Capstone Presentation

Project: Adversarial Learning - Medical Tomography

Advisor(s): Kris Kitani, Min Xu

Name: Qiqi Xiao

Date: 04/28/2018

Outline

1. ADA(Adversarial data augmentation): A Game-Theoretic Perspective on Data Augmentation for Object Detection

2. Detection of diabetic retinopathy(DR).

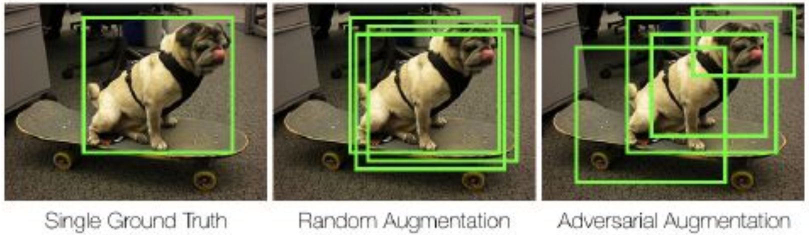
3. Difficulties and next steps

ADA(Adversarial data augmentation): A Game-Theoretic Perspective on Data Augmentation for Object Detection

- Introduce an adversarial function to generate (some distribution of) maximally perturbed version of the groundtruth which is hardest for the predictor to learn.
    - Why data augmentation: ground-truth wrong/not accurate….
    - How to add data augmentation: random translation, flipping, scaling…(manually add perturbations)
    - Problems: can be error-prone

- Adversary is not free but with constraints [e.g. features(new bb) ≈ features(ori bb)].

- First work to provide theoretic basis for data augmentation in terms of an adversarial two player zero-sum game.
    - predictor(maximize performance) *vs* constrained adversary(minimize expected performance).



Game Formulation

The value/payoff of the game for x (the expected loss)

$$\mathbb{E}_{\substack{y'|\boldsymbol{x} \sim f \\ y|\boldsymbol{x} \sim P}} [\ell(y', y)] = \sum_{y', y} f(y'|x)\ell(y', y)P(y|x)$$
$$= \mathbf{f}^\top \mathbf{G}\mathbf{p}.$$

f: the vector of probabilities obtained from the predictor over all labels

G: the game matrix where each element contains the loss between two labels

p: the augmentor distribution vector

Definition

Primal Adversarial Data Augmentation(ADA-P):

$$\min_f \max_P \mathbb{E}_{\substack{\boldsymbol{x} \sim \mathcal{D}, \\ y'|\boldsymbol{x} \sim f, \\ y|\boldsymbol{x} \sim P}} [\ell(y', y)] \textit{ such that:}$$

$$\mathbb{E}_{\substack{\boldsymbol{x} \sim \mathcal{D}, \\ y|\boldsymbol{x} \sim P}} [\phi(y, \boldsymbol{x})] = \mathbb{E}_{y, \boldsymbol{x} \sim \mathcal{D}} [\phi(y, \boldsymbol{x})]$$ where

$$\boxed{\mathbb{E}_{y, \boldsymbol{x} \sim \mathcal{D}} [\phi(y, \boldsymbol{x})] = \frac{1}{N} \sum_{n=1}^{N} \phi(y_n, \boldsymbol{x}_n),}$$

The Dual Adversarial Data Augmentation(ADA-D):

$$\min_\theta \mathbb{E}_{\mathbf{x}, y^* \sim \mathcal{D}} \Big[\min_f \max_P \mathbb{E}_{\substack{y' \sim f, \\ y \sim P}} \Big[\ell(y', y) + \theta^\top \{\phi(y, \boldsymbol{x}) - \phi(y^*, \boldsymbol{x})\} \Big] \Big].$$

Constraint Generation for Large Games

To solve the Games:

Nash Equilibrium → Linear Programming

To solve ADA-D without explicitly constructing the entire payoff matrix G.

Key idea: To use a set of the most violated constraints to grow a game matrix that supports the equilibrium distribution, but is much smaller than the full game matrix.

→ Double Oracle Algorithm

Algorithm and Implementation of ADA
(One iteration)

Algorithm 1 ADA Equilibrium Computation

Input: Image x; Parameters θ; Ground Truth y^*
Output: Nash equilibrium, $(\mathbf{f}, \mathbf{p})$

$\mathcal{Y} \leftarrow \text{EdgeBox}(x)$
$\Phi = \text{CNN}(\mathcal{Y}, x)$
$\psi \leftarrow \boxed{\theta^\top}(\Phi - \Phi(y^*))$
$\mathcal{S}_p \leftarrow \mathcal{S}_f \leftarrow \text{argmax}_y\, \psi(y)$
repeat
  $(\mathbf{f}, \mathbf{p}, v_p) \leftarrow \texttt{solveGame}(\psi(\mathcal{S}_p), \text{loss}(\mathcal{S}_f, \mathcal{S}_p))$
  $(y_{\text{new}}, v_{\max}) \leftarrow \max_y \mathbb{E}_{y' \sim f}[\text{loss}(y, y') + \psi(y)]$
  if $(v_p \neq v_{\max})$ **then**
    $\mathcal{S}_p \leftarrow \mathcal{S}_p \cup y_{\text{new}}$
  end if
  $(\mathbf{f}, \mathbf{p}, v_f) \leftarrow \texttt{solveGame}(\psi(\mathcal{S}_p), \text{loss}(\mathcal{S}_f, \mathcal{S}_p))$
  $({y'}_{\text{new}}, v_{\min}) \leftarrow \min_{\hat{y}} \mathbb{E}_{y \sim p}[\text{loss}(y, y')]$
  if $(v_f \neq v_{\min})$ **then**
    $\mathcal{S}_f \leftarrow \mathcal{S}_f \cup {y'}_{\text{new}}$
  end if
until $v_p = v_{\max} = v_f = v_{\min}$
return $(\mathbf{f}, \mathbf{p})$

Preprocess steps to extract features

Solve Nash equilibrium using linear programming
[Gurobi library]

$$\min_\theta \mathbb{E}_{\mathbf{x}, y^* \sim \mathcal{D}} \Big[\min_f \max_P \mathbb{E}_{\substack{y' \sim f, \\ y \sim P}} \Big[\ell(y', y) + \boxed{\theta^\top} \{ \phi(y, \boldsymbol{x}) - \phi(y^*, \boldsymbol{x}) \} \Big] \Big].$$

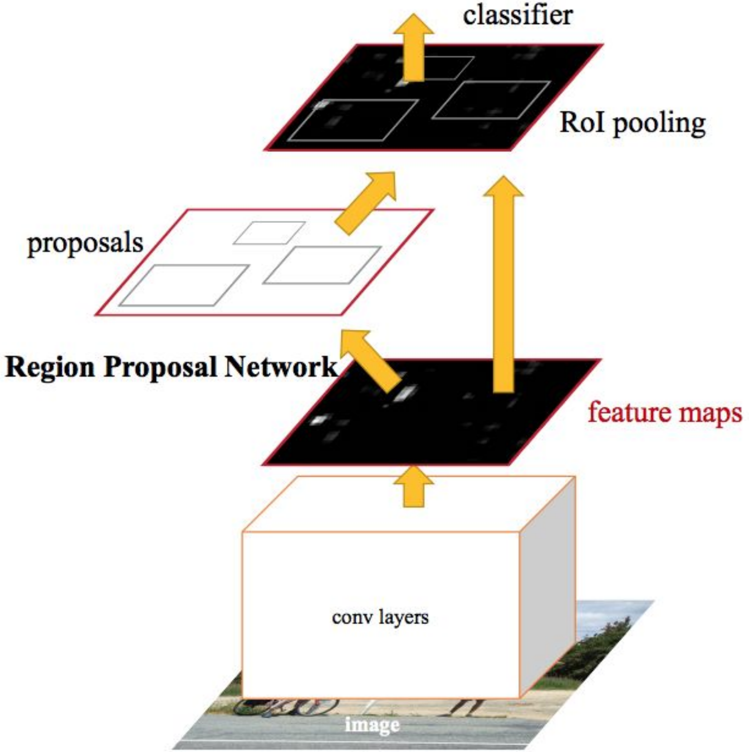
Update **θ** :

∇ **θ** : from (f, p)

Preprocess Steps

EdgeBox:
https://github.com/dculibrk/edge_boxes_with_python

CNN features:
https://github.com/longcw/faster_rcnn_pytorch
roi_pooling(vgg5_3 features, bounding boxes) ⇒ fc6 ⇒ fc7



Part Results of VOC2007 dataset

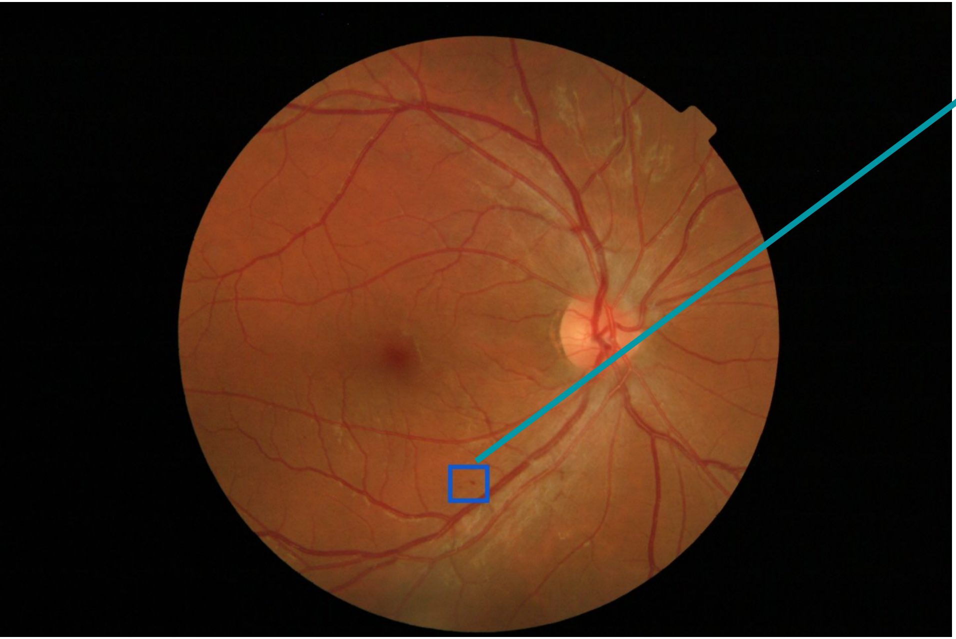
IOU	aero	bike	bird	boat	bott	bus	car	cat	chair	cow	dint	dog	horse	mbike	person	plant	sheep	sofa	traun	tv
0.5	--	--	--	--	--	0.803	0.711	--	--	--	--	--	--	0.760	0.689	--	--	--	--	--
0.7	--	--	--	--	--	0.628	0.461	--	--	--	--	--	--	0.636	0.403	--	--	--	--	--

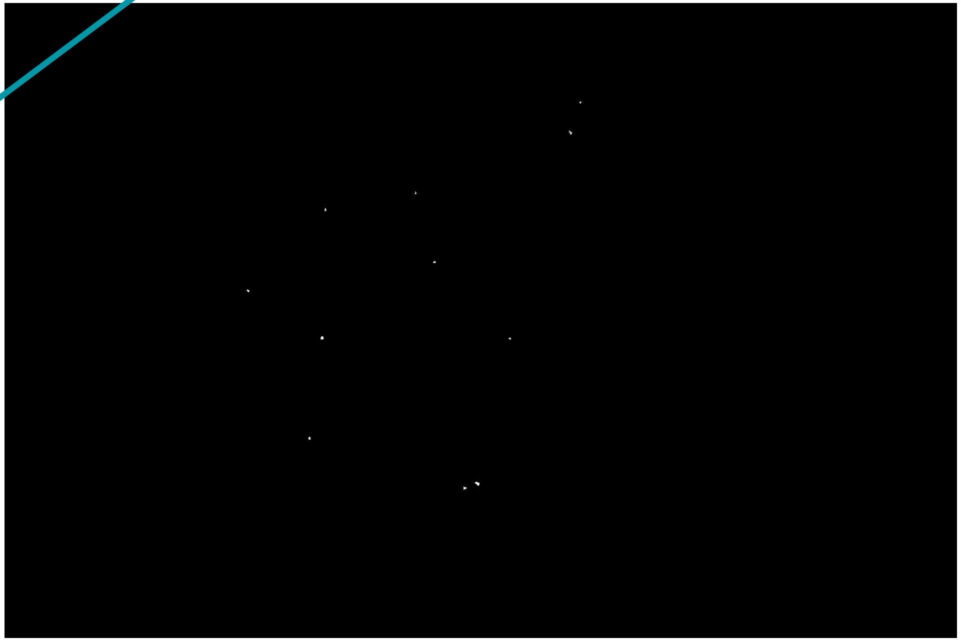
Table 6. ADA Generalization Across Deep Architectures. VOC2007 mAP for IoU>0.5.

Model	**VOC 2007 Object Category**																				
	Aero	Bike	Bird	Boat	Bott	Bus	Car	Cat	Chair	Cow	DinT	Dog	Horse	mbike	person	Plant	Sheep	Sofa	Train	TV	mAP
ADA+VGG16	68.5	71.5	67.8	63.3	48.6	76.5	78.8	80.9	**50.9**	78.5	64.5	79.6	71.8	73.2	66.4	30.2	70.6	72.6	80.8	62.8	67.9
SSVM+VGG16	73.6	76.4	63.7	46.1	44.0	76.0	78.4	80.0	41.6	74.2	62.8	79.8	78.0	72.5	64.3	35.0	67.2	67.2	70.8	**71.4**	66.1
SVM+VGG16 [7]	73.4	**77.0**	63.4	45.4	44.6	75.1	78.1	79.8	40.5	73.7	62.2	79.4	**78.1**	73.1	64.2	35.6	66.8	67.2	70.4	71.1	66.0
ADA+AlexNet fc7	62.4	70.0	63.6	63.0	44.8	72.2	75.5	79.5	44.6	81.6	64.0	81.5	70.2	68.5	71.4	**69.5**	65.0	71.2	81.4	59.8	68.0
SSVM+AlexNet fc7	68.2	72.9	57.3	44.2	41.8	66.0	74.3	69.2	34.6	54.7	54.3	61.3	69.8	68.7	58.5	34.6	63.6	52.5	62.6	63.5	58.6
SVM+AlexNet fc7 [7]	68.1	72.8	56.8	43.0	36.8	66.3	74.2	67.6	34.4	63.5	54.5	61.2	69.1	68.6	58.7	33.4	62.9	51.1	62.5	64.8	58.5
ADA+ResNet101	**76.4**	74.8	**72.4**	**64.0**	**52.5**	**84.0**	**81.9**	**86.0**	48.5	**83.5**	**64.8**	**82.0**	73.5	**77.0**	**72.4**	36.6	**74.4**	**74.8**	**81.4**	65.6	**71.3**
SSVM+ResNet101	68.0	70.2	69.3	54.3	46.5	76.2	78.8	85.0	46.8	80.2	63.2	78.1	69.5	71.4	61.8	36.8	68.1	69.1	73.6	64.5	66.5

Detection of diabetic retinopathy(DR)

Apply ADA to eye images for detection of diabetic retinopathy(DR detection)





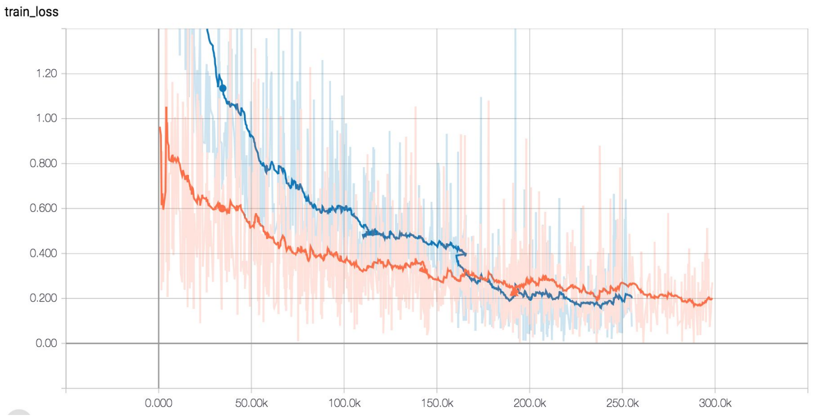
Detection of diabetic retinopathy(DR)

1. Dataset: e-optha-MA(microaneurysm)
    a. 148 unhealthy images → 134 for training, 14 for test
    b. 233 healthy images → 199 for training, 24 for test
2. Generate baseline for detection(MA only)
    a. Faster RCNN based methods
    b. Modifications
        i. Generate bounding boxes: connect annotation lump and double the size of bounding boxes.
        ii. Random crop 200 x 200 on the original images ~1000 x1000
        iii. Enable faster rcnn to train on input images without any positive bounding boxes by only considering classification loss.
        iv. Only Use small anchor sizes [2, 4, 8, 16] → [1, 2, 4]

Experiments

E1 training data: patches with positive bounding boxes(standard).

E2 training data: E1 data + without bounding boxes(my approach).

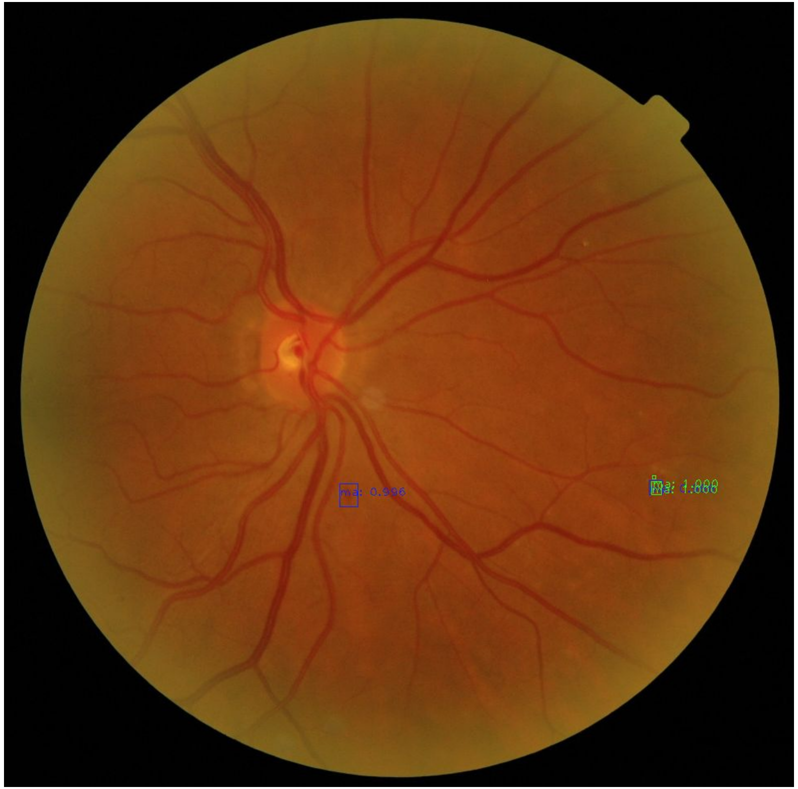


iou = 0.5 bbox_thresh = 0.5	bbox thresh	E1(all positive)	E2(half negative)
	avg_recall	0.608	0.575
	tot_recall	0.556	0.481
	avg_precision	0.197	0.375
	tot_precision	0.169	0.382

Comparison among different thresholds of E2

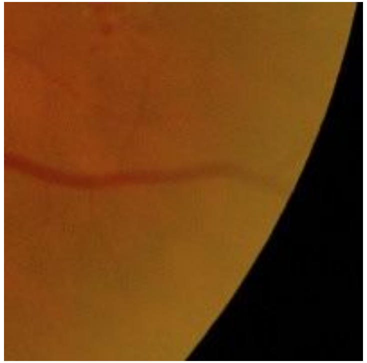
iou = 0.5	bbox thresh	0.	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
E2	AUC(image level)	0.5	0.860	0.860	0.878	0.878	0.878	0.881	0.887	0.881	0.892
	AP(image level)	0.368	0.804	0.800	0.819	0.912	0.805	0.803	0.820	0.793	0.798
	avg_recall	0.641	0.595	0.580	0.580	0.587	0.575	0.585	0.565	0.527	0.528
	tot_recall	0.602	0.519	0.491	0.491	0.5	0.481	0.491	0.472	0.454	0.454
	avg_precision	0.001	0.268	0.294	0.322	0.360	0.375	0.390	0.404	0.480	0.537
	tot_precision	0.001	0.257	0.286	0.313	0.353	0.382	0.408	0.418	0.480	0.544

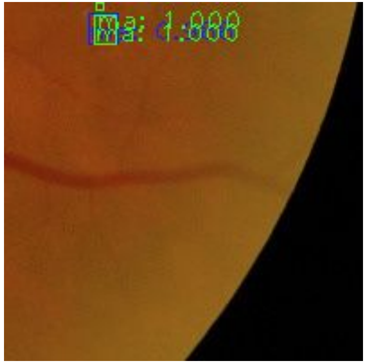
False Positives

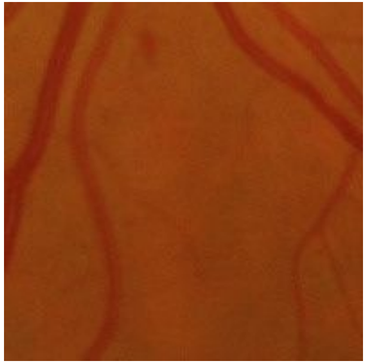


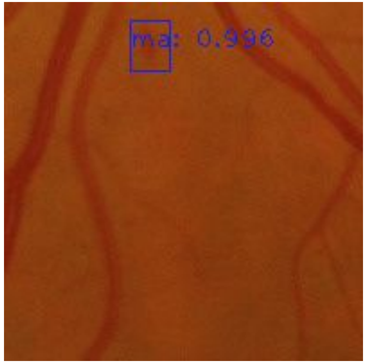
Original patches

Groundtruth
Predicted

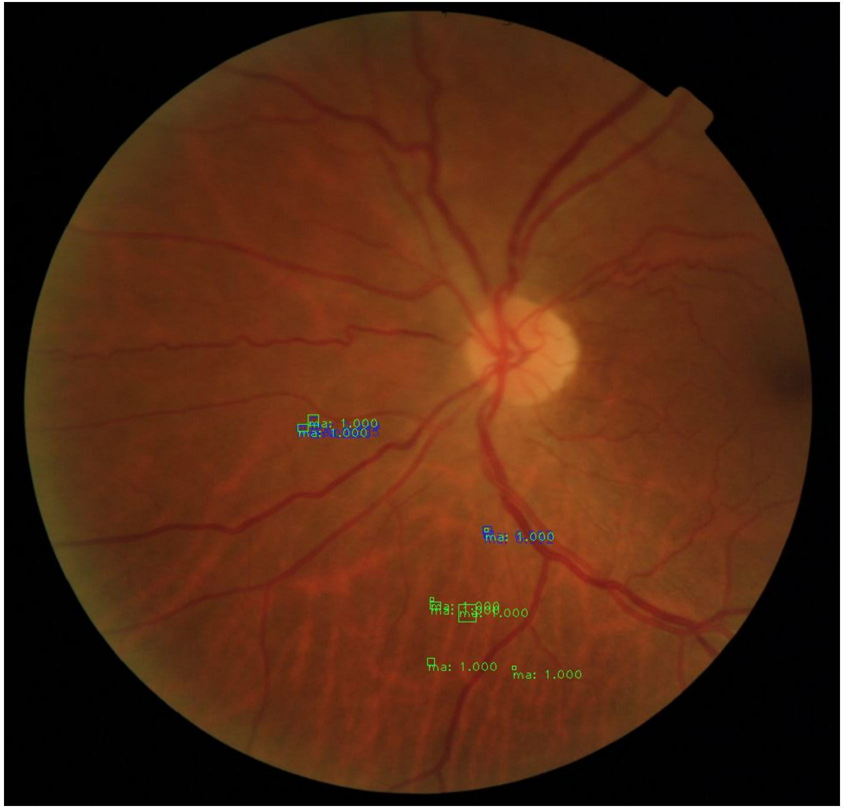






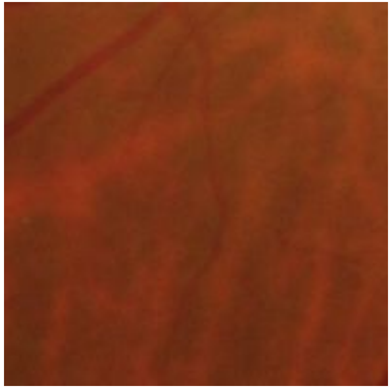
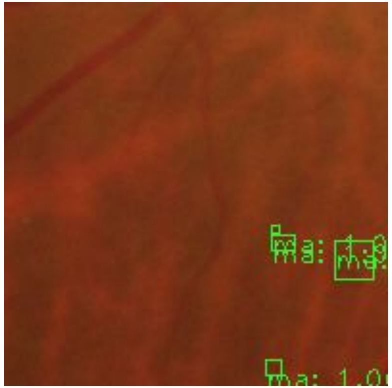
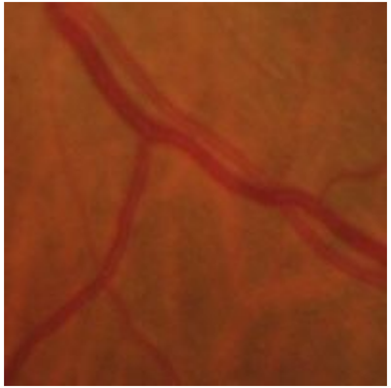
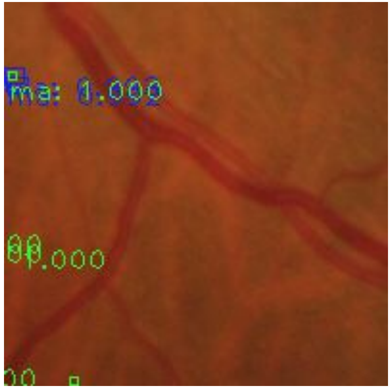


False Negatives



Original patches

Groundtruth
Predicted

Next steps

- Pretrain VGG with image-level labeled DR dataset like Messidor
- Apply ADA to improve the detection performance, e.g. using maximally disturbed contrast or resolution as data augmentation.
- Apply Adversarial Robust Cuts for segmentation
    a. ARC: Adversarial Robust Cuts for Semi-Supervised and Multi-Label Classification
    b. Another adversarial learning algorithm, can be used for pixel classification
    c. Compare with Mask RCNN

Q&A