May. 9, 2024

Learning CNN on ViT: A Hybrid Model to Explicitly Class-specific Boundaries for Domain Adaptation

Ba Hung Ngo [1,*], *Nhat-Tuong Do-Tran* [2,*], *Tuan-Ngoc Nguyen* [3], *Hae-Gon Jeon* [4], *Tae Jong Choi* [1,†]

[1]Graduate School of Data Science, Chonnam National University, South Korea
[2]Department of Computer Science, National Yang Ming Chiao Tung University, Taiwan
[3]Digital Transformation Center, FPT Telecom, VietNam, [4]AI Graduate School, GIST, South Korea

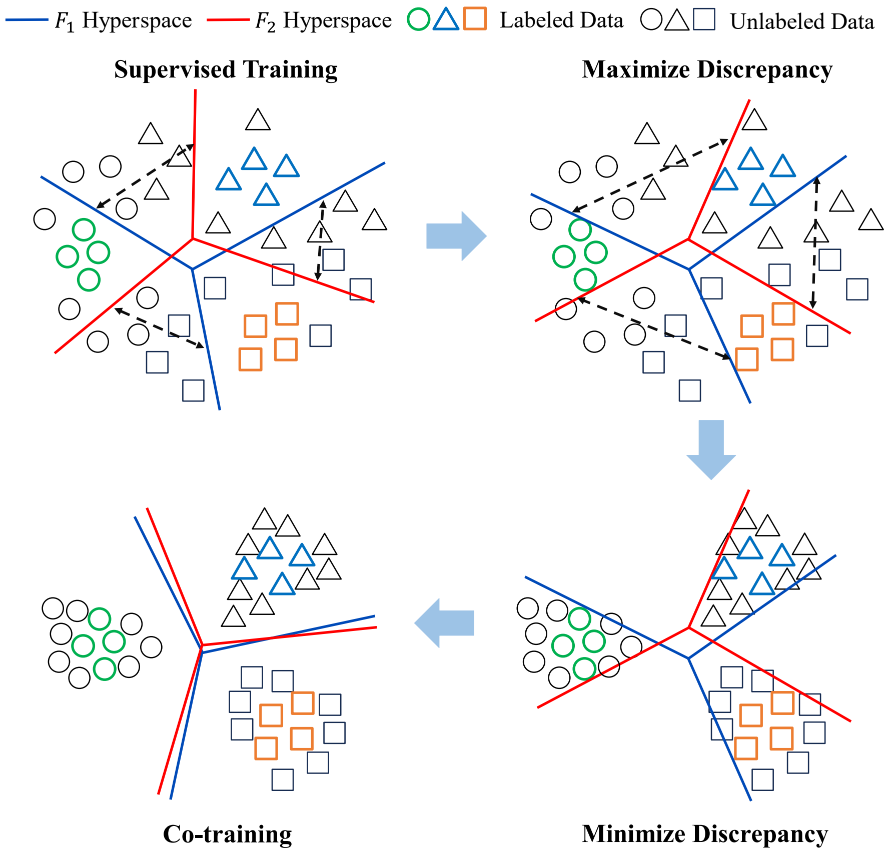
* Co-first author † Corresponding author

Project Page

CVPR
JUNE 17-21, 2024
SEATTLE, WA

Motivation

- Data setting
  - Unsupervised Domain Adaptation:
    - Rich labeled source samples
    - Large unlabeled target samples
  - Semi-supervised Domain Adaptation:
    - Rich labeled source samples
    - A few labeled target samples
    - Large unlabeled target samples



Motivation

- Goal
  - Learning CNN on ViT
  - Complement properties of CNN and ViT in capturing local and global information
  - Improve the quality and quantity of generated pseudo labels
  - Allivate data bias toward the source domain

- Solutions
  - Builld a new hybrid framework
  - Define new upper class-specific decision boundaries
  - Co-training to improve the quality of pseudo labels and reduce knowledge discrepancies

Proposed Method

- Network Architecture
  - ViT branch includes a ViT encoder $E_1(\cdot,\theta_{E_1})$ and a classifier $F_1(\cdot,\theta_{F_1})$
  - CNN branch includes a CNN encoder $E_2(\cdot,\theta_{E_2})$ and a classifier $F_2(\cdot,\theta_{F_2})$

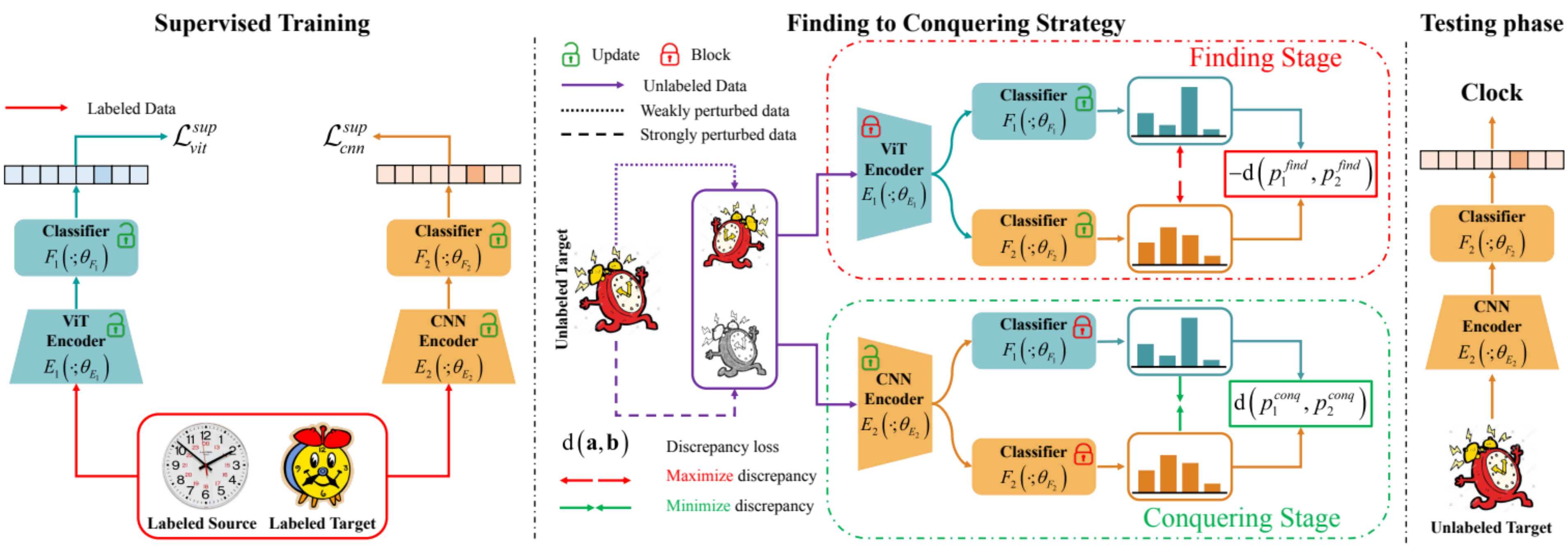


Figure 2. Illustration of a hybrid network with the proposed Finding to Conquering strategy. We use ViT to build E_1 that drives two classifiers F_1 and F_2 to expand class-specific boundaries comprehensively. Besides, we select CNN for the second encoder E_2 to cluster target features based on the boundaries identified by ViT. These encoders all use two classifiers F_1, F_2.

Proposed Method

- Training Strategy
  - **Step 1:** Supervised Training on labeled samples
    - Labeled source domain

$$\mathcal{D}_{\mathcal{S}} = \{(x_i^{\mathcal{S}}, y_i^{\mathcal{S}})\}_{i=1}^{\mathcal{N}_{\mathcal{S}}}$$

    - Labeled target domain

$$\mathcal{D}_{\mathcal{T}_l} = \{(x_i^{\mathcal{T}_l}, y_i^{\mathcal{T}_l})\}_{i=1}^{\mathcal{N}_{\mathcal{T}_l}}$$

    - Labeled set = labeled source domain + labeled target domain

(Notably, the labeled target domain is empty in UDA)

$$\mathcal{D}_l = \mathcal{D}_{\mathcal{S}} \cup \mathcal{D}_{\mathcal{T}_l}$$

    - Supervised training for the ViT branch

$$\mathcal{L}_{vit}^{sup}(\boldsymbol{\theta}_{E_1}, \boldsymbol{\theta}_{F_1}) = \frac{1}{\mathcal{N}_l} \sum_{i=1}^{\mathcal{N}_l} H(y_i^l, p_1^l(x_i^l))$$

    - Supervised training for the CNN branch

$$\mathcal{L}_{cnn}^{sup}(\boldsymbol{\theta}_{E_2}, \boldsymbol{\theta}_{F_2}) = \frac{1}{\mathcal{N}_l} \sum_{i=1}^{\mathcal{N}_l} H(y_i^l, p_2^l(x_i^l))$$

$H(\cdot)$: the standard cross-entropy loss

σ : the softmax function

$$p_1^l(x_i^l) = \sigma(F_1(E_1(x_i^l)))$$

$$p_2^l(x_i^l) = \sigma(F_2(E_2(x_i^l)))$$

Proposed Method

- Training Strategy
  - **Step 2:** Finding to Conquering (FTC) Strategy

- Unlabeled target data

$$\mathcal{D}_{\mathcal{T}_u} = \{(x_i^{\mathcal{T}_u}, y_i^{\mathcal{T}_u})\}_{i=1}^{\mathcal{N}_{\mathcal{T}_u}}$$

- Discrepancy Loss

$$d(\mathbf{a}, \mathbf{b}) = \frac{1}{K}\sum_{k=1}^{K} |a_k - b_k|$$

$p_1^{find}(x_i^{\mathcal{T}_u})$: the probability outputs of F_1 with ViT encoder

$p_2^{find}(x_i^{\mathcal{T}_u})$: the probability outputs of F_2 with ViT encoder

$p_1^{conq}(x_i^{\mathcal{T}_u})$: the probability outputs of F_1 with CNN encoder

$p_2^{conq}(x_i^{\mathcal{T}_u})$: the probability outputs of F_2 with CNN encoder

- Finding Stage

$$\mathcal{L}_{find}(\boldsymbol{\theta}_{F_1}, \boldsymbol{\theta}_{F_2}) = \mathcal{L}_{vit}^{sup} + \mathcal{L}_{cnn}^{sup} - \frac{1}{\mathcal{N}_{\mathcal{T}_u}}\sum_{i=1}^{\mathcal{N}_{\mathcal{T}_u}} d(p_1^{find}(x_i^{\mathcal{T}_u}), p_2^{find}(x_i^{\mathcal{T}_u}))$$

- Conquering Stage

$$\mathcal{L}_{conq}(\boldsymbol{\theta}_{E_2}) = \frac{1}{\mathcal{N}_{\mathcal{T}_u}}\sum_{i=1}^{\mathcal{N}_{\mathcal{T}_u}} d(p_1^{conq}(x_i^{\mathcal{T}_u}), p_2^{conq}(x_i^{\mathcal{T}_u}))$$

Proposed Method

- Training Strategy

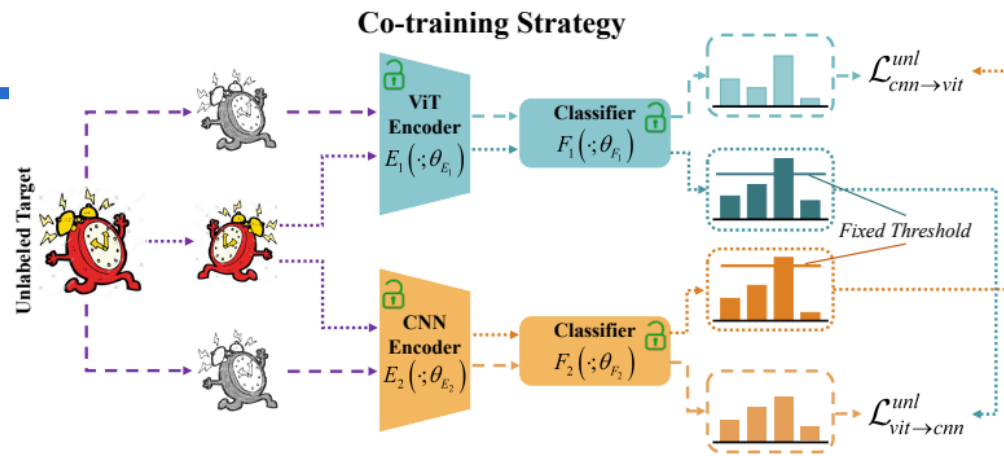


Figure 3. Illustration of co-training strategy.

- **Step 3:** Co-training
- ViT branch teaches CNN branch

$$\mathcal{L}^{unl}_{vit\to cnn}(\boldsymbol{\theta}_{E_2},\boldsymbol{\theta}_{F_2}) = \frac{1}{\mathcal{N}_{\mathcal{T}_u}}\sum_{i=1}^{\mathcal{N}_{\mathcal{T}_u}} \mathbb{1}[\max(\mathbf{q}_i^v) \geq \tau_{vit}] H\big(\hat{q}_i^v, p^c(x_{i,str}^{\mathcal{T}_u})\big)$$

- CNN branch teaches ViT branch

$$\mathcal{L}^{unl}_{cnn\to vit}(\boldsymbol{\theta}_{E_1},\boldsymbol{\theta}_{F_1}) = \frac{1}{\mathcal{N}_{\mathcal{T}_u}}\sum_{i=1}^{\mathcal{N}_{\mathcal{T}_u}} \mathbb{1}[\max(\mathbf{q}_i^c) \geq \tau_{cnn}] H\big(\hat{q}_i^c, p^v(x_{i,str}^{\mathcal{T}_u})\big)$$

- **Inference Stage**

$$\hat{y}_i^{\mathcal{T}_u} = argmax\Big(\big(F_2(E_2(x_i^{\mathcal{T}_u}))\big)\Big)$$

$\hat{q}_i^v$: pseudo label is generated by the ViT

$p^c(x_{i,str}^{\mathcal{T}_u})$: output prediction of the CNN branch

$\hat{q}_i^c$: pseudo label is generated by the CNN

$p^v(x_{i,str}^{\mathcal{T}_u})$: output prediction of the ViT branch

Experimental Results

◆ Accuracy **(%)** on **Office-Home** of UDA setting

Method	$A \to C$	$A \to P$	$A \to R$	$C \to A$	$C \to P$	$C \to R$	$P \to A$	$P \to C$	$P \to R$	$R \to A$	$R \to C$	$R \to P$	Mean
DANN [8]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
MCD [30]	48.9	68.3	74.6	61.3	67.6	68.8	57.0	47.1	75.1	69.1	52.2	79.6	64.1
BNM [4]	52.3	73.9	80.0	63.3	72.9	74.9	61.7	49.5	79.7	70.5	53.6	82.2	67.9
MDD [37]	54.9	73.7	77.8	60.0	71.4	71.8	61.2	53.6	78.1	72.5	60.2	82.3	68.1
MCC [12]	55.1	75.2	79.5	63.3	73.2	75.8	66.1	52.1	76.9	73.8	58.4	83.6	69.4
GVB [5]	57.0	74.7	79.8	64.6	74.1	74.6	65.2	55.1	81.0	74.6	59.7	84.3	70.4
DCAN [18]	54.5	75.7	81.2	67.4	74.0	76.3	67.4	52.7	80.6	74.1	59.1	83.5	70.5
DALN [2]	57.8	79.9	82.0	66.3	76.2	77.2	66.7	55.5	81.3	73.5	60.4	85.3	71.8
FixBi [22]	58.1	77.3	80.4	67.7	79.5	78.1	65.8	57.9	81.7	76.4	62.9	86.7	72.7
DCAN+SCDA [19]	60.7	76.4	<u>82.8</u>	69.8	77.5	78.4	68.9	59.0	82.7	74.9	61.8	84.5	73.1
ATDOC [20]	60.2	77.8	82.2	68.5	78.6	77.9	68.4	58.4	83.1	74.8	61.5	<u>87.2</u>	73.2
EIDCo [38]	<u>63.8</u>	<u>80.8</u>	82.6	<u>71.5</u>	<u>80.1</u>	<u>80.9</u>	<u>72.1</u>	<u>61.3</u>	<u>84.5</u>	<u>78.6</u>	<u>65.8</u>	87.1	<u>75.8</u>
ECB (CNN)	**68.5**	**85.4**	**88.3**	**79.2**	**86.8**	**89.0**	**79.3**	**66.4**	**88.5**	**81.0**	**71.1**	**90.4**	**81.2**

Table 1. **Accuracy (%) on Office-Home** of UDA methods across different domain shifts. **ECB (CNN)** represents the performance of our method when applied to ResNet-50. The top and second-best accuracy results are highlighted in **bold** and <u>underline</u> for easy identification.

Experimental Results

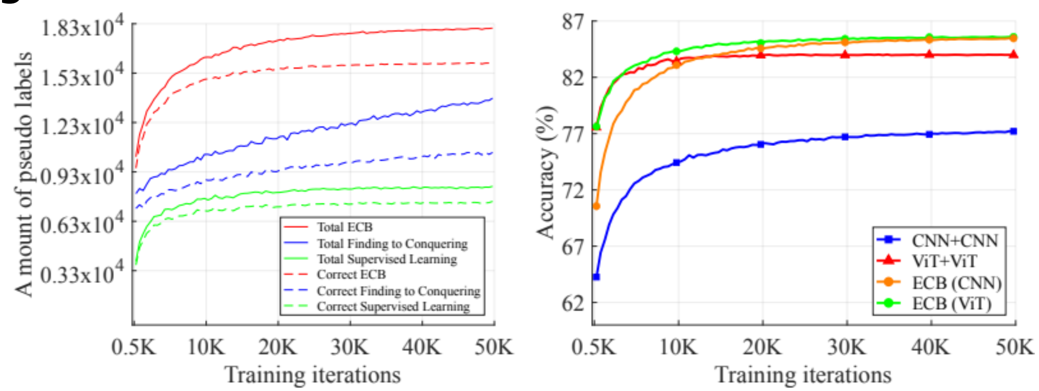
◆ Accuracy **(%)** on **DomainNet** of SSDA setting

Method	*rel*→*clp*		*rel*→*pnt*		*pnt*→*clp*		*clp*→*skt*		*skt*→*pnt*		*rel*→*skt*		*pnt*→*rel*		Mean	
	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3_{shot}	1_{shot}	3_{shot}
ENT [9]	65.2	71.0	65.9	69.2	65.4	71.1	54.6	60.0	59.7	62.1	52.1	61.1	75.0	78.6	62.6	67.6
MME [31]	70.0	72.2	67.7	69.7	69.0	71.7	56.3	61.8	64.8	66.8	61.0	61.9	76.1	78.5	66.4	68.9
S^3D [35]	73.3	75.9	68.9	72.1	73.4	75.1	60.8	64.4	68.2	70.0	65.1	66.7	79.5	80.3	69.9	72.1
ATDOC [20]	74.9	76.9	71.3	72.5	72.8	74.2	65.6	66.7	68.7	70.8	65.2	64.6	81.2	81.2	71.4	72.4
MAP-F [24]	75.3	77.0	74.0	75.0	74.3	77.0	65.8	69.5	73.0	73.3	67.5	69.2	81.7	83.3	73.1	74.9
DECOTA [34]	79.1	80.4	74.9	75.2	76.9	78.7	65.1	68.6	72.0	72.7	69.7	71.9	79.6	81.5	73.9	75.6
CDAC [16]	77.4	79.6	74.2	75.1	75.5	79.3	67.6	69.9	71.0	73.4	69.2	72.5	80.4	81.9	73.6	76.0
ASDA [28]	77.0	79.4	75.4	76.7	75.5	78.3	66.5	70.2	72.1	74.2	70.9	72.1	79.7	82.3	73.9	76.2
CDAC+SLA [36]	79.8	81.6	75.6	76.0	77.4	80.3	68.1	71.3	71.7	73.5	71.7	73.5	80.4	82.5	75.0	76.9
ProML [11]	78.5	80.2	75.4	76.5	77.8	78.9	70.2	72.0	74.1	75.4	72.4	73.5	84.0	84.8	76.1	77.4
MVCL [23]	78.8	79.8	76.0	77.4	78.0	80.3	70.8	73.0	75.1	76.7	72.4	74.4	82.4	85.1	76.2	78.1
G-ABC [17]	80.7	82.1	76.8	76.7	79.3	81.6	72.0	73.7	75.0	76.3	73.2	74.3	83.4	83.9	77.2	78.4
ECB (CNN)	**83.8**	**87.4**	**85.4**	**85.6**	**86.4**	**87.3**	**79.7**	**80.6**	**83.4**	**85.6**	**79.5**	**81.7**	**88.7**	**90.3**	**83.8**	**85.5**

Table 2. **Accuracy (%) on DomainNet** of SSDA methods in both 1-shot and 3-shot settings using ResNet-34.

Analyses

- Ablation Study
  - "Quality and quantity of generated pseudo labels" and "Comparison between backbone settings"



(a) (b)

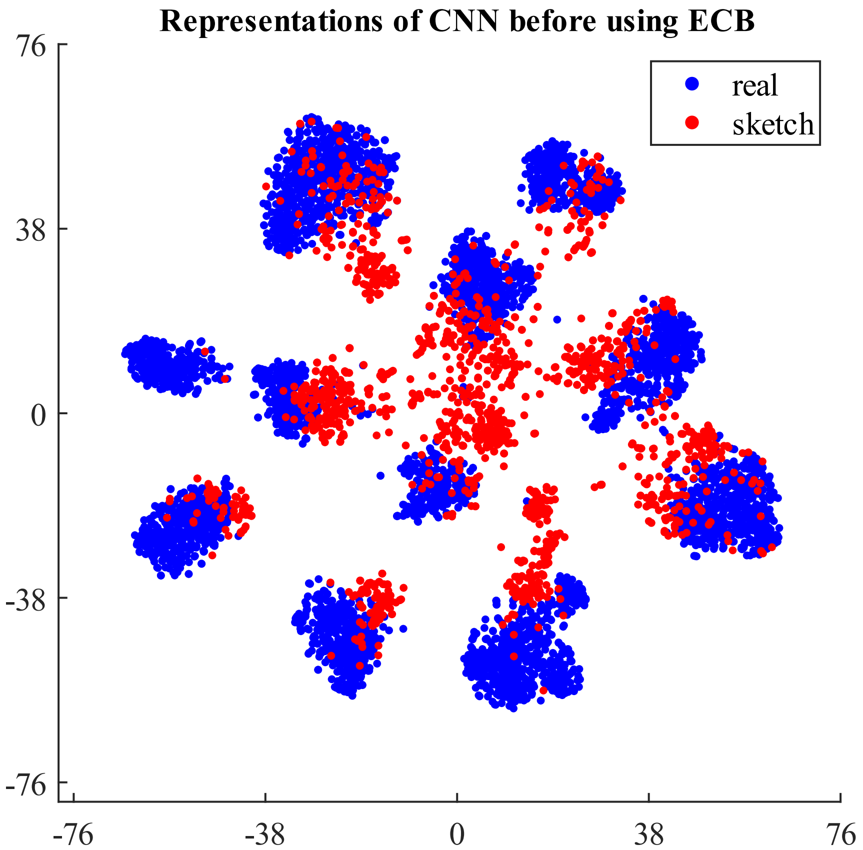
  - Effectiveness of co-training

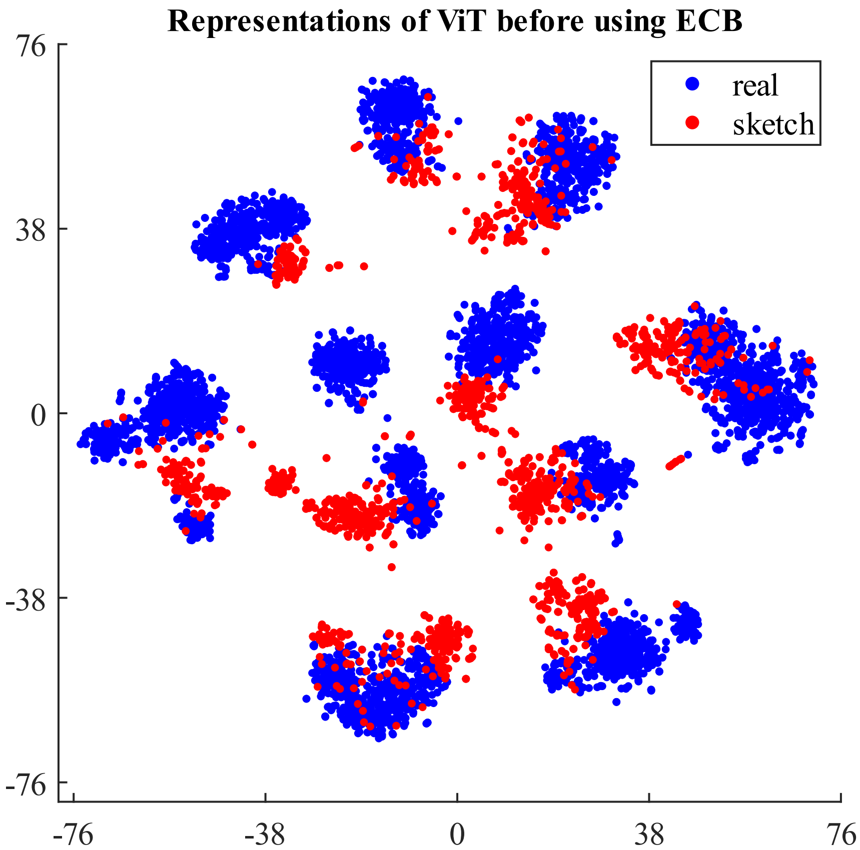
Method	*rel*→*clp*		*rel*→*pnt*		*pnt*→*clp*		*clp*→*skt*		*skt*→*pnt*		*rel*→*skt*		*pnt*→*rel*		Mean	
	ViT	CNN	ViT	CNN	ViT	CNN	ViT	CNN	ViT	CNN	ViT	CNN	ViT	CNN	ViT	CNN
vit→*cnn*	73.3	79.0	78.8	81.0	75.1	79.2	71.6	74.7	78.6	80.8	67.2	72.0	88.1	88.8	76.1	79.4
cnn→*vit*	74.2	61.9	76.8	66.8	76.1	67.4	69.5	57.2	74.9	64.6	67.4	54.8	86.0	76.1	75.0	64.1
co-training	87.4	87.4	85.8	85.6	87.3	87.3	80.7	80.6	85.8	85.6	81.7	81.7	90.9	90.3	85.7	85.5

Table 3. **Ablation study on DomainNet** between co-training and one-direction teaching under 3-shot settings.

Analyses

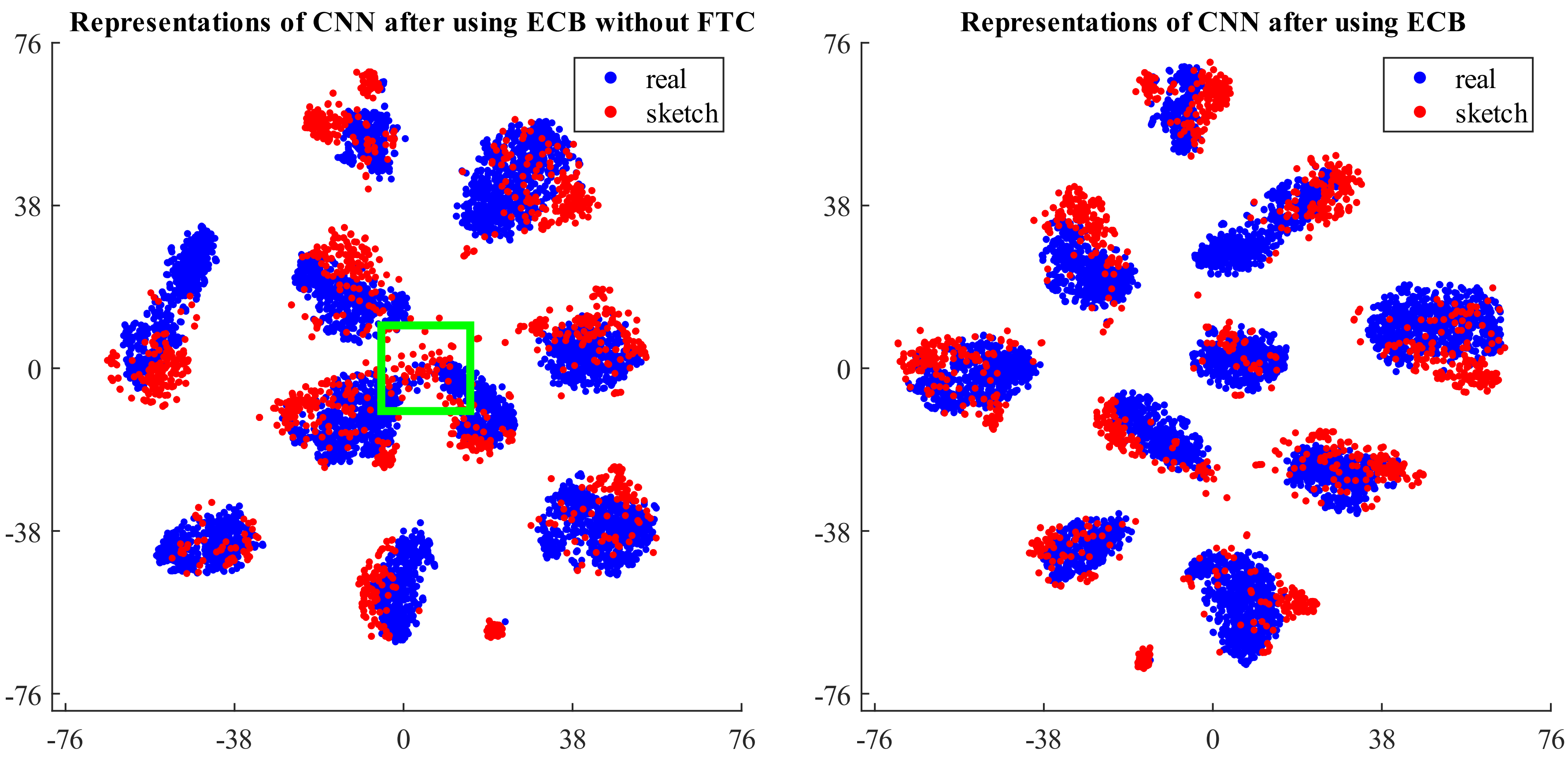
◆ Visualization t-SNE





Analyses

◆ Visualization t-SNE



Analyses

◆ Attention map visualization

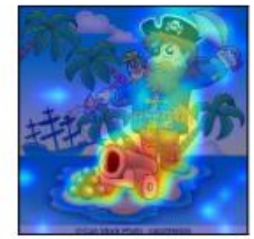
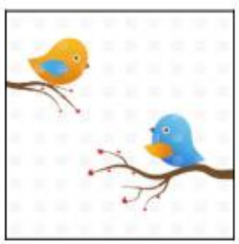
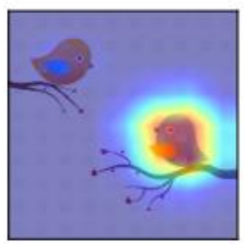
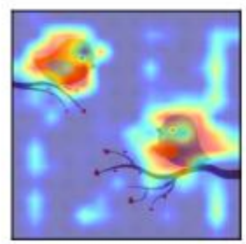
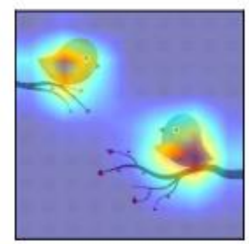
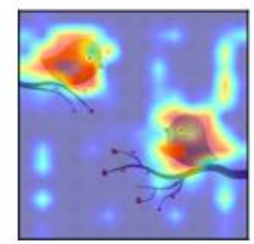
Cannon	CNN (Before)	ViT (Before)	CNN (After)	ViT (After)
				
Bird	CNN (Before)	ViT (Before)	CNN (After)	ViT (After)
				

Table 4. **Visualize the feature maps** for the '*Cannon*' and '*Bird*' examples to investigate the learning behaviors of CNN and ViT with and without using the proposed method ECB.

Thank you for listening