Deep Co-Training for Semi-Supervised Image Segmentation

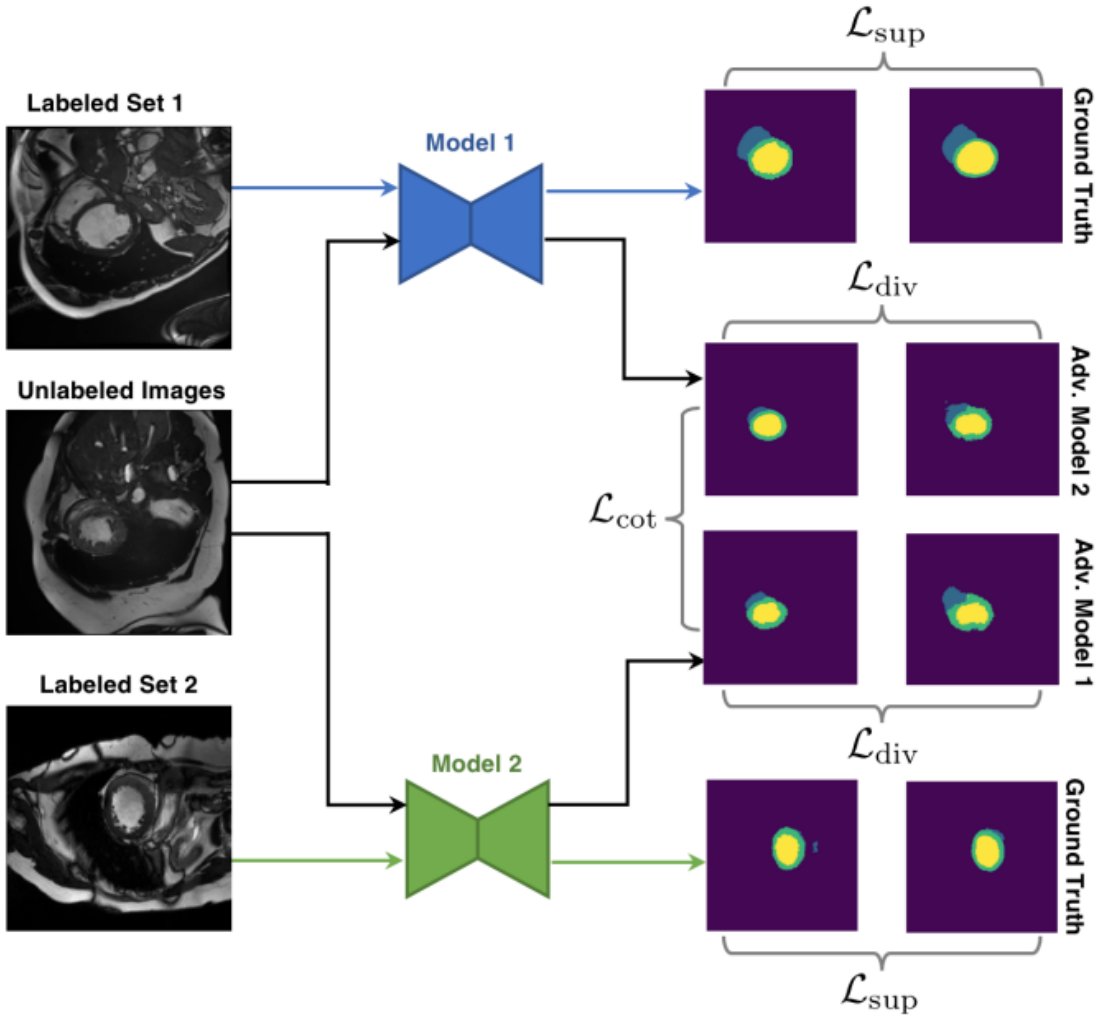
Jizong Peng[a,*], Guillermo Estrada[b], Marco Pedersoli[a], Christian Desrosiers[a]

[a] *ETS Montreal, 1100 Notre-Dame W., Montreal, Canada*
[b] *PUC-Rio, 225 Marquês de São Vicente Street, Rio de Janeiro, Brazil*

Co-training

- based on the idea that training examples can be described by two complementary (conditionally independent given the corresponding class labels) sets of features, called views.
- The general principle of this type of method is to simultaneously train classifiers for each view, using the labeled data, such that their predictions agree for unlabeled examples.
- Problem: Their application to visual tasks has so far been limited
  - there is no effective way to construct these sets from individual images. (multiplanar images)
  - Previous work: Using adversarial examples, built from both labeled and unlabeled images, for imposing diversity among the different classifiers.

Framework


Labeled Set 1
Unlabeled Images
Labeled Set 2
Model 1
Model 2
$\mathcal{L}_{\text{sup}}$
$\mathcal{L}_{\text{div}}$
$\mathcal{L}_{\text{cot}}$
$\mathcal{L}_{\text{div}}$
$\mathcal{L}_{\text{sup}}$
Ground Truth
Adv. Model 2
Adv. Model 1
Ground Truth

Loss

$$\mathcal{L}(\theta; \mathcal{D}) = \mathcal{L}_{\text{sup}}(\theta; \mathcal{S}) + \lambda_{\text{cot}} \mathcal{L}_{\text{cot}}(\theta; \mathcal{U}) + \lambda_{\text{div}} \mathcal{L}_{\text{div}}(\theta; \mathcal{D}).$$

$$\mathcal{L}_{\text{sup}}(\theta; \mathcal{S}) = \mathcal{L}^1_{\text{sup}}(\theta^1; \mathcal{S}^1) + \mathcal{L}^2_{\text{sup}}(\theta^2; \mathcal{S}^2).$$

$$\mathcal{L}^i_{\text{sup}}(\theta^i; \mathcal{S}^i) = \mathbb{E}_{(x,y)\in\mathcal{S}^i} \left[\sum_{j\in\Omega} \sum_{c\in\mathcal{C}} y_{jc} \log f^i_{jc}(x; \theta^i) \right]$$

$$\mathcal{L}_{\text{cot}}(\theta; \mathcal{U}) = \mathbb{E}_{x\in\mathcal{U}} \left[D_{\text{KL}}\Big(f^1(x; \theta^1) \,||\, \overline{f}(x; \theta) \Big) + D_{\text{KL}}\Big(f^2(x; \theta^2) \,||\, \overline{f}(x; \theta) \Big) \right]$$

$$= \mathbb{E}_{x\in\mathcal{U}} \left[\mathcal{H}\Big(\tfrac{1}{2}(f^1(x; \theta^1) + f^2(x; \theta^2)) \Big) - \frac{1}{2}\Big(\mathcal{H}(f^1(x; \theta^1)) + \mathcal{H}(f^2(x; \theta^2)) \Big) \right].$$

Jensen-Shannon divergence (JSD): average Kullack Liebler divergence (D_{KL}), distance between the class distributions(Shannon entropy)

$$\mathcal{L}_{\text{div}}(\theta; \mathcal{D}) = \mathbb{E}_{x\in\mathcal{D}} \left[\mathcal{H}\Big(f^1(x; \theta^1),\, f^2(g^1(x); \theta^2) \Big) + \mathcal{H}\Big(f^2(x; \theta^2),\, f^1(g^2(x); \theta^1) \Big) \right]$$

gi(x) is an adversarial example targeted on model fi(·; θi)

Adversarial example

- FGSM: Fast Gradient Sign Method (labeled)

$$L_{adv} := D[q(y|x_l), p(y|x_l + r_{adv}, \theta)]$$
$$where \quad r_{adv} := \arg\max_{r:\|r\|\leq\epsilon} D[q(y|x_l), p(y|x_l + r, \theta)]$$

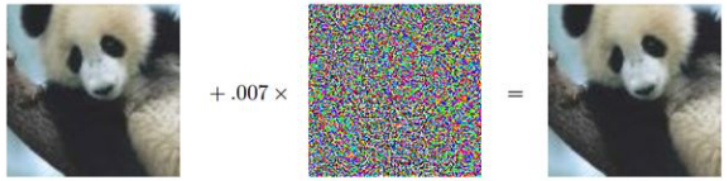


Figure 1: A demonstration of fast adversarial example generation applied to GoogLeNet (Szegedy et al., 2014a) on ImageNet. By adding an imperceptibly small vector whose elements are equal to the sign of the elements of the gradient of the cost function with respect to the input, we can change GoogLeNet's classification of the image. Here our ϵ of .007 corresponds to the magnitude of the smallest bit of an 8 bit image encoding after GoogLeNet's conversion to real numbers.

- VAT: virtual adversarial training (unlabeled)

$$L_{qadv} := D[q(y|x_*), p(y|x_* + r_{qadv}, \theta)]$$
$$where \quad r_{qadv} := \arg\max_{r:\|r\|\leq\epsilon} D[q(y|x_*), p(y|x_* + r, \theta)]$$

FGSM中的y是one-hot向量的真实标签，
而VAT中的y是预测分布p(y | x)

Algorithm

Algorithm 1: Deep Co-Training Segmentation (*training*)

Input: Labeled images $\mathcal{S} = \{(x_1, y_1), \ldots, (x_m, y_m)\}$;
Input: Unlabeled images $\mathcal{U} = \{x_1, \ldots, x_n\}$;
Input: Number of views k;
Output: Network parameters $\{\theta^i\}_{i=1}^k$;

Initialize network parameters θ^i, $i = 1, \ldots, k$;

for epoch $= 1, \ldots, E_{\max}$ **do**

  for iter $= 1, \ldots, T_{\max}$ **do**

    Randomly choose two different networks θ^{i_1} and θ^{i_2};

    Draw two batches $\mathcal{S}^{i_1}$, $\mathcal{S}^{i_2} \subset \mathcal{S}$ of b labeled images (x, y) (with replacement);

    Draw a single batch $\mathcal{U}^b \subset \mathcal{U}$ of b unlabeled images x;

    Compute adversarial examples $g^{i_1}(x)$ for all $x \in \mathcal{S}^{i_1} \cup \mathcal{U}^b$, and $g^{i_2}(x)$ for all $x \in \mathcal{S}^{i_2} \cup \mathcal{U}^b$, using Eq. (6) or (7);

    Let $\mathcal{L} = \mathcal{L}_{\text{sup}} + \lambda_{\text{cot}} \mathcal{L}_{\text{cot}} + \lambda_{\text{div}} \mathcal{L}_{\text{div}}$, as defined in Eq. (2)-(5), using $\mathcal{S}^{i_j}$ for the supervised loss of model i_j;

    Compute gradients w.r.t. $\mathcal{L}$ and update parameters θ^{i_j}, $j = 1, 2$, using back-propagation;

  Update learning rate and parameters $\lambda_{\text{cot}}, \lambda_{\text{div}}$ as in Eq. (8);

return $\{\theta^i\}_{i=1}^k$;

Diversity!

ACDC dataset

Method		DSC (%)			
		RV	Myo	LV	Mean
Full		81.96 (0.15)	85.39 (0.20)	91.82 (0.15)	86.39 (0.10)
Pseudo Label [46]		74.60 (0.32)	78.91 (0.21)	85.79 (0.17)	79.77 (0.14)
VAT [59]		72.78 (0.39)	80.81 (0.21)	87.60 (0.18)	80.39 (0.15)
Mean Teacher [29]		74.62 (1.10)	80.66 (0.61)	86.75 (0.27)	80.68 (0.41)
Independent	avg	68.82 (1.90)	78.30 (1.55)	85.92 (0.62)	77.68 (1.48)
	voting	68.28 (1.61)	79.94 (1.00)	86.41 (0.29)	78.21 (0.89)
JSD	avg	74.75 (1.69)	81.85 (0.42)	89.73 (0.58)	82.11 (0.44)
	voting	75.06 (1.87)	82.64 (0.57)	**90.31 (0.47)**	82.67 (0.67)
DCT-Seg (ours)	avg	77.51 (0.69)	82.43 (0.27)	89.85 (0.26)	83.26 (0.16)
	voting	**78.20 (0.70)**	**83.11 (0.20)**	90.22 (0.24)	**83.84 (0.10)**

hyper-parameter of L_{div}

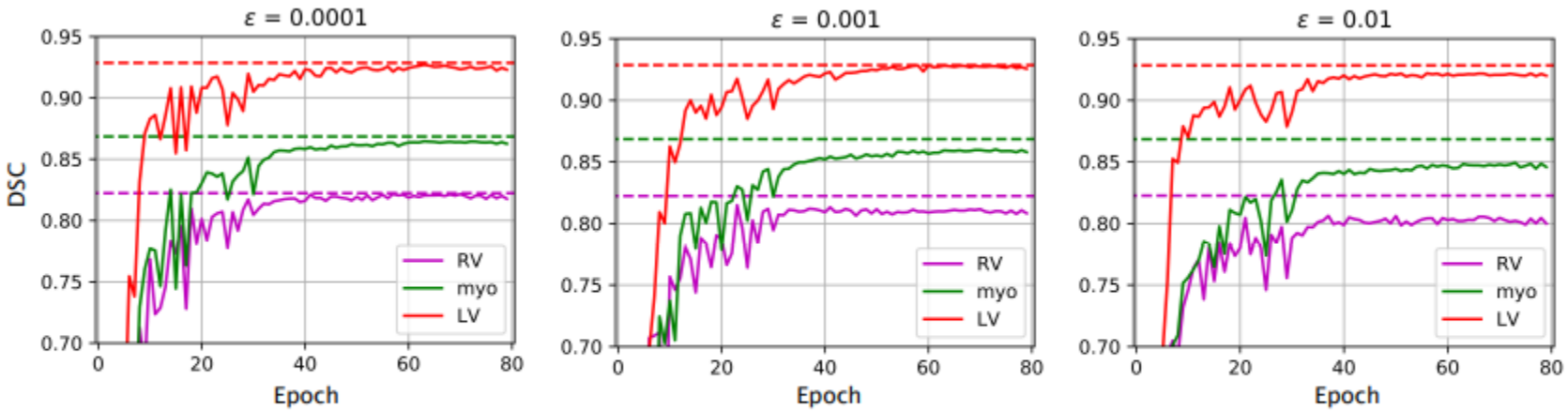


Figure 8: DSC score for models trained from scratch using only $\mathcal{L}_{\text{div}}$ with different ϵ. It can be seen that $\mathcal{L}_{\text{div}}$ acts as a similarity loss, especially when ϵ is small.

reference model (dashed line) and model trained from scratch (solid line),

Impact of adversarial noise

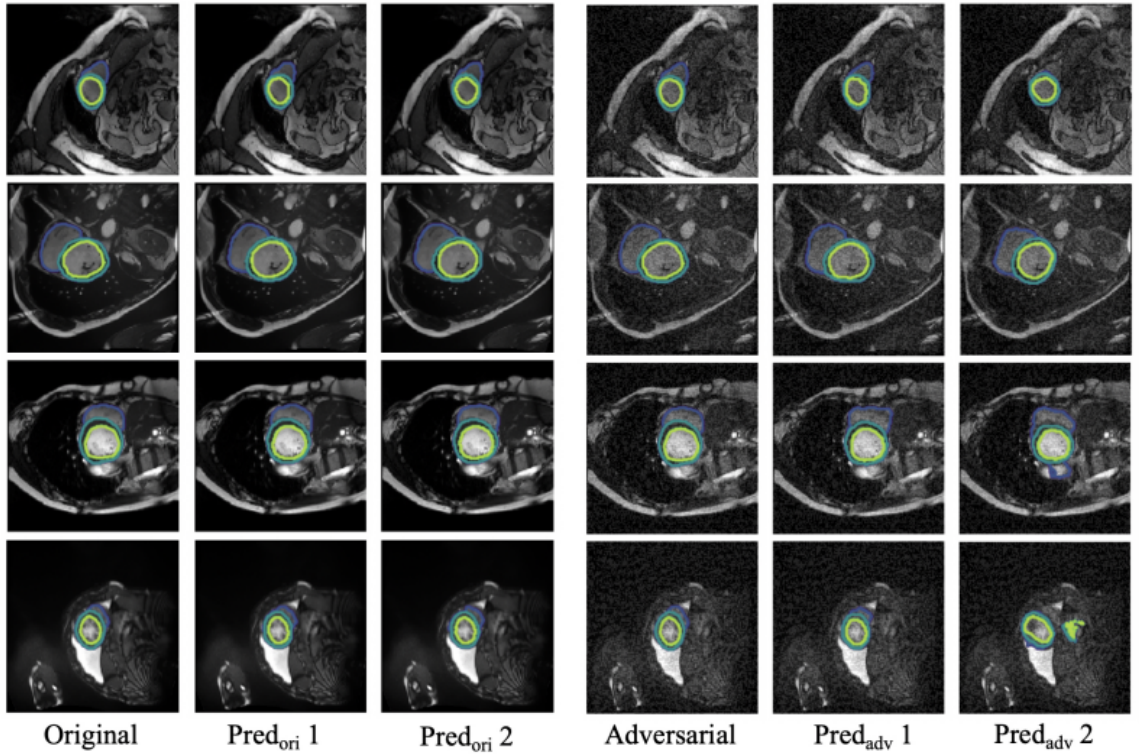


Figure 10: Impact of adversarial noise on prediction diversity. From left to right: original image (with GT contours), predictions of models 1 and 2 for the original image, adversarial image for model 2 (with GT contour), and predictions of model 1 and 2 for the adversarial image.

The End