ISTITUTO ITALIANO
DI TECNOLOGIA
PATTERN ANALYSIS
AND COMPUTER VISION

UNIVERSITÀ
di VERONA
Dipartimento
di INFORMATICA

Domain Adaptation and Generalization

Vittorio Murino, Pietro Morerio

April 8, 2022

Session 3
Beyond Domain Adaptation

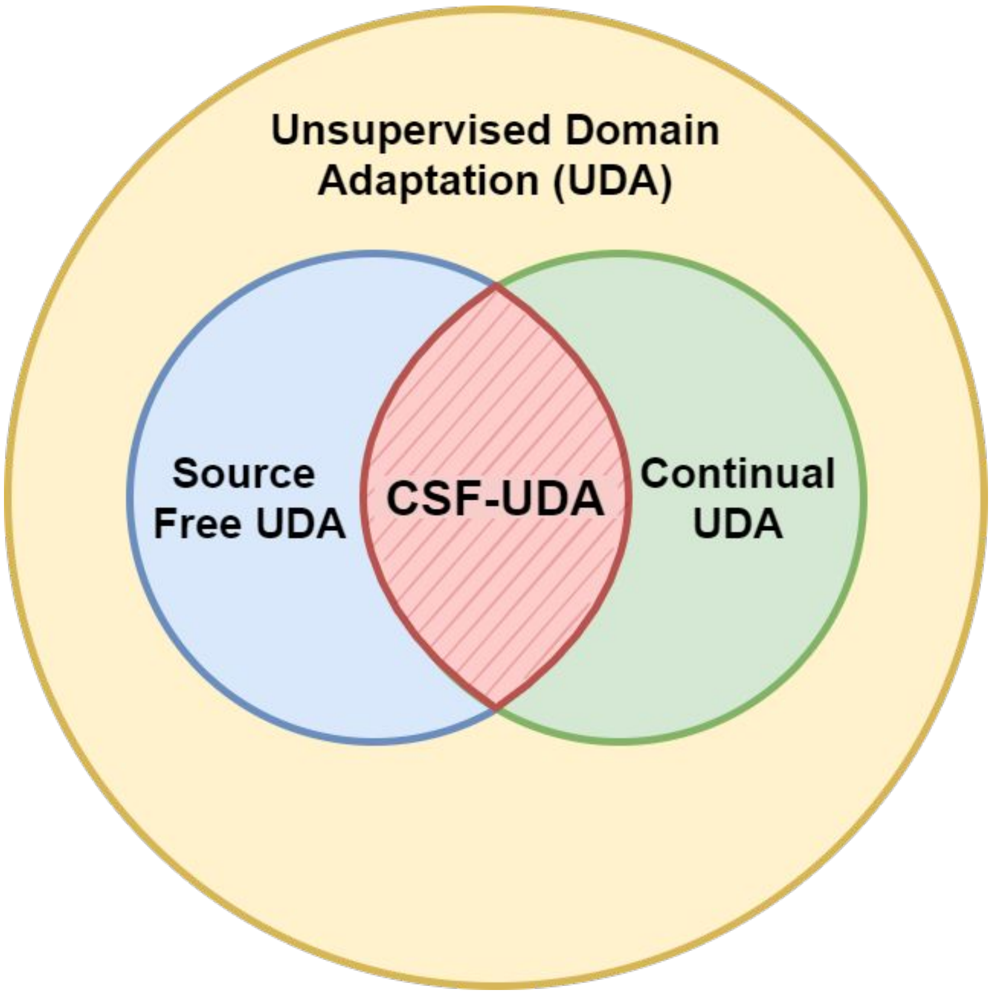
Outline

Session 3 - Beyond Domain Adaptation (1h)

- Source Free UDA
- Domain Discovery
- Continuous DA
- Predictive DA
- Validation issues in Unsupervised Domain adaptation

Source-free UDA

Continual & Source Free UDA



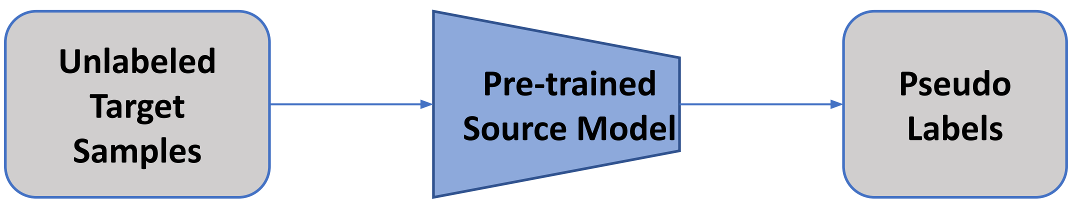
E.g. Pseudo-Labeling methods are source-free

Continual UDA: adapt to target with limited drop in performance on the source

Challenge: how to limit the drop on Source if no samples are available?

Negative ensemble Learning

Stage 1: Inferring Pseudo-Labels for the target set



Given noisy label : Car

Cleaning Noisy Labels by Negative Ensemble Learning for Source-Free Unsupervised Domain Adaptation
Ahmed W, Morerio P, Murino V. WACV 2022

Negative ensemble Learning

The concept of Negative Learning

Given noisy label : Car

Complementary Labels

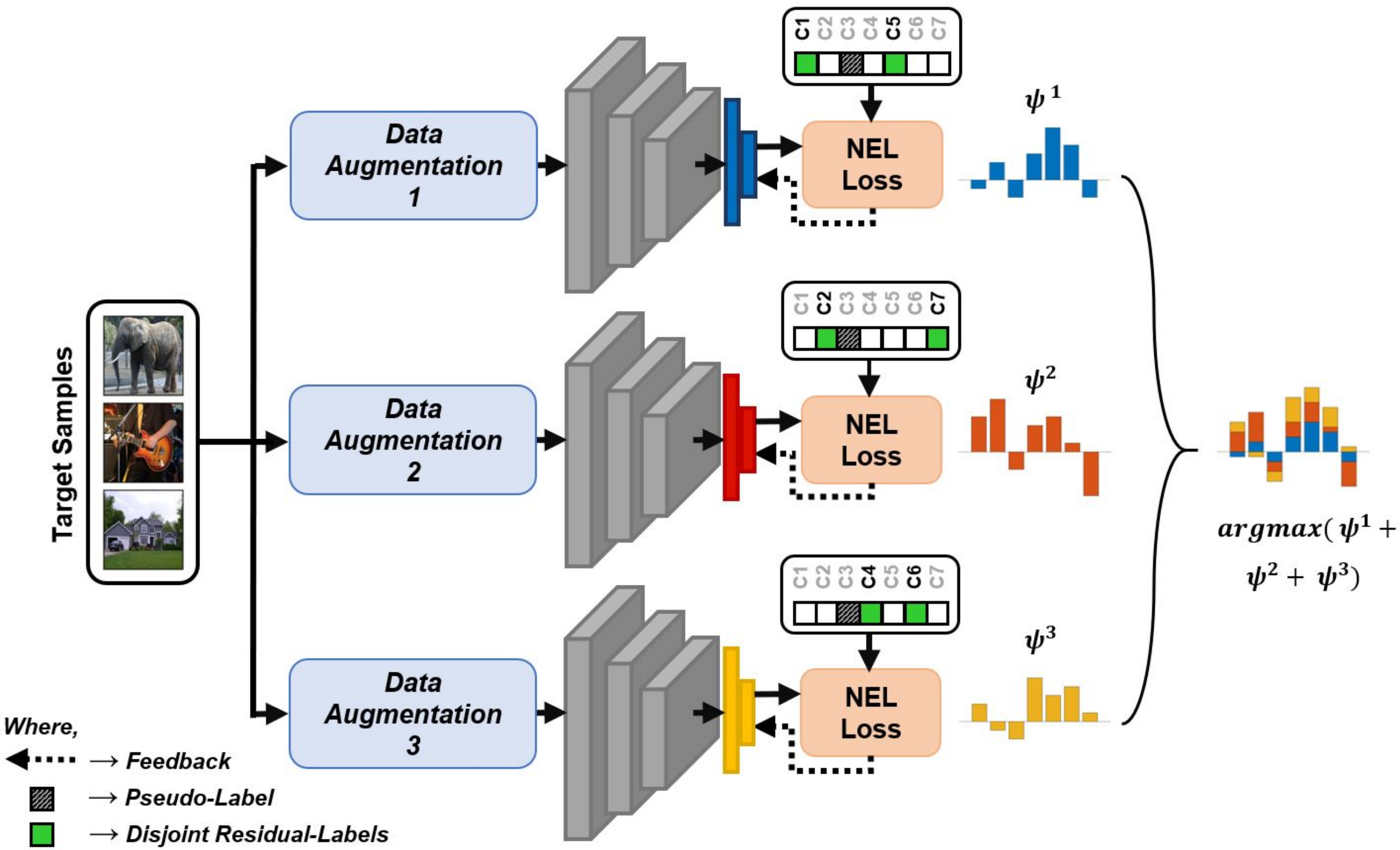
Car	Bird	Horse	Cat	Dog	Chair	Truck

$$\mathcal{L}_{NL}(\mathcal{D}_t) = -\mathbb{E}_{x_t\sim\mathcal{D}_t}\sum_{c=1}^{C}\mathbb{1}_{[c=\bar{y}]}\log(1-p^c)$$

$\bar{y}^j$ (randomly selected from $\{1,\ldots,C\}\setminus\{\tilde{y}\}$)

Negative ensemble Learning

Stage 2: Pseudo-Label Refinement

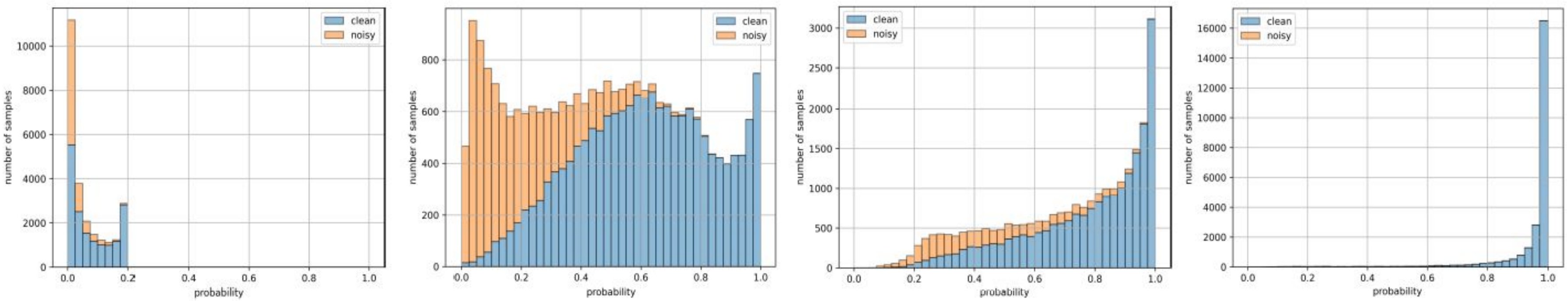


Negative ensemble Learning

Adaptive Reassignment Rule

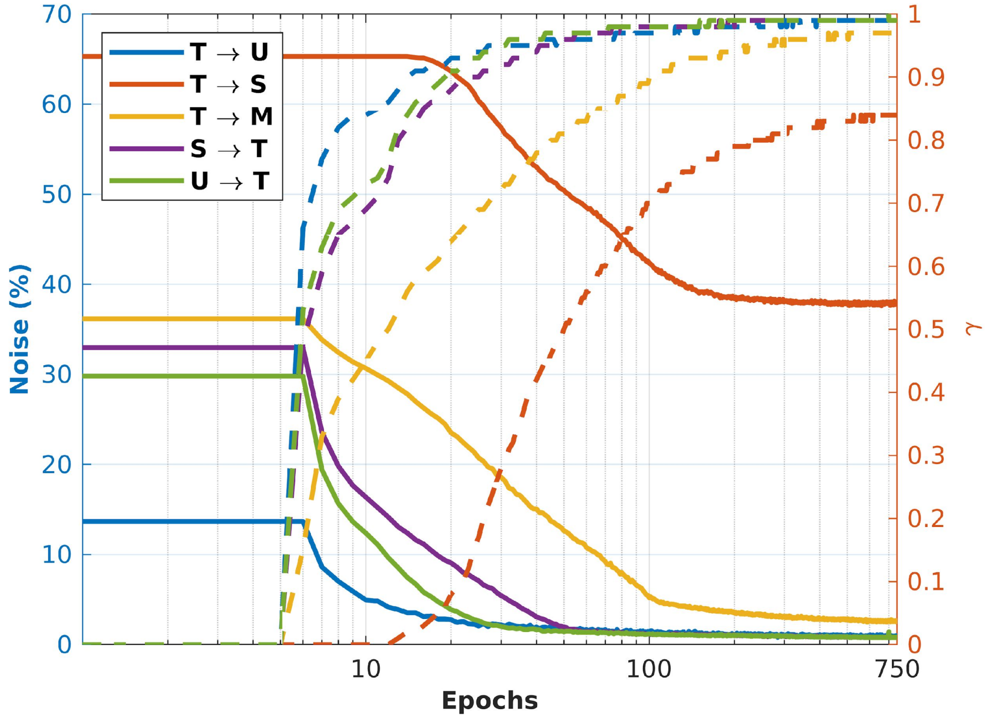
$$\gamma = \frac{\# \; of \; HCS}{N_t} \qquad \tilde{p} > \alpha \text{ as High Confidence Samples (HCS)}$$

$$\tilde{y}^j(n) = \begin{cases} \text{argmax}(\boldsymbol{p}^j), & \text{if } \tilde{p}^j < \gamma \\ \tilde{y}^j(n-1), & \text{otherwise} \end{cases} \quad \forall j$$



(a) epochs = 1 (b) epochs = 10 (c) epochs = 25 (d) epochs = 50

Results



	Single-Source UDA							Multi-Source UDA					
Source	*T*	*T*	*T*	*S*	*U*	Avg.		*M,S, D,U*	*T,S, D,U*	*T,M, D,U*	*T,M, S,U*	*T,M, S,D*	Avg.
Target	*U*	*S*	*M*	*T*	*T*			*T*	*M*	*S*	*D*	*U*	
ATT [37]	–	52.8	94.0	85.8	–	–	DCTN [43]	–	70.9	77.5	–	–	–
SBA [36]	97.1	50.9	**98.4**	74.2	87.5	81.6	MM [33]	98.4	72.8	81.3	89.5	96.1	87.6
MALT [28]	97.0	**78.7**	71.4	98.7	20.7	73.3	OML [24]	98.7	71.7	84.8	91.1	97.8	88.8
MTDA [16]	94.2	52.0	85.5	84.6	91.5	81.5	CMSS [45]	99.0	75.3	88.4	**93.7**	97.7	90.8
GPLR [29]	89.3	63.4	94.3	97.3	91.8	87.5							
AdaPLR	**97.4**	61.6	95.4	**99.2**	**99.2**	**90.6**		**99.1**	**95.5**	**89.6**	90.0	**97.8**	**94.4**

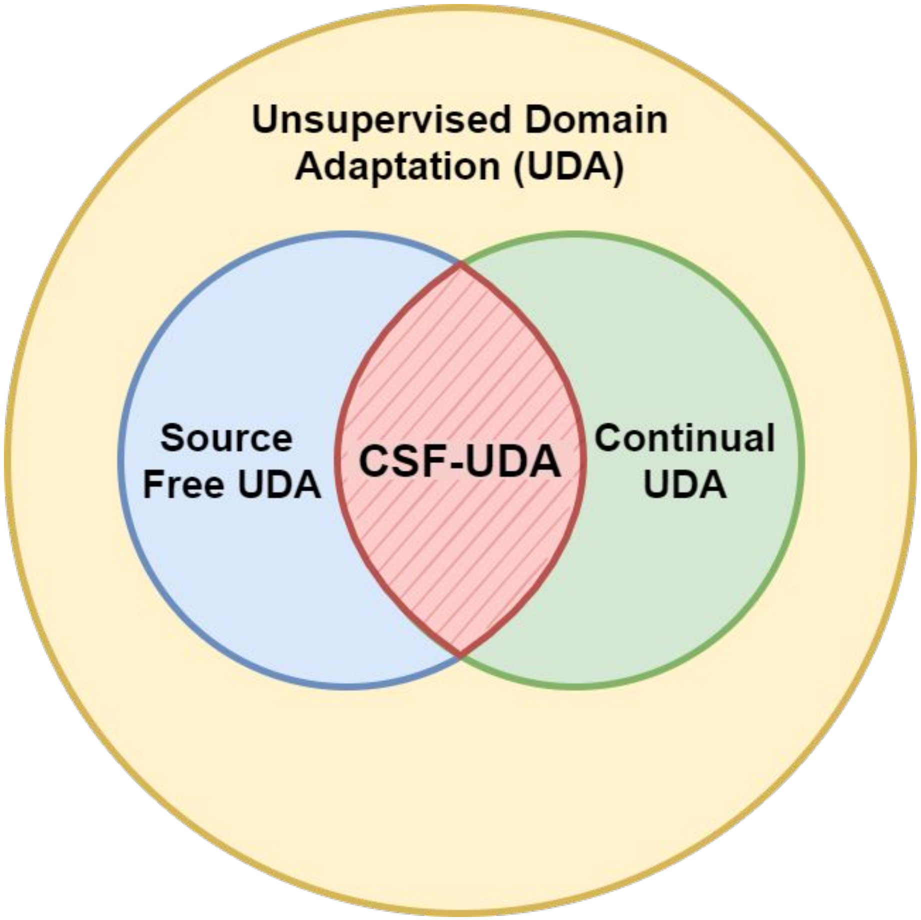
Table 1: Classification accuracy on Digit5 with a naive 3-layer CNN. Legend: *T*: *MNIST*, *S*: *SVHN*, *U*: *USPS*, *M*: *MNIST-M, and* *D*: *Synthetic-Digits.*

	Multi-Target UDA								Multi-Source UDA				
Source	*P*			*A*			Avg.		*C,P,S*	*A,P,S*	*A,C,S*	*A,C,P*	Avg.
Target	*A*	*C*	*S*	*P*	*C*	*S*			*A*	*C*	*P*	*S*	
1-NN*	15.2	18.1	25.6	22.7	19.7	22.7	20.7	DD [27]	87.5	87.0	96.6	71.6	85.7
ADDA*	24.3	20.1	22.4	32.5	17.6	18.9	22.6	SIB [18]	88.9	89.0	98.3	82.2	89.6
DSN*	28.4	21.1	25.6	29.5	25.8	24.6	25.8	OML [24]	87.4	86.1	97.1	78.2	87.2
ITA*	31.4	23.0	28.2	35.7	27.0	28.9	29.0	RABN [42]	86.8	86.5	98.0	71.5	85.7
KD [1]	24.6	32.2	**33.8**	35.6	46.6	**57.5**	46.6	JiGen [2]	84.8	81.0	97.9	79.0	85.7
								CMSS [45]	88.6	**90.4**	96.9	82.0	89.5
AdaPLR	**80.1**	**76.1**	25.9	**96.0**	**82.8**	49.8	**68.4**		**90.8**	89.5	**98.8**	**85.2**	**91.1**

Table 2: Classification accuracy on PACS with ResNet18. * results are taken from [16]. Legend: *A*: *Art-Painting*, *C*: *Cartoon*, *P*: *Photo, and* *S*: *Sketch.*

Continual Source Free UDA

The adapted model suffers a drop in performance when tested back on the Source Domain



Continual Source Free Unsupervised Domain Adaptation
Ahmed W, Morerio P, Murino V. *Under Review*

Continual Source Free UDA

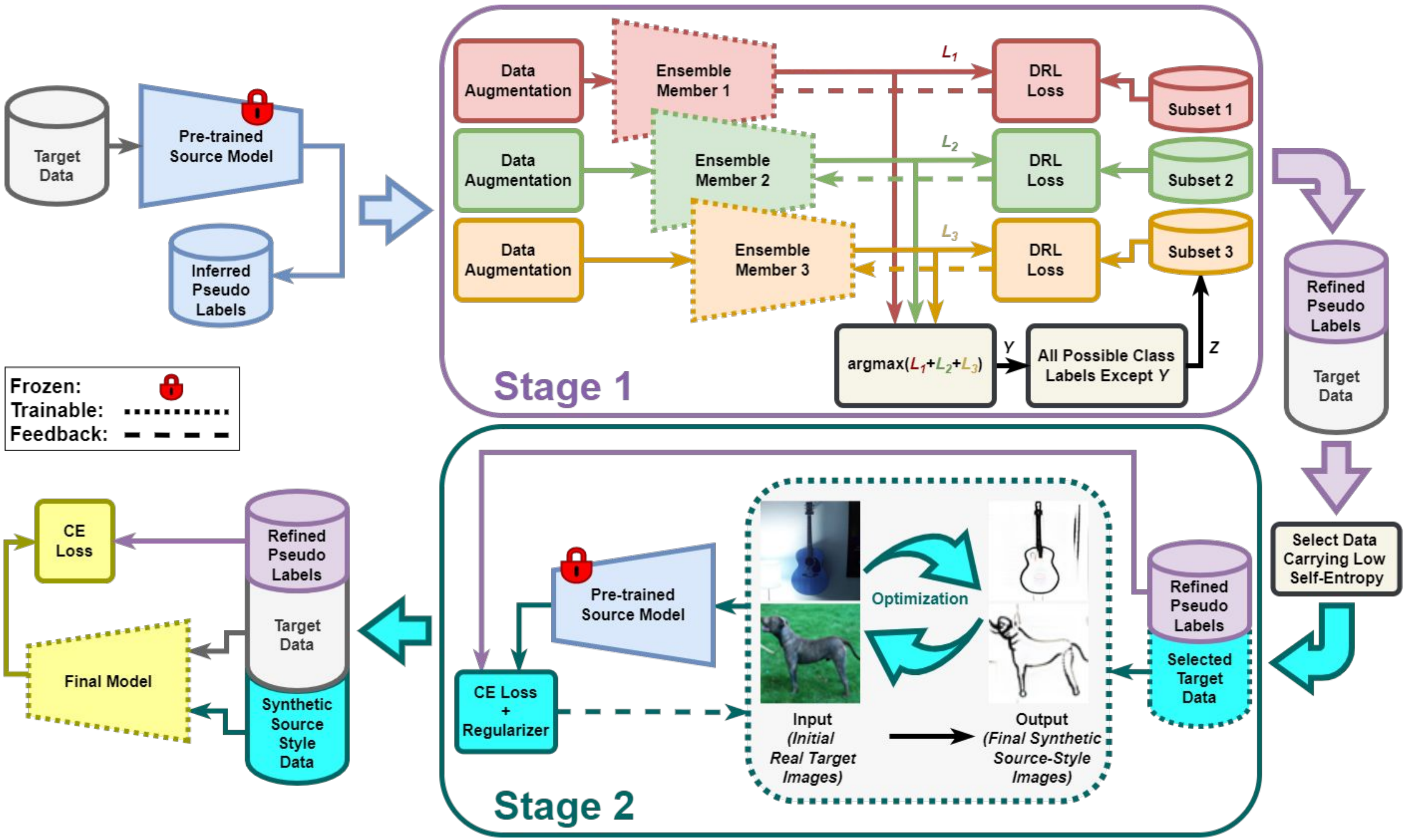
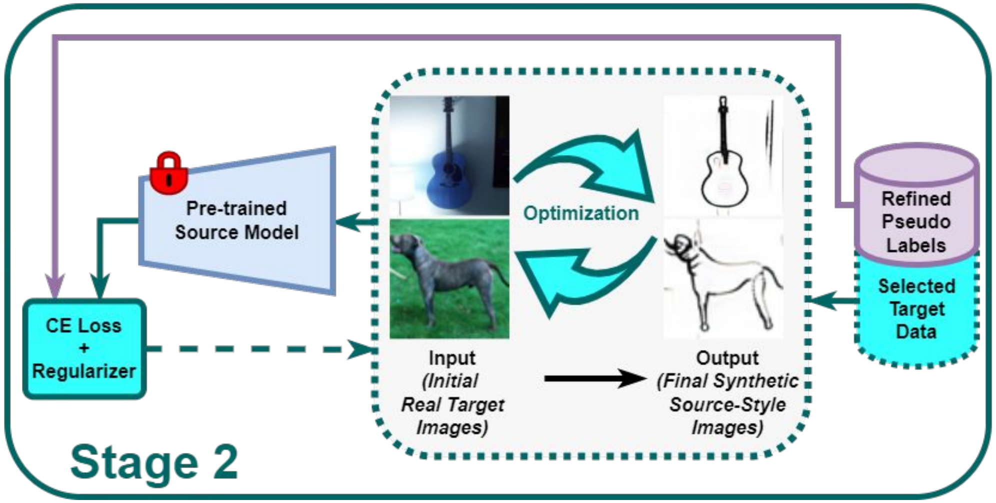


Image Synthesis for Continual Adaptation

Target Images are **directly optimized in the pixel space** in order to minimize the classification loss for the source model.



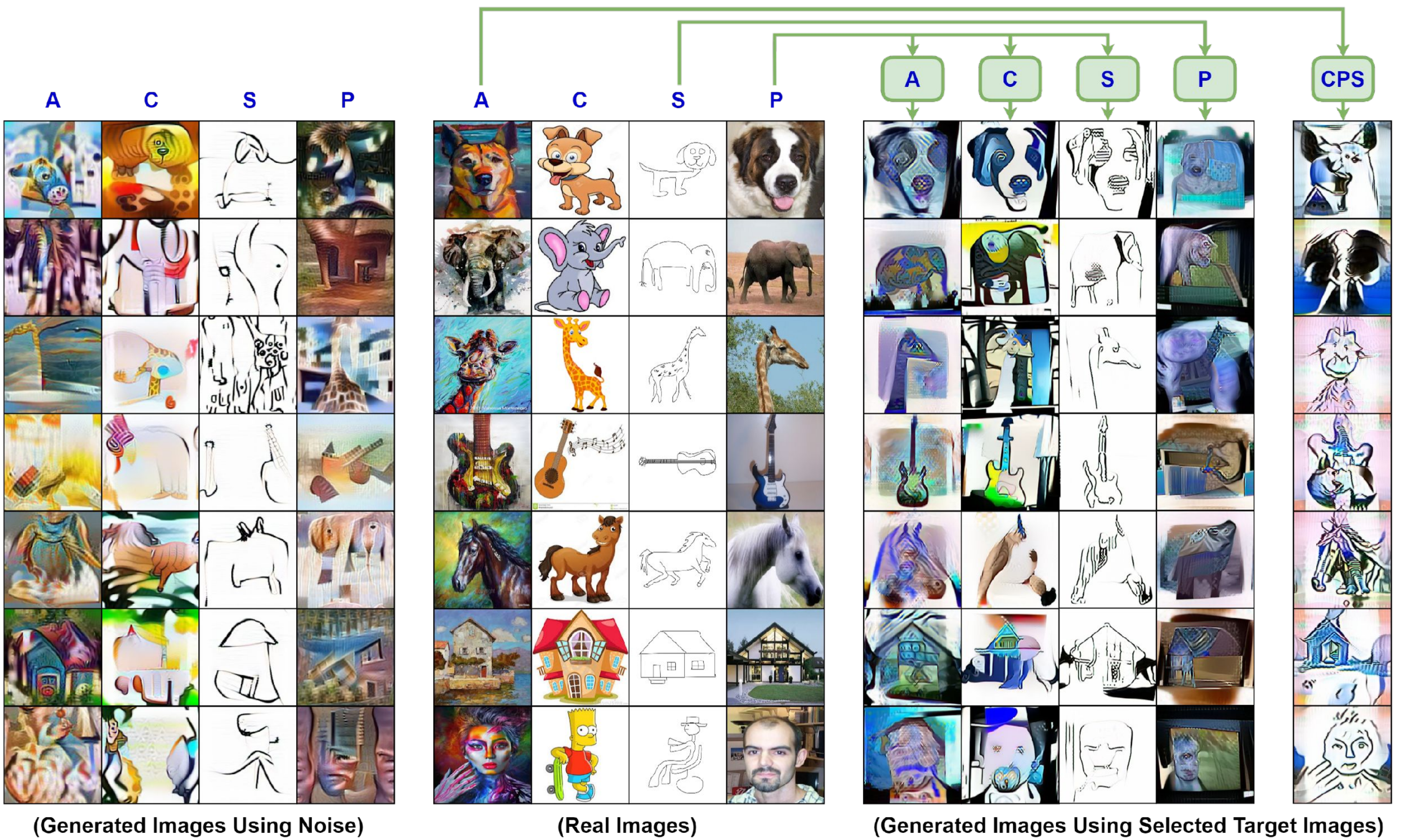
$$\boldsymbol{x} \leftarrow \boldsymbol{x} - \eta \nabla_{\boldsymbol{x}} \mathcal{L}(f_s(\boldsymbol{x}), \tilde{y}), \quad \boldsymbol{x}_0 = \boldsymbol{x}_t^j$$

$$\begin{aligned}\mathcal{L}(f_s(\boldsymbol{x}), \tilde{y}) =& \ell_{CE}((f_s(\boldsymbol{x})), \tilde{y}) + \\ & \lambda_{TV} \mathcal{R}_{TV}(\boldsymbol{x}) + \lambda_{BN} \mathcal{R}_{BN}(\boldsymbol{x}),\end{aligned}$$

$$\mathcal{R}_{TV}(\boldsymbol{x}) = \sum_{u,v} ((\boldsymbol{x}_{u,v+1} - \boldsymbol{x}_{uv})^2 + (\boldsymbol{x}_{u+1,v} - \boldsymbol{x}_{uv})^2)^{\frac{1}{2}},$$

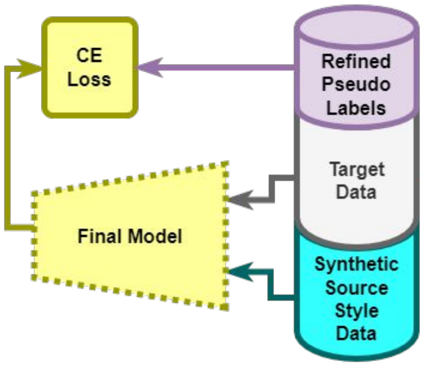
$$\mathcal{R}_{BN} = \sum_{l,j} \parallel \mu_l(\boldsymbol{x}^j) - \mu_l \parallel + \parallel \sigma_l^2(\boldsymbol{x}^j) - \sigma_l^2) \parallel_2,$$

Image Synthesis for Continual Adaptation



Continual Source Free UDA

Synthetic images are then fed back to the model



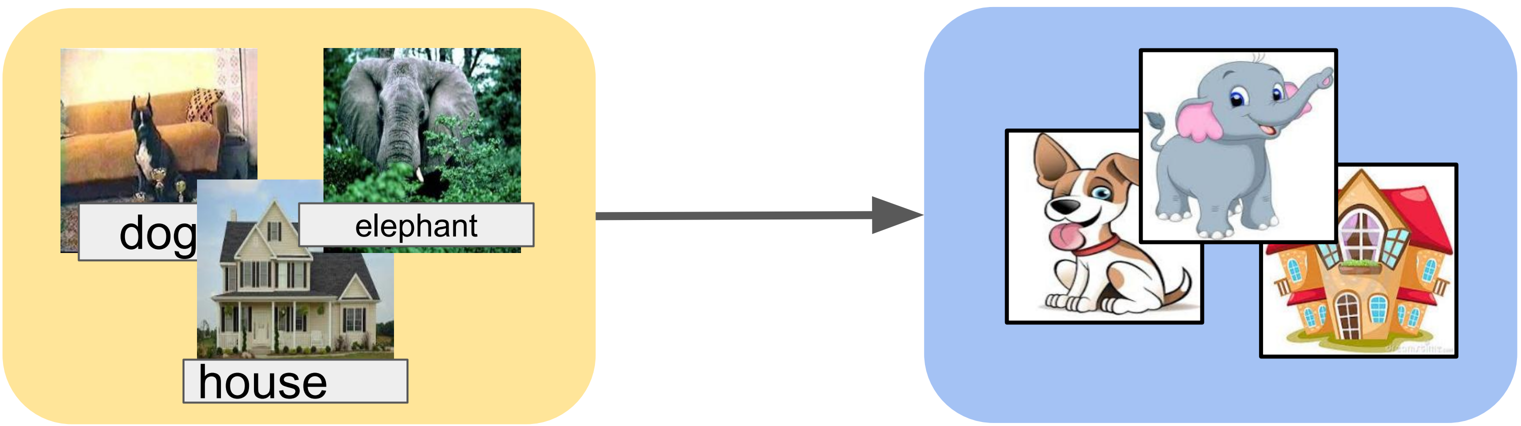
Results

Methods	*plane*	*bcycl*	*bus*	*car*	*horse*	*knife*	*mcycl*	*person*	*plant*	*skate*	*train*	*truck*	Avg.
Inferred	64.2	6.3	75.2	21.7	55.9	95.7	22.8	1.4	79.8	0.7	82.8	19.8	46.3
GPDA [16]	83.0	74.3	80.4	66.0	87.6	75.3	83.8	73.1	90.1	57.3	80.2	37.9	73.3
DADA [40]	92.9	74.2	82.5	65.0	90.9	**93.8**	87.2	74.2	89.9	71.5	86.5	48.7	79.8
SHOT [25]	94.3	**88.5**	80.1	57.3	93.1	94.9	80.7	80.3	91.5	89.1	86.3	58.2	82.9
A^2Net [44]	94.0	87.8	85.6	66.8	93.7	95.1	85.8	81.2	91.6	88.2	86.5	56.0	84.3
Sc→Tg	64.2	6.3	75.2	21.7	55.9	95.7	22.8	1.4	79.8	0.7	82.8	19.8	46.3
PLR	**95.2**	64.8	**90.8**	**89.7**	87.4	93.7	**91.5**	**88.5**	56.4	82.9	**97.1**	**93.8**	**85.1**
SF-UDA	**94.8**	68.1	**89.5**	**88.1**	86.5	90.4	**87.4**	**89.0**	53.2	81.5	**96.9**	**93.0**	**84.8**
CSF-UDA	**94.9**	67.3	**89.2**	**87.8**	86.1	90.0	86.6	**88.7**	53.1	80.9	**96.5**	**94.6**	**84.6**
SF-UDA*	*45.2*	*18.5*	*55.9*	*52.7*	*54.8*	*44.3*	*12.5*	*41.4*	*24.6*	*35.1*	*40.2*	*51.2*	*39.7*
CSF-UDA*	***47.6***	***21.4***	***58.2***	***54.3***	***61.1***	***49.5***	***27.9***	***41.9***	***44.8***	***36.2***	***43.1***	***55.4***	***45.1***

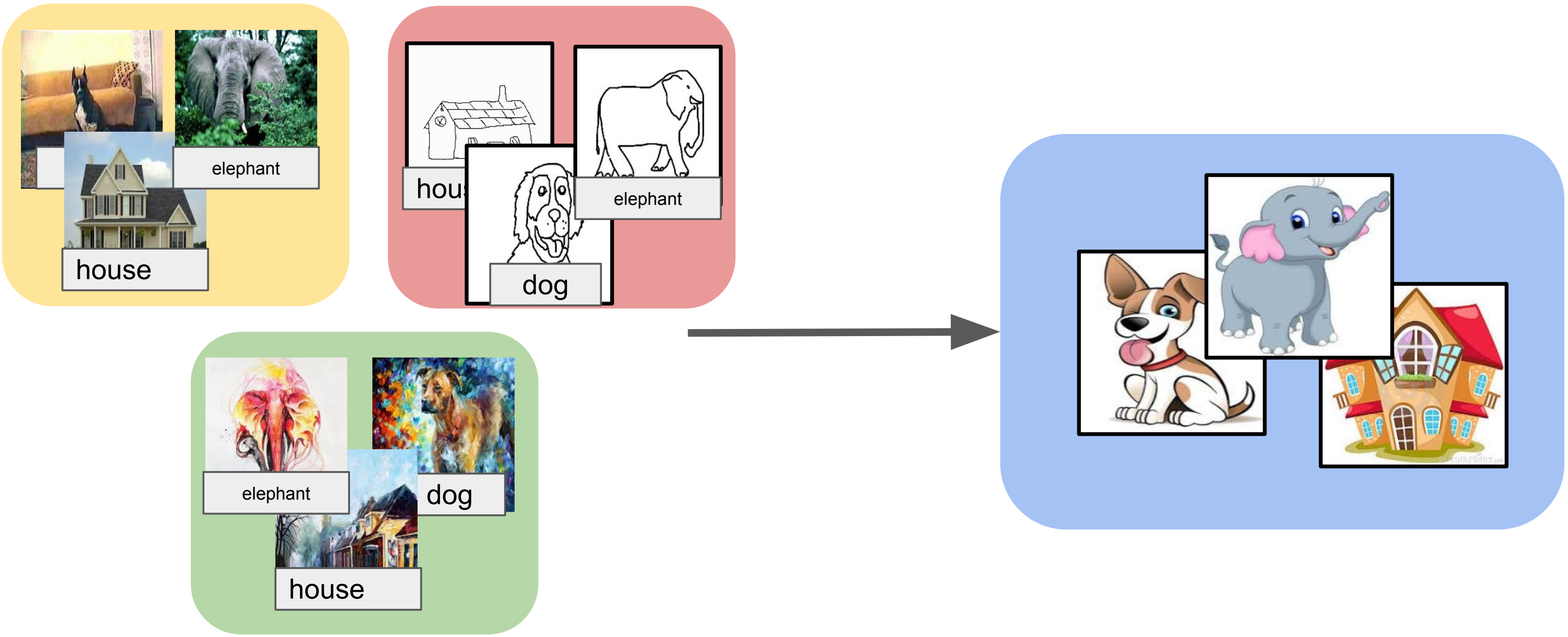
Table 3: Classification accuracy on Visda-C with ResNet101. Legend: ***Sc:*** *Source (real),* ***Tg:*** *Target (real),* ***Sc→Tg****: Inferred pseudo-labels,* ***PLR****: Pseudo-label refinement (output of Stage 1),* ***SF-UDA:*** *Source-free UDA,* ***CSF-UDA:*** *Continual source-free UDA,* $\star$*: Accuracy on real-source.*

Latent Domain Discovery
(Towards Domain Generalization)

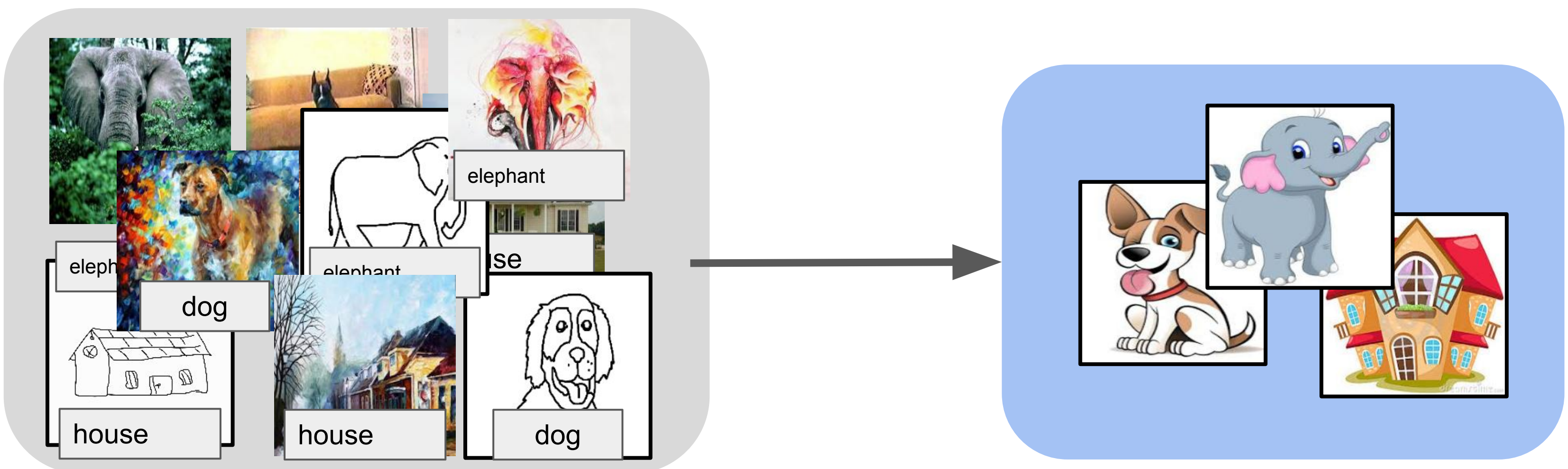
Unsupervised Domain Adaptation



Multi-source Domain Adaptation

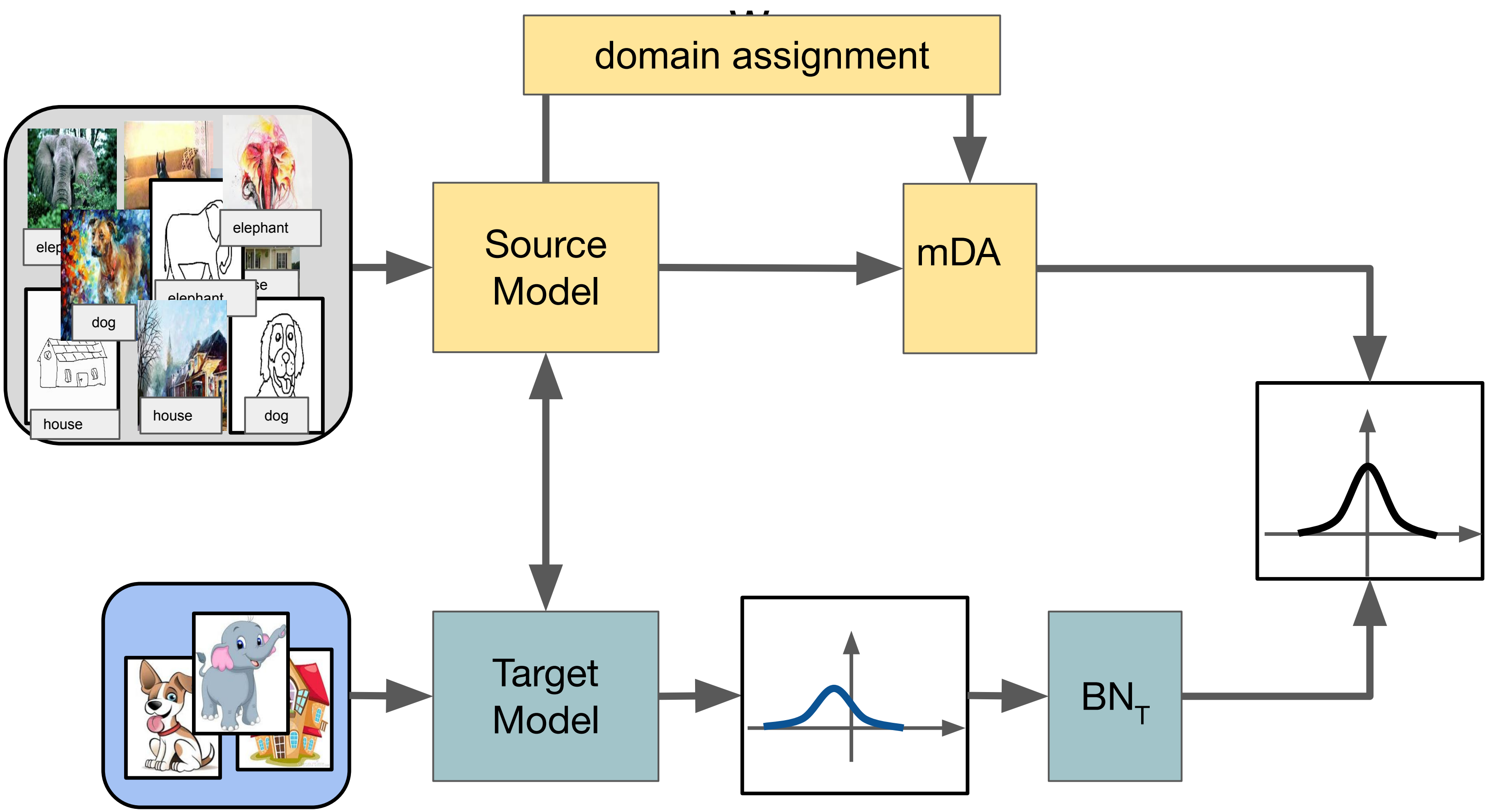


Domain Discovery + Adaptation



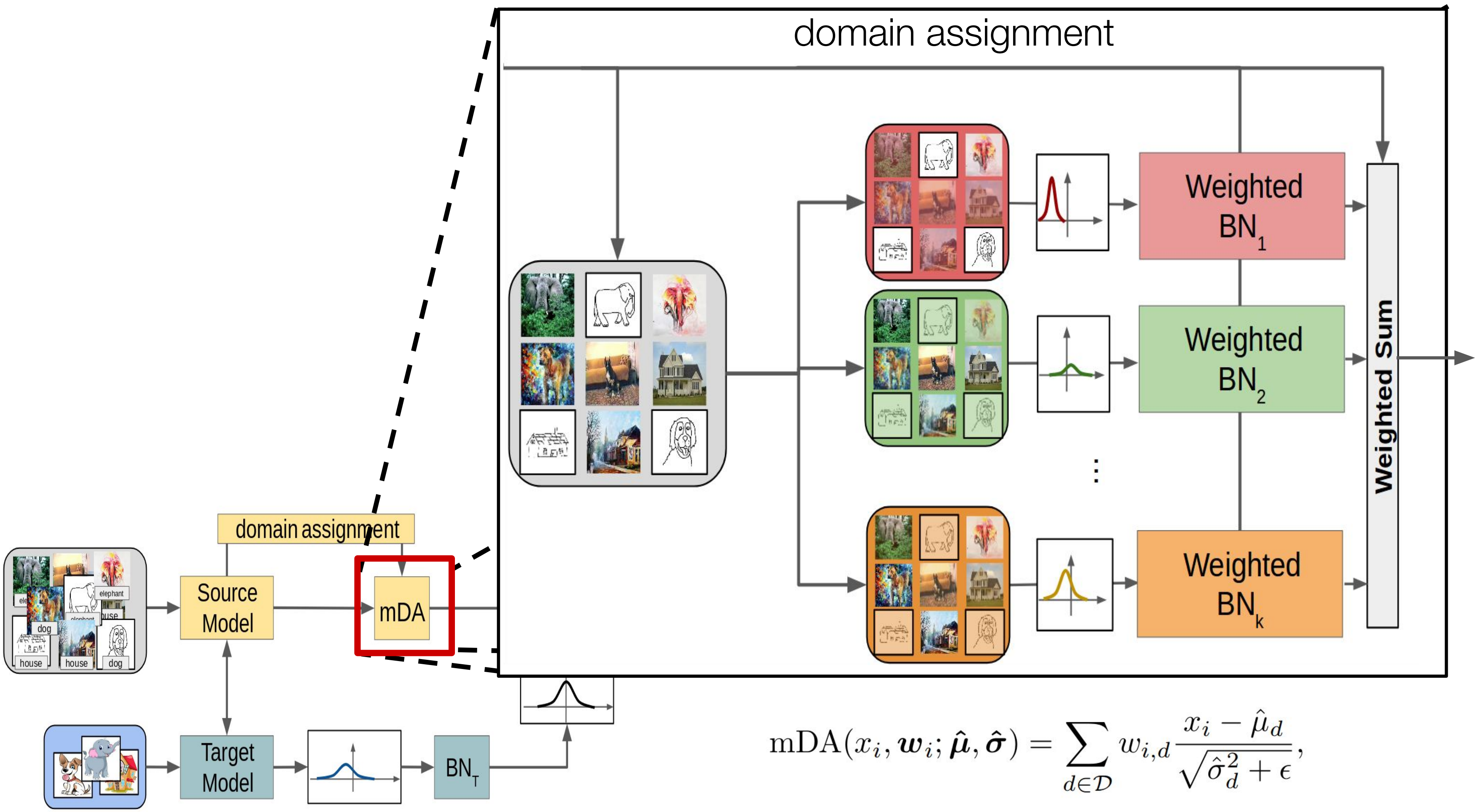
?

Multi Domain Alignment Layer (mDA)



Mancini, M., Porzi, L., Rota Bulò, S., Caputo, B., & Ricci, E. "Boosting Domain Adaptation by Discovering Latent Domains". *CVPR 2018.*

Multi Domain Alignment Layer (mDA)



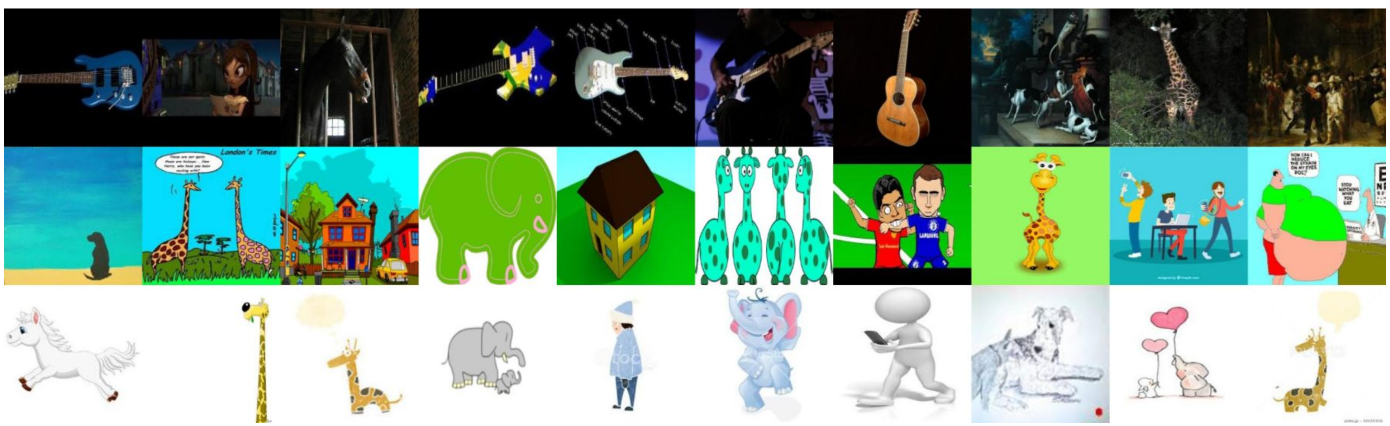
$$\mathrm{mDA}(x_i, \boldsymbol{w}_i; \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\sigma}}) = \sum_{d \in \mathcal{D}} w_{i,d} \frac{x_i - \hat{\mu}_d}{\sqrt{\hat{\sigma}_d^2 + \epsilon}},$$

Mancini, M., Porzi, L., Rota Bulò, S., Caputo, B., & Ricci, E. "Boosting Domain Adaptation by Discovering Latent Domains". *CVPR 2018.*

Results

Cartoon as Target

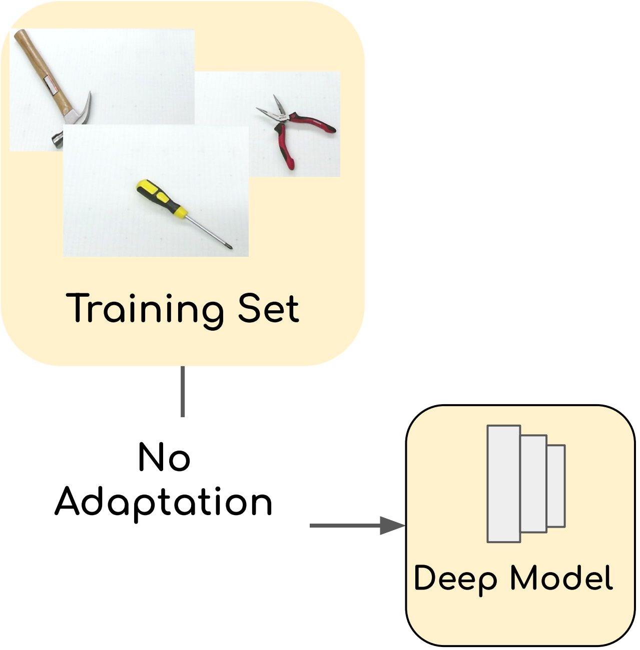
Sketch as Target

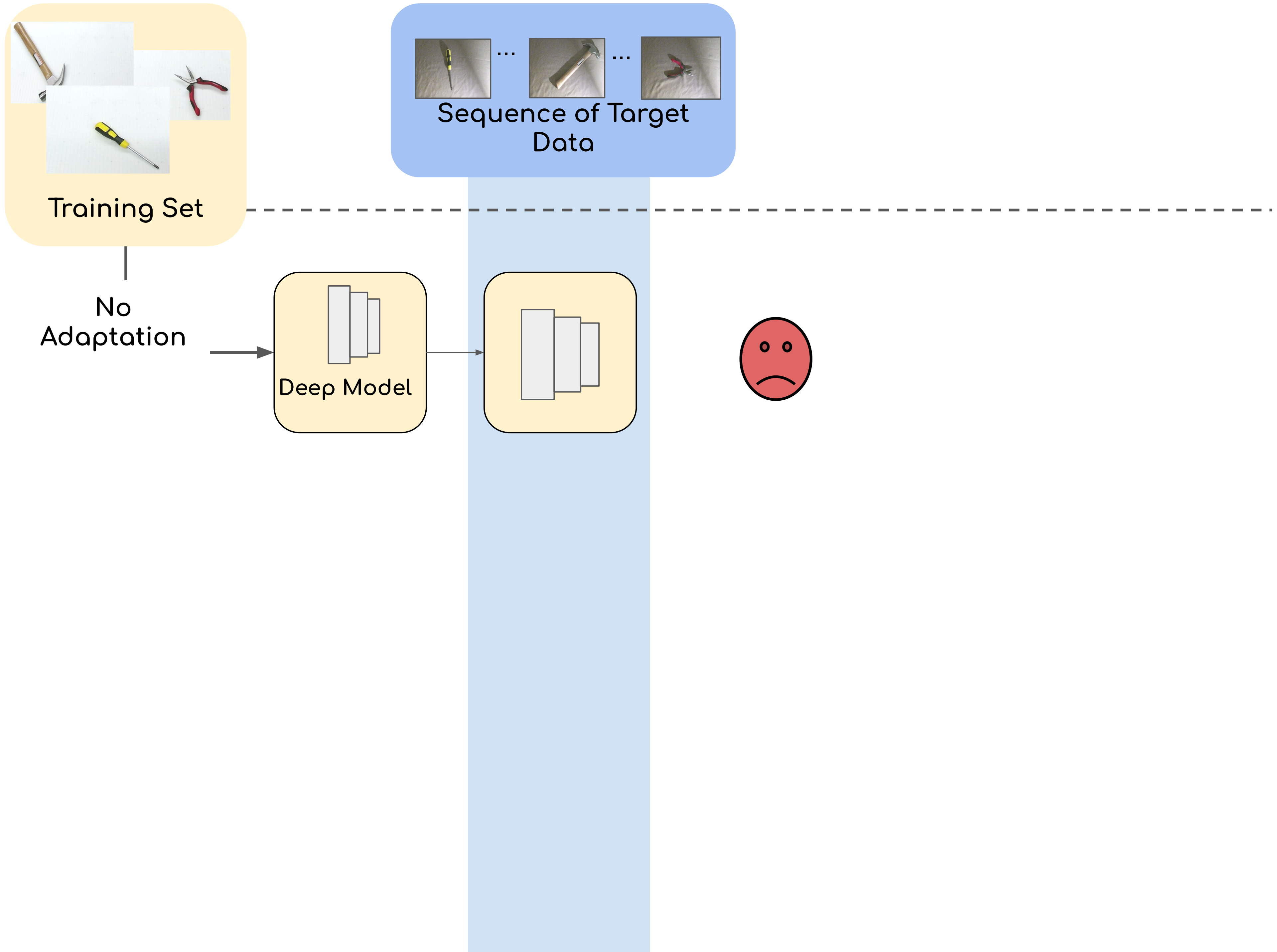


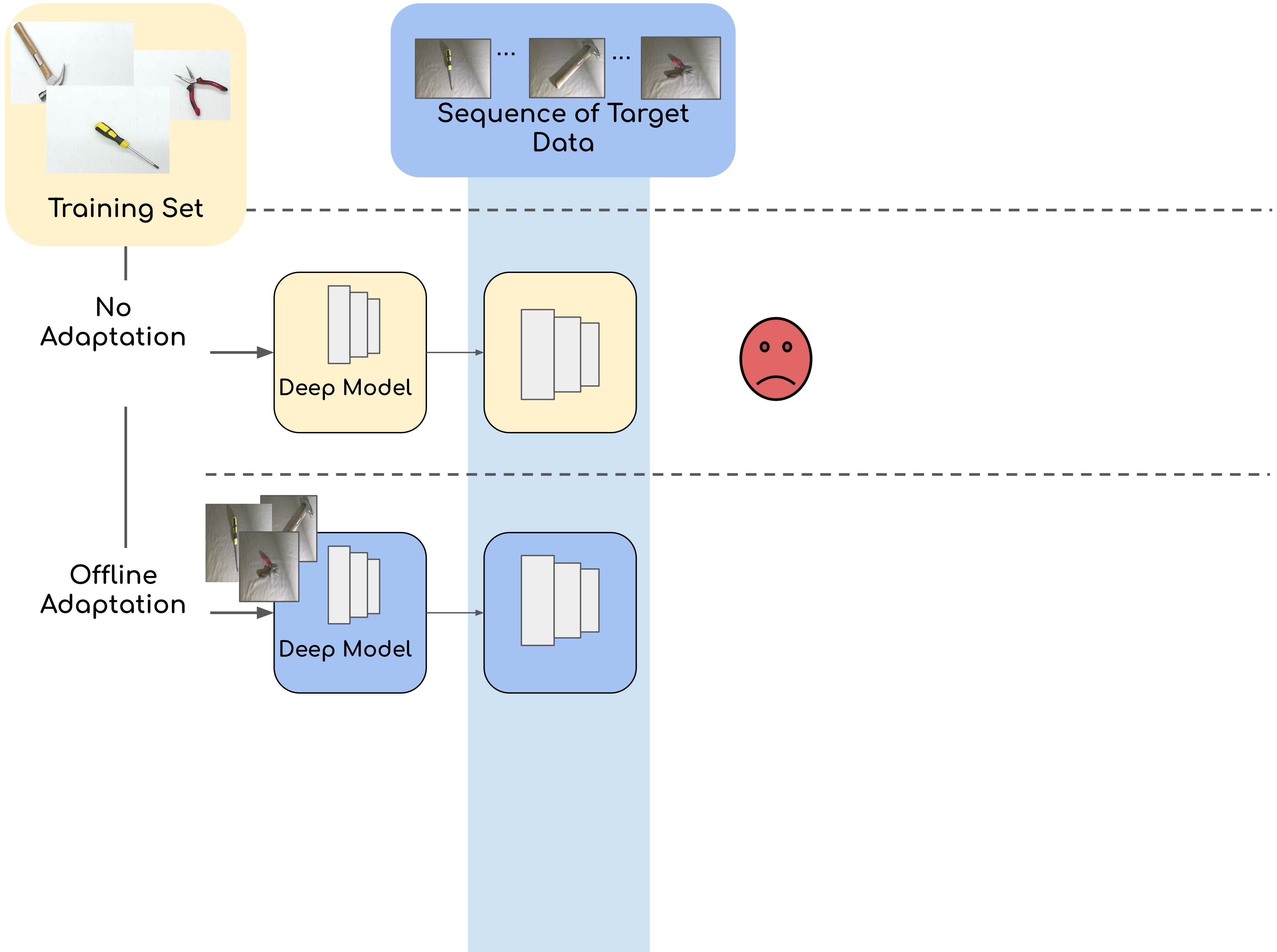
Mancini, M., Porzi, L., Rota Bulò, S., Caputo, B., & Ricci, E. "Boosting Domain Adaptation by Discovering Latent Domains". *CVPR 2018.*

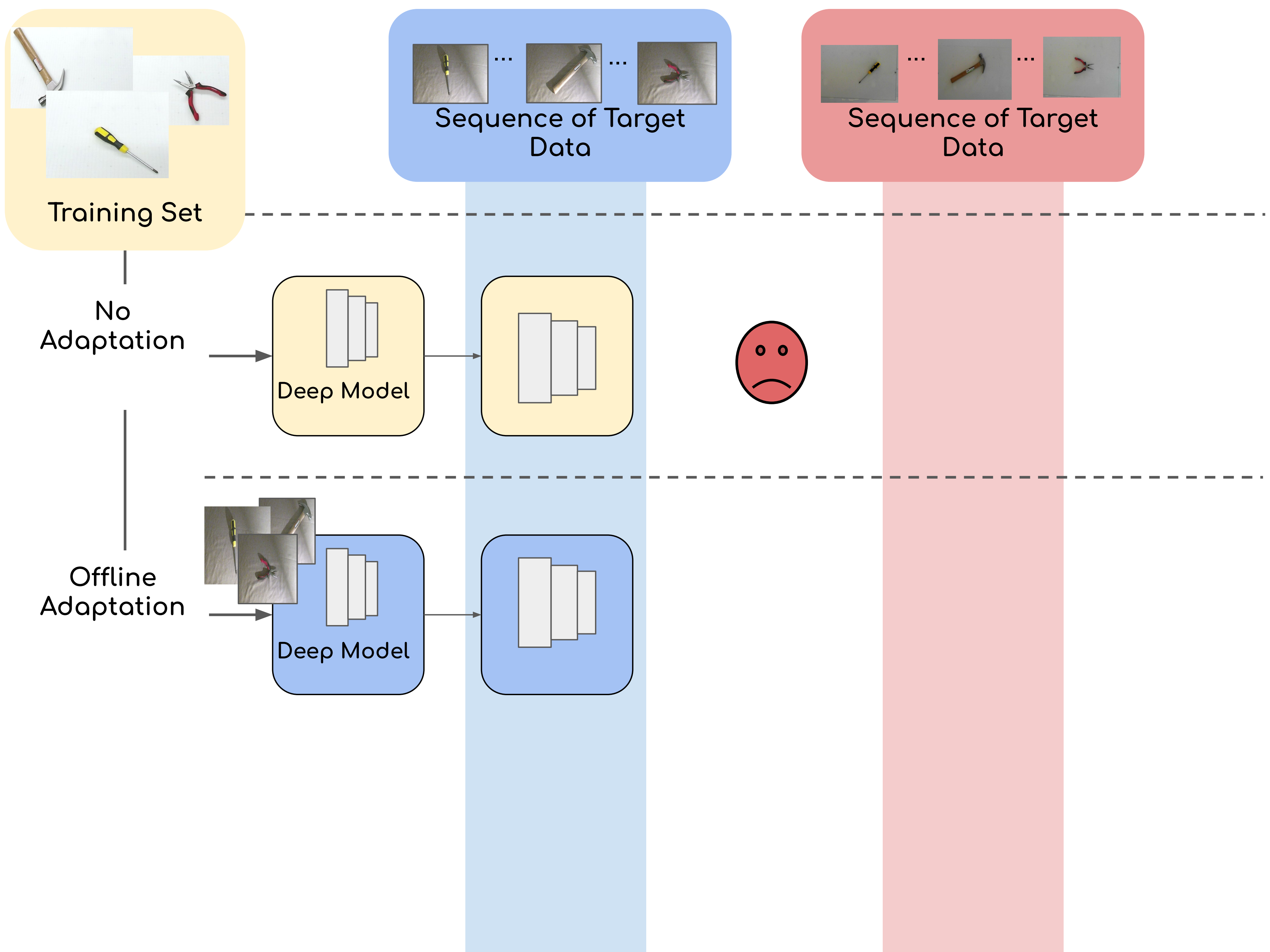
Continuous DA

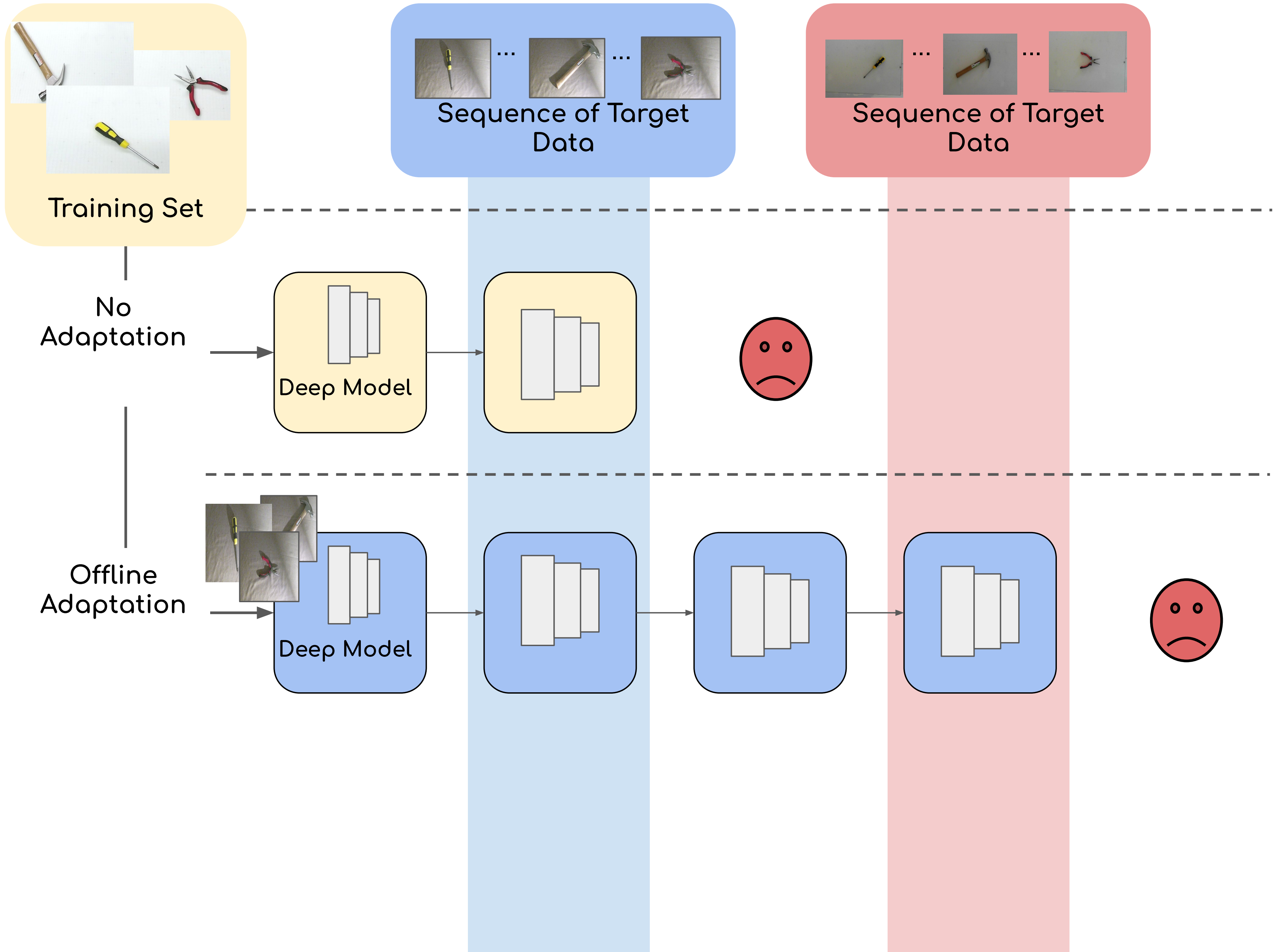
Training Set

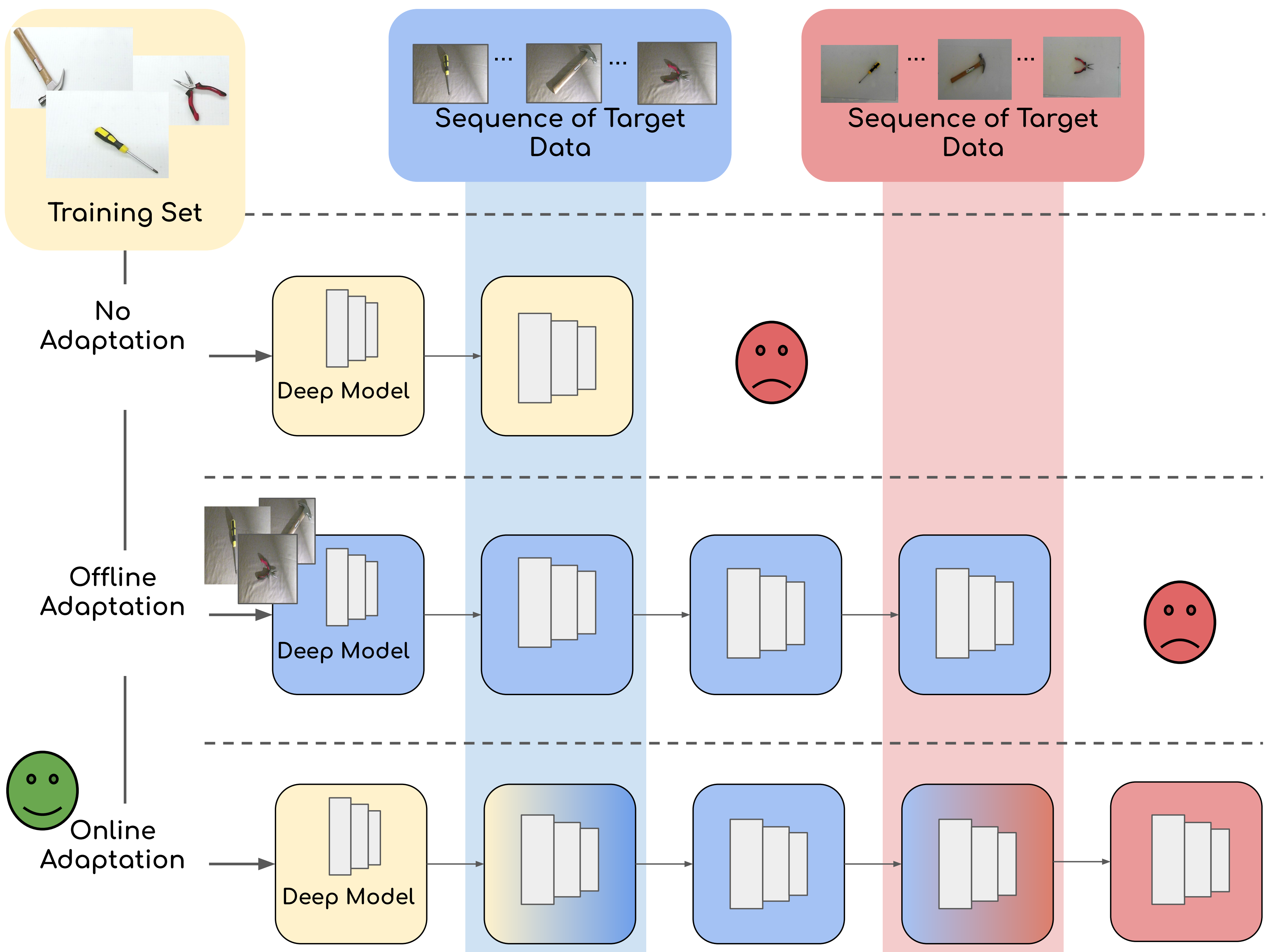

Training Set
No
Adaptation
Deep Model


Training Set
Sequence of Target Data
No Adaptation
Deep Model


Training Set
Sequence of Target Data
No Adaptation
Deep Model
Offline Adaptation
Deep Model


Training Set
Sequence of Target Data
Sequence of Target Data
No Adaptation
Deep Model
Offline Adaptation
Deep Model


Training Set
Sequence of Target Data
Sequence of Target Data
No Adaptation
Deep Model
Offline Adaptation
Deep Model


Training Set
Sequence of Target Data
Sequence of Target Data
No Adaptation
Deep Model
Offline Adaptation
Deep Model
Online Adaptation
Deep Model

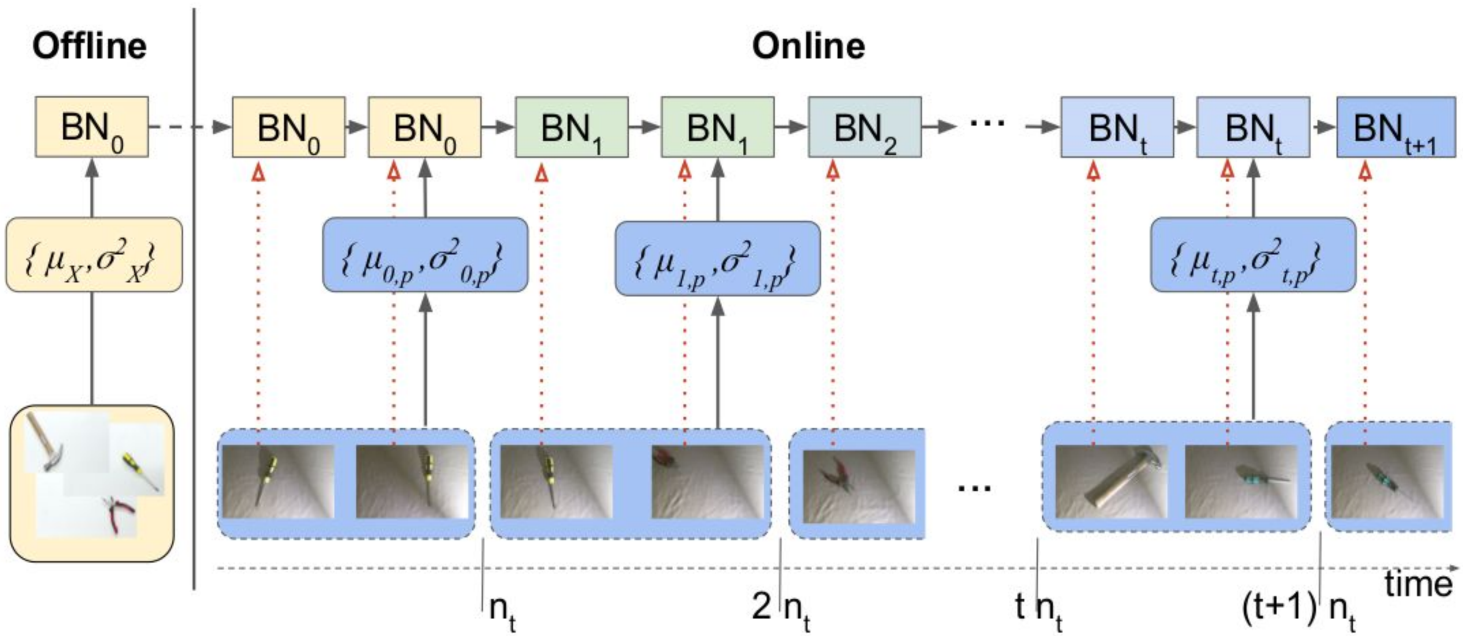
Online DA with Batch Normalization (ONDA)

1. Accumulate n_t frames
2. Use them to compute the statistics

$$\hat{\mu_t} = \frac{1}{n_t}\sum_{i=1}^{n_t} x_i \qquad \hat{\sigma_t}^2 = \frac{1}{n_t}\sum_{i=1}^{n_t}(x_i - \hat{\mu}_t)^2$$

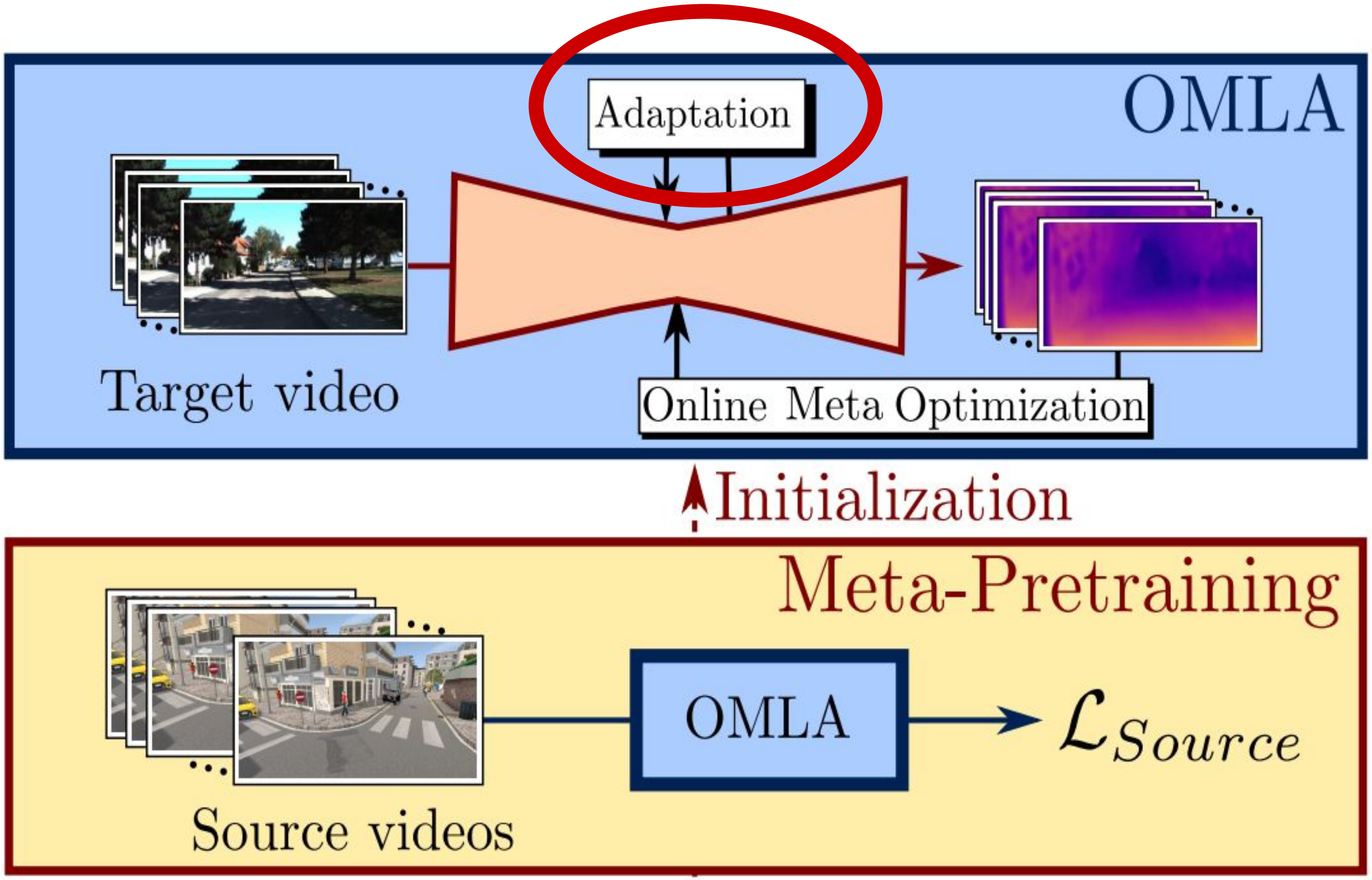
3. Update the BN layers

$$\mu_t = (1-\alpha)\mu_{t-1} + \alpha\hat{\mu}_t \qquad \sigma_t^2 = (1-\alpha)\sigma_{t-1}^2 + \alpha\frac{n_t}{n_t - 1}\hat{\sigma}_t^2$$



Mancini, M., Karaoguz, H., Ricci, E., Jensfelt, P., & Caputo, B. “Kitting in the Wild through Online Domain Adaptation”. *IROS 2018.*

...beyond classification!

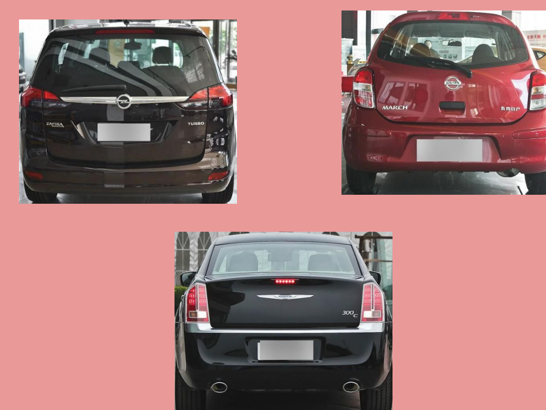


Zhang, Z., Lathuilière, S., Pilzer, A., Sebe, N., Ricci, E., & Yang, J. (2019). Online Adaptation through Meta-Learning for Stereo Depth Estimation. *arXiv preprint arXiv:1904.08462.*

Predictive DA

Standard DA

Source Domain



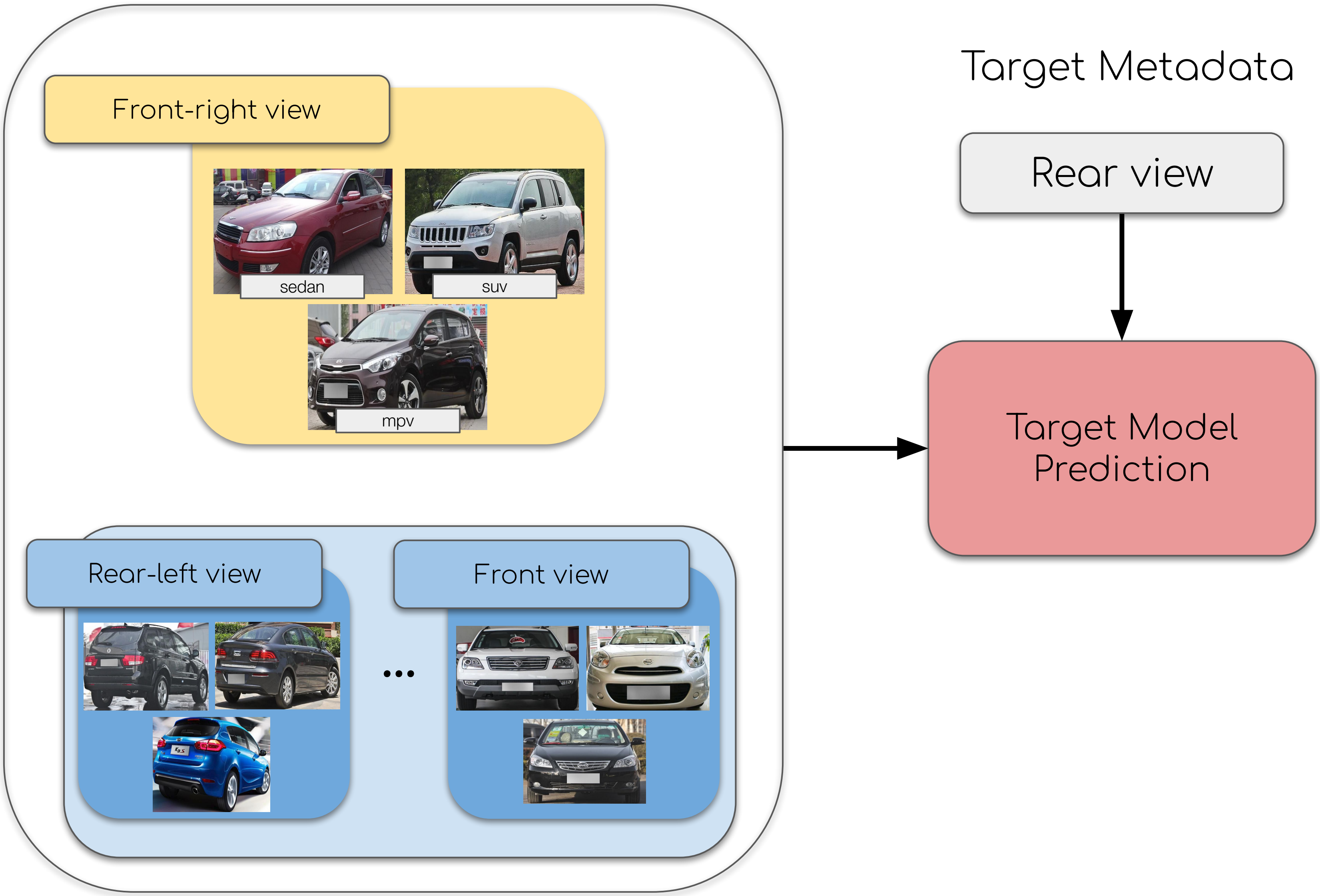
Target Domain

Predictive DA

Source Domain

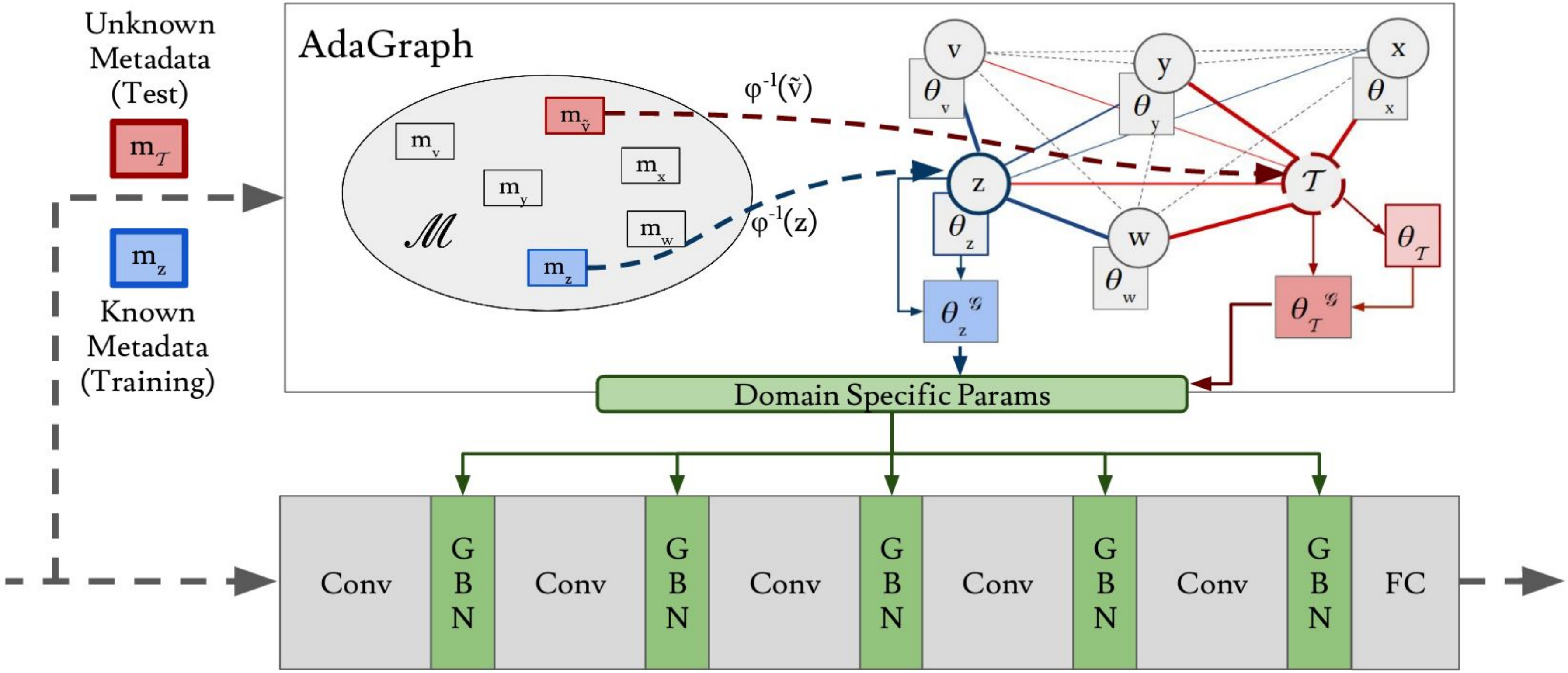
Rear view

Predictive DA



AdaGraph

$$\mathrm{GBN}(x, v, \mathcal{G}) = \gamma_v^{\mathcal{G}} \cdot \frac{x - \mu_v}{\sqrt{\sigma_v^2 + \epsilon}} + \beta_v^{\mathcal{G}}$$

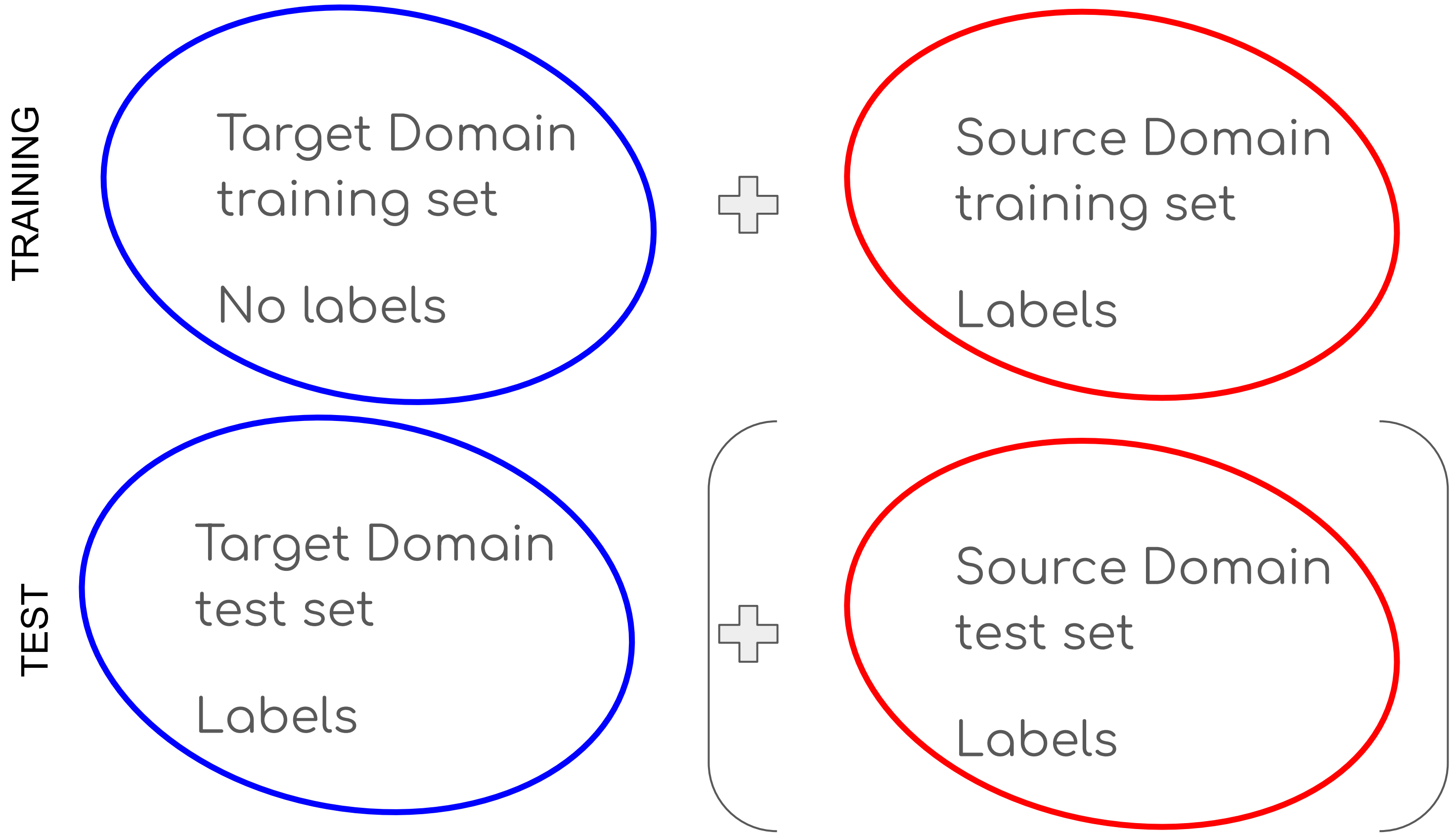


Mancini, M., Rota Bulò, S., Caputo, B., & Ricci, E. "AdaGraph: Unifying Predictive and Continuous Domain Adaptation through Graphs". *CVPR 2019.*

Unsupervised Domain Adaptation

Validation issues

Unsupervised Domain Adaptation - recap



Validation in UDA

How to ...

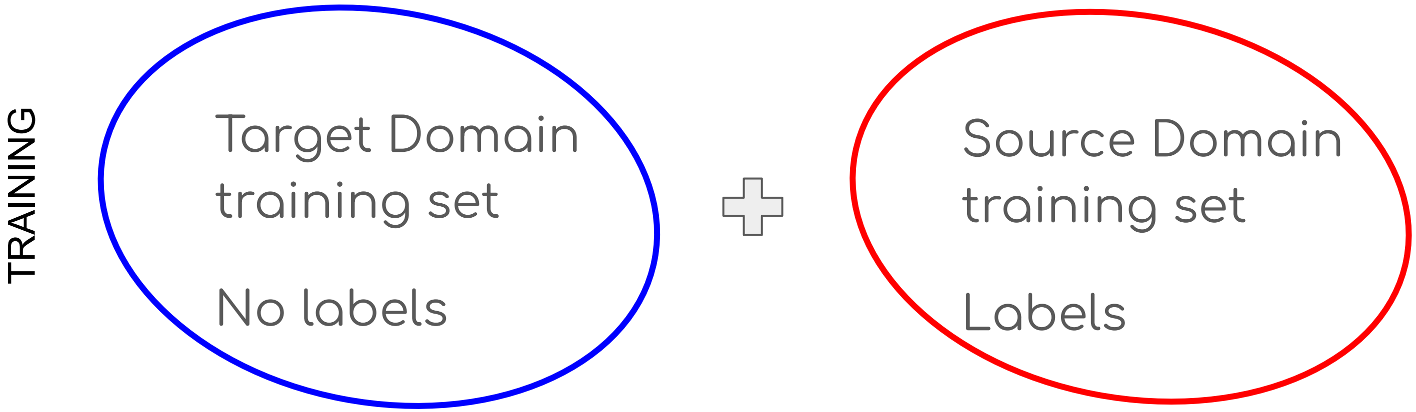
- choose/validate hyperparameters?
- Early-stop your training

Remember:

every time you peek at performance on the *target* set, you are actually using target labels.

Is your UDA method really UNSUPERVISED?

Validation set



Validation set - **a subset of your training set**

- Subset of the labelled source
- Subset of the unlabelled target
- Both the above
- (Small fraction of the target set *with labels)*

A step back: aligning distributions

Many methods try to **align source and target distribution** (in some appropriate feature space).

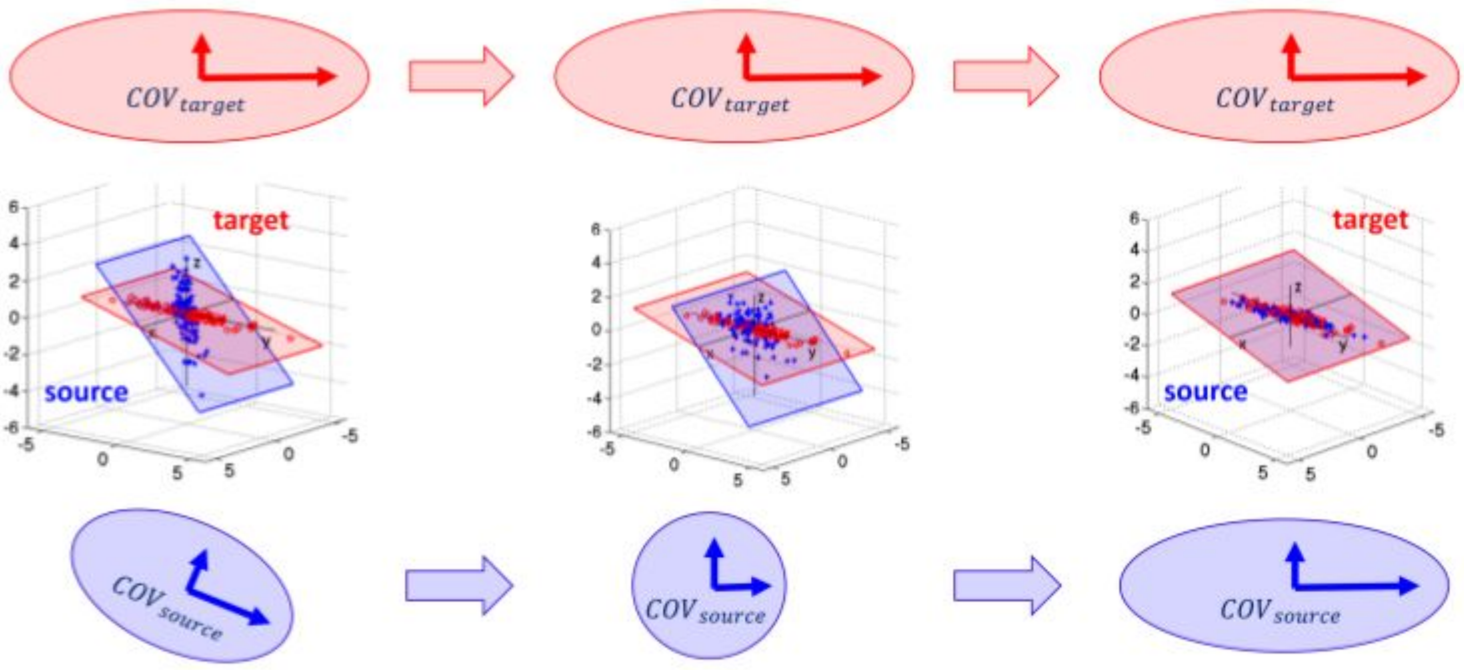
$$P(X^s) \Longleftrightarrow P(X^t)$$

Align distributions:

- First order momentum (zero mean features)
- Second order momentum (align covariances)
- ...

E.g. CORAL, deep CORAL

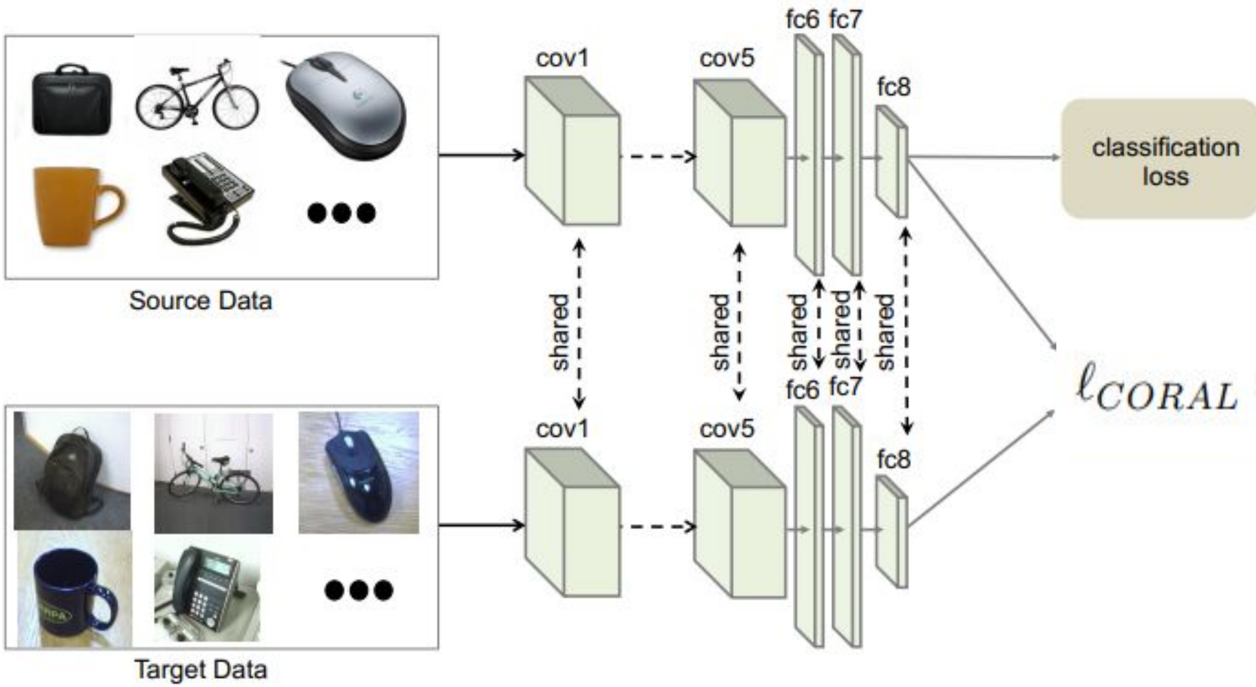
Aligning Covariances



Closed form alignment:

- Requires matrix inversion - not scalable to big datasets
- Requires a fixed feature representation

Deep correlation aligment



$$\ell_{CORAL} = \frac{1}{4d^2} \|C_S - C_T\|_F^2$$

$$\boxed{\ell = \ell_{CLASS.} + \sum_{i=1}^{t} \lambda_i \ell_{CORAL}}$$

- <u>How to validate lambda?</u>

Sun and Saenko "Deep CORAL: Correlation Alignment for Deep Domain Adaptation" *ECCV 2016 workshops*.

Validation on the (unlabelled) target set

(Euclidean) Correlation Alignment.

$$\min_\theta \left[H(\mathbf{X}_\mathcal{S}, \mathbf{Y}_\mathcal{S}) + \lambda \cdot \underbrace{\frac{1}{4d^2} \|\mathbf{C}_\mathcal{S} - \mathbf{C}_\mathcal{T}\|_F^2}_{\text{CORAL loss}} \right]. \quad (1)$$

Entropy Regularization.

$$\min_\theta \left[H(\mathbf{X}_\mathcal{S}, \mathbf{Y}_\mathcal{S}) + \gamma \cdot \underbrace{\left(- \sum_{\mathbf{z} \in \mathbf{Z}_\mathcal{T}} \langle f(\mathbf{z}; \theta), \log f(\mathbf{z}; \theta) \rangle \right)}_{\text{target entropy } E(\mathbf{Z}_\mathcal{T})} \right]. \quad (2)$$

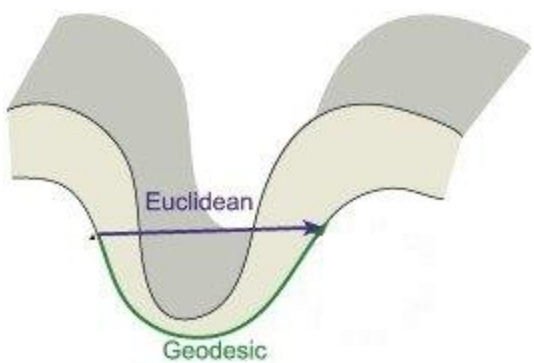
Theorem 1. If $\theta^\star$ optimally aligns correlation in (1), then, $\theta^\star$ minimizes (2) for every $\gamma > 0$.

Morerio et al. "Minimal entropy correlation alignment for unsupervised deep domain adaptation" *ICLR*, 2018.

Optimal correlation aligment on the manifold of SPD matrices

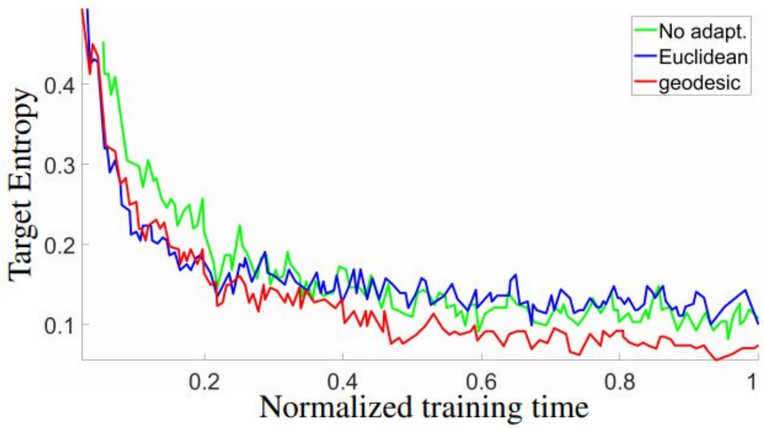
Minimal Entropy Correlation Alignment. *A more principled Riemannian alignment between covariance representations which give for free an unsupervised criterion for hyper-parameters tuning.*

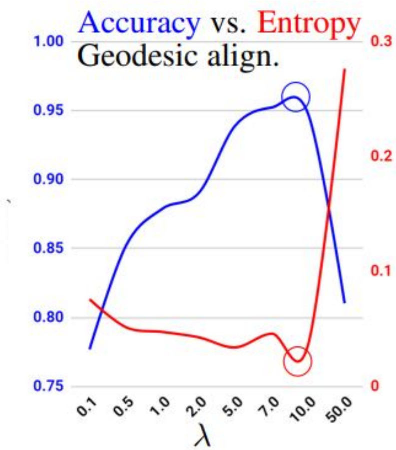
$$\boxed{\min_{\theta} \left[H(\mathbf{X}_{\mathcal{S}}, \mathbf{Y}_{\mathcal{S}}) + \lambda \cdot \underbrace{\frac{1}{4d^2} \|\log \mathbf{C}_{\mathcal{S}} - \log \mathbf{C}_{\mathcal{T}}\|_F^2}_{\text{Riemannian alignment}} \right] \quad \text{subject to} \quad \lambda \text{ minimizes } E(\mathbf{Z}_{\mathcal{T}}).}$$

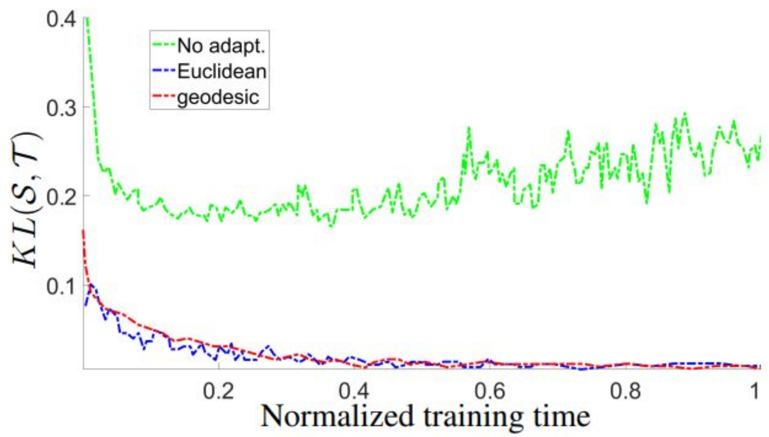


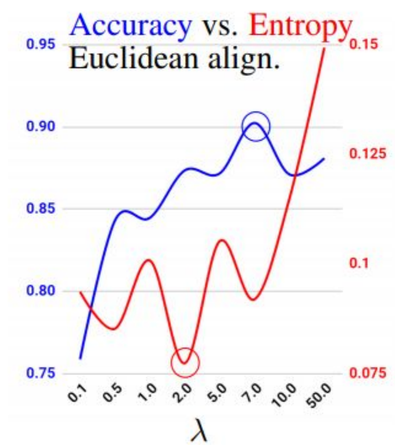
Morerio et al. "Minimal entropy correlation alignment for unsupervised deep domain adaptation" *ICLR*, 2018.

Optimal aligment induces minimal entropy









Morerio et al. “Minimal entropy correlation alignment for unsupervised deep domain adaptation” *ICLR*, 2018.