CyCADA: Cycle-Consistent Adversarial Domain Adaptation

Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu
BAIR, UC Berkeley
{jhoffman,etzeng,taesung_park,junyanz}@eecs.berkeley

Phillip Isola
OpenAI*
isola@eecs.berkeley

Kate Saenko
CS, Boston University
saenko@bu

Alexei A. Efros, Trevor Darrell
BAIR, UC Berkeley
{efros,trevor}@eecs.berkeley

Source image (GTA5)

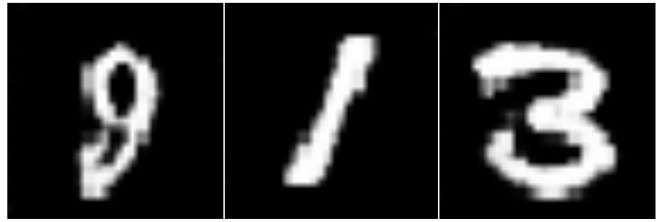
Adapted source image (**Ours**)

Target image (CityScapes)

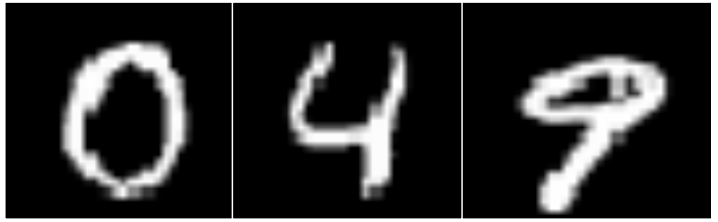
Pixel accuracy on target

Source-only:	54.0%
Adapted (**ours**):	**83.6%**

Source images (SVHN)



Adapted source images (**Ours**)



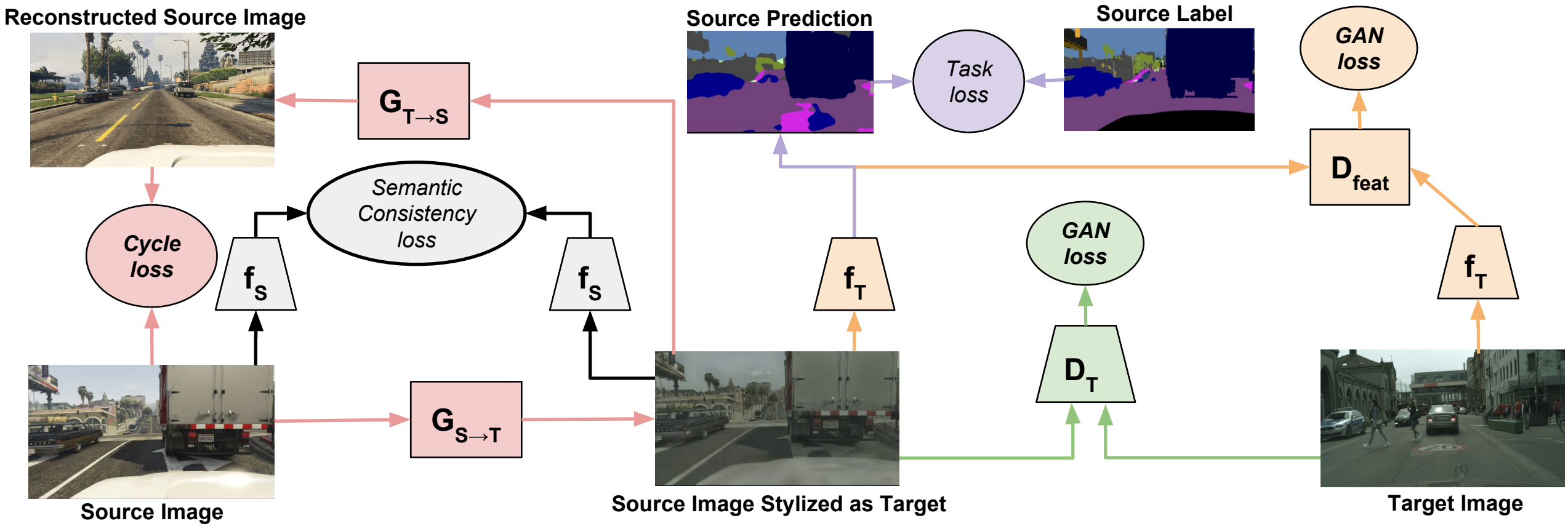
Target images (MNIST)

Accuracy on target

Source-only:	67.1%
Adapted (**ours**):	**90.4%**

Unsupervised Domain Adaptation

- Source data – Yes
- Source label – Yes
- Target data – Yes
- Target label – No


Reconstructed Source Image
Source Prediction
Source Label
GAN loss
Task loss
$G_{T\to S}$
D_{feat}
Semantic Consistency loss
Cycle loss
f_S
f_S
f_T
f_T
GAN loss
D_T
$G_{S\to T}$
Source Image
Source Image Stylized as Target
Target Image

Straightforward Approach

We can begin by simply learning a source model f_S that can perform the task on the source data. For K-way classification with a cross-entropy loss, this corresponds to

$$\mathcal{L}_{\text{task}}(f_S, X_S, Y_S) = -\mathbb{E}_{(x_s,y_s)\sim(X_S,Y_S)} \sum_{k=1}^{K} \mathbb{1}_{[k=y_s]} \log\left(\sigma(f_S^{(k)}(x_s))\right) \quad (1)$$

To this end, we introduce a mapping from source to target $G_{S\to T}$ and train it to produce target samples that fool an adversarial discriminator D_T. Conversely, the adversarial discriminator attempts to classify the real target data from the source target data. This corresponds to the loss function

$$\mathcal{L}_{\text{GAN}}(G_{S\to T}, D_T, X_T, X_S) = \mathbb{E}_{x_t\sim X_T}\left[\log D_T(x_t)\right] + \mathbb{E}_{x_s\sim X_S}\left[\log(1 - D_T(G_{S\to T}(x_s)))\right] \quad (2)$$

Problem: Resemble data drawn from target domain, but not content!

Proposed Cycle Consistence

sample, thereby enforcing cycle-consistency. In other words, we want $G_{T\to S}(G_{S\to T}(x_s)) \approx x_s$ and $G_{S\to T}(G_{T\to S}(x_t)) \approx x_t$. This is done by imposing an L1 penalty on the reconstruction error, which is referred to as the *cycle-consistency loss*:

$$\begin{aligned}\mathcal{L}_{\text{cyc}}(G_{S\to T}, G_{T\to S}, X_S, X_T) = &\mathbb{E}_{x_s\sim X_S}\left[||G_{T\to S}(G_{S\to T}(x_s)) - x_s||_1\right] \\ &+ \mathbb{E}_{x_t\sim X_T}\left[||G_{S\to T}(G_{T\to S}(x_t)) - x_t||_1\right].\end{aligned} \tag{3}$$

Semantic Consistence and Feature GAN Loss

can define the semantic consistency before and after image translation as follows:

$$\mathcal{L}_{\text{sem}}(G_{S\to T}, G_{T\to S}, X_S, X_T, f_S) = \mathcal{L}_{\text{task}}(f_S, G_{T\to S}(X_T), p(f_S, X_T)) + \mathcal{L}_{\text{task}}(f_S, G_{S\to T}(X_S), p(f_S, X_S)) \quad (4)$$

network. This would amount to an additional feature level GAN loss (see Figure 2 orange portion):

$$\mathcal{L}_{\text{GAN}}(f_T, D_{\text{feat}}, f_S(G_{S\to T}(X_S)), X_T). \quad (5)$$

Overall Loss Function

Taken together, these loss functions form our complete objective:

$$\begin{aligned}\mathcal{L}_{\text{CyCADA}}(f_T, X_S, X_T, Y_S, G_{S\to T}, G_{T\to S}, D_S, D_T) \\ = \mathcal{L}_{\text{task}}(f_T, G_{S\to T}(X_S), Y_S) \\ + \mathcal{L}_{\text{GAN}}(G_{S\to T}, D_T, X_T, X_S) + \mathcal{L}_{\text{GAN}}(G_{T\to S}, D_S, X_S, X_T) \\ + \mathcal{L}_{\text{GAN}}(f_T, D_{\text{feat}}, f_S(G_{S\to T}(X_S)), X_T) \\ + \mathcal{L}_{\text{cyc}}(G_{S\to T}, G_{T\to S}, X_S, X_T) + \mathcal{L}_{\text{sem}}(G_{S\to T}, G_{T\to S}, X_S, X_T, f_S).\end{aligned} \tag{6}$$

This ultimately corresponds to solving for a target model f_T according to the optimization problem

$$f_T^* = \arg\min_{f_T} \min_{\substack{G_{S\to T} \\ G_{T\to S}}} \max_{D_S, D_T} \mathcal{L}_{\text{CyCADA}}(f_T, X_S, X_T, Y_S, G_{S\to T}, G_{T\to S}, D_S, D_T). \tag{7}$$

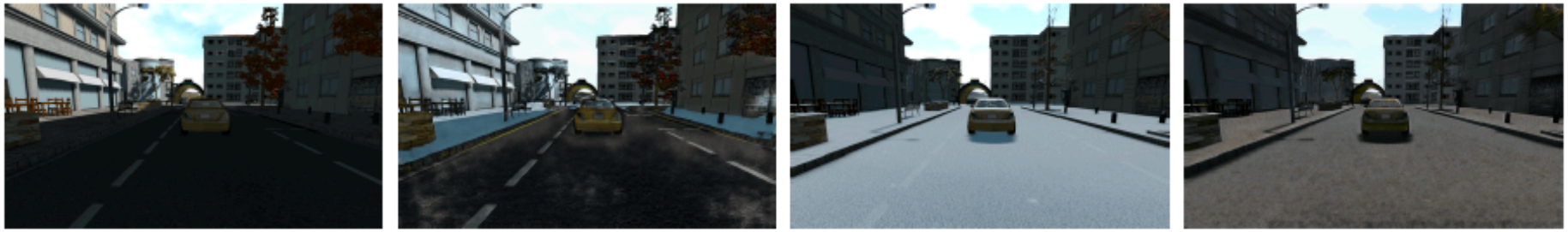
We implement G as a pixel-to-pixel convnet, f as a convnet classifier or a Fully-Convolutional Net (FCN) and D as a convnet with binary outputs.

Unsupervised domain adaptation across digit datasets

Model	MNIST → USPS	USPS → MNIST	SVHN → MNIST
Source only	82.2 ± 0.8	69.6 ± 3.8	67.1 ± 0.6
DANN (Ganin et al., 2016)	-	-	73.6
DTN (Taigman et al., 2017a)	-	-	84.4
CoGAN (Liu & Tuzel, 2016a)	91.2	89.1	-
ADDA (Tzeng et al., 2017)	89.4 ± 0.2	90.1 ± 0.8	76.0 ± 1.8
CyCADA pixel only	**95.6 ± 0.2**	96.4 ± 0.1	70.3 ± 0.2
CyCADA pixel+feat	**95.6 ± 0.2**	**96.5 ± 0.1**	**90.4 ± 0.4**
Target only	96.3 ± 0.1	99.2 ± 0.1	99.2 ±0.1

(a) Without Semantic Loss (b) Without Cycle Loss

Figure 3: **Ablation: Effect of Semantic or Cycle Consistency** Examples of translation failures without the semantic consistency loss. Each triple contains the original SVHN image (*left*), the image translated into MNIST style (*middle*), and the image reconstructed back into SVHN (*right*). (a) Without semantic loss, both the GAN and cycle constraints are satisfied (translated image matches MNIST style and reconstructed image matches original), but the image translated to the target domain lacks the proper semantics. (b) Without cycle loss, the reconstruction is not satisfied and though the semantic consistency leads to some successful semantic translations (*top*) there are still cases of label flipping (*bottom*).



(a) Fall (b) Fall → Winter (c) Winter (d) Winter → Fall

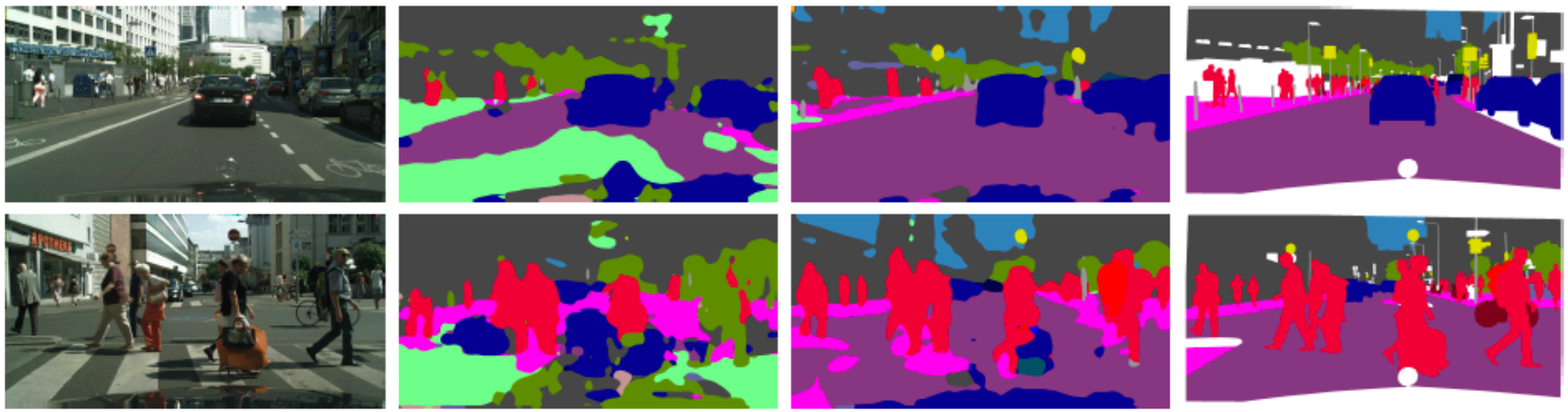
Figure 4: **Cross Season Image Translation.** Example image-space conversions for the SYNTHIA seasons adaptation setting. We show real samples from each domain (Fall and Winter) alongside conversions to the opposite domain.

SYNTHIA Fall → Winter																
	sky	building	road	sidewalk	fence	vegetation	pole	car	traffic sign	pedestrian	bicycle	lanemarking	traffic light	**mIoU**	**fwIoU**	**Pixel acc.**
Source only	91.7	80.6	79.7	12.1	71.8	44.2	26.1	42.8	49.0	38.7	45.1	41.3	24.5	49.8	71.7	82.3
FCNs in the wild	92.1	86.7	91.3	20.8	72.7	**52.9**	**46.5**	64.3	**50.0**	**59.5**	**54.6**	**57.5**	**26.1**	59.6	—	—
CyCADA pixel-only	**92.5**	**90.1**	**91.9**	**79.9**	**85.7**	47.1	36.9	**82.6**	45.0	49.1	46.2	54.6	21.5	**63.3**	**85.7**	**92.1**
Oracle (Train on target)	93.8	92.2	94.7	90.7	90.2	64.4	38.1	88.5	55.4	51.0	52.0	68.9	37.3	70.5	89.9	94.5

Table 3: Adaptation between seasons in the SYNTHIA dataset. We report IoU for each class and mean IoU, freq-weighted IoU and pixel accuracy. Our CyCADA method achieves state-of-the-art performance on average across all categories. *FCNs in the wild is by Hoffman et al. (2016).

(a) GTA5 (b) GTA5 → Cityscapes (c) CityScapes (d) CityScapes → GTA5

Figure 6: **GTA5 to CityScapes Image Translation.** Example images from the GTA5 (a) and Cityscapes (c) datasets, alongside their image-space conversions to the opposite domain, (b) and (d), respectively. Our model achieves highly realistic domain conversions.



(a) Test Image (b) Source Prediction (c) CyCADA Prediction (d) Ground Truth

Figure 5: **GTA5 to CityScapes Semantic Segmentation.** Each test CityScapes image (a) along with the corresponding predictions from the source only model (b) and our CyCADA model (c) are shown and may be compared against the ground truth annotation (d).

GTA5 → Cityscapes																							
	Architecture	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorbike	bicycle	**mIoU**	**fwIoU**	**Pixel acc.**
Source only	A	26.0	14.9	65.1	5.5	12.9	8.9	6.0	2.5	70.0	2.9	47.0	24.5	0.0	40.0	12.1	1.5	0.0	0.0	0.0	17.9	41.9	54.0
FCNs in the wild*	A	70.4	32.4	62.1	14.9	5.4	10.9	14.2	2.7	79.2	21.3	64.6	44.1	4.2	70.4	8.0	7.3	0.0	3.5	0.0	27.1	—	—
CyCADA feat-only	A	**85.6**	30.7	74.7	14.4	13.0	17.6	13.7	5.8	74.6	15.8	**69.9**	38.2	3.5	72.3	16.0	5.0	0.1	3.6	0.0	29.2	71.5	82.5
CyCADA pixel-only	A	83.5	**38.3**	**76.4**	20.6	**16.5**	22.2	**26.2**	**21.9**	**80.4**	28.7	65.7	49.4	4.2	74.6	16.0	26.6	2.0	8.0	0.0	34.8	73.1	82.8
CyCADA pixel+feat	A	85.2	37.2	**76.5**	**21.8**	15.0	**23.8**	22.9	21.5	**80.5**	**31.3**	60.7	**50.5**	**9.0**	**76.9**	**17.1**	**28.2**	**4.5**	**9.8**	0.0	**35.4**	**73.8**	**83.6**
Oracle - Target Super	A	96.4	74.5	87.1	35.3	37.8	36.4	46.9	60.1	89.0	54.3	89.8	65.6	35.9	89.4	38.6	64.1	38.6	40.5	65.1	60.3	87.6	93.1
Source only	B	42.7	26.3	51.7	5.5	6.8	13.8	23.6	6.9	75.5	11.5	36.8	49.3	0.9	46.7	3.4	5.0	0.0	5.0	1.4	21.7	47.4	62.5
CyCADA feat-only	B	78.1	31.1	71.2	10.3	14.1	29.8	28.1	20.9	74.0	16.8	51.9	53.6	6.1	65.4	8.2	20.9	1.8	13.9	5.9	31.7	67.4	78.4
CyCADA pixel-only	B	63.7	24.7	69.3	21.2	17.0	30.3	33.0	**32.0**	80.5	25.3	62.3	62.0	**15.1**	73.1	19.8	23.6	5.5	16.2	**28.7**	37.0	63.8	75.4
CyCADA pixel+feat	B	**79.1**	**33.1**	**77.9**	**23.4**	**17.3**	**32.1**	**33.3**	31.8	**81.5**	**26.7**	**69.0**	**62.8**	14.7	**74.5**	**20.9**	**25.6**	**6.9**	**18.8**	20.4	**39.5**	**72.4**	**82.3**
Oracle - Target Super	B	97.3	79.8	88.6	32.5	48.2	56.3	63.6	73.3	89.0	58.9	93.0	78.2	55.2	92.2	45.0	67.3	39.6	49.9	73.6	67.4	89.6	94.3