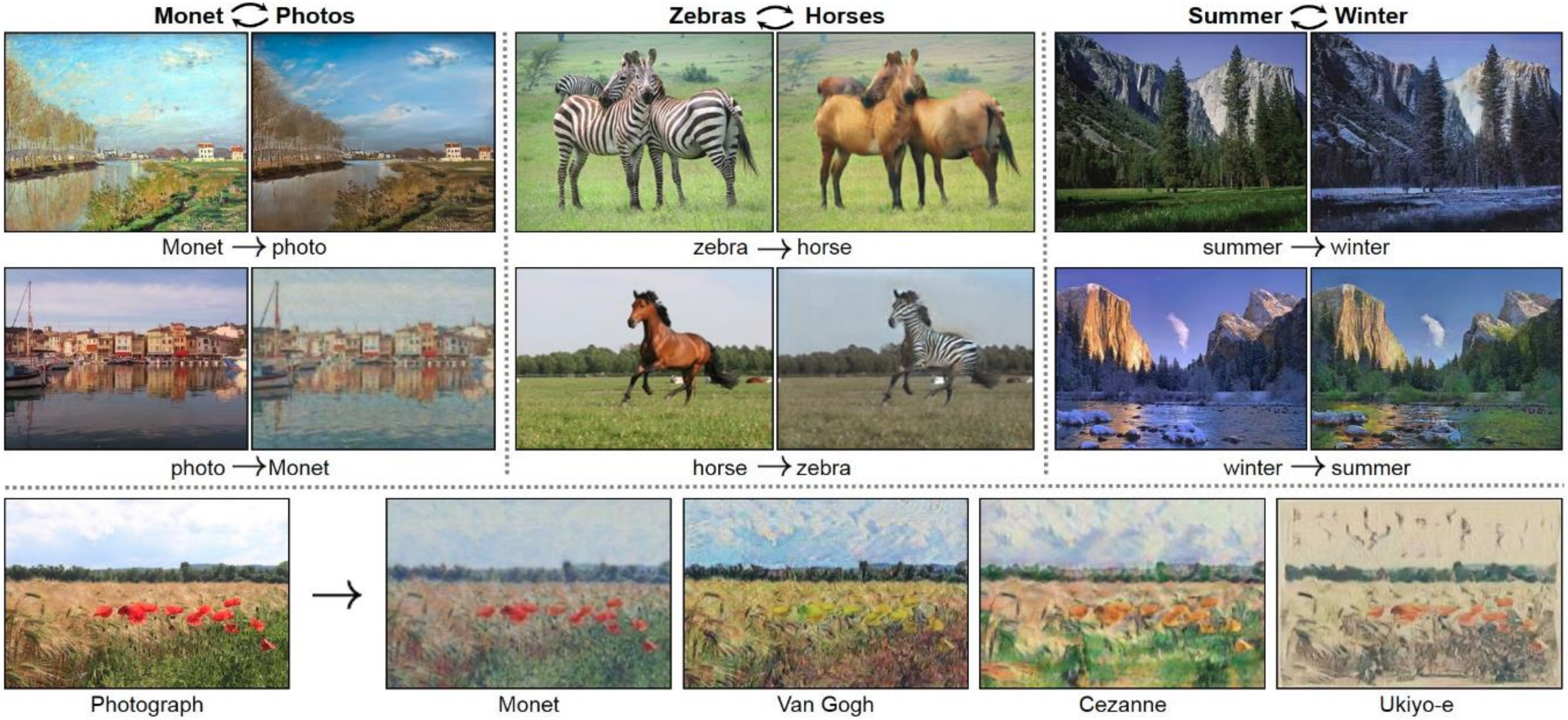
One-Shot Unsupervised Cross Domain Translation

Sagie Benaim and Lior Wolf

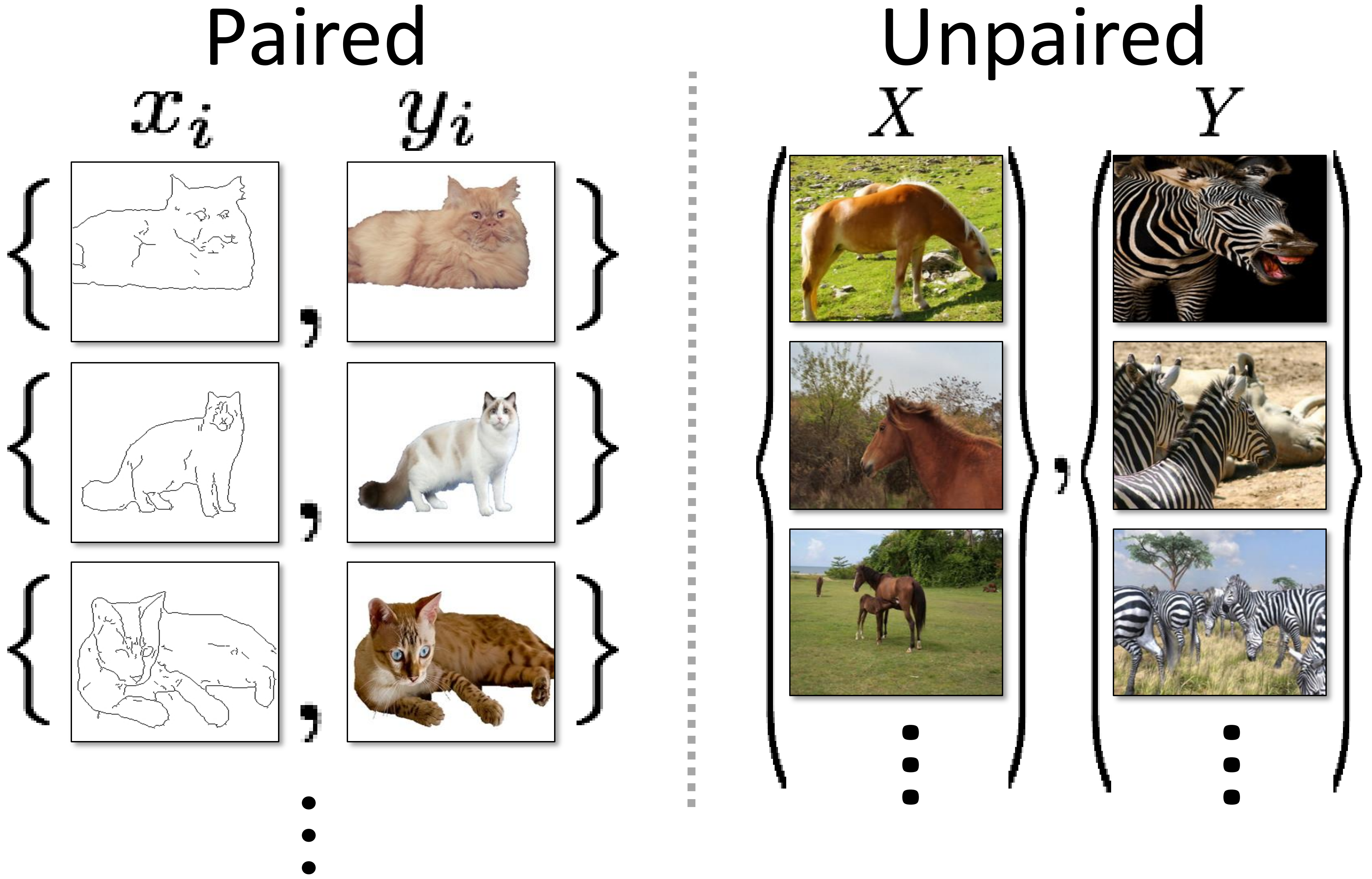
NeurIPS 2018

Image to Image Translation






	Supervised	Unsupervised
Unimodal	Pix2pix, CRN, SRGAN	DistanceGAN, CycleGAN, DiscoGAN, DualGAN, UNIT, DTN, StarGAN, OST
Multimodal	pix2pixHD, BicycleGAN	MUNIT, Augmented CycleGAN

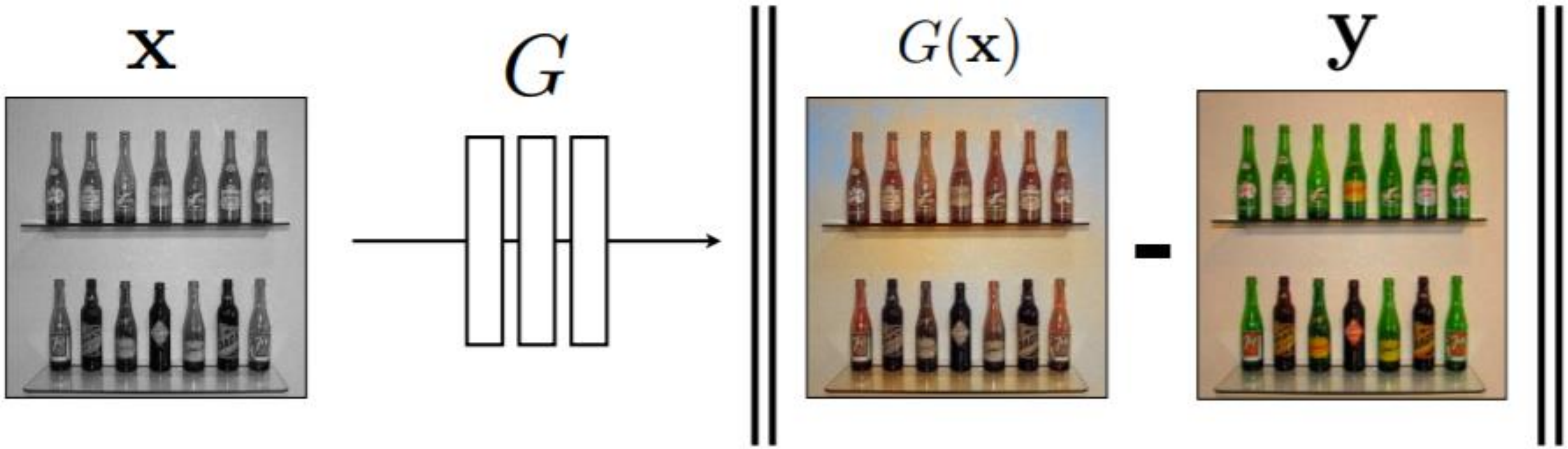


Paired
x_i
y_i
Unpaired
X
Y

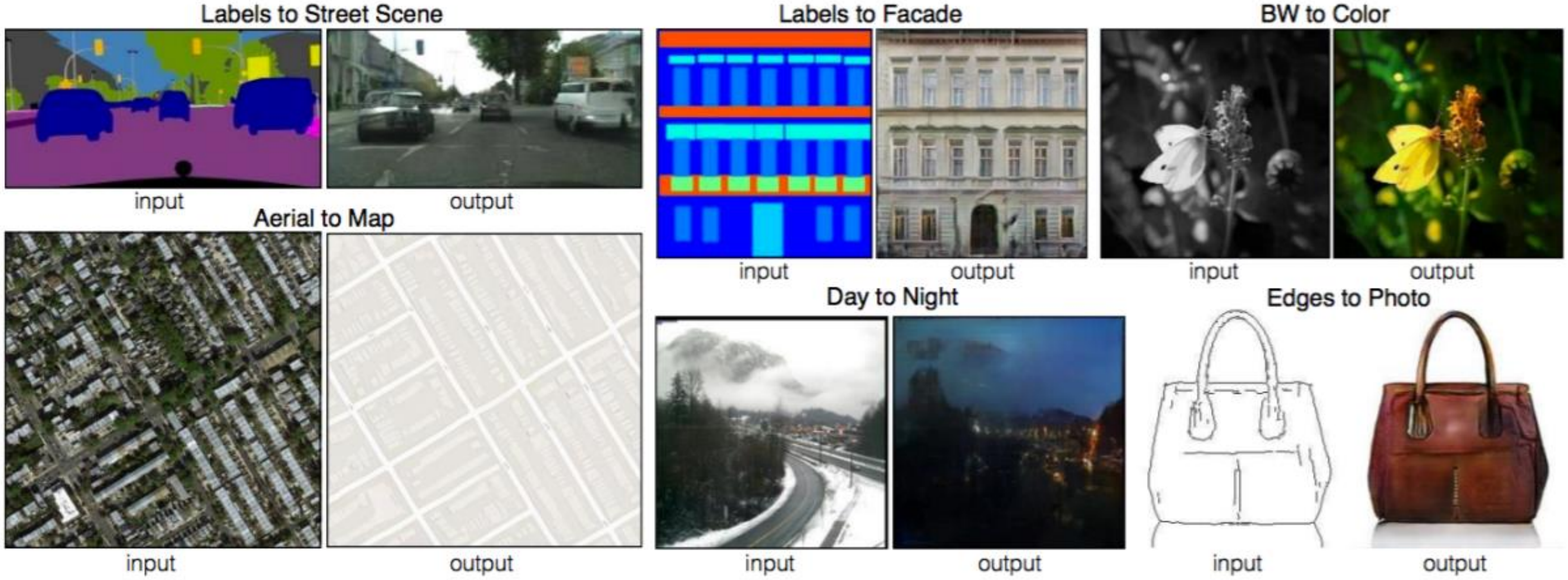
Fully Supervised: pix2pix

Conditional GAN

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G).$$



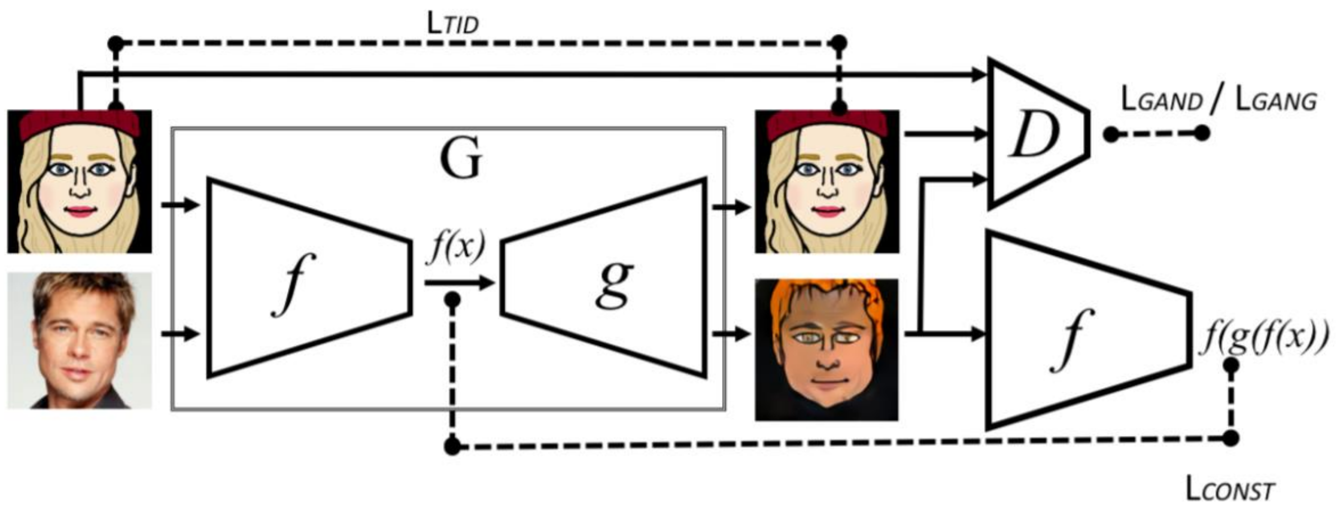
[Isola et al., CVPR 2017]



[Isola et al., CVPR 2017]

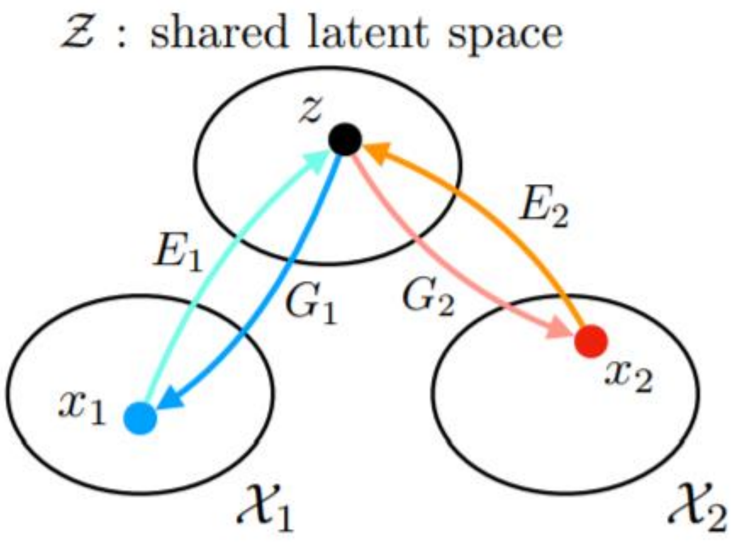
Partially Supervised Alignment

- "Unsupervised Cross-Domain Image Generation" Taigman et al.



Unsupervised Alignment

- Highly related domains
  - “Unsupervised Image-to-Image Translation Networks” Liu et al.



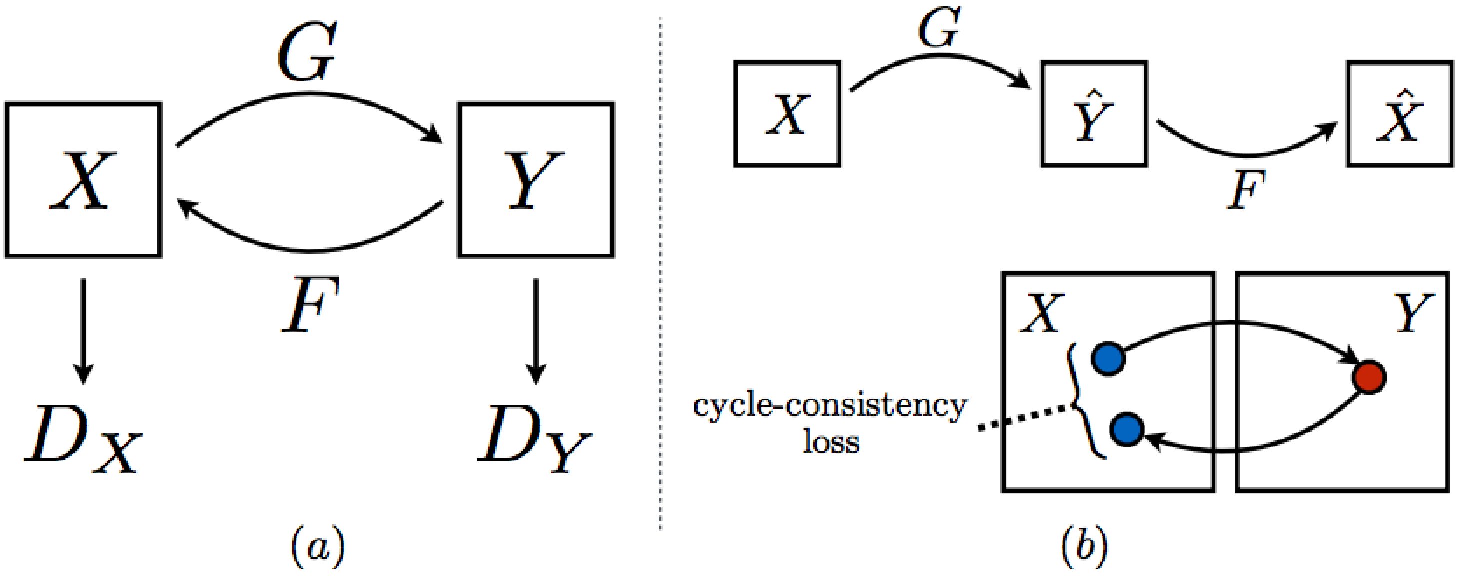
Circular GANs

DiscoGAN: “Learning to Discover Cross-Domain Relations with Generative Adversarial Networks”. Kim et al. ICML’17.

CycleGAN: “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks”. Zhu et al. arXiv:1703.10593, 2017.

DualGAN: “ Unsupervised Dual Learning for Image-to-Image Translation”. Zili et al. arXiv:1704.02510, 2017.

Circular GANs



Circular GANs (DiscoGAN, CycleGAN, DualGAN)

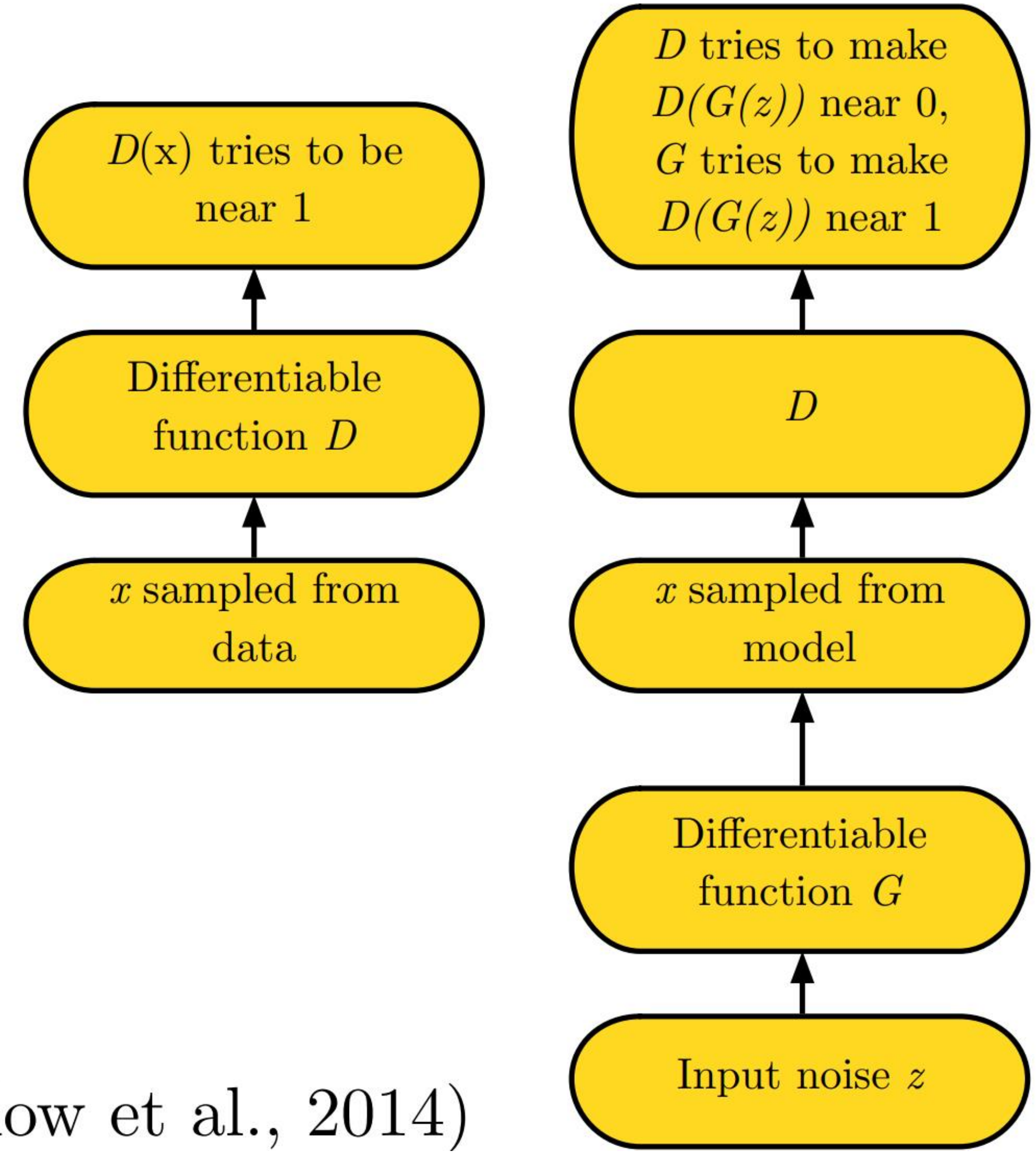
$$x \sim F(G(x))$$
$$y \sim G(F(y))$$

Generative Modeling: Sample Generation

Training Data
(CelebA)

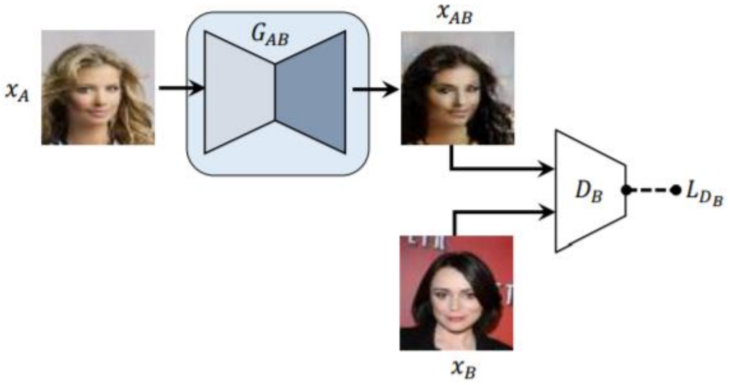
Sample Generator
(Karras et al, 2017)

Adversarial Nets Framework



(Goodfellow et al., 2014)

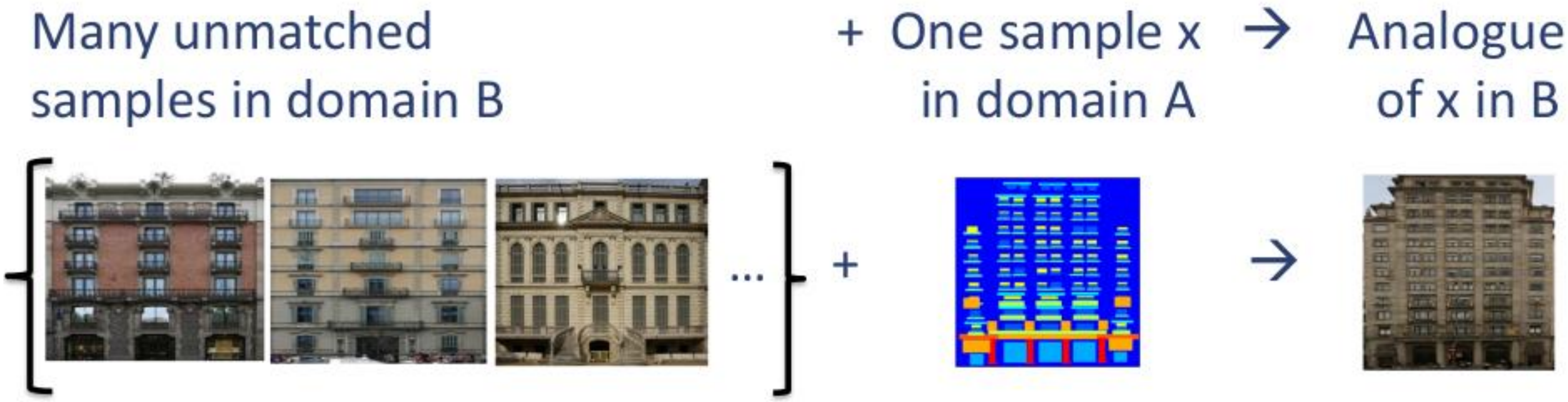
Building Block: Conditional GAN



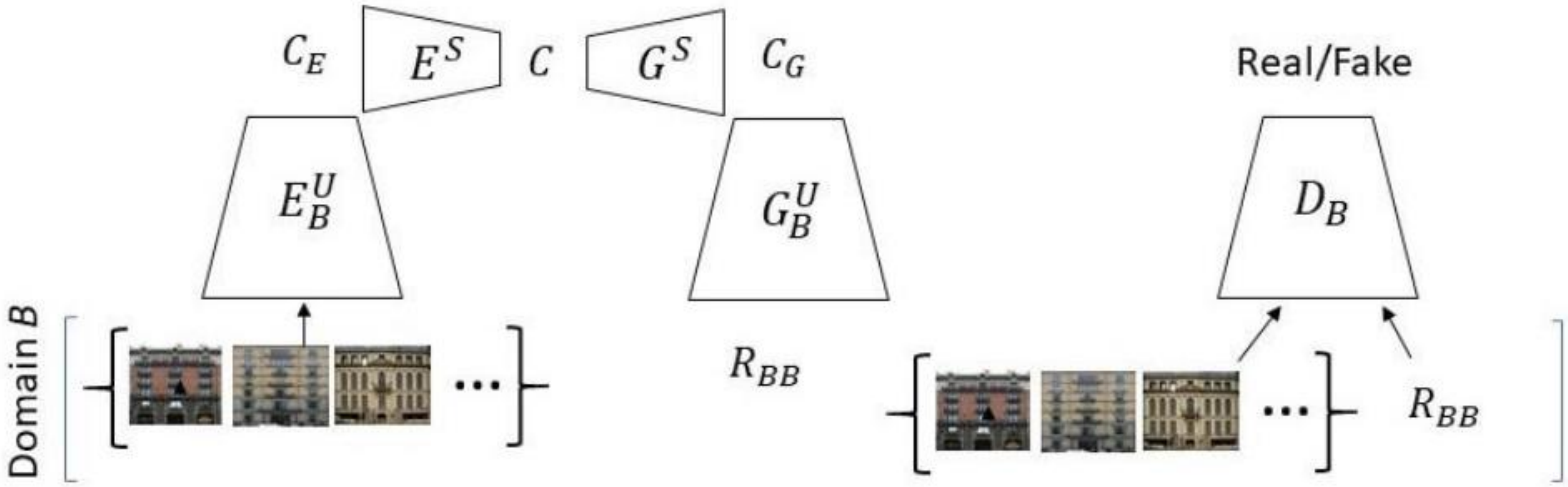
$$\mathcal{L}_{\mathrm{GAN}}(G_{AB}, D_B, \hat{p}_A, \hat{p}_B) = \mathbb{E}_{x_B \sim \hat{p}_B}[\log D_B(x_B)] + \mathbb{E}_{x_A \sim \hat{p}_A}[\log(1 - D_B(G_{AB}(x_A)))]$$

- Other GAN variants can be used: w-gan, improved w-gan, BEGAN, etc.

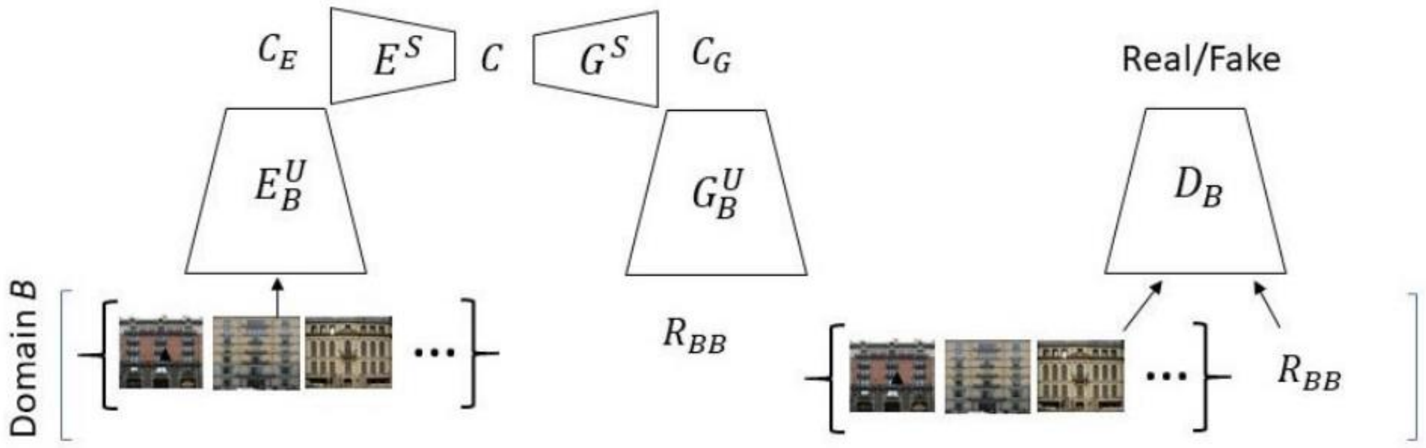
Our Contribution: Only a single image in domain A



Phase I



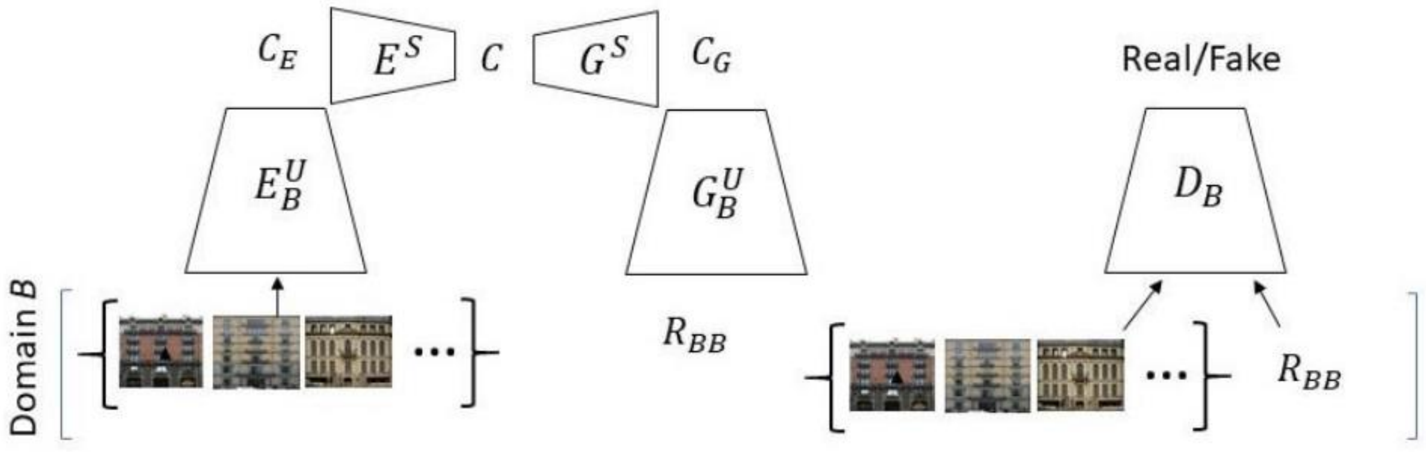
Phase I



$$\mathcal{L}_{REC_B} = \sum_{s \in P(\Lambda)} \|G_B(E_B(s)) - s\|_1$$

$$\mathcal{L}_{VAE_B} = \sum_{s \in P(\Lambda)} \mathrm{KL}(E_B \circ P(\Lambda) \| \mathcal{N}(0, I))$$

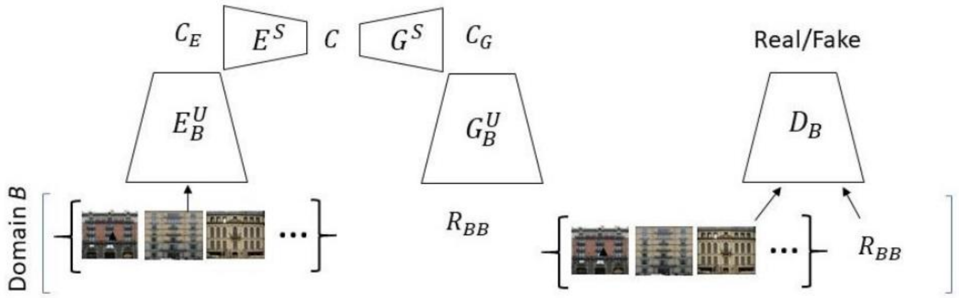
Phase I



$$\mathcal{L}_{\mathrm{GAN}_B} = \sum_{s \in P(\Lambda)} -\ell(\overline{D_B}(G_B(E_B(s))), 0)$$

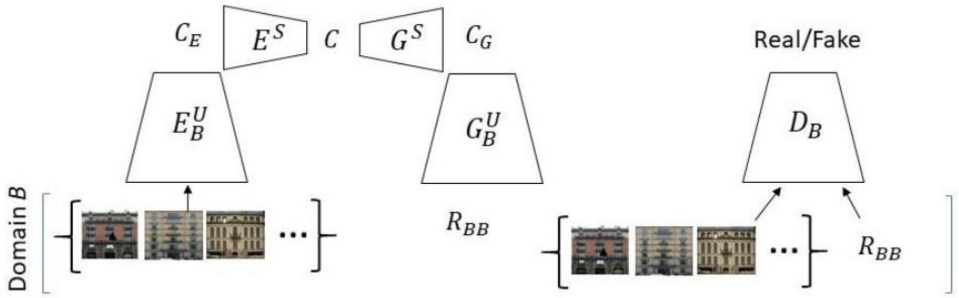
$$\mathcal{L}_{\mathrm{D}_B} = \sum_{s \in P(\Lambda)} +\ell(D_B(\overline{G_B}(\overline{E_B}(s))), 0) + \ell(D_B(s), 1)$$

Phase I



• Shared Latent Space assumption (UNIT Liu et al, CoGAN Liu et al, etc): Upper layers of the encoder and lower layers of the decoder should be shared to achieve successful translation.

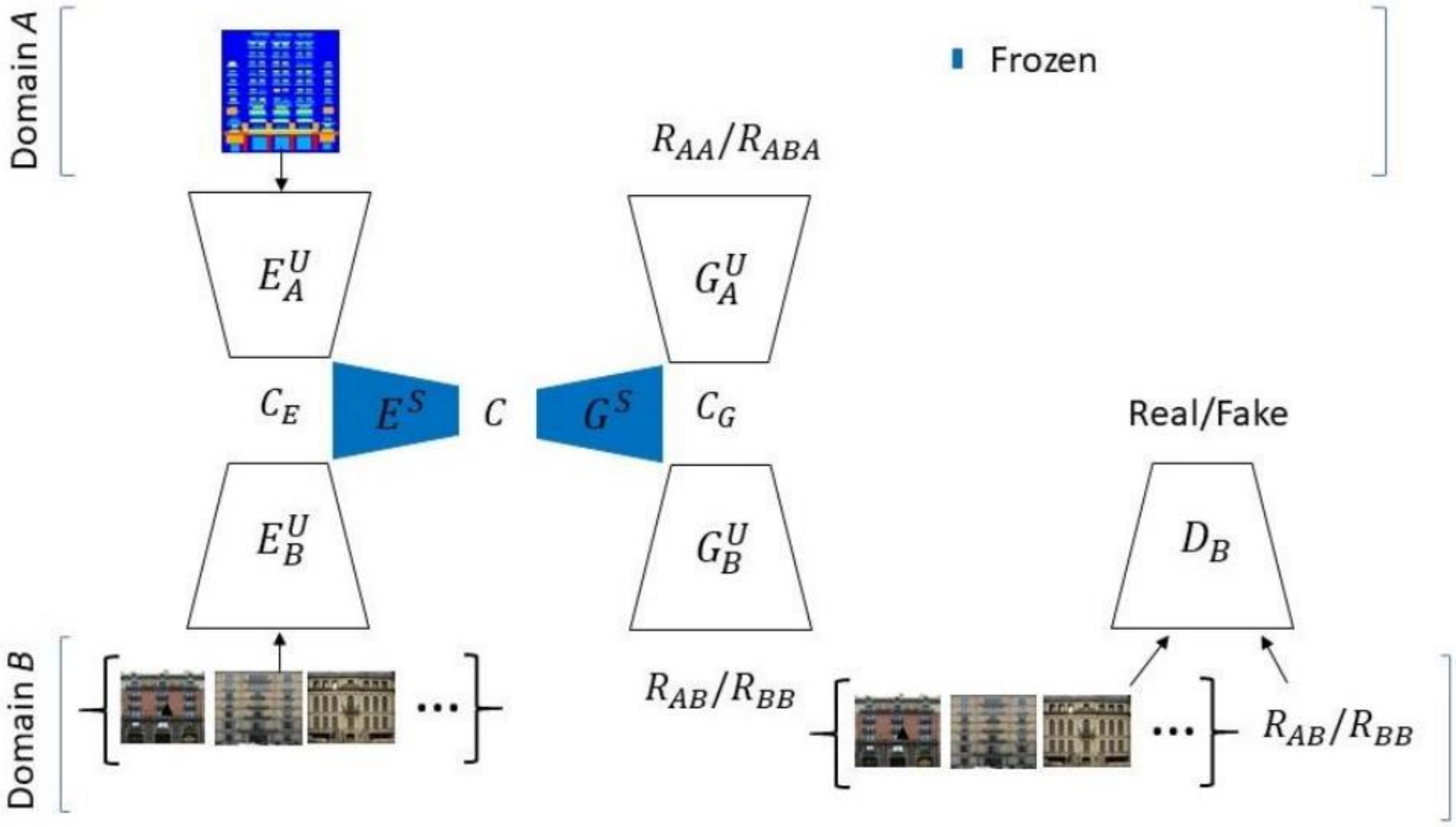
Phase I



- Shared Latent Space assumption (UNIT Liu et al, CoGAN Liu et al, etc): Upper layers of the encoder and lower layers of the decoder should be shared to achieve successful translation.

- In fact, as we only have a single sample in A, these layers, represented by the shared encoder (Es) and shared decoder (Gs) can be trained with domain B samples **only**

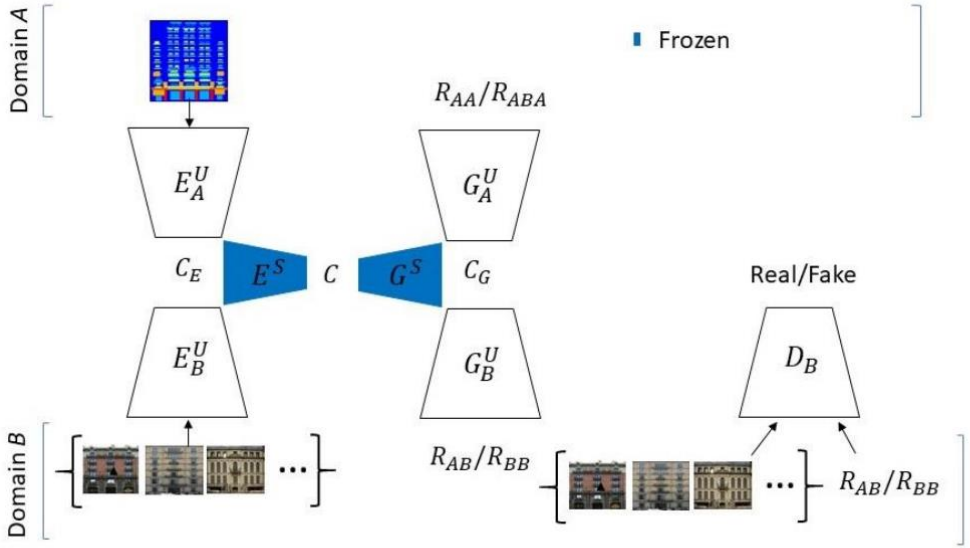
Phase II



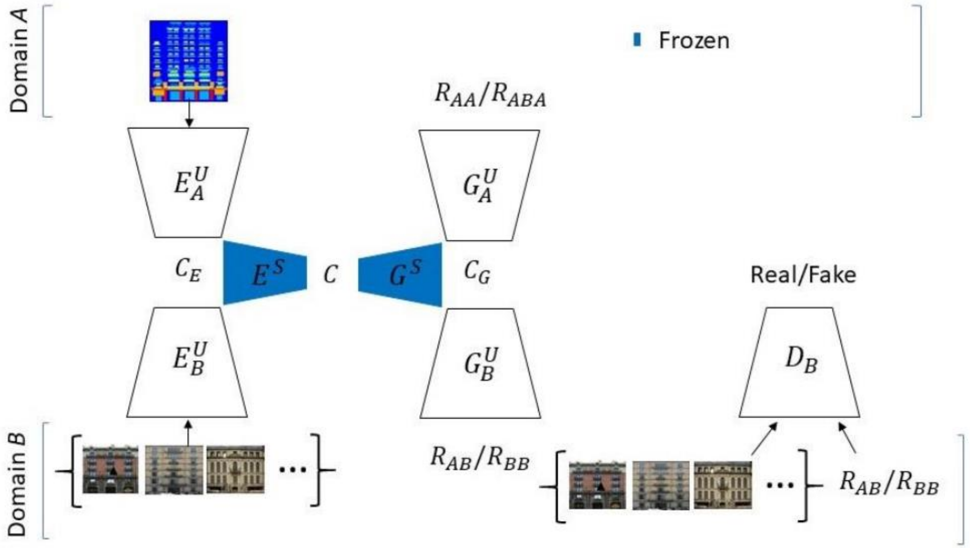
Domain A
Frozen
R_{AA}/R_{ABA}
E_A^U
G_A^U
C_E
E^S
C
G^S
C_G
Real/Fake
E_B^U
G_B^U
D_B
Domain B
R_{AB}/R_{BB}
R_{AB}/R_{BB}

Phase II

1. $$\mathcal{L}_{REC_A} = \sum_{s \in P(x)} \|T_{AA}(s) - s\|_1$$



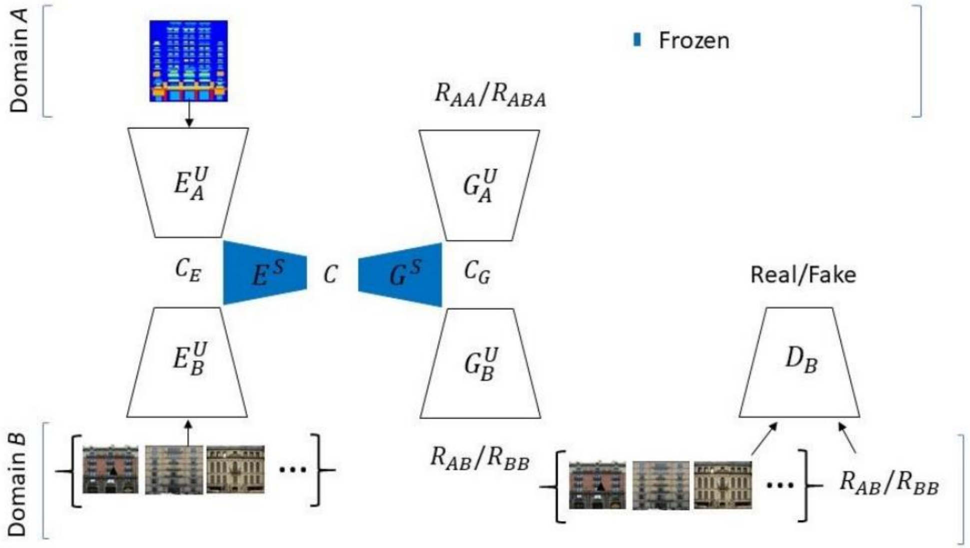
Phase II



1. $$\mathcal{L}_{REC_A} = \sum_{s \in P(x)} \|T_{AA}(s) - s\|_1$$

2. $$\mathcal{L}_{\text{cycle}} = \sum_{s \in P(x)} \|T_{BA}(T_{AB}(s)) - s\|_1$$

Phase II



1. $$\mathcal{L}_{REC_A} = \sum_{s \in P(x)} \|T_{AA}(s) - s\|_1$$

2. $$\mathcal{L}_{\text{cycle}} = \sum_{s \in P(x)} \|T_{BA}(T_{AB}(s)) - s\|_1$$

3. $$\mathcal{L}_{\text{GAN}_{AB}} = \sum_{s \in P(x)} -\ell(\overline{D_B}(T_{AB}(s)), 0)$$

$$\mathcal{L}_{\text{D}_{AB}} = \sum_{s \in P(x)} +\ell(D_B(\overline{T_{AB}}(s)), 0) + \ell(D_B(s), 1)$$

Selective Backpropagation

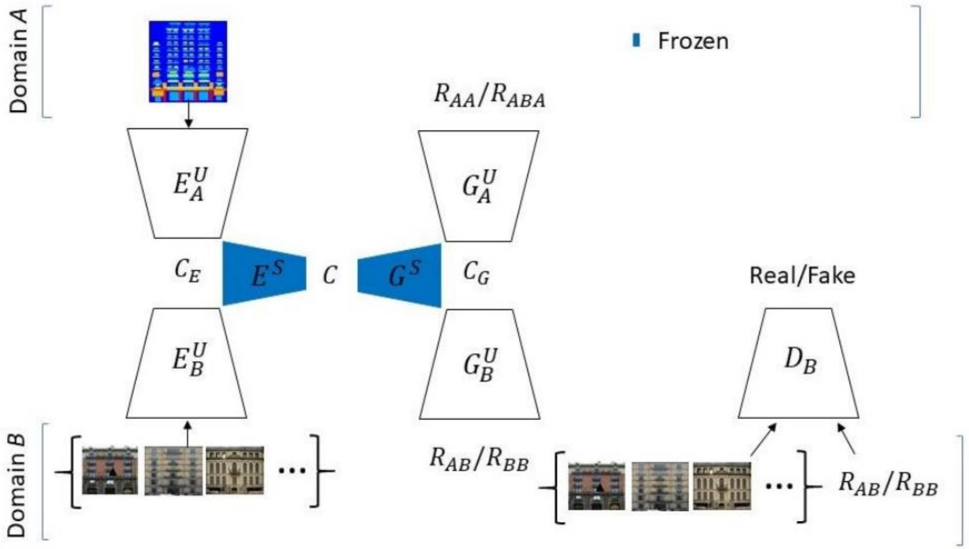
When training our network with x and its augmentations, backpropagation is applied **selectively** on the separate encoders and decoders only.

$$T_{BB} = G^U_B(\overline{G^S}(\overline{E^S}(E^U_B(x))))$$

$$T_{BA} = G^U_A(\overline{G^S}(\overline{E^S}(E^U_B(x))))$$

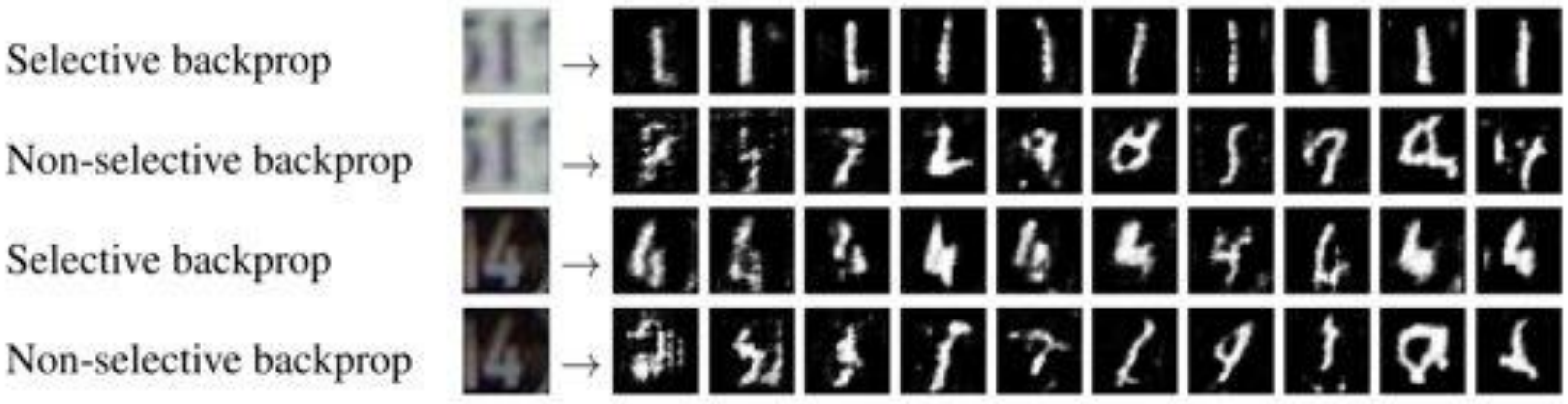
$$T_{AA} = G^U_A(\overline{G^S}(\overline{E^S}(E^U_A(x))))$$

$$T_{AB} = G^U_B(\overline{G^S}(\overline{E^S}(E^U_A(x))))$$



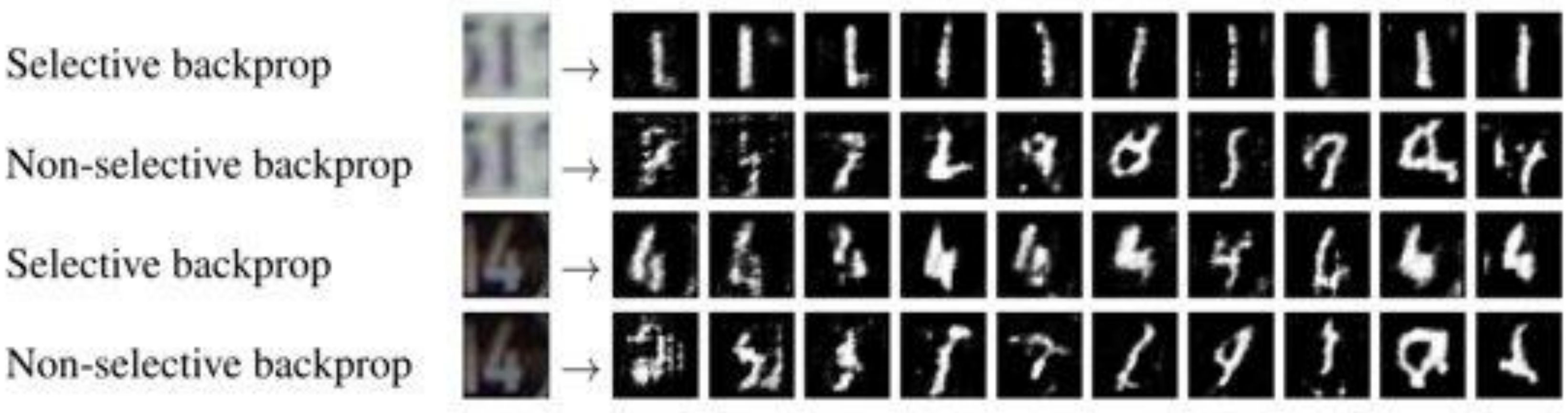
Selective Backpropagation

- Updating the shared encoder (Es) and decoder (Gs) with selective backpropagation turned off leads to **overfitting** on x, since for every shared representation, the unshared layers in domain A can still reconstruct this one sample.



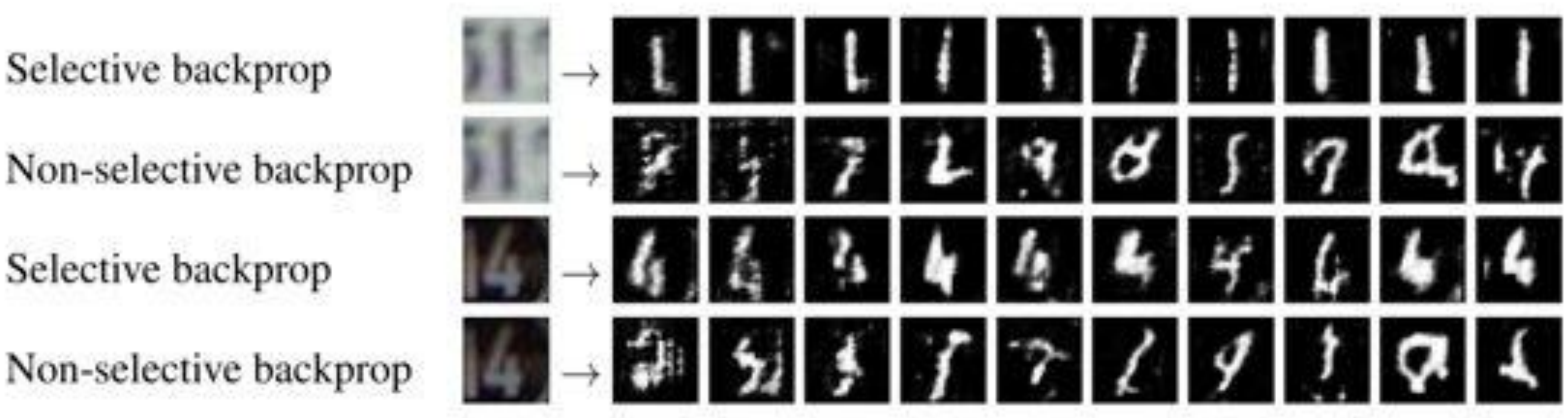
Selective Backpropagation

• Updating the shared encoder (Es) and decoder (Gs) with selective backpropagation turned off leads to **overfitting** on x, since for every shared representation, the unshared layers in domain A can still reconstruct this one sample.
• However, as the shared encoder (Es) and decoder (Gs) can be trained with domain B samples **only**, translation from domain A to B is still possible.

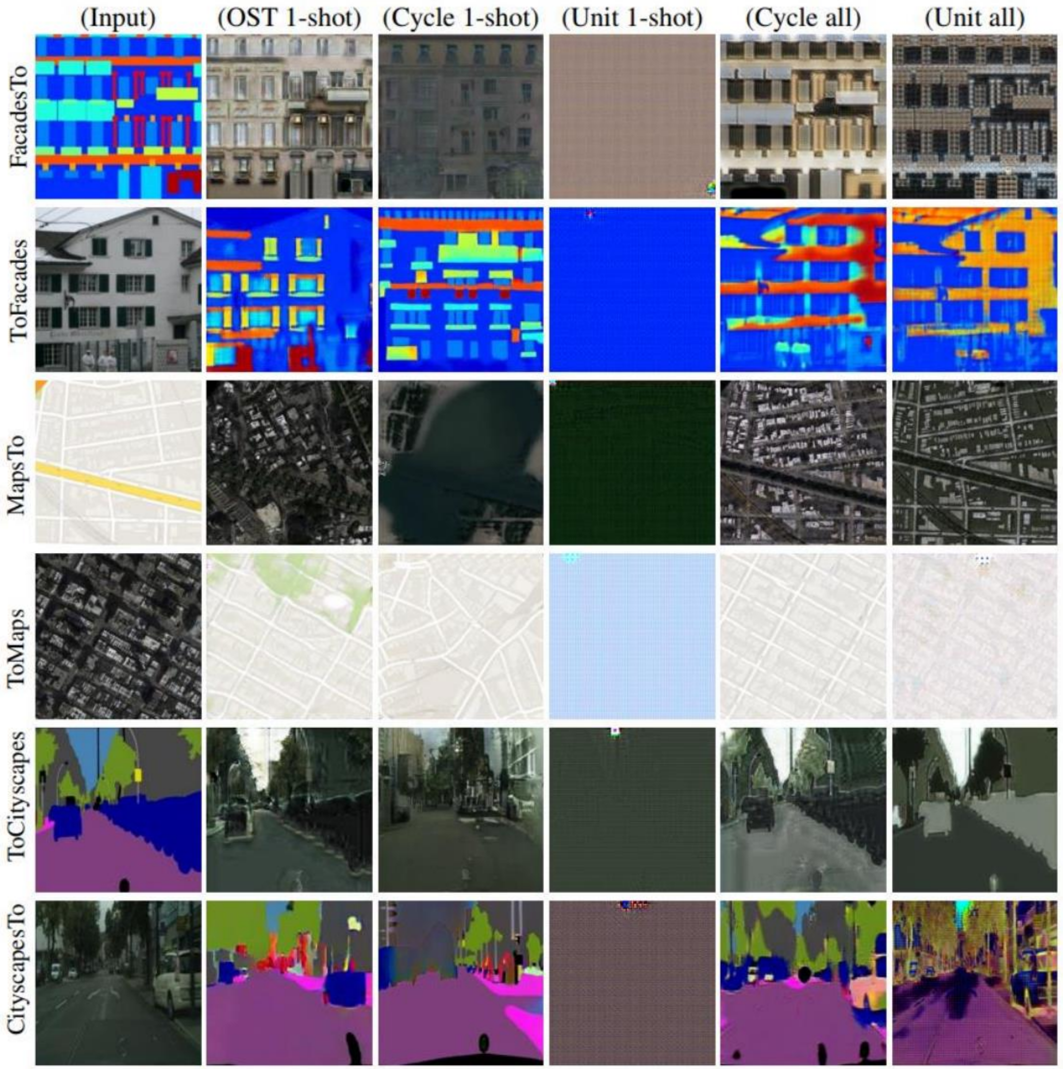


Selective Backpropagation

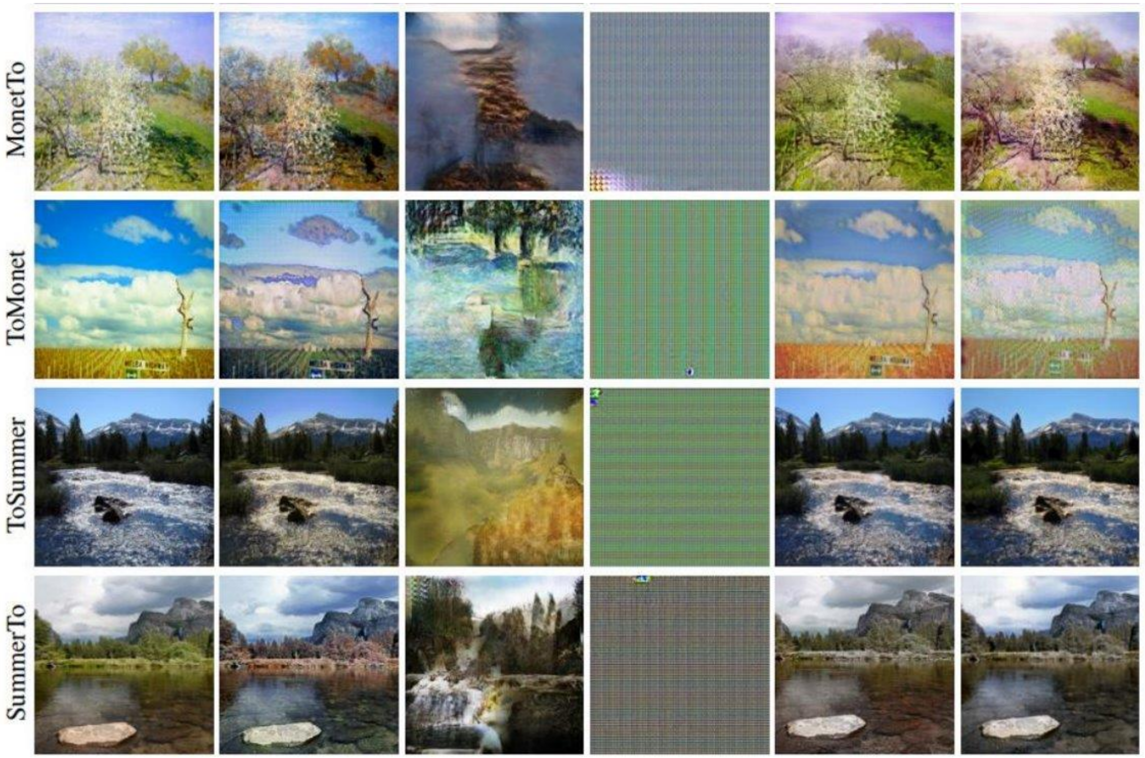
• Updating the shared encoder (Es) and decoder (Gs) with selective backpropagation turned off leads to **overfitting** on x, since for every shared representation, the unshared layers in domain A can still reconstruct this one sample.
• However, as the shared encoder (Es) and decoder (Gs) can be trained with domain B samples **only**, translation from domain A to B is still possible.
•Use of a Patch GAN as well as convolutional layers induces further prior on the network that allows for succesful translation given one input from domain A



Qaulitative Results



Qaulitative Results



Quantitative Results

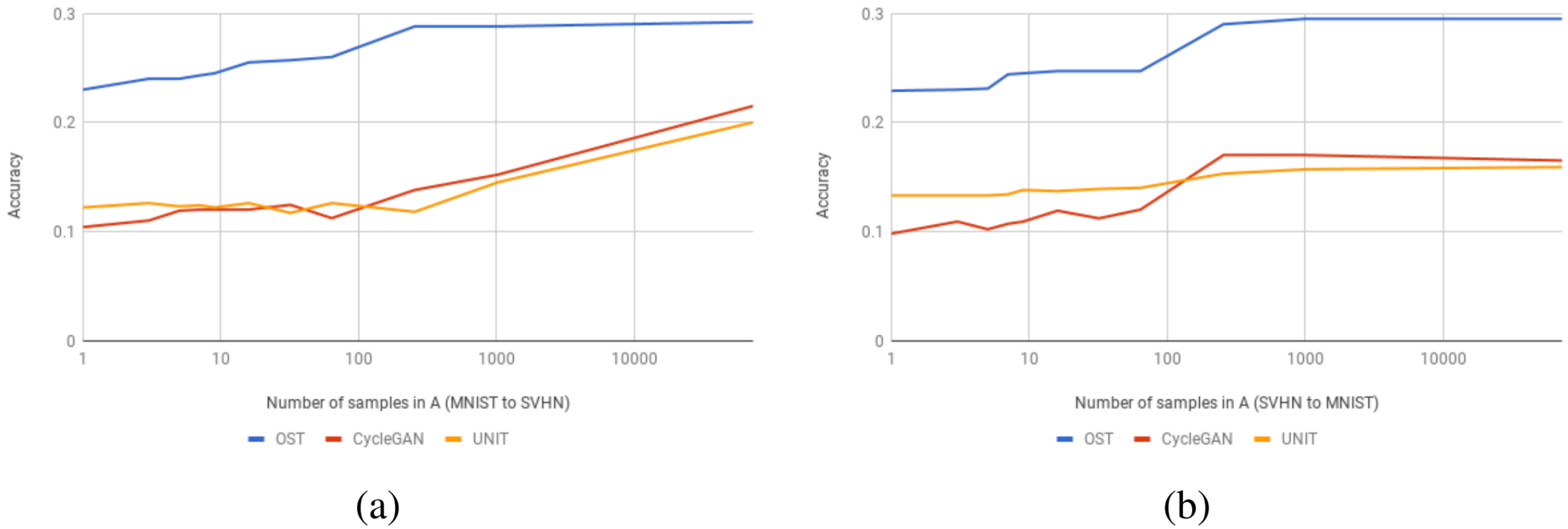


Figure 3: (a) Translating MNIST images to SVHN images. x-axis is the number of samples in A (log-scale), y-axis is the accuracy of a pretrained classifier on the resulting translated images. The accuracy is averaged over 1000 independent runs for different samples. Blue: Our OST method. Yellow: UNIT [7]. Red: CycleGAN [2] . (b) The same graph in the reverse direction.

Quantitative Results

Table 1: Ablation study for the MNIST to SVHN translation (and vice versa). We consider the contribution of various parts of our method on the accuracy. Translation is done for one sample.

Augment-ation	One-way cycle	Selective backprop	Accuracy (MNIST to SVHN)	Accuracy (SVHN to MNIST)
False	False	False	0.07	0.10
True	False	False	0.11	0.11
False	True	False	0.13	0.13
True	True	False	0.14	0.14
False	False	True	0.19	0.20
True	False	True	0.20	0.20
False	True	True	0.22	0.23
True	True	No Phase II update of E^S and G^S	0.16	0.15
True	Two-way cycle	True	0.20	0.13
True	Two-way cycle	False	0.11	0.12
True	True	True	**0.23**	**0.23**

Quantitative Results

Table 2: (i) Measuring the perceptual distance [29], between inputs and their corresponding output images of different style transfer tasks. Low perceptual loss indicates that much of the high-level content is preserved in the translation. (ii) Measuring the style difference between translated images and images from the target domain. We compute the average Gram matrix of translated images and images from the target domain and find the average distance between them, as described in [29].

Component	Dataset Samples in A	OST 1	UNIT [7] 1	CycleGAN [2] 1	UNIT [7] All	CycleGAN [2] All
(i) Content	Summer2Winter	0.64	3.20	3.53	1.41	0.41
	Winter2Summer	0.73	3.10	3.48	1.38	0.40
	Monet2Photo	3.75	6.82	5.80	1.46	1.41
	Photo2Monet	1.47	2.92	2.98	2.01	1.46
(ii) Style	Summer2Winter	1.64	6.51	1.62	1.69	1.69
	Winter2Summer	1.58	6.80	1.31	1.69	1.66
	Monet2Photo	1.20	6.83	0.90	1.21	1.18
	Photo2Monet	1.95	7.53	1.91	2.12	1.88

Quantitative Results

Table 3: (i) Perceptual distance [29] between the inputs and corresponding output images, for various drawing tasks. (ii) Style difference between translated images and images from the target domain. (iii) Correctness of translation as evaluated by a user study.

	Method	Images to Facades	Facades to Images	Images To Maps	Maps to Images	Labels to Cityscapes	Cityscapes to Labels
(i)	OST 1	4.76	5.05	2.49	2.36	3.34	2.39
	UNIT [7] All	3.85	4.80	2.42	2.30	2.61	2.18
	CycleGAN [2] All	3.79	4.49	2.49	2.11	2.73	2.28
(ii)	OST 1	3.57	7.88	2.24	1.50	0.67	1.13
	UNIT [7] All	3.92	7.42	2.56	1.59	0.69	1.21
	CycleGAN [2] All	3.81	7.03	2.33	1.30	0.77	1.22
(iii)	OST 1	91%	90%	83%	67%	66%	56%
	UNIT [7] ALL	86%	83%	81%	75%	63%	37%
	CycleGAN [2] ALL	93%	84%	97%	81%	72%	45%

Future reseach

- One Shot Domain Adaptation
- One Shot Image to Image translation in the reverse direction
- Other Domains: Audio, Video?
- Online Setting

Thank You! Questions?

Minimality

- Potentially Infinitely many solutions preserving distance correlations

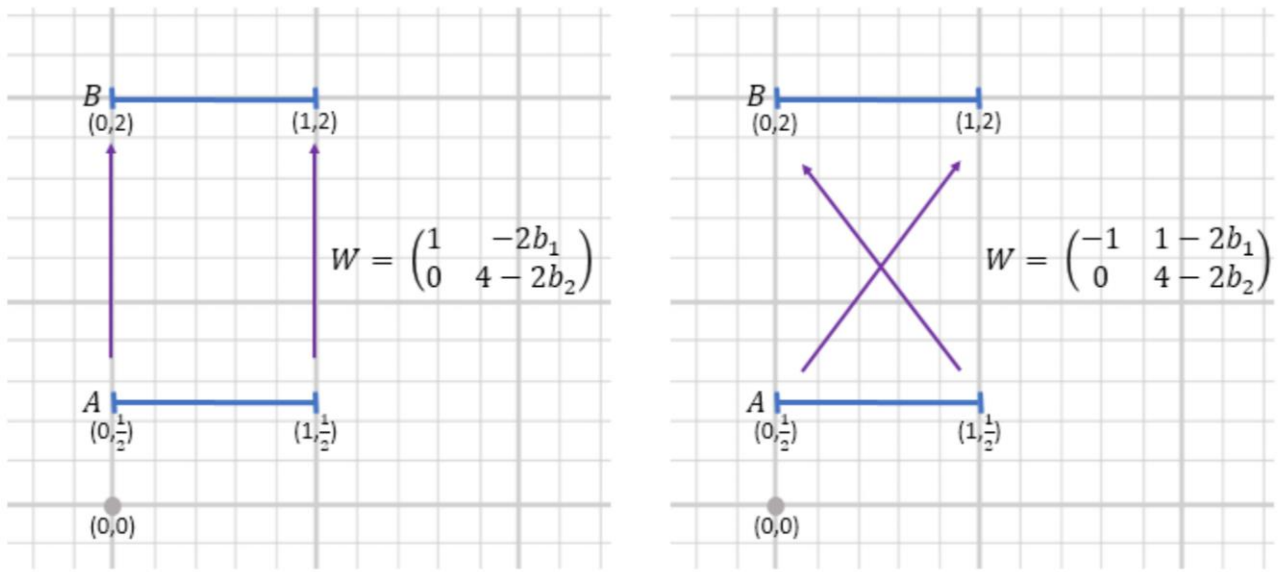


Figure 1: An illustrative example where the two domains are line segments in $\mathbb{R}^2$. There are infinitely many mappings that preserve the uniform distribution on the two segments. However, only two stand out as "semantic". These are exactly the two mappings that can be captured by a neural network with only two hidden neurons and Leaky ReLU activations, i.e., by a function $h(x) = \sigma_a(Wx + b)$, for a weight matrix W and the bias vector b.