Data Selection in Deep Learning

唐英鹏

2019.07.26

OUTLINE

MICCAI 19

Assessing Reliability and Challenges of Uncertainty Estimations for Medical Image Segmentation

Alain Jungo and Mauricio Reyes, University of Bern, Switzerland

MICCAI 18

Efficient Active Learning for Image Classification and Segmentation Using a Sample Selection and Conditional Generative Adversarial Network

Dwarikanath Mahapatra, Behzad Bozorgtabar, Jean-Philippe Thiran, Mauricio Reyes

MICCAI 18

Training Multi-organ Segmentation Networks with Sample Selection by Relaxed Upper Confident Bound

Yan Wang, Yuyin Zhou, Peng Tang, Wei Shen, Elliot K. Fishman, and Alan L. Yuille

Arvix 19

BAOD: Budget-Aware Object Detection

Alejandro Pardo, Mengmeng Xu, Ali Thabet, Pablo Arbelaez, and Bernard Ghanem

Arvix 19

Active Adversarial Domain Adaptation

Jong-Chyi Su, Yi-Hsuan Tsai, Kihyuk Sohn, Buyu Liu, Subhransu Maji, Manmohan Chandraker

Assessing Reliability and Challenges of Uncertainty Estimations for Medical Image Segmentation

Alain Jungo[1](✉)[0000−0001−8327−4653] and Mauricio Reyes[1]

Healthcare Imaging A.I., Insel Data Science Center, Inselspital, Bern University Hospital, University of Bern, Switzerland
alain.jungo@artorg.unibe.ch

Uncertainty in Deep Learning

- **Baseline uncertainty: Softmax entropy**

$$H = -\sum_{c \in C} p_c log(p_c) / log(|C|) \in [0, 1]$$

- **MC dropout**[1]

  - Test time drop out.

- **Aleatoric uncertainty**[2]

  - Obtained by defining a network f with two outputs $[\hat{x}, \sigma^2] = f(x)$ and input x, where the outputs $\hat{x}$ and σ^2 are the mean and variance of the logits perturbed with Gaussian noise.

- **Ensembles**

  - QBC

- **Auxiliary network**

  - An auxiliary network is used to predict predict voxel-wise uncertainties of the segmentation model

1. ICML'16 **Gal, Y., Ghahramani, Z.:** Dropout as a bayesian approximation: Representing model uncertainty in deep learning.
2. NIPS'17 **Kendall, A., Gal, Y.**: What uncertainties do we need in bayesian deep learning for computer vision?

Assessing quality of uncertainties

- **Calibration**

  A model is said to be perfectly calibrated if its predictions $f(x)$ with confidence p do occur with a fraction p of the time ($P(y = 1|f(x)) = p) = p$ for the binary case).

- **Uncertainty-Error overlap**

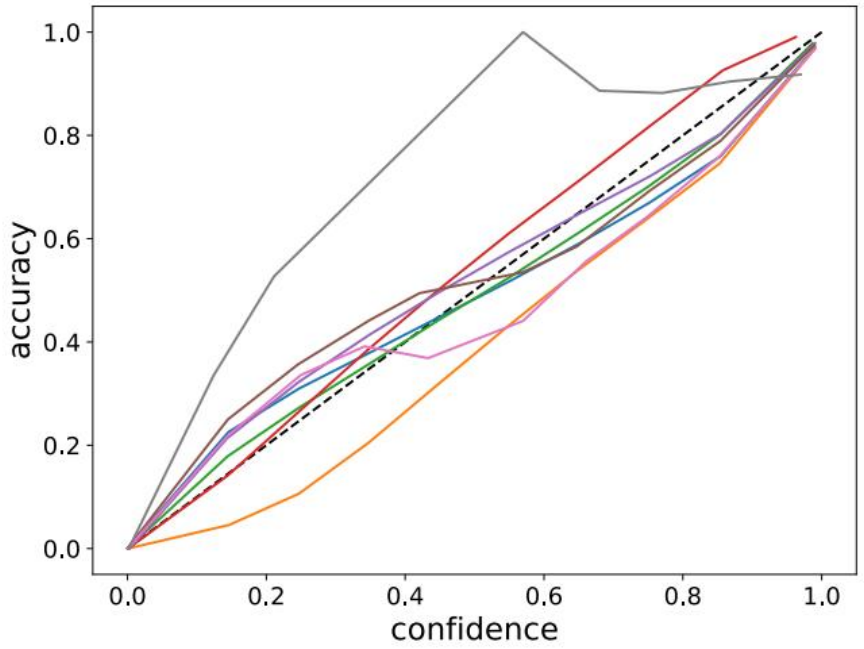
  The overlap (determined by the Dice coefficient) between the segmentation error and the thresholded uncertainty.

- **Corrections**

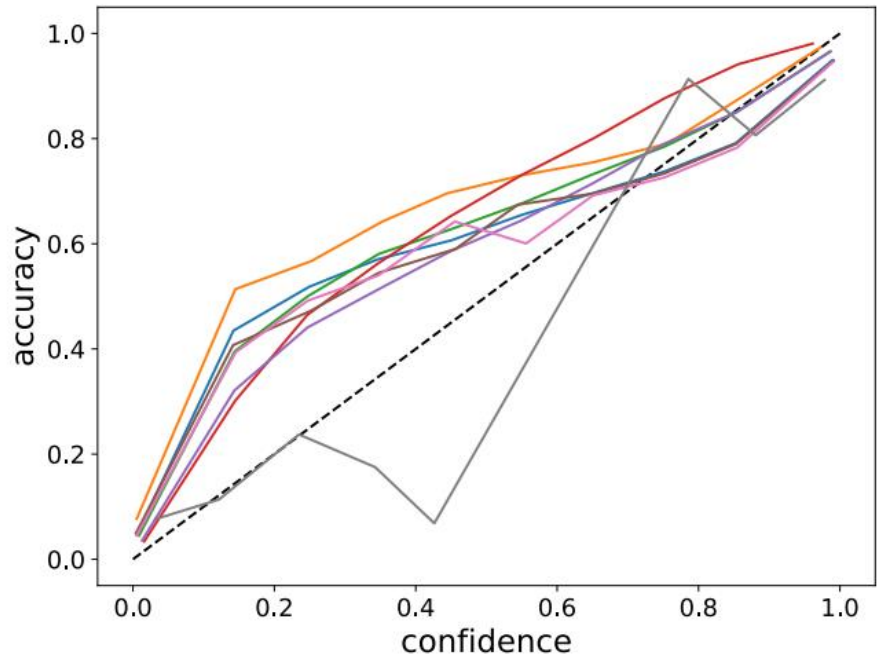
  Model performance improves by removing pixels with uncertainty larger than various percentile thresholds .

Calibration at the dataset level

BraTS



ISIC



baseline center baseline+MC center+MC ensemble auxiliary feat. auxiliary segm. aleatoric

Uncertainty in Segmentation

	BraTS				ISIC			
	ECE % ↓	U-E ↑	BnF ↑	Dice ↑	ECE % ↓	U-E ↑	BnF ↑	Dice ↑
baseline	0.925 (4)	0.432 (2)	0.39 (3)	0.874 (2)	7.256 (4)	0.424 (4)	0.26 (4)	0.814 (3)
center	1.758 (7)	0.409 (5)	**0.5** (1)	0.866 (5)	9.415 (8)	0.411 (6)	0.27 (3)	0.78 (6)
baseline+MC	**0.9** (1)	**0.433** (1)	0.36 (4)	0.874 (2)	7.36 (5)	0.428 (3)	0.24 (5)	0.813 (4)
center+MC	1.233 (6)	**0.433** (1)	0.27 (6)	0.868 (4)	8.766 (7)	0.428 (3)	0.17 (6)	0.794 (5)
ensemble	0.919 (2)	**0.433** (1)	0.32 (5)	**0.879** (1)	**7.131** (1)	0.431 (2)	0.31 (2)	**0.831** (1)
auxiliary feat.	0.923 (3)	0.427 (3)	0.48 (2)	0.874 (2)	7.216 (3)	0.421 (5)	**0.33** (1)	0.814 (3)
auxiliary segm.	0.925 (4)	0.412 (4)	0.48 (2)	0.874 (2)	7.212 (2)	**0.433** (1)	0.27 (3)	0.814 (3)
aleatoric	1.134 (5)	0.054 (6)	0.06 (7)	0.872 (3)	7.837 (6)	0.058 (7)	0.12 (7)	0.82 (2)

No overall best uncertainty measure was found among the studied methods. From our experiments we can conclude that methods that aggregate voxel-wise uncertainty to provide subject-level estimations are not reliable enough to be used as a mechanism to detect failed segmentations.

Efficient Active Learning for Image Classification and Segmentation Using a Sample Selection and Conditional Generative Adversarial Network

Dwarikanath Mahapatra[1(✉)], Behzad Bozorgtabar[2], Jean-Philippe Thiran[2], and Mauricio Reyes[3]

[1] IBM Research Australia, Melbourne, Australia
dwarim@au1.ibm.com
[2] Ecole Polytechnique Federale de Lausanne, Lausanne, Switzerland
{behzad.bozorgtabar,jean-philippe.thiran}@epfl.ch
[3] University of Bern, Bern, Switzerland
mauricio.reyes@istb.unibe.ch

Methods

1. The sample generator (cGAN) takes a test image and a manually segmented mask (and its variations) as input and generates realistic looking images.

2. A Bayesian neural network (BNN) [6] calculates generated images' informativeness and highly informative samples are added to the labeled image set.

3. Fine-tune the model with labeled set.

4. Test on a separate test set.

Experiment (image classification)

Table 1. Classification and Segmentation results for active learning framework of Xray images. DM-Dice metric and HD- Hausdorff distance

	Active learning (% labeled + Classifier)										FSL			
	10%		15%		25%		30%		35%		5-fold		35%	
	VGG16 [12]	ResNet18 [4]	[12]	[4]	[12]	[4]	[12]	[4]	[12]	[4]	[12]	[4]	[12]	[4]
Sens	70.8	71.3	75.3	76.2	89.2	89.7	91.5	91.8	91.7	91.9	92.1	92.4	78.1	78.5
Spec	71.1	71.9	76.0	76.8	89.9	90.5	92.1	92.4	92.4	92.5	92.9	93.1	78.4	78.7
AUC	74.3	75.0	78.7	79.4	92.5	93.0	94.9	95.1	95.2	95.3	95.7	95.9	80.6	81.0
DM	68.2		74.1		86.4		90.4		91.0		91.3		79.3	
HD	18.7		14.3		9.3		8.1		7.9		7.5		15.1	

Experiment (segmentation)

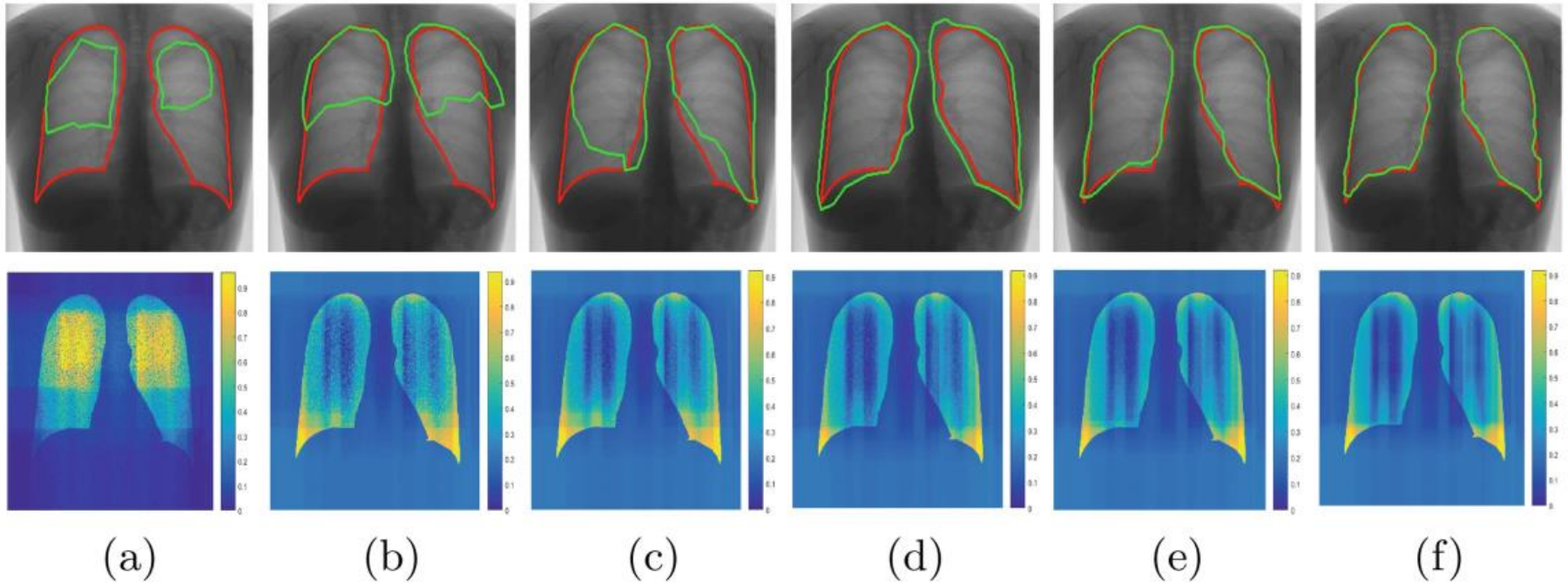


Fig. 3. Segmentation (top row) and uncertainty map (bottom row) results for different numbers of labeled examples in the training data (a) 5%; (b) 10%; (c) 20%; (d) 30%; (e) 35%; (f) 40%. Red contour is manual segmentation and green contour is the UNet generated segmentation.

Training Multi-organ Segmentation Networks with Sample Selection by Relaxed Upper Confident Bound

Yan Wang[1(✉)], Yuyin Zhou[1], Peng Tang[2], Wei Shen[1,3], Elliot K. Fishman[4], and Alan L. Yuille[1]

[1] Johns Hopkins University, Baltimore, USA
ywang372@jhu.edu
[2] Huazhong University of Science and Technology, Wuhan, China
[3] Shanghai University, Shanghai, China
[4] Johns Hopkins University School of Medicine, Baltimore, USA

Key idea

1. Employ data sampling strategy to train model efficiently.

2. Hard example mining yields faster training, higher accuracy.

3. Upper Confident Bound (UCB) exploits a range of hard samples rather than being stuck with a small set of very hard ones, which mitigates the influence of annotation errors during training.

4. In our RUCB, we relax this policy by selecting hard samples from a larger range.

Multi-armed bandit

- **Multi-armed bandit (MAB) problem**

  In a K-armed bandit problem, each arm k = 1, ..., K is recorded by an unknown distribution associated with an unknown expectation. In each trial t = 1, ..., T, a learner takes an action to choose one of K alternatives g(t) ∈ {1, ..., K} and collects a reward $x_{g(t)}^{(t)}$. The objective of this problem is to maximize the long-run cumulative expected reward $\sum_{t=1}^{T} x_{g(t)}^{(t)}$. But, as the expectations are unknown, the learner can only make a judgement based on the record of the past trails.

- **Sample selection as MAB**

  - Each sample is an arm.
  - The reward of a sample is defined as the network loss function.
  - Exploiting hard samples and exploring less frequently visited samples.

Upper Confident Bound

- **UCB strategy**

exploitation exploration

At trial *t*, the UCB selects the alternative *k* maximizing $\bar{x}_k + \sqrt{\frac{2\ln n}{n_k}}$ where $\bar{x}_k = \sum_{t=1}^{n} x_k^{(t)} / n_k$

n_k is the number of times alternative *k* has been selected so far and *n* is the total number of trail done.

Reward:

- **RUCB strategy**
  - Hard samples are regarded as slices whose UCB scores are larger than μ.
  - We count the number of samples that lie in the range $[\mu + \alpha \cdot std\left(q^{(M)}\right), +\infty]$, denoted by K, where α is drawn from a uniform distribution [0, a] (a = 3 in our experiment).
  - Then a sample is selected randomly from the set whose UCB score is in the range.

$$\mu = \sum_{i=1}^{M} q_i^{(M)} / M \qquad q_i^{(n)} = \tilde{J}_i^{(n)} + \sqrt{\frac{2\ln n}{n_i}}$$

Experiment

Table 1. DSC (%) of sixteen segmented organs (mean ± standard deviation).

Organs	Uniform	OHEM	UCB	RUCB (ours)
Aorta	81.53 ± 4.50	77.49 ± 5.90	81.02 ± 4.50	81.03 ± 4.40
Adrenal gland	29.33 ± 16.26	31.44 ± 16.71	33.75 ± 16.26	36.76 ± 17.28
Celiac AA	34.49 ± 12.92	33.34 ± 13.86	35.89 ± 12.92	38.45 ± 12.53
Colon	77.51 ± 7.89	73.20 ± 8.94	76.40 ± 7.89	77.56 ± 8.65
Duodenum	63.39 ± 12.62	59.68 ± 12.32	63.10 ± 12.62	64.86 ± 12.18
Gallbladder	79.43 ± 23.77	77.82 ± 23.58	79.10 ± 23.77	79.68 ± 23.46
IVC	78.75 ± 6.54	73.73 ± 8.59	77.10 ± 6.54	78.57 ± 6.69
Left kidney	95.35 ± 2.53	94.24 ± 8.95	95.53 ± 2.53	95.57 ± 2.29
Right kidney	94.48 ± 9.49	94.23 ± 9.19	94.39 ± 9.49	95.40 ± 3.62
Liver	96.03 ± 1.70	90.43 ± 4.74	95.68 ± 1.70	96.00 ± 1.28
Pancreas	77.86 ± 9.92	75.32 ± 10.42	78.25 ± 9.92	78.48 ± 9.86
SMA	45.36 ± 14.36	47.18 ± 12.75	44.63 ± 14.36	49.59 ± 13.62
Small bowel	72.35 ± 13.30	67.44 ± 13.22	72.16 ± 13.30	72.88 ± 13.98
Spleen	95.32 ± 2.17	94.56 ± 2.41	95.16 ± 2.17	95.09 ± 2.44
Stomach	90.62 ± 6.51	86.37 ± 8.53	90.70 ± 6.51	90.92 ± 5.62
Veins	64.95 ± 19.96	60.87 ± 19.02	62.70 ± 19.96	65.13 ± 20.15
AVG	73.55 ± 10.28	71.08 ± 11.20	73.47 ± 10.52	74.75 ± 9.88

* Online hard example mining (OHEM)

BAOD: Budget-Aware Object Detection

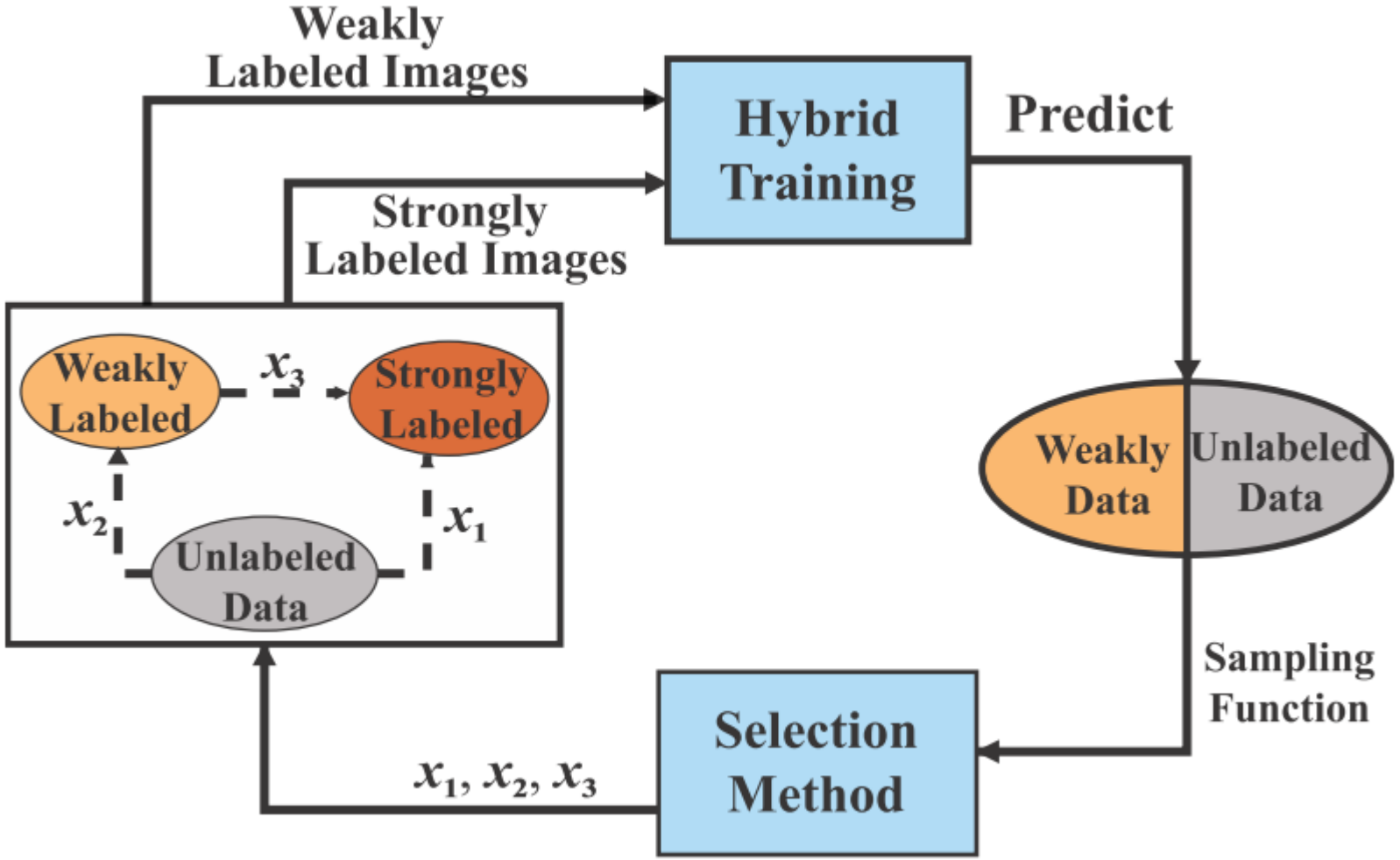
Alejandro Pardo *[1], Mengmeng Xu *[2], Ali Thabet[2], Pablo Arbeláez[1], and Bernard Ghanem[2]

[1]Universidad de los Andes, Colombia
[2]King Abdullah University of Science and Technology (KAUST), Saudi Arabia

Key idea

- **Motivation**
  - Drawing bounding box is expensive.
  - Strongly annotating (bounding box & label) some examples is not necessary.
- **Proposed method**
  - Select both training images and their annotation scheme.
  - Propose a hybrid supervised learning strategy that combines category and localization information to train a robust detection model that handles both weak and strong annotations.

Pipeline


Weakly
Labeled Images
Strongly
Labeled Images
Hybrid
Training
Predict
Weakly
Labeled
x_3
Strongly
Labeled
x_2
x_1
Unlabeled
Data
Weakly
Data
Unlabeled
Data
Sampling
Function
Selection
Method
x_1, x_2, x_3

Selection criterion

- **Uncertainty of a detection example: mean entropy of M bounding boxes**

$$s_k = \frac{1}{M} \sum_{i=1}^{M} \sum_{p \in \boldsymbol{p}_i} -p \log(p)$$

- **Select images and their annotation scheme**

$$\max_{\boldsymbol{x_1}, \boldsymbol{x_2}, \boldsymbol{x_3} \in \{0,1\}^N} \boldsymbol{s}^\top (\boldsymbol{x_1} + \boldsymbol{x_3}) + (\mu \mathbf{1} - \boldsymbol{s})^\top \boldsymbol{x_2}$$

Only weakly annotated samples can take action 3

$$\text{s.t.} \begin{cases} \boldsymbol{x_3} \leq \boldsymbol{\psi} \\ \boldsymbol{x_1} + \boldsymbol{x_2} \leq \mathbf{1} - \boldsymbol{\psi} \end{cases}$$

Enforce that the budget be used as much as possible in each active step

$$\begin{cases} \mathbf{1}^\top (a\boldsymbol{x_1} + b\boldsymbol{x_2} + c\boldsymbol{x_3}) \leq d \\ \mathbf{1}^\top (a\boldsymbol{x_1} + b\boldsymbol{x_2} + c\boldsymbol{x_3}) \geq d - a \end{cases}$$

- x1: indicator of annotate strongly
- x2: indicator of annotate weakly
- x3: indicator of weak to strong
- a : cost of strongly annotation
- b : cost of weakly annotation
- c : cost of weak to strong
- d : budget
- ψ : indicator vector for weakly annotated (if yes 1, otherwise 0)

Alternative informativeness metric

- **Learning Active Learning (LAL)**
  - Simulate active learning process in another dataset.
  - Design meta feature, collect pairs T = {meta feature, model improvement}.
  - Train a example scorer with T.
  - Predict the model improvement for unlabeled example.
- **Optimization Based Active Selection using LAL**
  - build a feature vector $v = [O_t; s]$ that concatenates both the current model state O_t, represented as the average precision curves under five different Intersection over Union (IoU) thresholds, and the uncertainty scores s.
  - Train a SVR to regress the actual increment in mAP performance for both weak and strong annotation actions. The outputs are denoted as $\boldsymbol{h}_w$ and $\boldsymbol{h}_s$

$$\delta_t \approx \boldsymbol{h}_w^\top \boldsymbol{x_2} + \boldsymbol{h}_s^\top (\boldsymbol{x_1} + \boldsymbol{x_3})$$

Hybrid Supervision for Object Detection

- **Teacher-student Model**

  - Use strongly annotated examples to train a teacher network.

  - Use teacher network to predict unlabeled and weakly labeled examples.

  - Use strongly annotated examples and unlabeled and weakly labeled examples with predicted pseudo-labels to train a student network.

I. Radosavovic, P. Dollar, R. Girshick, G. Gkioxari, and ´ K. He. Data distillation: Towards omni-supervised learning. Tech report, 2017.

Experiment - Advantages of Uncertainty Sampling and Hybrid Training

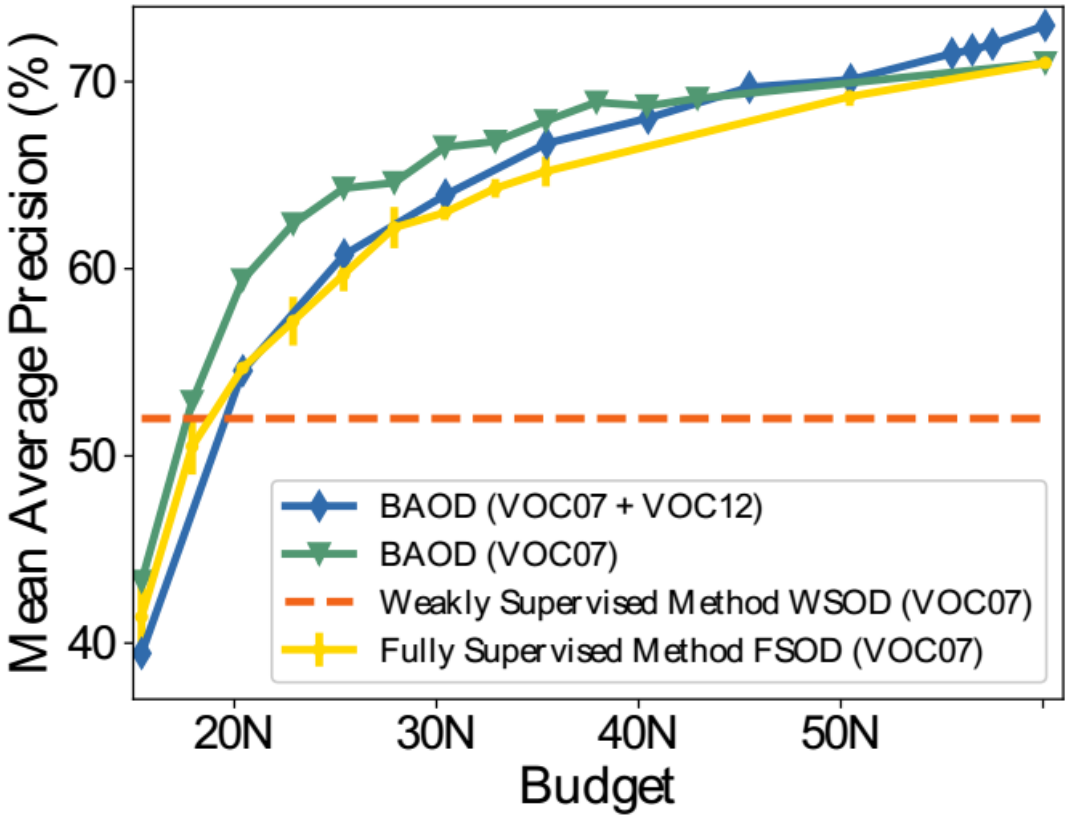
Table 1: **Budget-Average mAP using fully and hybrid training pipelines with random and uncertainty selection.** Uncertainty sampling is always better than random sampling selection, and hybrid training is always better than FSOD.

Selection Method	FSOD		Hybrid	
Sampling Function	RS	US	RS	US
Low Budget Range	52.5	53.1	**56.4**	55.1
Mid Budget Range	62.5	63.6	64.5	**65.8**
High Budget Range	67.9	68.7	68.7	**69.3**

Budget range: We take three ranges [10%; 30%], [30%; 50%], and [50%; 100%] to evaluate our experiment

a = 34.5 ; *b* = 1.6 ; *c* = *a* - *b*

Experiment - Optimization-Based Active Selection



at most budgets. Given a larger unlabeled image pool (Blue -◇- curve, VOC07+VOC12), our BAOD can reach a higher mAP using the same budget needed to annotate VOC07 with instance-level labels. Since the dataset is finite, WSOD cannot increase its per-

Table 2: **Budget-Average mAP using simple hybrid training and optimization methods.** US based optimization is slightly better than LAL one. The optimization methods perform better than the simple hybrid random selection and uncertainty selection methods in the three budget ranges.

Selection Method	**Hybrid**		**Optimization**	
Sampling Function	**RS**	**US**	**LAL**	**US** (BAOD)
Low Budget Range	56.4	55.1	56.3	**57.1**
Mid Budget Range	64.5	65.8	65.9	**66.0**
High Budget Range	68.7	69.3	69.3	**69.5**

Learning curves

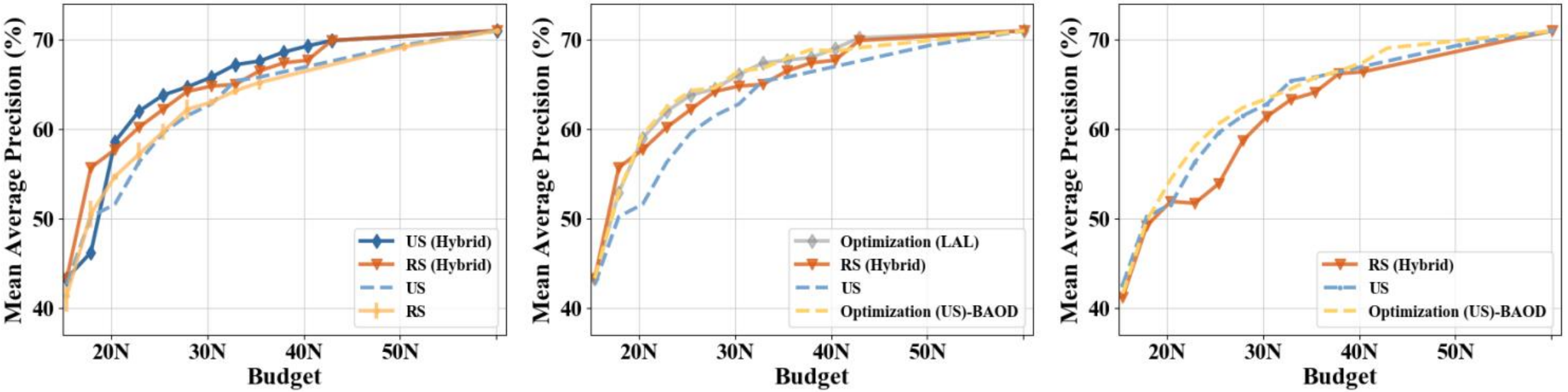


Figure 6: *left*: **Budget-mAP curves using fully and hybrid training pipelines.** Orange bars compare the two training pipelines using random sample while Blue bars show both pipelines using US sampling. *Middle*: **Budget-mAP curves using FSOD, hybrid training, and optimization methods.** Blue bars show US using FSOD. Orange bars show RS hybrid baseline. Gold bars show US Optimization (BAOD). Dark Grey Bars show LAL Optimization. *right*: **Budget-mAP curves using a lower cost for strong annotations**. Blue Bars represent FSOD with US sampling. Orange Bars represent Hybrid training with RS sampling. Gold Bars represent US Optimization. All the methods use a smaller gap between the weak and strong annotation costs.

Experiment - Easy Images and Weak Annotation First

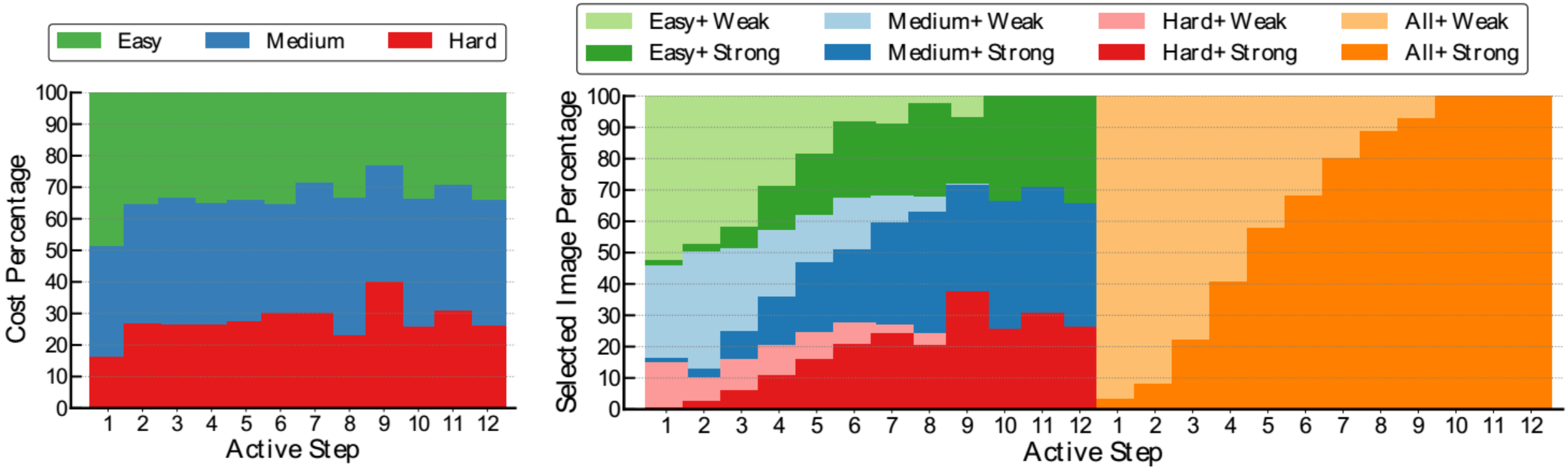


Figure 4: **Comparison of the cost and image number on every active selection step.** *Left*: Budget usage distribution to learn different difficulty categories. More budget is used to annotate Easy images (green area) at beginning. The cost spent on Hard images (red area) grows up when the active model is mature. *Right*: Selected images distribution in different difficulty categories and different annotation type. The selection agent gives more weak annotations (light color) at the first steps. Given more budget, the proportion of strong annotations (dark color) increase. We run out of unlabeled images after 9-th step. The mapping is motivated and shown in the supplementary material.

Experiment - Qualitative Results of the Active Selection

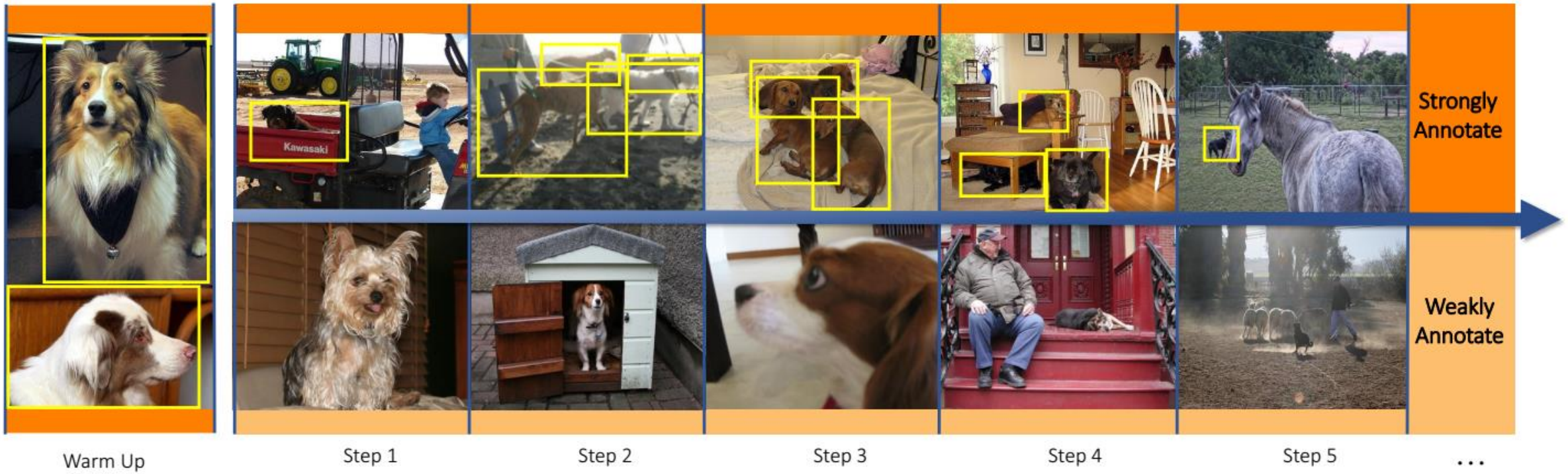


Figure 5: **Visualization of the selected images in each step.** *Left*: Two examples in the warm-up set which is fully annotated by 10% budget. *Up-Right*: Strongly annotated images per step. They are hard examples including occlusion, multiple instance or tiny scale. *Bottom-Right*: Weakly annotated images per step. They are simple in the beginning but the difficulty increases when the detector is mature.

Experiment - Sample from Small Uncertainty Images

Selection Method	FSOD		FSOD	Opt.
Sampling Function	LUS	SUS	RS	US (BAOD)
Low Budget Range	52.3	53.1	53.1	57.1
Mid Budget Range	63.1	62.8	62.8	66.0
High Budget Range	68.6	67.7	68.3	69.5

When the budget is low (10%-30%), collecting images with small uncertainty score (shown as SUS) is more effective. However, when the dataset size is large enough, choosing large uncertainty scores (shown as LUS) is preferable.

Experiment - Reinforcement Learning

- **State** : O_t (represented as the average precision curves under five different Intersection over Union (IoU) thresholds)

- **Action** : (1) asking for low/medium/high uncertainty images and (2) giving a strongly/weakly annotation. In conclusion, the RL agent gives a choice a from three by two possible actions matrix.

- **Reward** : model increment performance.

- **Learning** : Q-learning, which learns to approximate Q-function $Q^{\pi}(a; O_t)$ with a three-layer neural network.

Selection Method	FSOD		RL	Opt.
Sampling Function	LUS	SUS		US (BAOD)
Low Budget Range	52.3	53.1	56.6	57.1
Mid Budget Range	63.1	62.8	64.1	66.0
High Budget Range	68.6	67.7	68.5	69.5

FSOD
RS
53.1
62.8
68.3

Active Adversarial Domain Adaptation

Jong-Chyi Su*[1] Yi-Hsuan Tsai[2] Kihyuk Sohn[2] Buyu Liu[2]
Subhransu Maji[1] Manmohan Chandraker[2]

[1]University of Massachusetts, Amherst [2]NEC Laboratories America

Qualitative Results of the Active Selection

- **Domain adversarial neural network (DANN)**

  - **Feature extractor**: G_f, extract feature for samples in source and target domain
  - **Discriminator**: G_d, Predict where the sample is coming from (source or target domain)
  - **Classifier**: G_y

$$\mathcal{L}_d = \mathbb{E}_{x \sim p_{\mathcal{S}}(x)} \left[\log G_d(G_f(x)) \right] + \mathbb{E}_{x \sim p_{\mathcal{T}}(x)} \left[\log(1 - G_d(G_f(x))) \right]$$

$$\min_{\theta_f, \theta_y} \max_{\theta_d} \mathcal{L}_c(G_y(G_f(x)), y) + \lambda \mathcal{L}_d$$

Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain adversarial training of neural networks. The Journal of Machine Learning Research, 17(1):2096-2030, 2016.

Importance sampling

- **Setting**

  labeled data is only available from the source domain, the goal of our sample selection is to find the most informative data from unlabeled target domain.

- **Importance weighted empirical risk minimization (IWERM)**

$$\min_{\theta_f,\theta_y} \mathbb{E}_{(x,y)\sim p_{\mathcal{S}}(x,y)}\left[\frac{p_{\mathcal{T}}(x)}{p_{\mathcal{S}}(x)}\mathcal{L}_c\big(G_y(G_f(x)),y\big)\right]$$

$$\text{where } w(x)=\frac{p_{\mathcal{T}}(x)}{p_{\mathcal{S}}(x)} \text{ is an importance of each labeled data}$$

Which data is more important during optimization?

1. the data with higher empirical risk.
2. the one with higher importance, i.e., larger density in the target distribution pT (x) but lower in the source pS(x).

Query strategy

- Q:The empirical risk of unlabeled data can not be computed?
- A:Use entropy instead.

$$s(x) = \frac{1 - G_d^*(G_f(x))}{G_d^*(G_f(x))} \mathcal{H}(G_y(G_f(x)))$$

- Q:the importance estimation of high-dimensional data is difficult?
- A:with adversarial training, the optimal discriminator [12] is obtained at:

$$G_d^*(\hat{x}) = \frac{p_{\mathcal{S}}(x)}{p_{\mathcal{S}}(x) + p_{\mathcal{T}}(x)} \Rightarrow w(x) = \frac{1 - G_d^*(\hat{x})}{G_d^*(\hat{x})}$$

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Advances in neural information processing systems, pages 2672-2680, 2014.

Query strategy

$$s(x) = \frac{1 - G_d^*(G_f(x))}{G_d^*(G_f(x))} \mathcal{H}(G_y(G_f(x)))$$

diversity cue　　uncertainty cue

select unlabeled target data which is less similar to the labeled ones in the source domain

Algorithm 1 AADA

Input: labeled source L_s; unlabeled target U_t;
labeled target $L_t = \emptyset$; budget per round b
Model: $\mathcal{M}$=$\{G_f, G_y, G_d\}$; feature extractor G_f;
class predictor G_y; discriminator G_d
Train $\mathcal{M}$ with (L_s, U_t)
for round $\leftarrow$ 1 to MaxRound **do**
  Compute $s(x)$ $\forall x \in U_t$ via (5)
  Select a set of b images z from U_t according to $s(z)$
  Get labels y_z from oracle
  $L_t \leftarrow L_t \cup (z, y_z)$
  $U_t \leftarrow U_t \setminus (z, y_z)$
  Train $\mathcal{M}$ with $(L_s \cup L_t, U_t)$

Experiment setting

- **Training schemes**

1) Adversarial Training: we train the classifier via (2) using $(L_s \cup L_t, U_t)$.

2) Joint Training: we train the classifier in a supervised way using $L_s \cup L_t$. Note that we still train a discriminator for sample selection but without adversarial training.

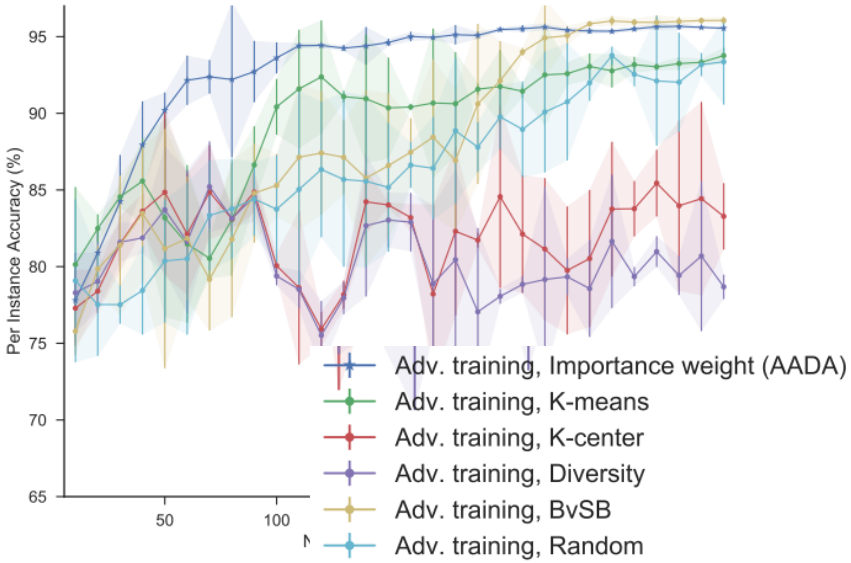
3) Fine-tuning: we train a classifier using L_s and then fine-tune it on L_t, both in a supervised way. Discriminator is trained in a similar manner to Joint Training.

4) Target Only: we train our classifier with L_t only.

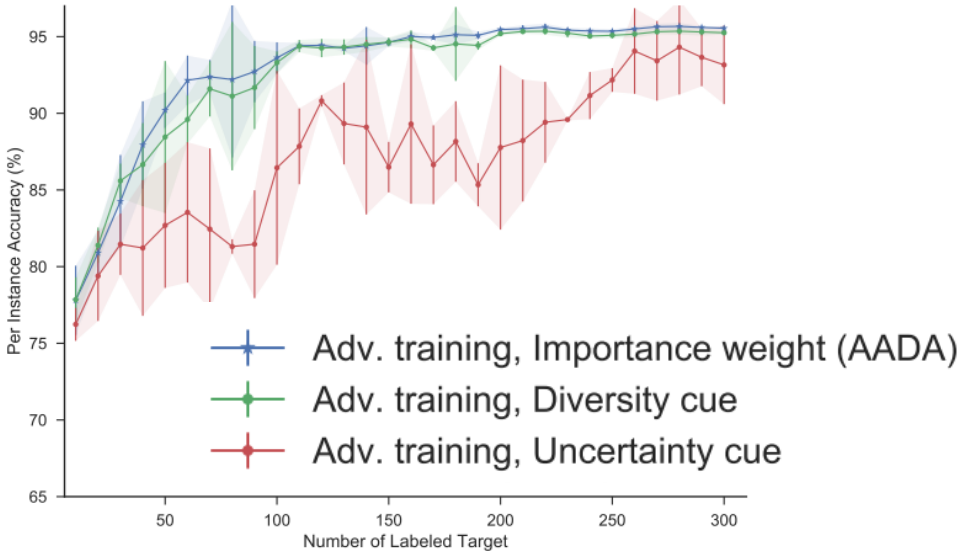
- **Sampling strategies**

1) Importance Weight
2) K-means Clustering
3) K-center (Core-set)
4) Diversity : Distance to all samples in *Lt* and obtain the average distance. Query top b.
5) Best-versus-Second Best (BvSB)
6) Random Selection

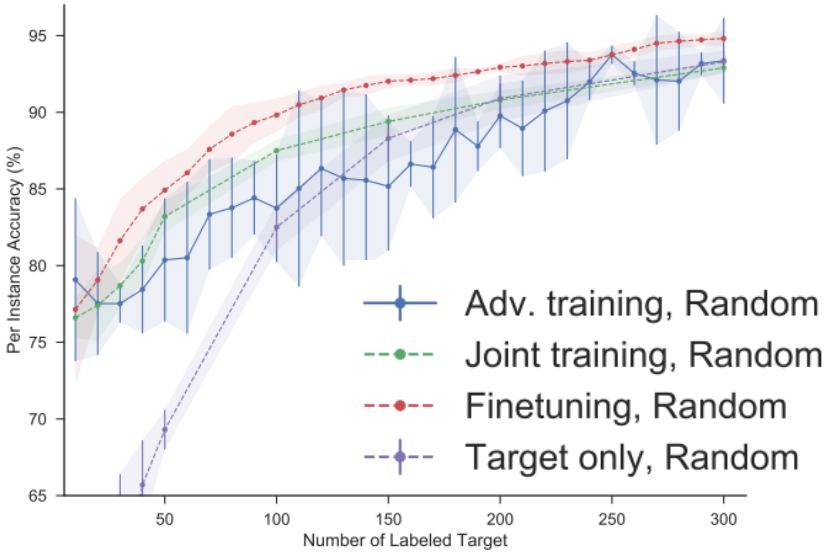
Exper



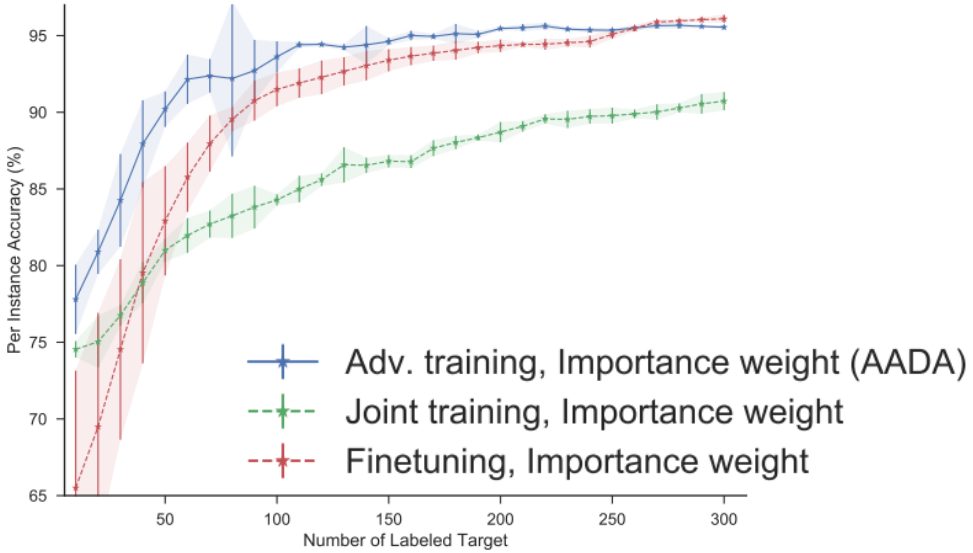
(a) Different sampling strategies with adversarial training.



(b) Different sampling cues with adversarial training.



(c) Different training schemes with random sampling.



(d) Different training schemes with importance weight.

Ablation studies on digit classification (SVHN -> MNIST)

Experiment

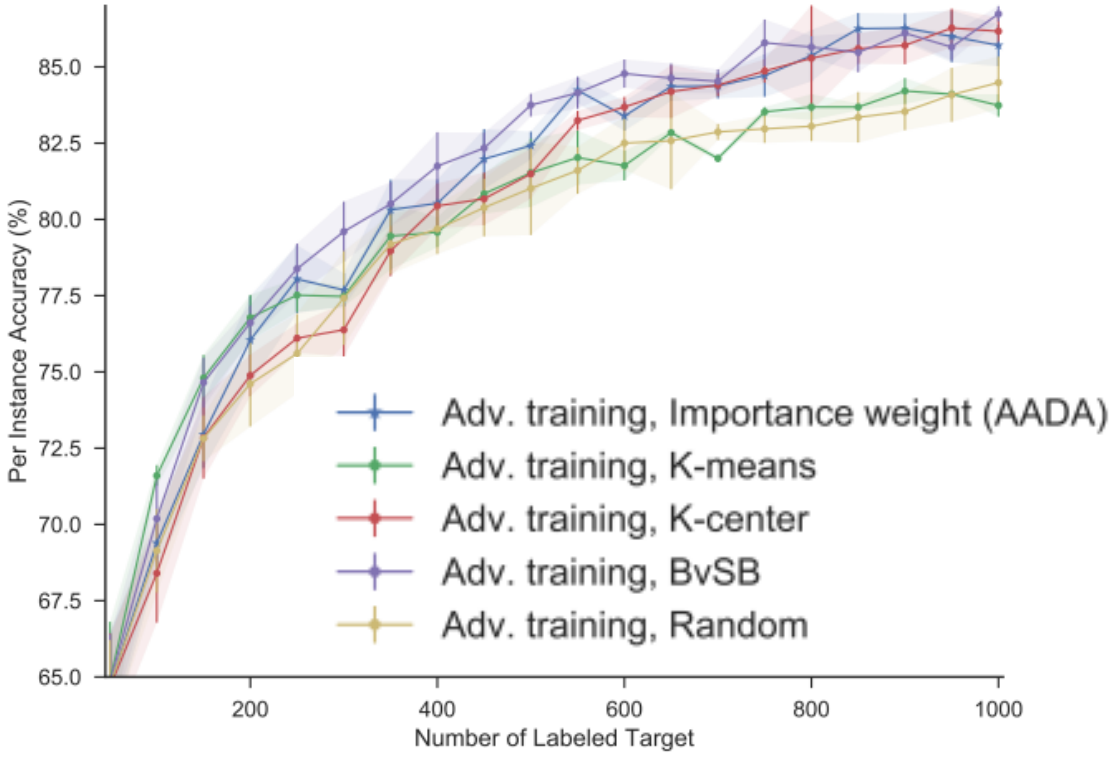


Figure 4: Object classification result (Office D → A). We compare different sampling methods with adversarial training. BvSB and AADA perform the best with 81.3% and 80.7% mean accuracy of 20 rounds separately.

Training	Sampling	Number of Labeled Target					
		10	20	30	50	100	200
Adversarial	Imp. weight	**49.4**	**53.3**	**54.6**	**57.4**	**60.4**	**62.3**
Adversarial	K-means	49.1	51.7	53.8	56.8	59.2	60.9
Adversarial	Entropy	48.9	50.9	52.3	54.3	58.1	61.0
Adversarial	Random	47.4	49.8	51.6	55.2	58.6	61.7
Joint	Imp. weight	48.5	52.1	53.5	56.2	58.6	60.5
Joint	Random	45.5	48.8	51.8	54.9	59.0	61.6
Fine-tuning	Random	41.0	46.0	48.7	51.4	56.0	59.8
Target only	Random	29.0	38.5	42.1	48.3	53.3	58.8

Table 1: Object detection results (KITTI → Cityscapes). Our AADA method (first row) outperforms all other baselines, including using adversarial training and other sample selection methods, as well as using different training schemes and random sampling.

Experiment

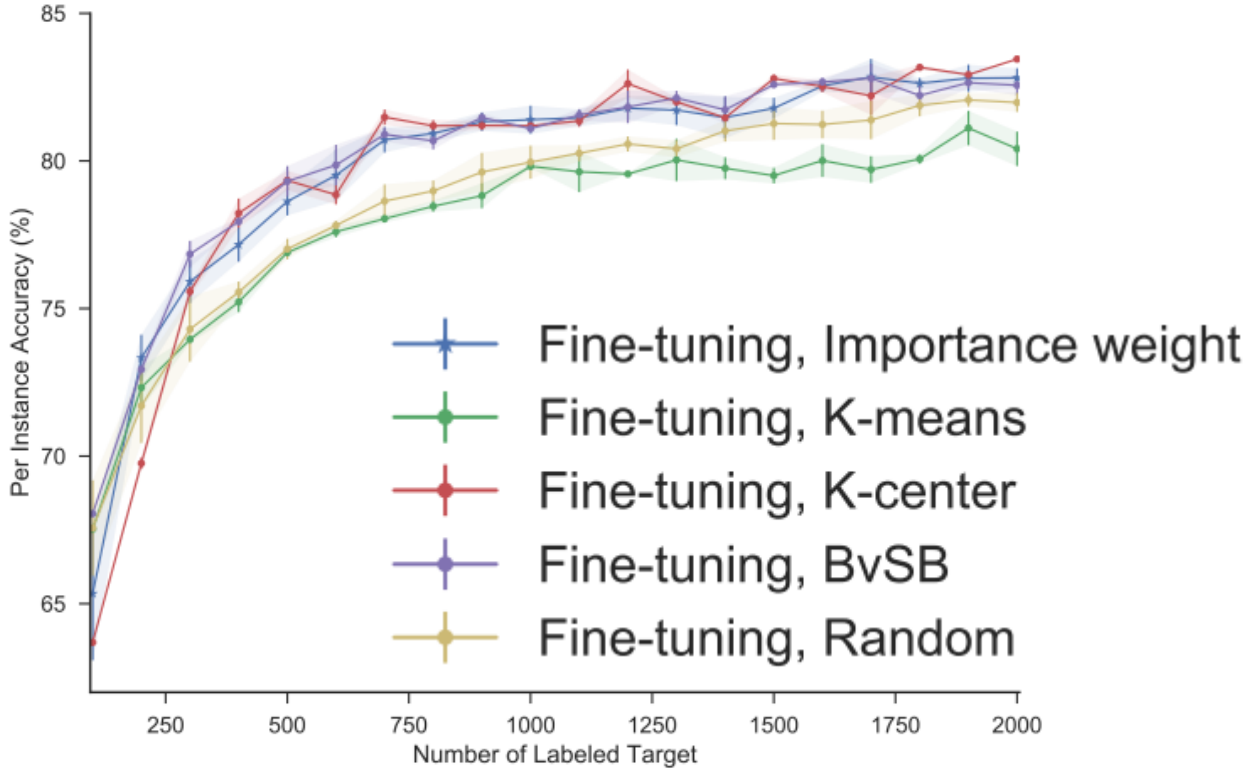


Figure 6: VisDA-18 result (synthetic → real). Here we use

- The source domain is composed of 78,222 synthetic images across 12 object categories rendered from 3D CAD models
- The target domain contains 5,534 real images

Experiment

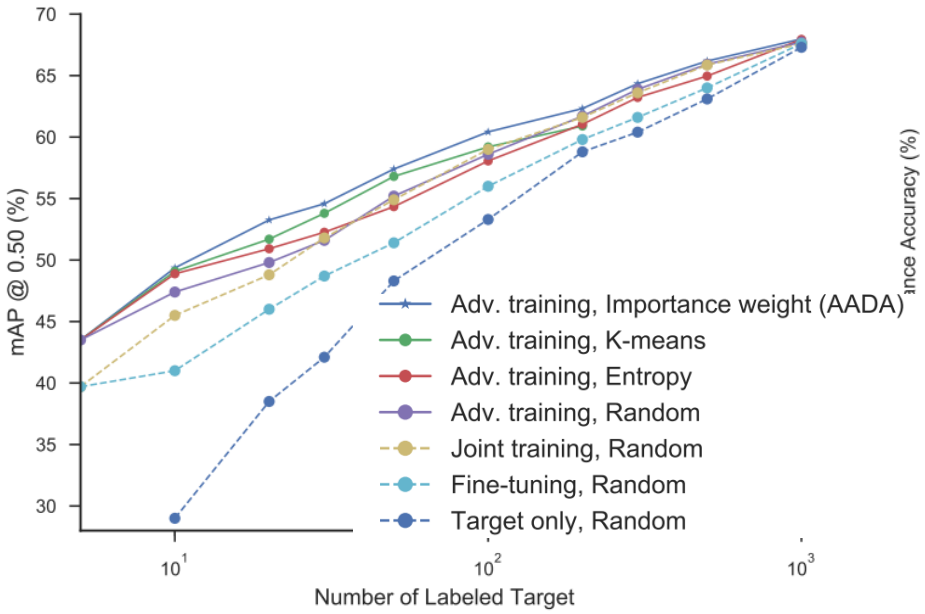


Figure A1: Object detection result (KITTI → Cityscapes) after 9 rounds. The x-axis is shown in log scale. The left-

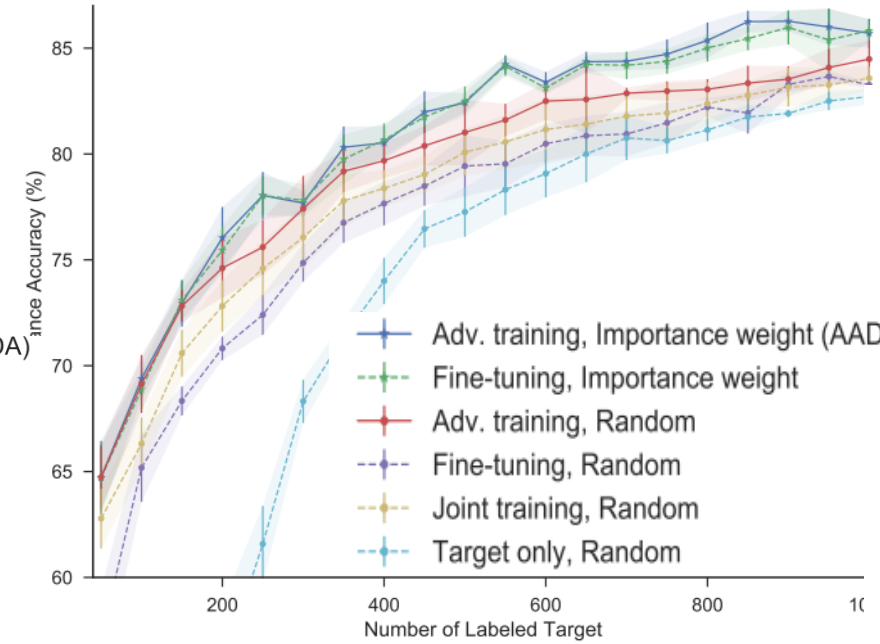


Figure A2: Comparing different training schemes on Office dataset (D → A). Adversarial training with importance weight for sampling (AADA) outperforms other baselines with different training schemes.

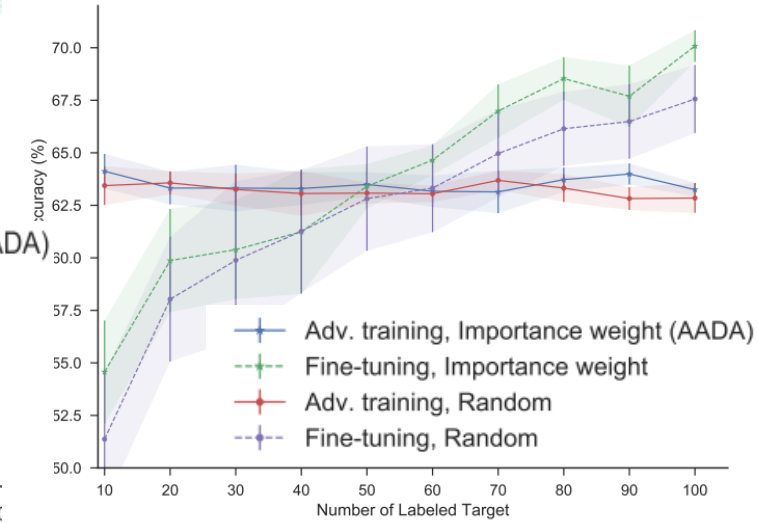


Figure A3: Comparing different training schemes on the VisDA dataset. Using adversarial training, the accuracy