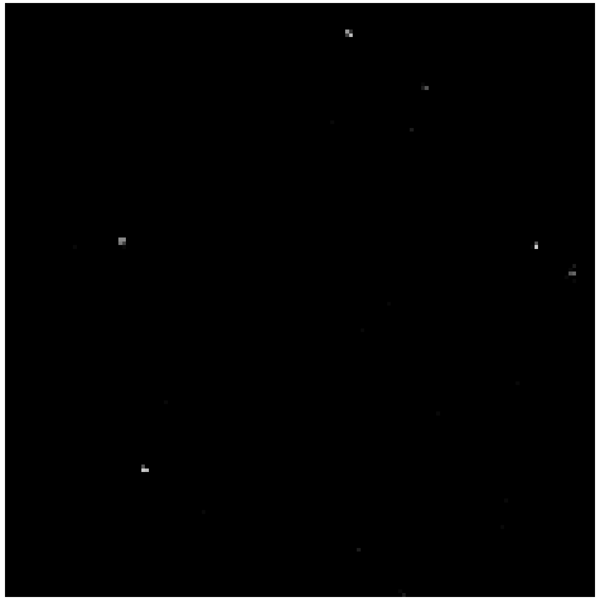
training



testing

## Training technique: Semi-supervised Learning





Step 1: Initial training set

- 66000 Mitosis
- 66000 Non-mitosis

- Positive Samples ---- Apply rotations and shifts on 200 mitosis.
- Negative Samples ----Randomly sampled from the non-mitosis pixels.

Sampling Probability Map

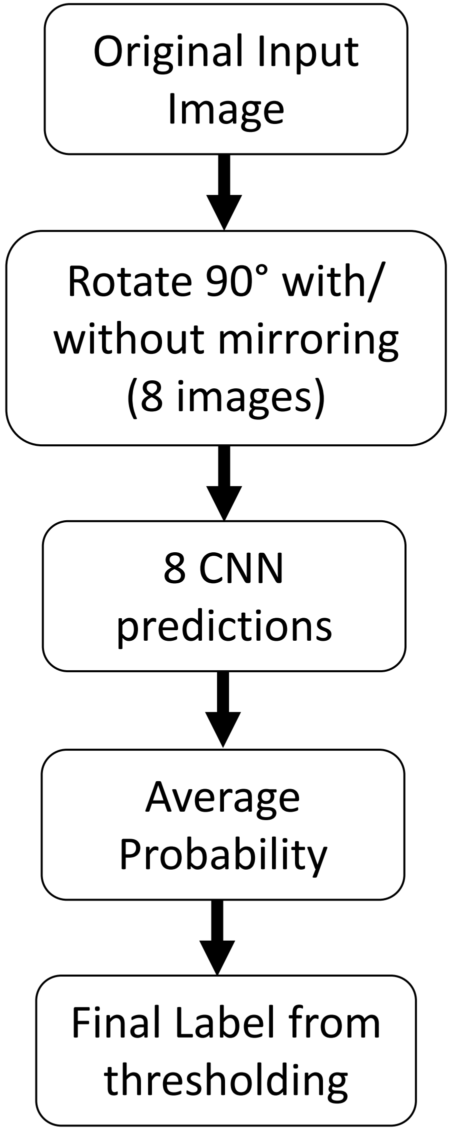
**Have the classifier to learn difficult samples.**

Step 2: Augmented training set

- 66000 Mitosis
- 1million Non-mitosis

- Positive Samples ---- Same samples as above
- Negative Samples ----Use the probability of false positive samples from last step as sampling probabilities to create a lot more difficult samples.

## Testing Process – Improving Rotational Invariance in the results




**Accuracy from CNN Model with/ without 8 direction Average**

| | Precision | Recall | F-Score |
|---|---|---|---|
| Single Direction | 0.78 | 0.74 | 0.758 |
| 8-direction Average | 0.78 | 0.79 | 0.784 |
| Original Paper | 0.88 | 0.70 | 0.782 |

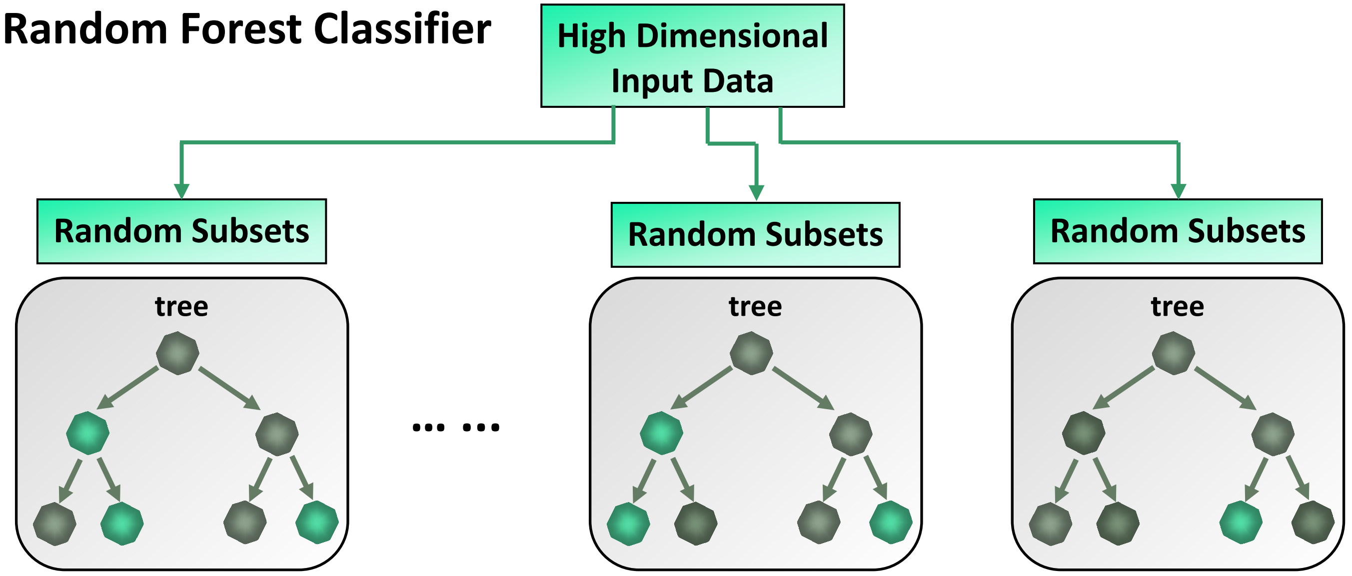
**Accuracy from CNN Model with different thresholds**

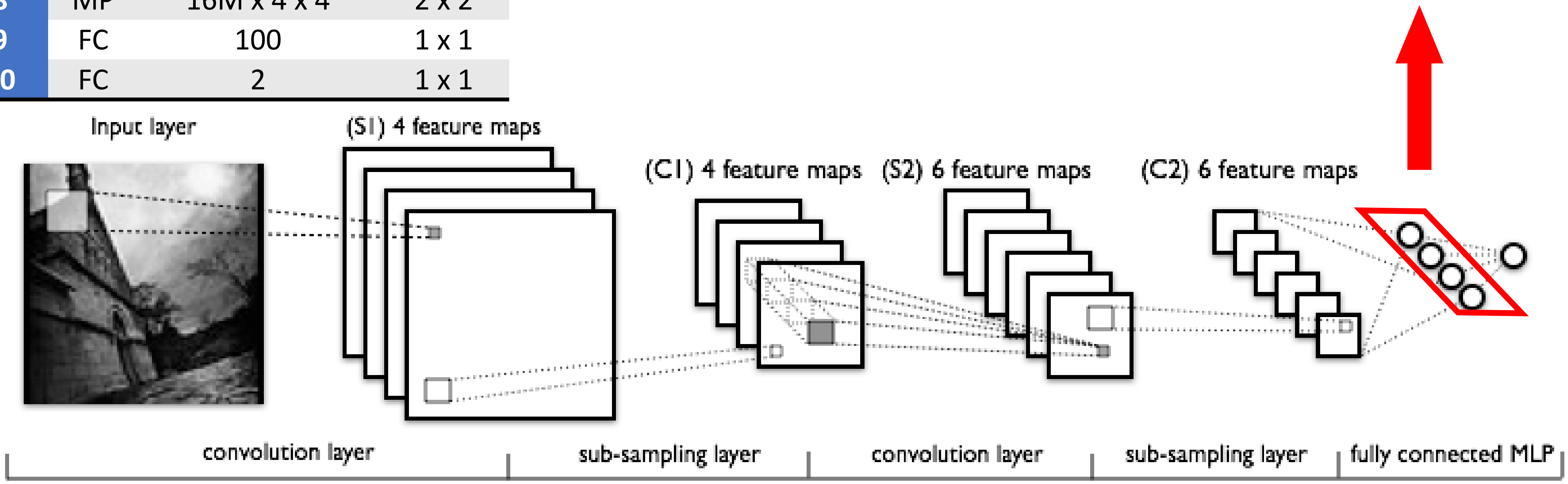
| Threshold Values | Precision | Recall | F-Score |
|---|---|---|---|
| 0.5 | 0.6905 | 0.8614 | 0.7665 |
| 0.6 | 0.7339 | 0.8345 | 0.7723 |
| 0.7 | 0.7767 | 0.7921 | 0.7843 |
| 0.8 | 0.8409 | 0.7327 | 0.7831 |
| 0.9 | 0.8659 | 0.7030 | 0.7760 |

## CNN extracted feature Classification -------Merging Different CNN models

| Layer | Type | Neurons | Filter Size |
|---|---|---|---|
| 0 | I | 3M x 101 x 101 | -- |
| 1 | C | 16M x 98 x 98 | 4 x 4 |
| 2 | MP | 16M x 49 x 49 | 2 x 2 |
| 3 | C | 16M x 46 x 46 | 4 x 4 |
| 4 | MP | 16M x 23 x 23 | 2 x 2 |
| 5 | C | 16M x 20 x 20 | 4 x 4 |
| 6 | MP | 16M x 10 x 10 | 2 x 2 |
| 7 | C | 16M x 8 x 8 | 3 x 3 |
| 8 | MP | 16M x 4 x 4 | 2 x 2 |
| 9 | FC | 100 | 1 x 1 |
| 10 | FC | 2 | 1 x 1 |

### Random Forest Classifier



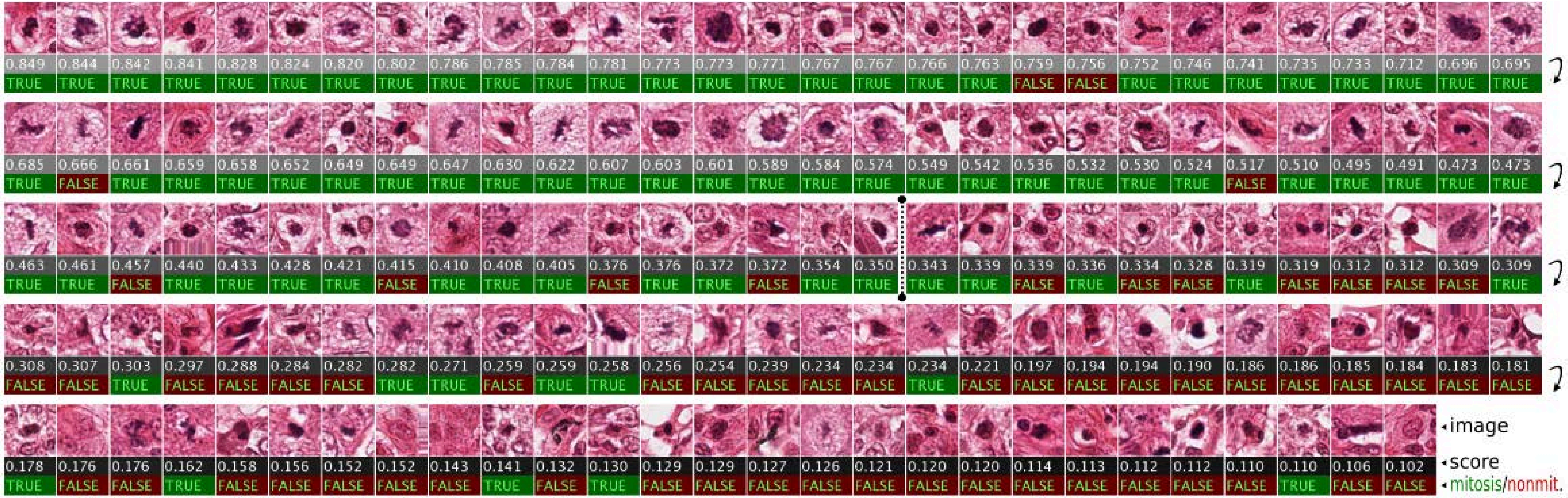






**Convolutional Neural Network**

## CNN Classification Accuracy

| | 10-layer DNN | | | 12-layer DNN | | |
|---|---|---|---|---|---|---|
| | Prec | Recall | F-meas | Prec | Recall | F-meas |
| Original Paper | 0.78 | 0.72 | 0.751 | 0.88 | 0.70 | 0.782 |
| Scanning window + Initial Samples | 0.74 | 0.13 | 0.221 | 0.93 | 0.04 | 0.084 |
| Scanning window + Augmented Samples | 0.80 | 0.74 | 0.769 | 0.78 | 0.79 | 0.784 |
| Extracted DNN feat. + RF classifier | 0.79 | 0.76 | 0.775 | 0.84 | 0.73 | 0.781 |
| Merged DNN feat. + RF classifier | Precision: 0.82 | | Recall: 0.76 | | F-meas: 0.789 | |




Convolutional Neural Network

**Training Efficiency**

Convolutional Neural Network (Caffe library)

| | ICPR Initial Set | ICPR Augmented Set | |
|---|---|---|---|
| | NVIDIA K40 | NVIDIA K40 | CPU |
| **10-layer CNN** | 40 min. | 8 hrs. | 20hrs |
| **12-layer CNN** | 2 hrs. | 12 hrs. | 45hrs |

CNN-feature-based Method

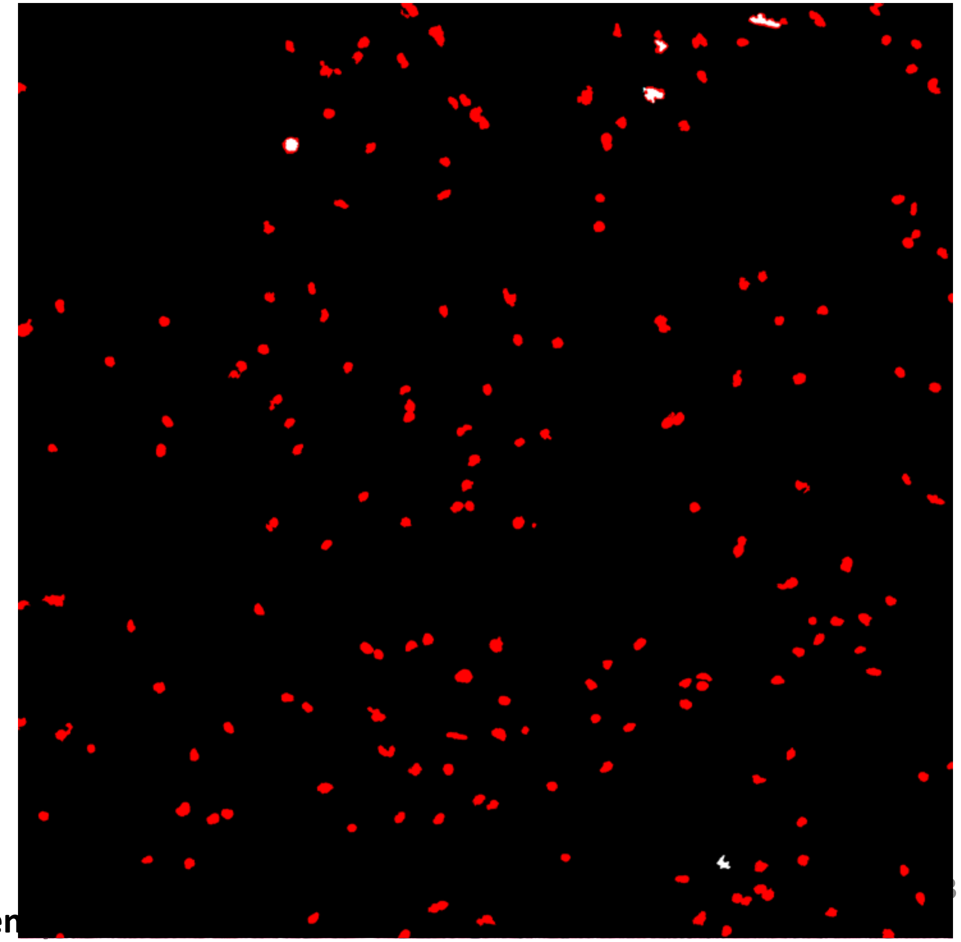
Random forest classifier is trained in less than 1 minute with extracted features on a CPU machine.

**Testing Efficiency**

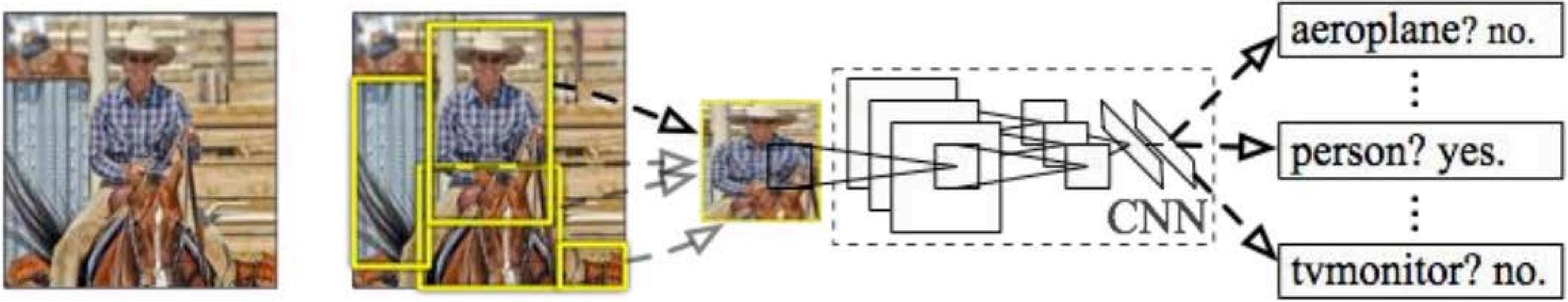
Convolutional Neural Network (5-pixel scanning interval)

Image tested with 8 rotations: **24min per image with CPU**

**Lots of unnecessary scans!!!**



Region-based Convolutional Neural Network




Input image

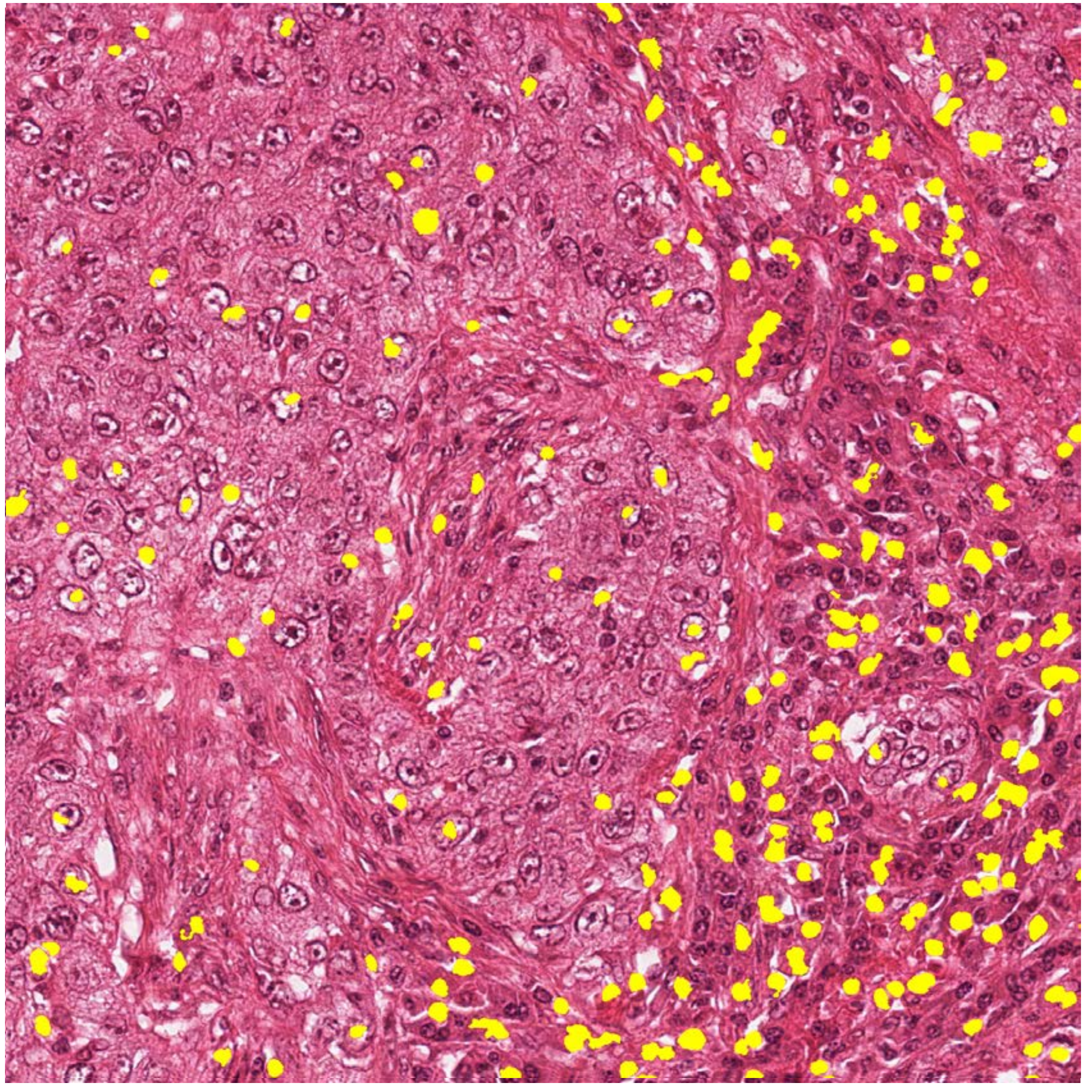
Extract region proposals (~2k / image)
e.g., selective search
[van de Sande, Uijlings et al.]

Compute CNN features on regions

Classify and refine regions

**Region-based convolutional network**

## Initial Mitotic Region Proposal Generation



**Mitosis Candidates**

**List of features (10 dimensions)**

- Multiscale Gaussian Smoothing
- Multiscale Laplacian of Gaussian
- Difference of Gaussians
- Structure Tensor Eigenvalues
- Hessian of Gaussian Eigenvalues

**Idea:** Only Scan in proposed regions

**Speed** (2000 * 2000 pixel RGB image):

Proposal Generation – 10sec/per image

CNN classification in 8 directions – 20sec/per image with cpu.

Calls CNN function 3200 times /per image instead of 320,000 times / per image