Supplement: Adaptation Across Extreme Variations using Unlabeled Bridges

Shuyang Dai[1], Kihyuk Sohn[2],
Yi-Hsuan Tsai[2], Lawrence Carin[1],
Manmohan Chandraker[2,3]

[1]Duke University
[2]NEC Labs America
[3]UC San Diego

S1 Proof of (8)

With source and bridging domains, $h^* = \arg\min_{h\in\mathcal{H}} \varepsilon(h)$, $h_1 = \arg\min_{h\in\mathcal{H}} \hat{\varepsilon}_S(h)$, an empirical minimizer of source error, and weight vector $\alpha = (0.5, 0.5)$, for any $\delta \in (0,1)$, with probability at least $1-\delta$, the target error can be bounded as follows:

$$\begin{aligned}\varepsilon_T(h_1) \le \varepsilon_T(h_T^*) + \tfrac{1}{2}\varepsilon_B(h_B^*) + 2\gamma + 2\eta \\ + d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_\alpha, \mathcal{D}_T) + \tfrac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_B)\end{aligned} \tag{S1}$$

where $h \in \mathcal{H}$ is a hypothesis and

$$d_{\mathcal{H}\Delta\mathcal{H}} = \sup_{h,h'\in\mathcal{H}} |P_{\mathcal{D}_S}(h(x) \neq h'(x)) - P_{\mathcal{D}_T}(h(x) \neq h'(x))| \tag{S2}$$

$$\gamma = \min_{h\in\mathcal{H}} \varepsilon_T(h) + \varepsilon_S(h) + \varepsilon_B(h) \tag{S3}$$

$$\eta = 2\sqrt{\left(\frac{2d\log(2m+1) + \log(\frac{4}{\delta})}{m}\right)} \tag{S4}$$

Proof. For the presentation clarity, we use $\varepsilon_\alpha = \frac{1}{2}\varepsilon_S + \frac{1}{2}\varepsilon_B$ interchangeably. Let $h_1, h_2, h_3 \in \mathcal{H}$, which will be defined later. We begin by bounding the target error ε_T by the mixture error ε_α and the divergence as follows:

$$\varepsilon_T(h_1) \le \varepsilon_T(h_2) + \varepsilon_\alpha(h_1, h_2) + \tfrac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(D_\alpha, D_T) \tag{S5}$$

The second term of RHS in (S5) is further bounded as follows:

$$\varepsilon_\alpha(h_1, h_2) = \tfrac{1}{2}\varepsilon_S(h_1, h_2) + \tfrac{1}{2}\varepsilon_B(h_1, h_2) \tag{S6}$$

$$\le \tfrac{1}{2}\left[\varepsilon_S(h_1) + \varepsilon_S(h_2)\right] + \tfrac{1}{2}\left[\varepsilon_B(h_1) + \varepsilon_B(h_2)\right] \tag{S7}$$

and $\varepsilon_B(h_1)$ is bounded as follows:

$$\varepsilon_B(h_1) \le \varepsilon_B(h_3) + \varepsilon_S(h_1, h_3) + \tfrac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) \tag{S8}$$

$$\le \varepsilon_B(h_3) + \varepsilon_S(h_1) + \varepsilon_S(h_3) + \tfrac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) \tag{S9}$$

© 2020. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

Plugging (S7) and (S9) into (S5), we get the following:

$$\varepsilon_T(h_1) \le \varepsilon_T(h_2) + \left[\tfrac{1}{2}\left[\varepsilon_S(h_1)+\varepsilon_S(h_2)\right] + \tfrac{1}{2}\left[\varepsilon_B(h_1)+\varepsilon_B(h_2)\right]\right] + \tfrac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_\alpha, D_T) \quad \text{(S10)}$$

$$= \left[\varepsilon_T(h_2) + \tfrac{1}{2}\varepsilon_S(h_2) + \tfrac{1}{2}\varepsilon_B(h_2)\right] + \tfrac{1}{2}\left[\varepsilon_B(h_3)+\varepsilon_S(h_3)\right] + \varepsilon_S(h_1) + \tfrac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_\alpha, D_T) + \tfrac{1}{4}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) \quad \text{(S11)}$$

Assuming $h_2 = \arg\min_h\{\varepsilon_T(h) + \frac{1}{2}\varepsilon_S(h) + \frac{1}{2}\varepsilon_B(h)\}$ and $h_3 = \arg\min_h\{\varepsilon_B(h) + \varepsilon_S(h)\}$, the RHS of (S11) is written as follows:

$$\varepsilon_T(h_1) \le \gamma_1 + \tfrac{1}{2}\gamma_2 + \varepsilon_S(h_1) + \tfrac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_\alpha, D_T) + \tfrac{1}{4}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) \quad \text{(S12)}$$

where $\gamma_1 = \min_h\{\varepsilon_T(h) + \frac{1}{2}\varepsilon_S(h) + \frac{1}{2}\varepsilon_B(h)\}$ and $\gamma_2 = \min_h\{\varepsilon_B(h) + \varepsilon_S(h)\}$. We further assume that $h_1 = \arg\min_{\hat{\varepsilon}_S(h)}$, an empirical minimizer of source error.

Now we are left with bounding the source error $\varepsilon_S(h_1)$ by the empirical source error $\hat{\varepsilon}_S(h_1)$, theoretical minimum errors of the target $\varepsilon_T(h_T^*)$ and the bridging $\varepsilon_B(h_B^*)$ domains. This is done by using Lemma 6 in [1] as follows:

$$\varepsilon_S(h_1) \le \hat{\varepsilon}_S(h_1) + \eta \quad \text{(S13)}$$

$$\le \tfrac{1}{2}\hat{\varepsilon}_S(h_T^*) + \tfrac{1}{2}\hat{\varepsilon}_S(h_B^*) + \eta \quad \text{(S14)}$$

$$\le \tfrac{1}{2}\varepsilon_S(h_T^*) + \tfrac{1}{2}\varepsilon_S(h_B^*) + 2\eta \quad \text{(S15)}$$

$$\le \tfrac{1}{2}\varepsilon_S(h_T^*) + \tfrac{1}{2}\left[\varepsilon_B(h_B^*) + \underbrace{\varepsilon_S(h_3)+\varepsilon_B(h_3)}_{=\gamma_2} + \tfrac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B)\right] + 2\eta \quad \text{(S16)}$$

$$\le \tfrac{1}{2}\varepsilon_S(h_T^*) + \tfrac{1}{2}\varepsilon_B(h_B^*) + \tfrac{1}{2}\gamma_2 + \tfrac{1}{4}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) + 2\eta \quad \text{(S17)}$$

$$\le \underbrace{\tfrac{1}{2}\varepsilon_S(h_T^*) + \tfrac{1}{2}\varepsilon_B(h_T^*)}_{=\varepsilon_\alpha(h_T^*)} + \tfrac{1}{2}\varepsilon_B(h_B^*) + \tfrac{1}{2}\gamma_2 + \tfrac{1}{4}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) + 2\eta \quad \text{(S18)}$$

$$\le \left[\varepsilon_T(h_T^*) + \underbrace{\varepsilon_T(h_2)+\varepsilon_\alpha(h_2)}_{=\gamma_1} + \tfrac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_\alpha, D_T)\right] + \tfrac{1}{2}\varepsilon_B(h_B^*) + \tfrac{1}{2}\gamma_2 + \tfrac{1}{4}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) + 2\eta \quad \text{(S19)}$$

$$= \varepsilon_T(h_T^*) + \tfrac{1}{2}\varepsilon_B(h_B^*) + \gamma_1 + \tfrac{1}{2}\gamma_2 + \tfrac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_\alpha, D_T) + \tfrac{1}{4}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) + 2\eta \quad \text{(S20)}$$

where the second inequality is due to the fact that $h_1 = \arg\min_h \hat{\varepsilon}_S$ and sixth is by adding $\frac{1}{2}\varepsilon_B(h_T^*)$ to RHS. η and 2η are introduced in the second and third inequalities using Lemma 6. Finally, plugging in (S20) into (S12), we get the following:

$$\varepsilon_T(h_1) \le \varepsilon_T(h_T^*) + \tfrac{1}{2}\varepsilon_B(h_B^*) + 2\gamma_1 + \gamma_2 + d_{\mathcal{H}\Delta\mathcal{H}}(D_\alpha, D_T) + \tfrac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) + 2\eta \quad \text{(S21)}$$

$$\le \varepsilon_T(h_T^*) + \tfrac{1}{2}\varepsilon_B(h_B^*) + 2\gamma + d_{\mathcal{H}\Delta\mathcal{H}}(D_\alpha, D_T) + \tfrac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_B) + 2\eta \quad \text{(S22)}$$

where the last inequality is given that $\min_h f(h) + \min_h g(h) \le \min_h\{f(h)+g(h)\}$ for any f and g. □

S2 Additional Experiments

S2.1 Digit Classification

In Table S1, we describe the model architecture used in this experiment.

Generator	Discriminator	Feature Extractor
Input feature f	Input feature f	Input X
MLP output 10	MLP output 128, ReLU MLP output 2	3×3 conv. 32 ReLU, stride 1 3×3 conv. 32 ReLU, stride 1, 2×2 max pool 2 3×3 conv. 64 ReLU, stride 1 3×3 conv. 64 ReLU, stride 1, 2×2 max pool 2 3×3 conv. 128 ReLU, stride 1 3×3 conv. 128 ReLU, stride 1, 2×2 max pool 2 Reshape to $128 \times 2 \times 2$ MLP output feature f with shape 128

Table S1: Architecture for Digit Classification Experiment

S2.2 Recognizing Cars in SV Domain at Night

Model architecture is listed in Table S2. Additional experiment results on Web→SVx:5, for $x = 1,2,3,4$ are shown in Figure S1, e.g., SV3:5 denotes SV3→SV4→SV5.

Generator	Discriminator	Feature Extractor
Input feature f	Input feature f	Input X
MLP output 431	MLP output 320, ReLU MLP output 2	7×7 conv. 64 ReLU, stride 2, 3×3 max pool 2 Resnet output 64 Resnet output 128 Resnet output 256 Resnet output 512 Resnet output 512 output feature f with shape 512

Table S2: Architecture for Car Recognition Experiment

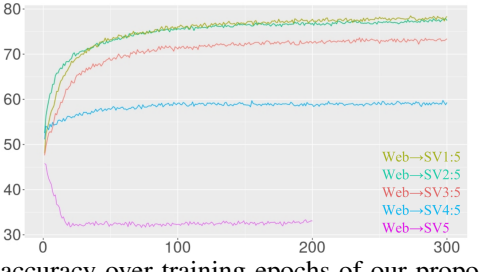

Figure S1: Validation accuracy over training epochs of our proposed domain adaptation framework with bridging domains. SV5 is used for validation set.

S2.3 Unsupervised Discovery of Bridging Domains

While works on unsupervised discovery of latent domains exist [2, 3, 5], the choice of bridging domains remains a hard, unsolved problem. In this section, we present several approaches that we have exploited along this direction. Our initial approach is to quantify the closeness to the source domain of each image in the target domain by using the discriminator score

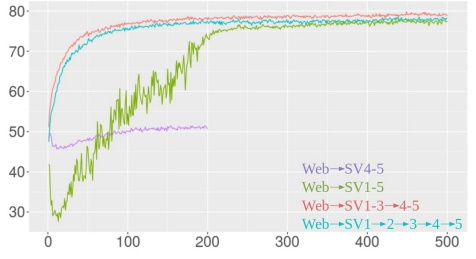

Figure S2: Performance on SV4–5 based on *supervised* bridging domain discovery using ground truth lighting conditions.

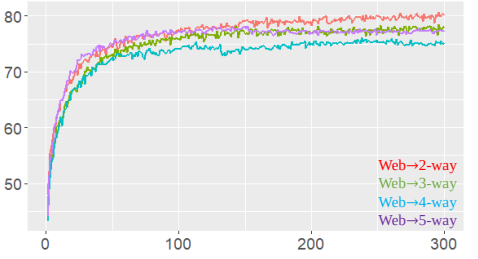

Figure S3: Performance on SV4–5 based on *unsupervised* bridging domain discovery using discriminator score of pretrained DANN.

$d_{\text{pre}}(f_{\text{pre}}(x))$ of pretrained DANN model as an indicator. This approach [10] intuitively makes sense as discriminator is trained to distinguish source and target domains, and those images from the target domain predicted as source domain are likely to be more similar to those images in the source domain, thus qualified as a bridging domain. Unfortunately, this is not necessarily true since the DANN is trained in an adversarial way and the discriminator at convergence should not be able to distinguish images from source and target domains [4]. Specifically, if we split the surveillance dataset into two domains based on the discriminator scores at each training epoch, and compute the AUC using the ground truth of day (SV1–3) and night (SV4–5) labels, we can see in Figure S7 that the AUC decreases as the number of training epochs increases. Meanwhile, as shown in Figure S4, S5 and S6, we visualize images from the surveillance domain based on the discriminator score from left-top the highest to right-bottom the lowest. With early stopping at the epoch 10, the discriminator of pretrained DANN model could be more discriminative in separating day and night images than those early stopped at epoch 50 and 150, which are closer to the convergence, thus cannot discriminate the images between the source and the target domains.

Based on our intuition and the visual inspection, we propose to construct bridging domains based on the discriminator score of the DANN model at epoch 10. By ranking the discriminator score $d_{\text{pre}}(f_{\text{pre}}(x))$ for $x \in \mathcal{D}_T$, we evenly split the unlabeled target data into m sub-domains, denoted as $\mathcal{D}_1, \cdots, \mathcal{D}_m$ for $m = 2, \cdots, 5$; $\mathcal{D}_1$ has the highest discriminator score and $\mathcal{D}_m$ the lowest. We then apply our proposed framework on $\mathcal{D}_1, \cdots, \mathcal{D}_m$ with $\mathcal{D}_m$ as the target and the rest as the bridging domains. Results are shown in Figure S3 (also Table 5 from the main paper). Note that we use the SV4–5 for validation and testing so that the results are comparable with the reported ones in the main paper. The performance of our framework using unsupervised bridging domain discovery is highly competitive to those using ground truth lighting condition to construct bridging domains. Moreover, our proposed framework with discovered bridging domains demonstrates much more stable training curve (Figure S3) comparing to the baseline DANN model (Figure S2, Web→SV1–5).

In addition, we evaluate the performance of our proposed adaptation framework with discovered bridging domains using DANN models at epoch 50 and 150. Using Web→2-way (78.62%) as a reference, the results are 76.31% and 67.46% respectively. This confirms our observation in Figure S7 that our framework is the most effective when the bridging domains are retrieved by the discriminator of DANN stopped early.

While using discriminator scores demonstrates the effectiveness in unsupervised bridging domain detection, additional model selection stage (i.e., early stopping) is required to find a reliable discriminator d_{pre}. To avoid this, we can directly use different measure of closeness

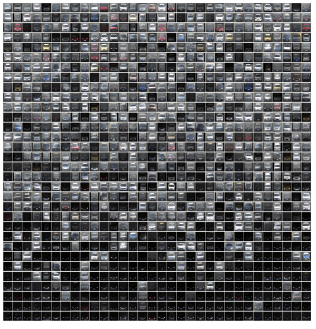

Figure S4: Early stopped at epoch 10.

Figure S5: Early stopped at epoch 50.

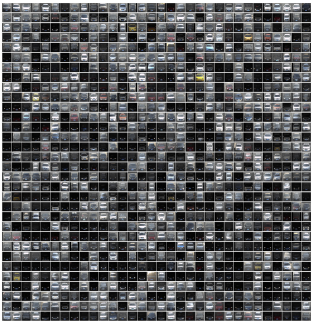

Figure S6: Early stopped at epoch 150.

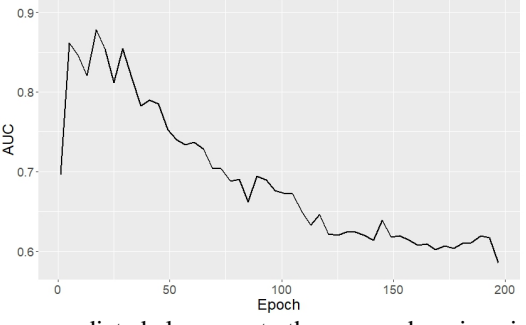

Figure S7: AUC between predicted closeness to the source domain using discriminator score and the ground-truth day/night labels at each training epoch.

in the feature space between the source and the target domains. Specifically, we propose two measures, namely, the maximum mean discrepancy (MMD) [6] and the out-of-distribution (OOD) sample detection score [7]. To evaluate these metrics, we first pretrain a classification model on the source domain only, with feature extractor f_{pre} and classifier C_{pre}. Then, the pretrained extractor is applied to each of the target domain data $f_{\text{pre}}(x), x \in \mathcal{D}_T$. We compute the MMD between a target feature $f_{\text{pre}}(x^{(T)})$ and the entire source domain distribution $\{f_{\text{pre}}(x^{(S)})\}$ as follows:

$$\text{MMD}(f_{\text{pre}}(x^{(T)}), \{f_{\text{pre}}(x^{(S)})\}) = \|\phi(f_{\text{pre}}(x^{(T)})) - \mathbb{E}_{x^{(S)}\sim\mathcal{D}_S}[\phi(f_{\text{pre}}(x^{(S)}))]\|_{\mathcal{H}}, \tag{S23}$$

where $\phi : \mathcal{D} \to \mathcal{H}$ is the kernel mapping, and $\mathcal{H}$ is the a reproducing kernel Hilbert space (RKHS). By ranking the target domain based on the MMD values (the smaller the MMD, the closer the target feature is to the source domain), we can split target domain into several sub domains, where the ones that are close to the source domain can be considered as the bridging domains.

Alternatively, we can use the out-of-distribution (OOD) sample detection methods [7]. Consider the output of the pretrained classifier for a target sample is $\hat{y}^{(T)} = C_{\text{pre}}(f_{\text{pre}}(x^{(T)}))$, such that $\hat{y}^{(T)} \sim \mathcal{Y}$ has N categories, denoted as $\hat{y}^{(T)} = \{\hat{y}_1^{(T)}, \cdots, \hat{y}_N^{(T)}\}$. Each $\hat{y}_i^{(T)}$ is the probability of $x^{(T)}$ being in category i. The OOD sample detection algorithm basically

calculates:

$$\text{OOD}(\hat{y}^{(T)}) = \max_i \sigma(\hat{y}_i^{(T)}), \tag{S24}$$

where $\sigma(\cdot)$ is the softmax function. The lower the value of $\text{OOD}(\hat{y}^{(T)})$ is, the more likely $x^{(T)}$ would be an out-of-distribution sample, and the further it is from the source domain. Similar to the MMD based approach, we can split the target domain based on the OOD score of every target domain sample.

As shown in Table 5 from the main paper, the discriminator score based approach achieves the highest AUC of 0.85. Without any requirement of early stopping, the MMD based approach provides an AUC of 0.79, and competitive model performance to the one from the discriminator score. The AUC from the OOD approach is relatively low at 0.69, and the model performance is lower than the other two. Moreover, we observe that the classification accuracy is well correlated with the AUC score, suggesting the importance of more advanced algorithms [8, 9] for measuring the closeness sensibly of the target example to the source domain.

References

[1] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 2010.

[2] Boqing Gong, Kristen Grauman, and Fei Sha. Reshaping visual datasets for domain adaptation. In *NIPS*, 2013.

[3] Boqing Gong, Kristen Grauman, and Fei Sha. Learning kernels for unsupervised domain adaptation with applications to visual object recognition. In *ICCV*, 2014.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[5] Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, 2011.

[6] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar): 723–773, 2012.

[7] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*, 2017.

[8] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NIPS*, 2018.

[9] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *ICLR*, 2018.

[10] Kihyuk Sohn, Sifei Liu, Guangyu Zhong, Xiang Yu, Ming-Hsuan Yang, and Manmohan Chandraker. Unsupervised domain adaptation for face recognition in unlabeled videos. In *ICCV*, 2017.