On disentangled and few shot visual generation and understanding

Sagie Benaim
School of Computer Science, Tel Aviv University

TEL AVIV UNIVERSITY

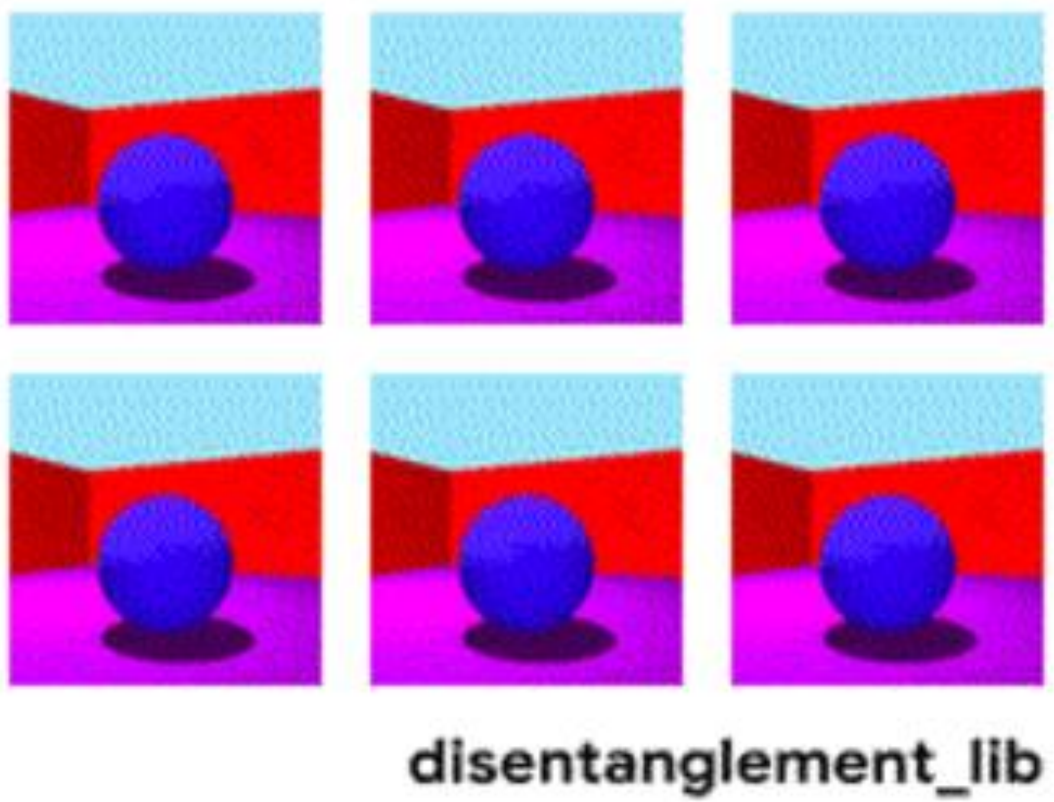
Outline

- Part I: Disentangled representation and generation
  - Semi-supervised setting
  - Unsupervised setting
- Part II: Few shot generation
  - Image to Image translation
  - Video generation

What is a ‘disentangled representation’?

- A real world **high-dimensional** observation x (image or video) can be represented by a **low-dimensional** latent variable z.
- z corresponds **to semantically meaningful factors** of variation of x such as: content, pose, style, etc.
- A change in a single factor of z should correspond to a change in a single underlying factor of variation of x.

What is a ‘disentangled representation’?



https://ai.googleblog.com/2019/04/evaluating-unsupervised-learning-of.html

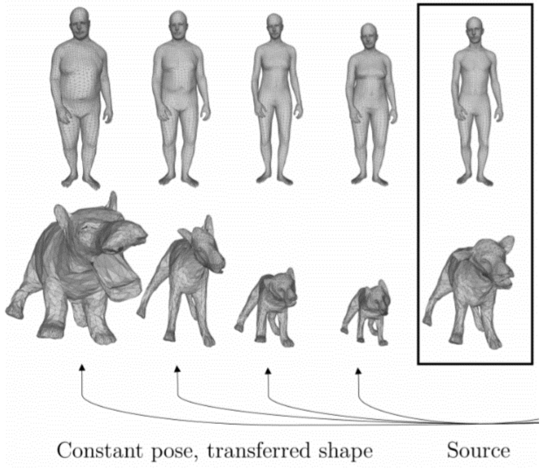
Why do we need it?

Style Transfer[1]

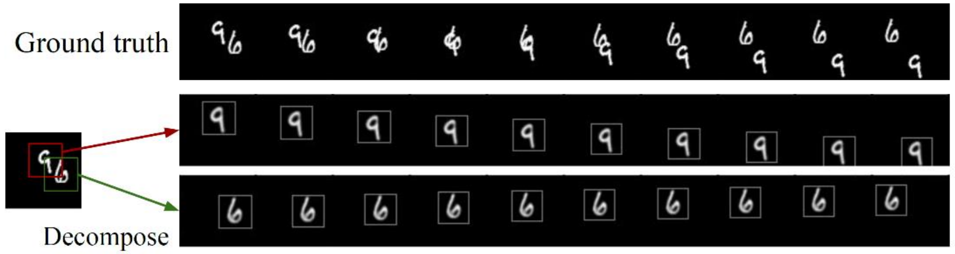
Content Transfer[2]

Pose Transfer[3]

Shape Transfer[4]



Video Prediction[5]



1. Huang et al., 2. Benaim et al., 3. Ren et al., 4. Zhou et al., 5. Hsieh et al.,

Disentanglement: Supervision Level

- <u>Fully Supervised:</u> Each image in the dataset appears with or without each factor of variation.
- <u>Semi-Supervised (Set Level):</u> Each set of images (which may be different), appear with or without each factor of variation.
- <u>Unsupervised:</u> Strong assumptions about data-set which are incorporated into the model design.

Disentanglement: Supervision Level

- <u>Fully Supervised:</u> Each image in the dataset appears with or without each factor of variation.
- <u>Semi-Supervised (Set Level):</u> Each set of images (which may be different), appear with or without each factor of variation.
- <u>Unsupervised:</u> Strong assumptions about data-set which are incorporated into the model design.

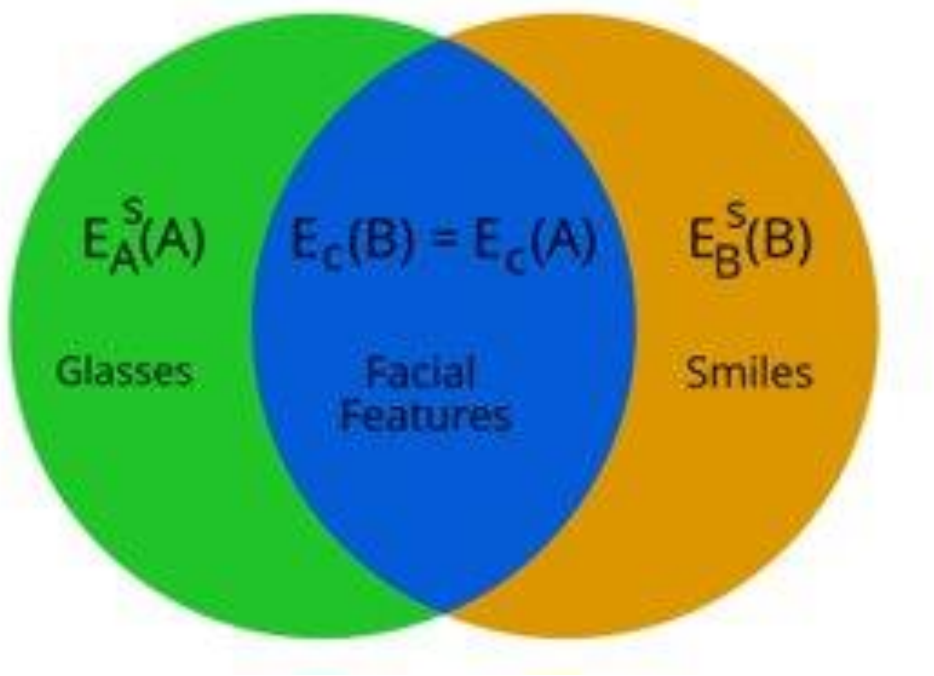
Domain Intersection and Domain Difference

S. Benaim, M. Khaitov, T. Galanti, L. Wolf. ICCV 2019.

Given two visual domains, disentangle the **separate (domain specific)** information and **common (domain invariant)** information.

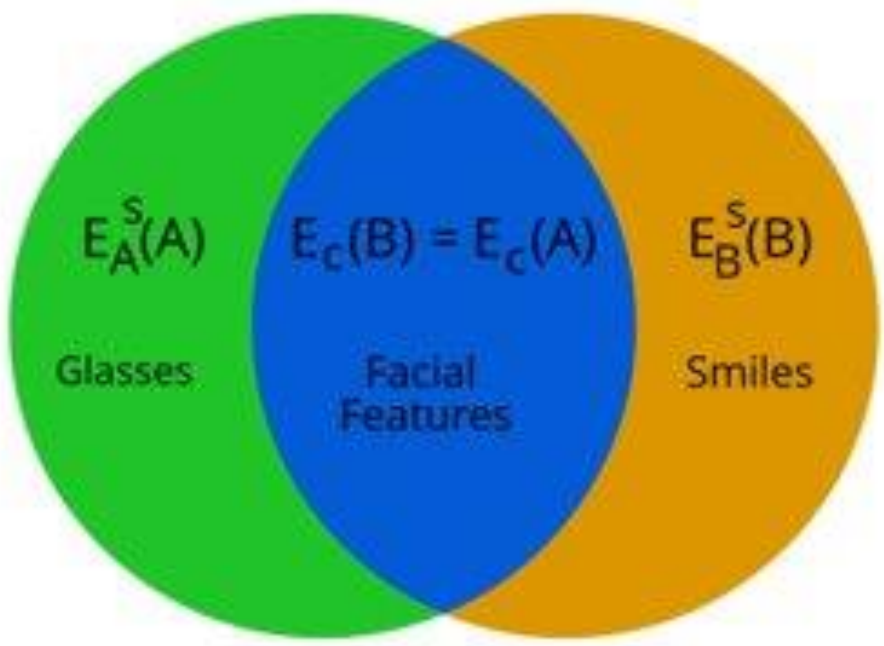
If A is **persons with glasses** and B is **smiling persons,** our method produces three latent spaces:

1. "Common" latent space, $E_c(A) = E_c(B)$. The space of **common facial features.** For $c \in A \cup B, E_c(c)$ is the **facial features of c**.



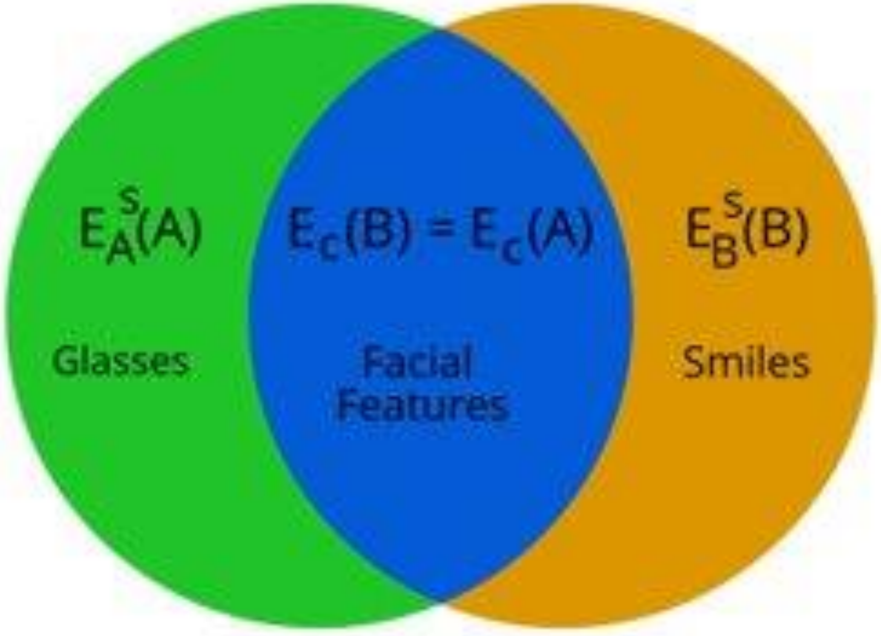
If A is **persons with glasses** and B is **smiling persons,** our method produces three latent spaces:

1. "Common" latent space, $E_c(A) = E_c(B)$. The space of **common facial features.** For $c \in A \cup B, E_c(c)$ is the **facial features of c**.
2. "Separate" latent space for domain A, $E_A^s(A)$. The **space of glasses.** $E_A^s(a)$ is the **glasses of $\boldsymbol{a}$**.

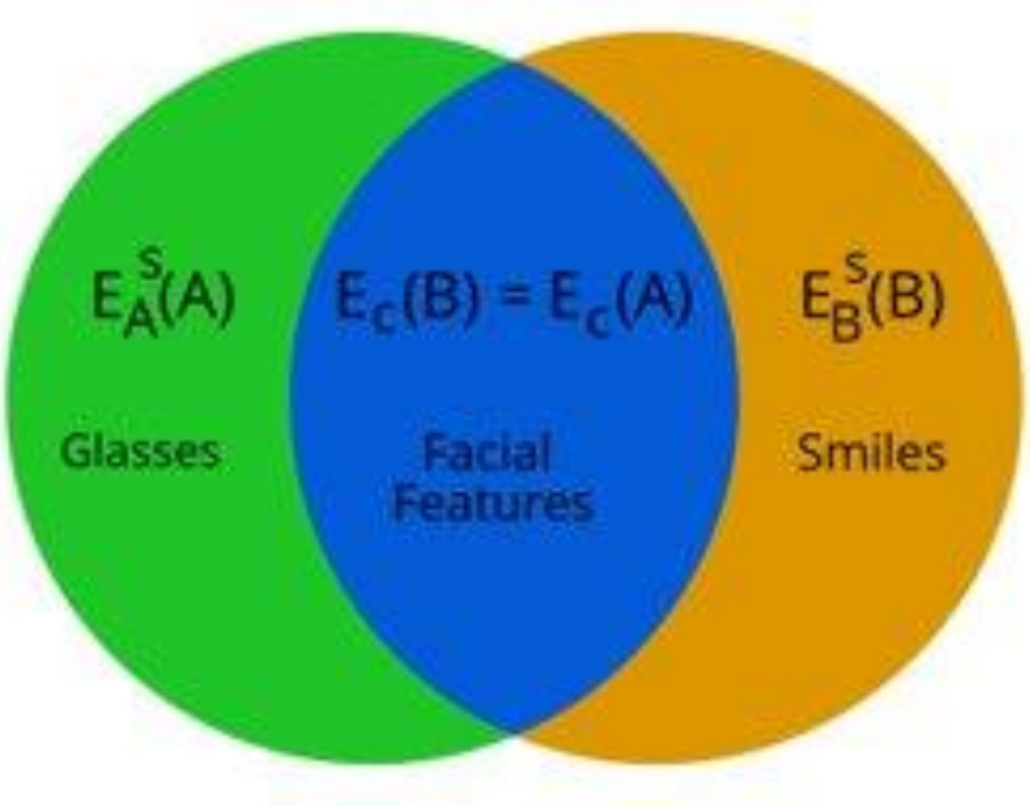


If A is **persons with glasses** and B is **smiling persons,** our method produces three latent spaces:

1. "Common" latent space, $E_c(A) = E_c(B)$. The space of **common facial features.** For $c \in A \cup B, E_c(c)$ is the **facial features of c**.
2. "Separate" latent space for domain A, $E_A^s(A)$. The **space of glasses.** $E_A^s(a)$ is the **glasses of $\boldsymbol{a}$**.
3. "Separate" latent space for domain B, $E_B^s(B)$. The **space of smiles.** $E_B^s(b)$ is the **smile of $\boldsymbol{b}$.**



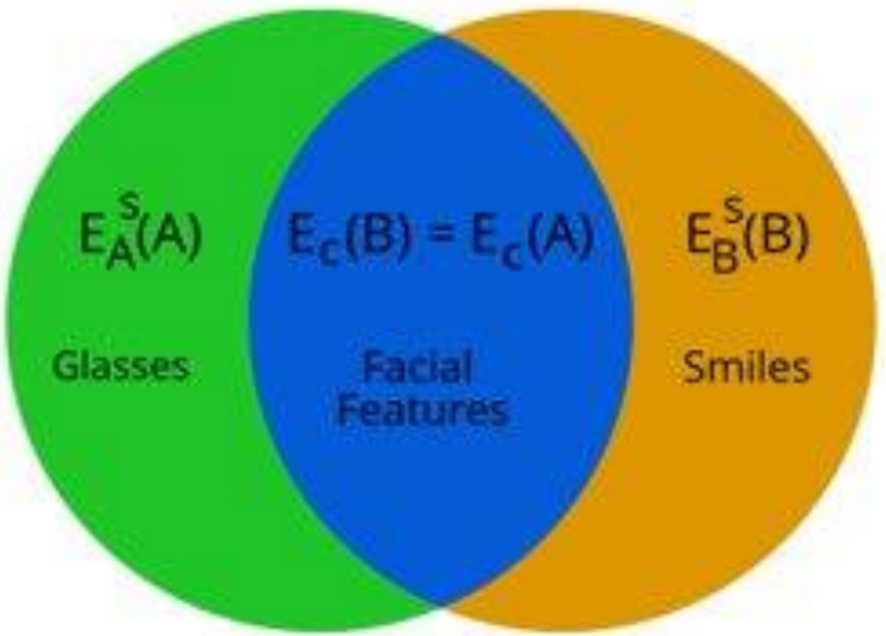
Given this disentangled representation, we generate a visual sample $G(E_c(c), E_A^s(a), E_B^s(b))$, having the **facial features of c, glasses of a, smile of b.**

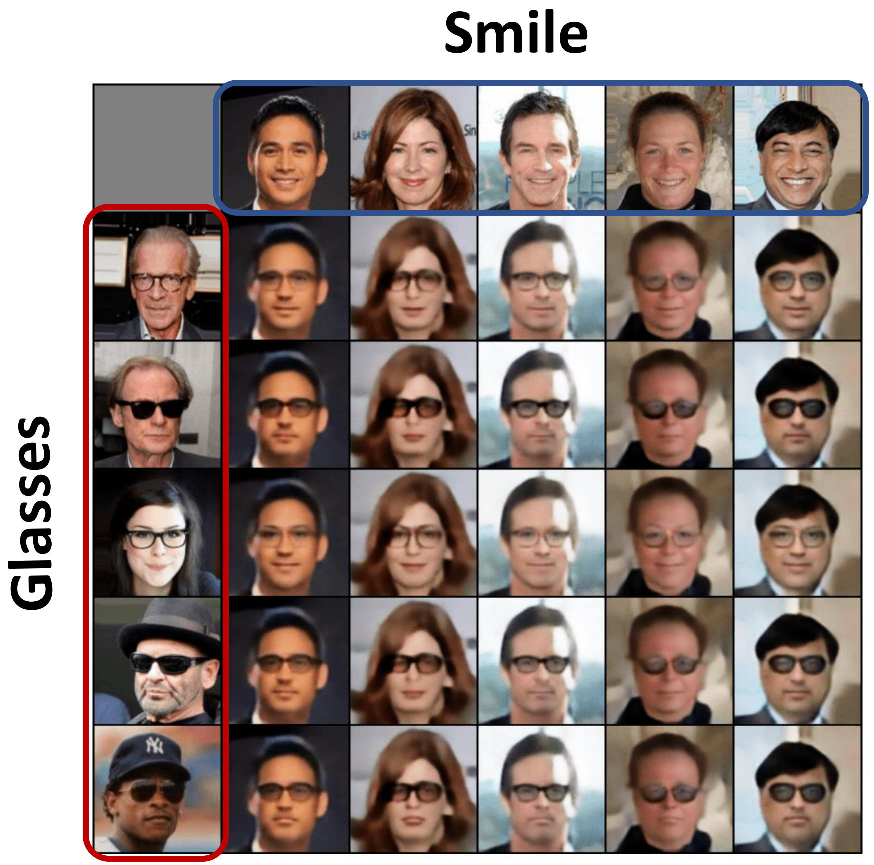


$$G(E_c(b), E_A^s(a), 0)$$

remove b's smile

add a's glasses





The "common" (or shared) Loss

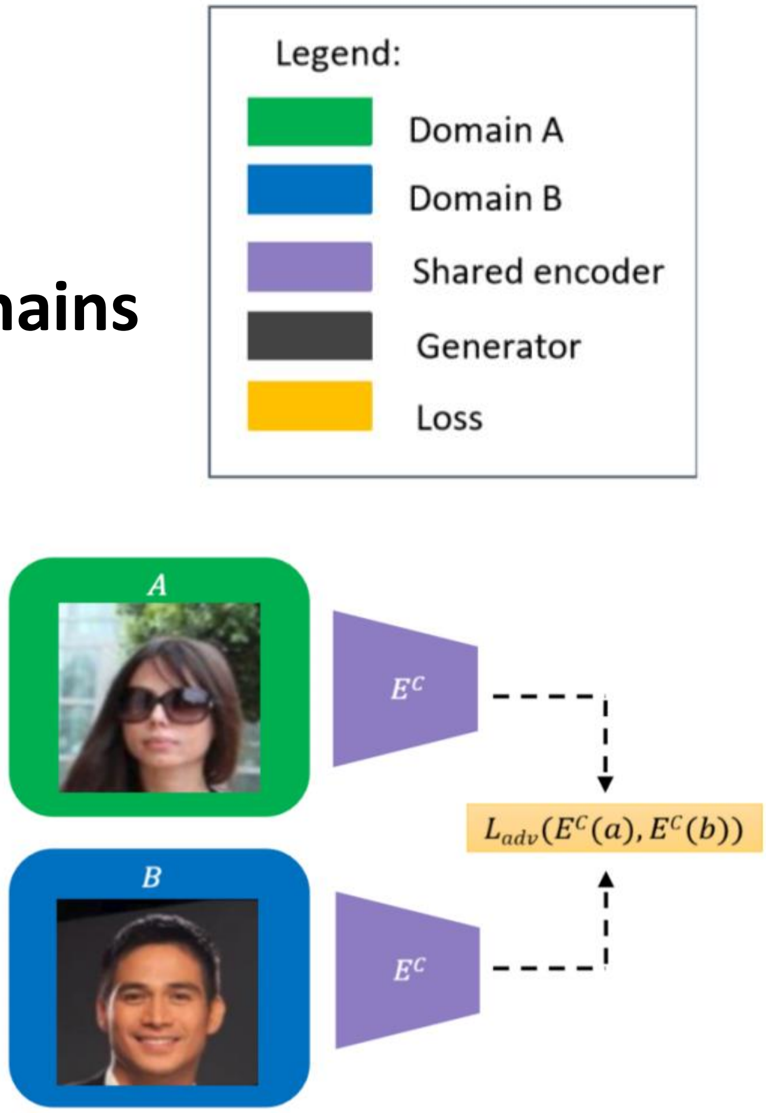
Ensures E_c encodes information common to both domains

Encoder E_c attempts to match distributions of $E_c(A)$ and $E_c(B)$:

$$\frac{1}{m_1}\sum_{i=1}^{m_1} l(d(E^c(a_i)),1) + \frac{1}{m_2}\sum_{j=1}^{m_2} l(d(E^c(b_j)),1)$$

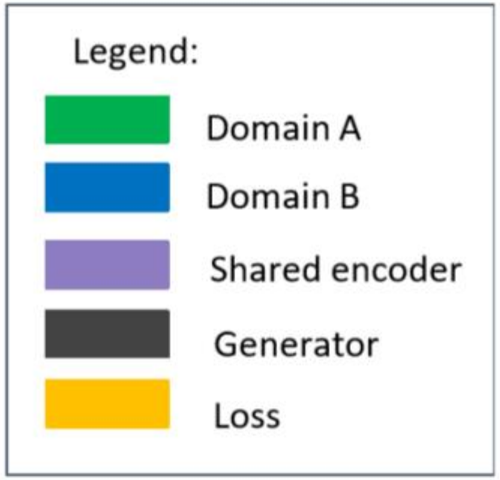
Discriminator d attempts to separate distributions:

$$\mathcal{L}_d := \frac{1}{m_1}\sum_{i=1}^{m_1} l(d(E^c(a_i)),0) + \frac{1}{m_2}\sum_{j=1}^{m_2} l(d(E^c(b_j)),1)$$

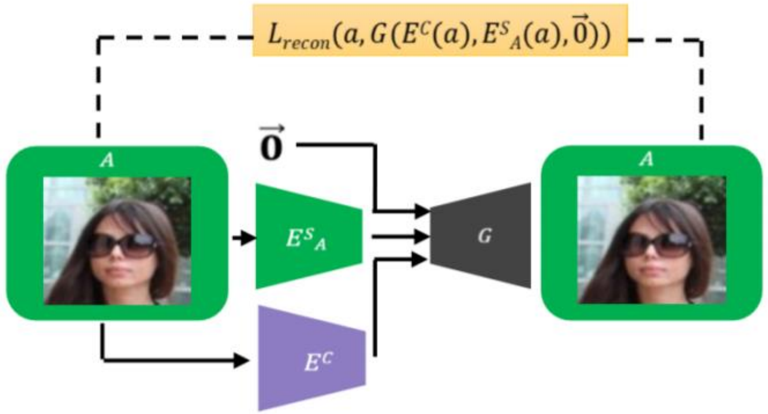


(c)

Reconstruction Losses



Ensures the “common” and “separate” encodings contain all the information in A



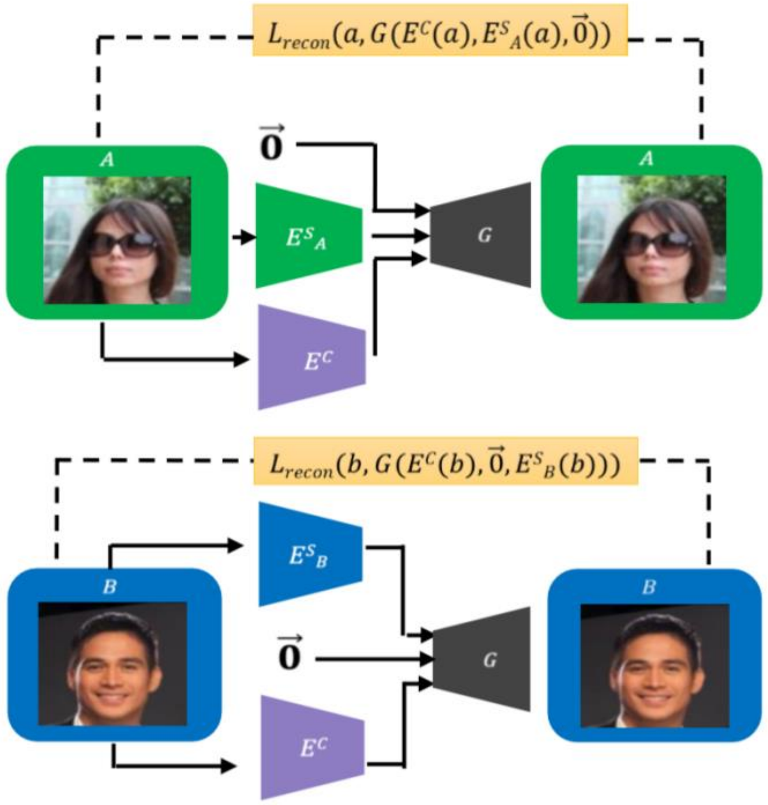
$$\mathcal{L}_{recon}^{A} := \frac{1}{m_1}\sum_{i=1}^{m_1} \|G(E^c(a_i), E_A^s(a_i), 0) - a_i\|_1$$

Reconstruction Losses

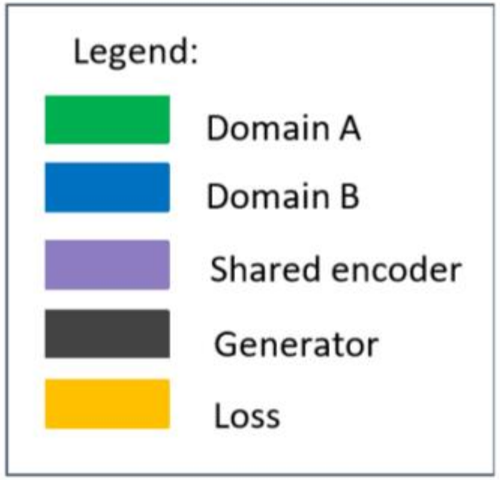
Ensures the "common" and "separate" encodings contain all the information in A or B

$$\mathcal{L}^{A}_{recon} := \frac{1}{m_1}\sum_{i=1}^{m_1} \|G(E^c(a_i), E^s_A(a_i), 0) - a_i\|_1$$

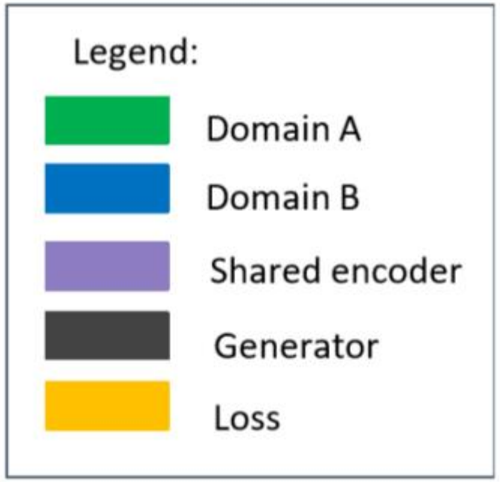
$$\mathcal{L}^{B}_{recon} := \frac{1}{m_2}\sum_{j=1}^{m_2} \|G(E^c(b_i), 0, E^s_B(b_j)) - b_j\|_1$$



(b)

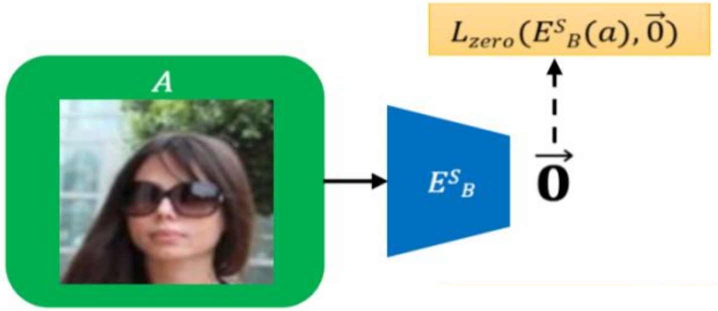


"Zero" Loss



Ensures the separate encoder of B does not encode information about A

$$\mathcal{L}^{A}_{zero} := \frac{1}{m_2}\sum_{j=1}^{m_2}\|E^{s}_{A}(b_j)\|_1$$

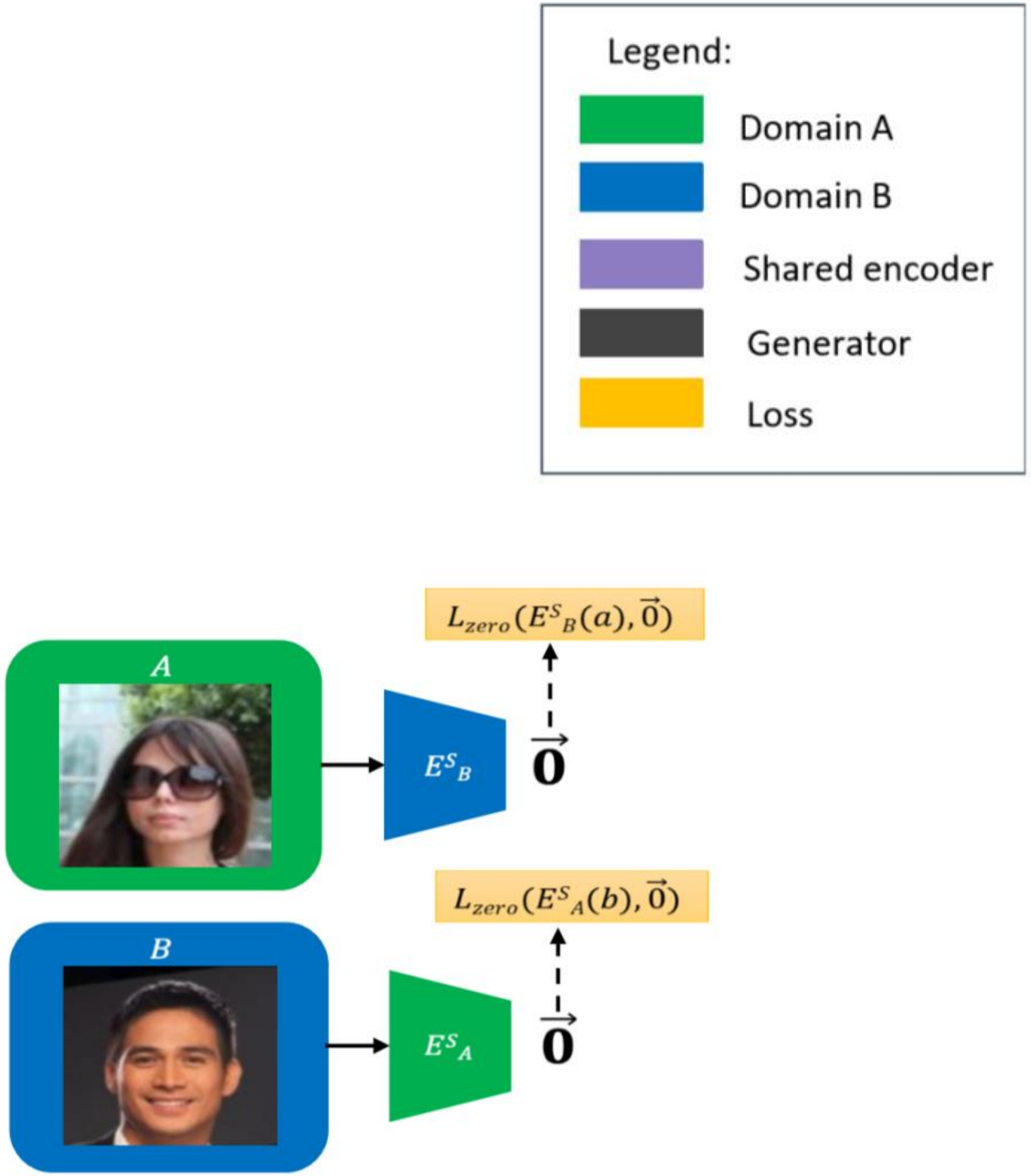


"Zero" Loss

Ensures the separate encoder of B (resp. A) does not encode information about A (resp. B)

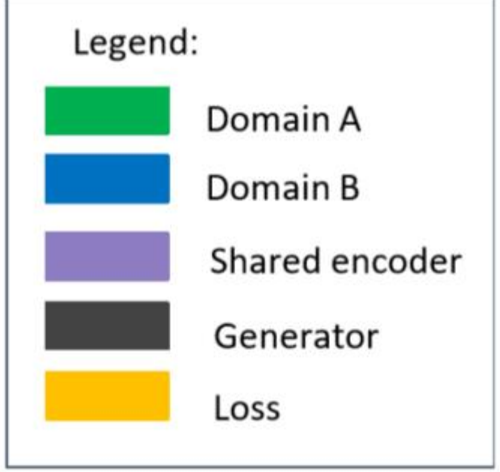
$$\mathcal{L}^A_{zero} := \frac{1}{m_2}\sum_{j=1}^{m_2}\|E^s_A(b_j)\|_1$$

$$\mathcal{L}^B_{zero} := \frac{1}{m_1}\sum_{i=1}^{m_1}\|E^s_B(a_i)\|_1$$



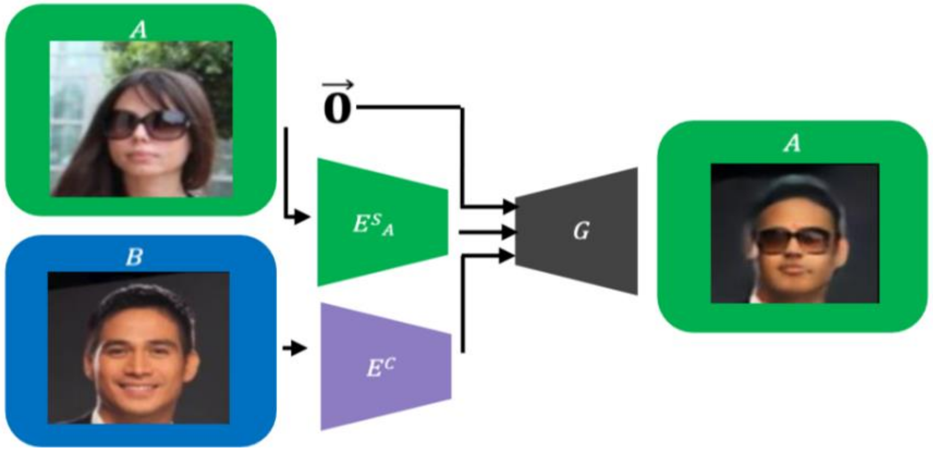
(a)

Inference:

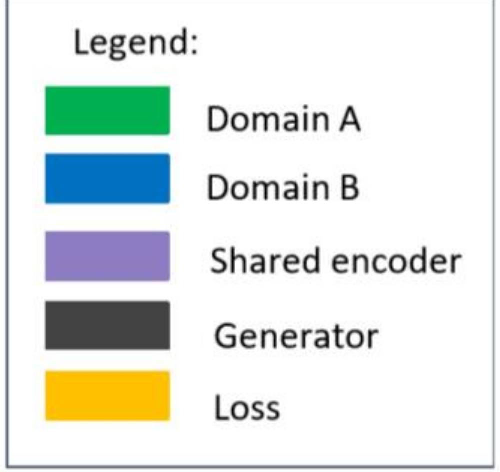


$$G(E_c(b), E_A^S(a), 0)$$

remove b's smile
add a's glasses



Inference:

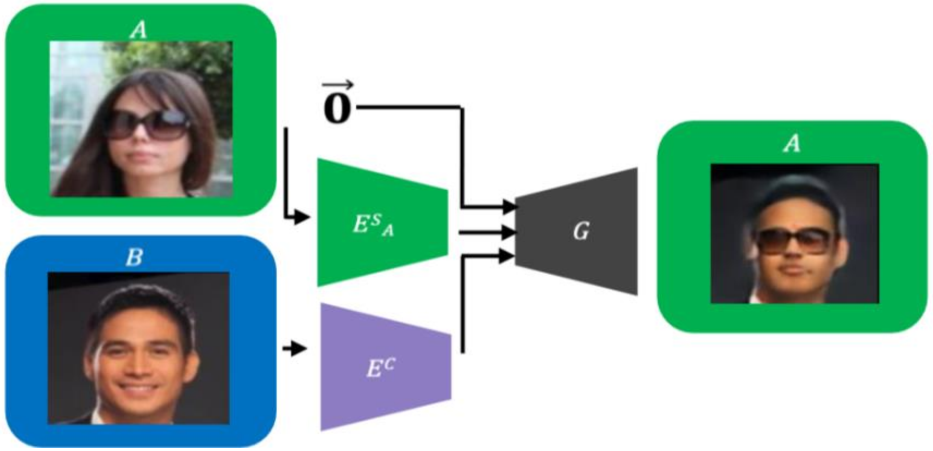


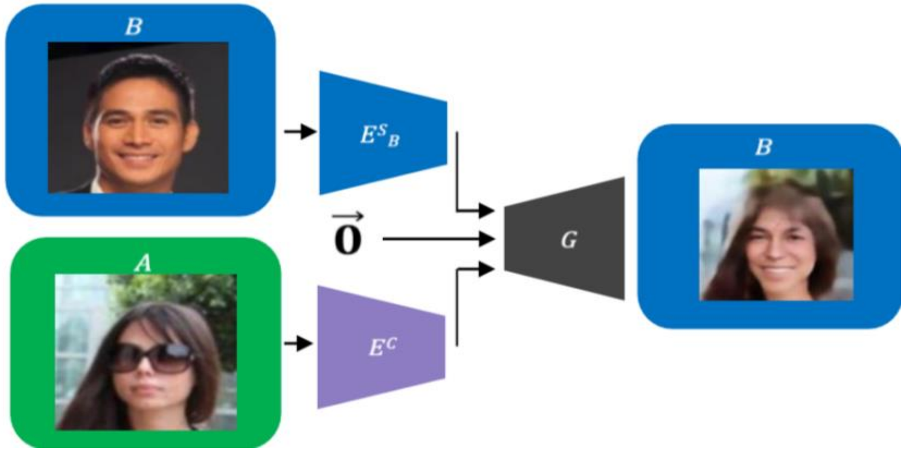
$$G(E_c(b), E_A^S(a), 0)$$

remove b's smile
add a's glasses

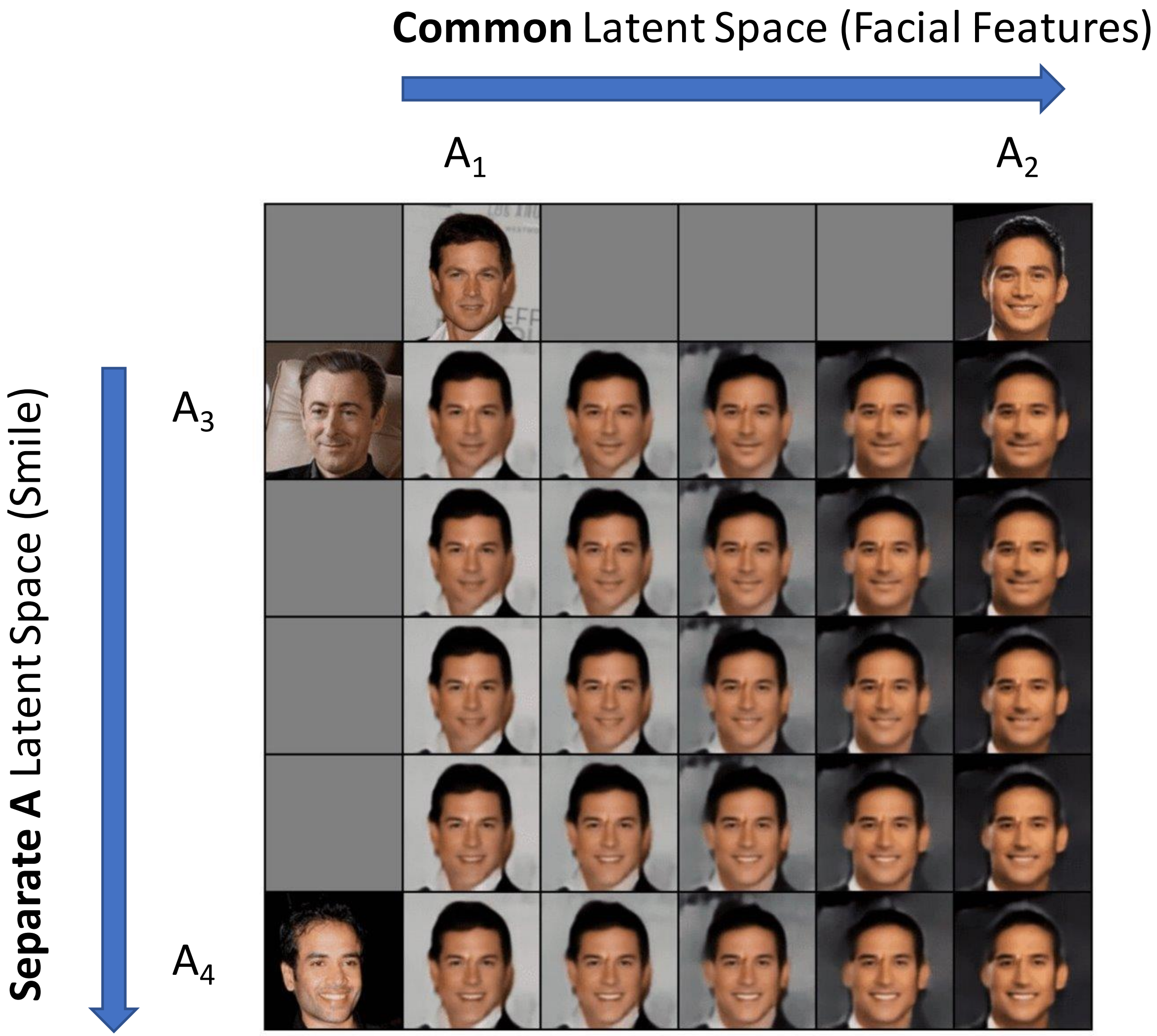
$$G(E_c(a), 0, E_A^S(b))$$

remove a's glasses
add b's smile

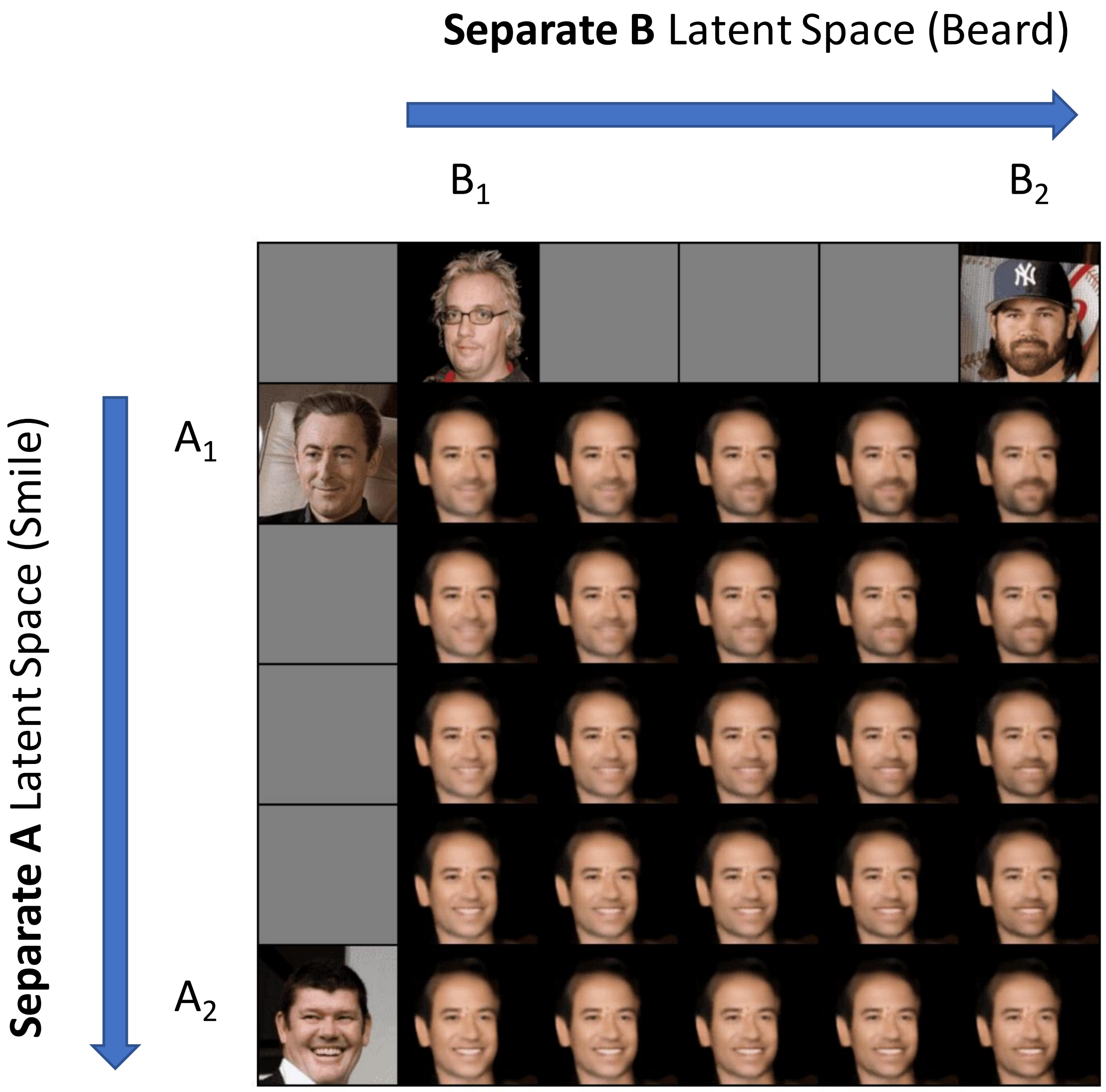




Interpolations



Interpolations



Losses "Necessary" and "Sufficient"

- Under mild assumptions (such as our losses being minimized):
  - $E^c(A)$ and $E^s_A(A)$ are independent (Similarly for B).
  - $E^c(A)$ captures the information underlying $e^c(A)$ (Similarly for B).
  - $E^s_A(A)$ holds the information underlying $e^s_A(A)$ (Similarly for B).
  - I.e. our losses are both **necessary and sufficient** for the desired **disentanglement**.

Masked Based Unsupervised Content Transfer

R. Mokady, S. Benaim, L. Wolf, A. Bermano. ICLR 2020.

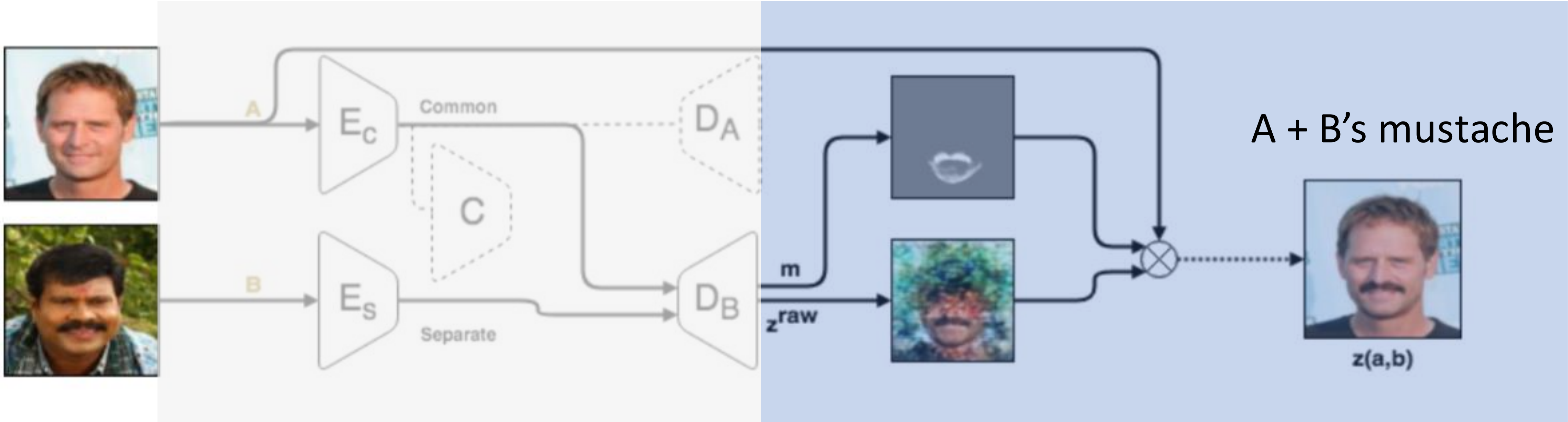
- Only a local change in the target is needed
- Learn a mask and adapt only the area in the masked area

Attention-based Masked Generation

A

B

A + B's mustache



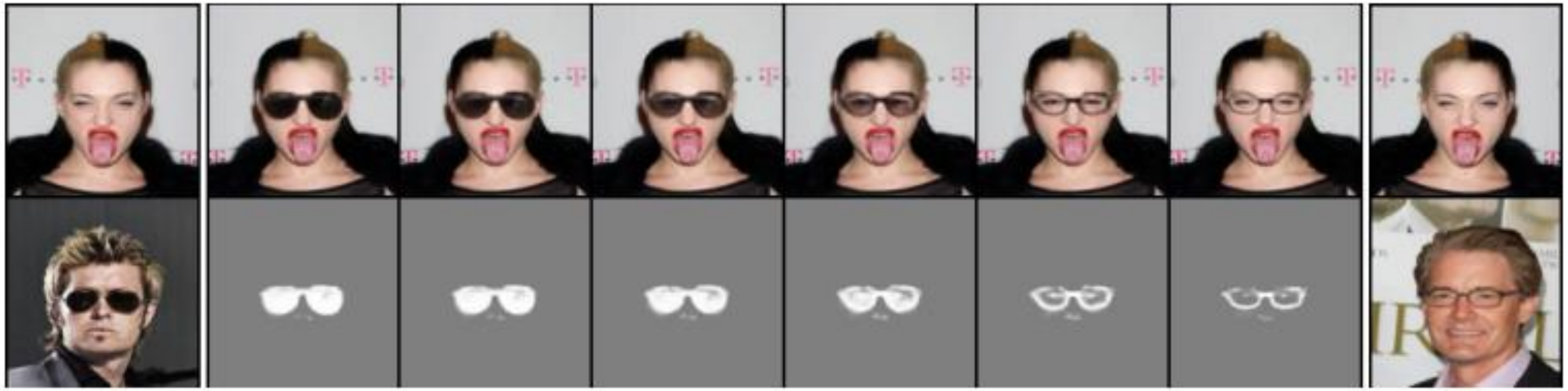
Two Attributes

Two Attributes

Smile to Glasses

Additional Content Transfer

Interpolation



Attribute Removal

Facial Hair

Smile

Glasses

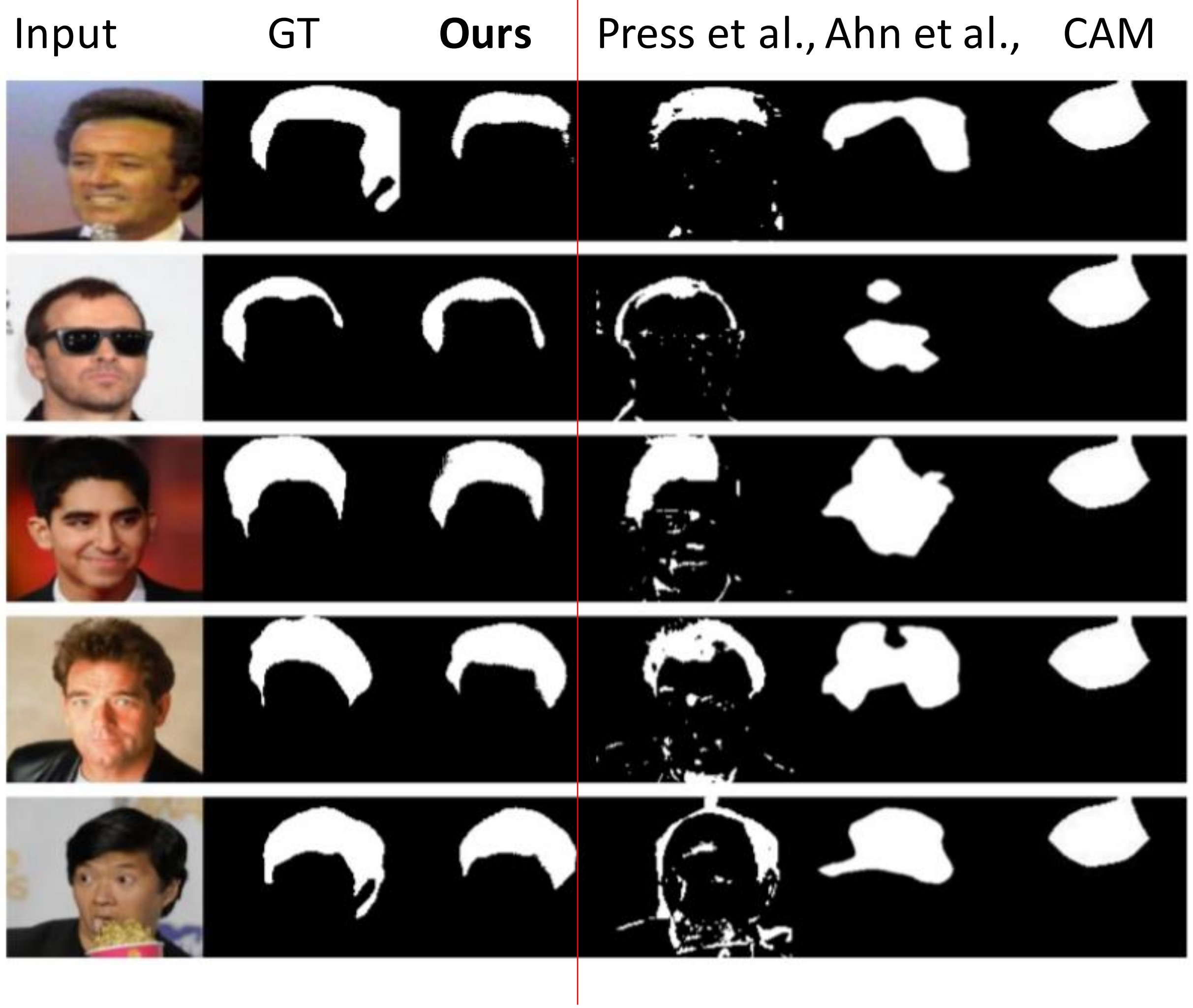
Table 6: Attribute removal for the task of Smile, Facial hair and Glasses.

Task	Method	KID	FID	Class.	Sim.
Smile	Ours	2.6 ± 0.4	120.0 ± 2.6	96.9%	0.96
	Press et al.	15.0 ± 0.6	167.7 ± 0.3	96.9%	0.81
	He et al.	4.1 ± 0.4	127.7 ± 4.5	96.9%	0.95
	Liu et al.	4.3 ± 0.3	129.0 ± 3	98.4%	0.92
	Fader	11.3 ± 0.7	155.6 ± 4.7	93.7 %	0.89
Mustache	Ours	1.9 ± 0.5	119.0 ± 0.8	95.3 %	0.95
	Press et al.	16.6 ±0.8	175.9 ± 1.4	100.0%	0.80
	He et al.	4.6 ± 0.5	130.0 ± 3.0	87.5%	0.96
	Liu et al.	14.0 ± 0.6	160.0 ± 3.3	87.5%	0.85
	Fader	14.1 ± 0.6	162.6 ± 1.5	98.4 %	0.76
Glasses	Ours	5.2± 0.5	136.5± 2.6	99.2%	0.87
	Press et al.	15.3± 0.5	172.0 ± 4.7	100.0%	0.73
	He et al.	8.3 ± 0.9	141.4±6.8	100.0%	0.84
	Liu et al.	6.8 ± 0.3	141.8 ± 4.8	98.4%	0.86
	Fader	12.5± 0.3	137.7± 4.2	100.0%	0.76

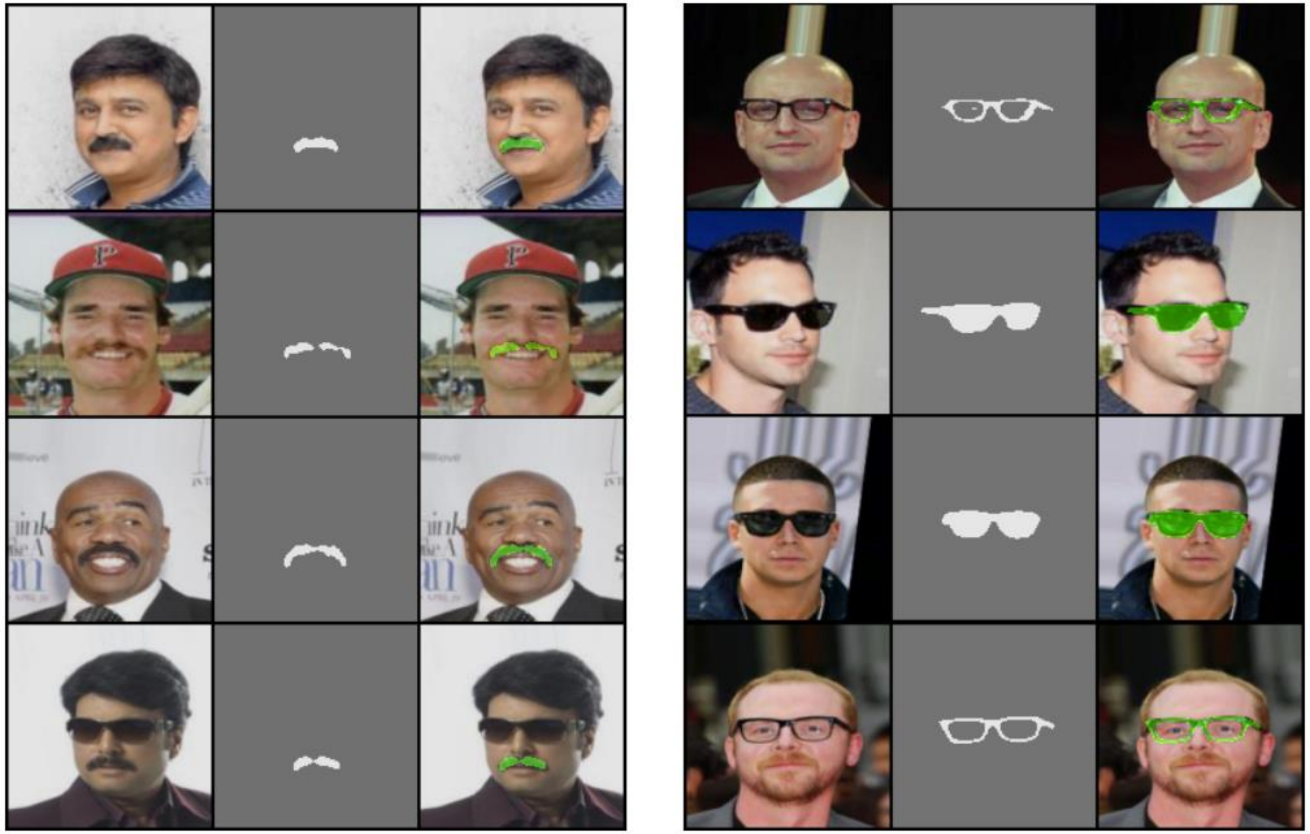
Out of Domain Manipulation

(a) (b)

Semi-Supervised Background-Foreground Segmentation Using Class Labels



Semi-Supervised Background-Foreground Segmentation Using Class Labels



Disentanglement: Supervision Level

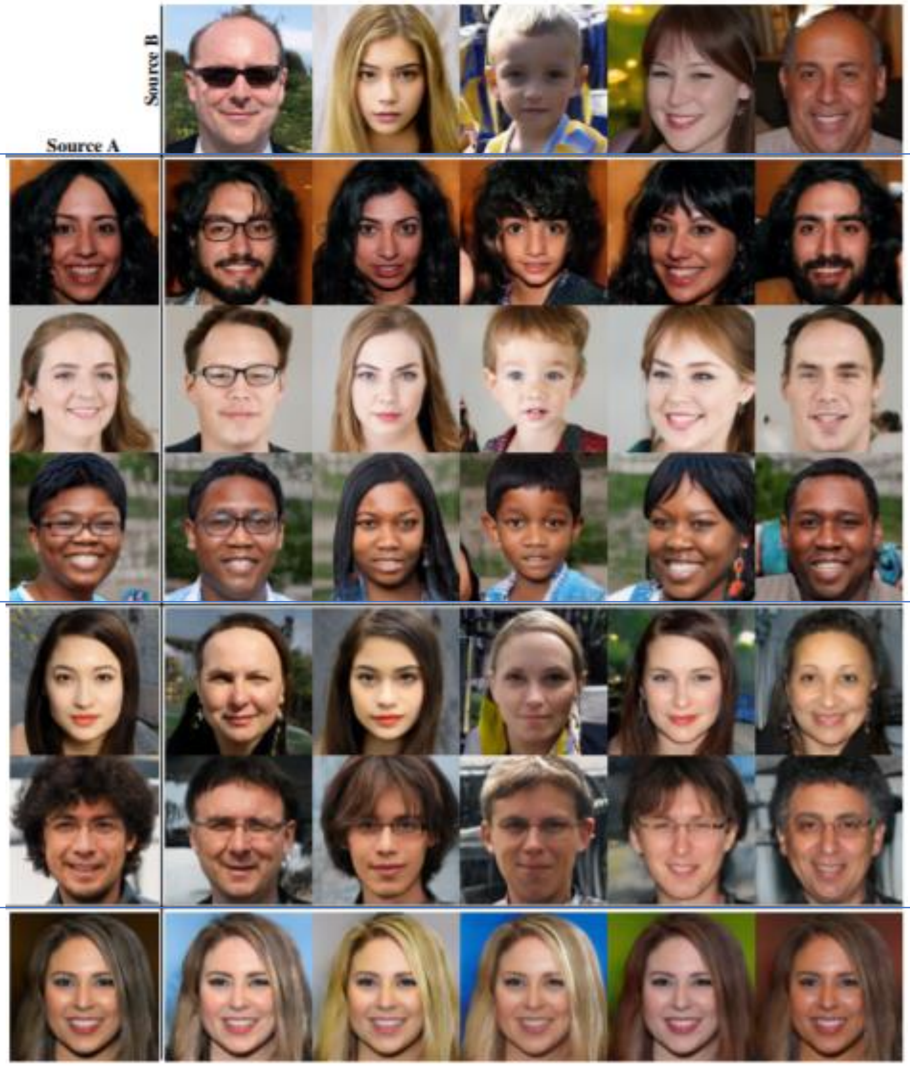
- Fully Supervised: Each image in the dataset appears with or without each factor of variation.
- Semi-Supervised (Set Level): Each set of images (which may be different), appear with or without each factor of variation.
- Unsupervised: Strong assumptions about data-set which are incorporated into the model design.

Unsupervised

"… **unsupervised** learning of disentangled representations is fundamentally **impossible without inductive biases** both on the considered **learning approaches** and the **data sets**."[1]

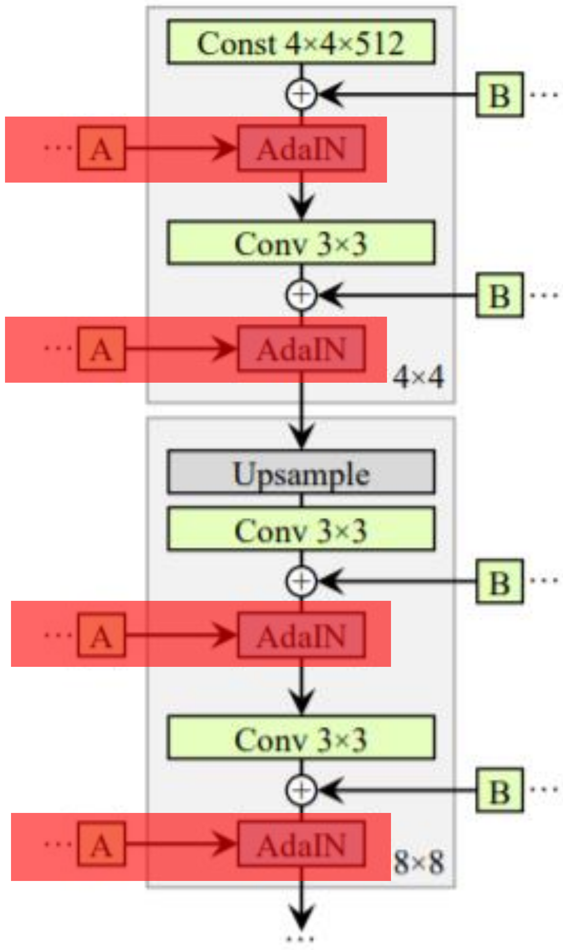
1. "Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations". ICML 2019. Locatello et al.,

Content Style Disentanglement: StyleGAN



Top
Middle
Bottom

Content Style Disentanglement: StyleGAN



Instance Normalization

- Let a be a representation of images I_a

$$IN(a) = \left(\frac{a\ - \mu(a)}{\sigma(a)}\right)$$

- μ and σ are computed along the spatial dimension of a.
- $\mu(a)$ and $\sigma(a)$ represent the **global statistics** of an image (such as brightness, contrast, lightning and global color changes)

Disentangle content from global statistics

- Let a be a representation of images I_a

$$a = \overbrace{\sigma(a)}^{\text{Global Statistics}} \underbrace{\left(\frac{a - \mu(a)}{\sigma(a)}\right)}_{\text{Content}} + \overbrace{\mu(a)}^{\text{Global Statistics}}$$

- $\mu(a)$ and $\sigma(a)$ represent the **global statistics** of an image (such as brightness, contrast, lightning and global color changes)
- **Content** represents information relating to shape and texture of objects.
- This gives unsupervised disentanglement of content and global statistics!

AdaIN – Adaptive Instance Normalization

- Let a, b be a representation of images I_a, I_b

$$AdaIN(a,b) = \overbrace{\sigma(b)}^{\text{Global Statistics}} \underbrace{\left(\frac{a - \mu(a)}{\sigma(a)}\right)}_{\text{Content}} + \overbrace{\mu(b)}^{\text{Global Statistics}}$$

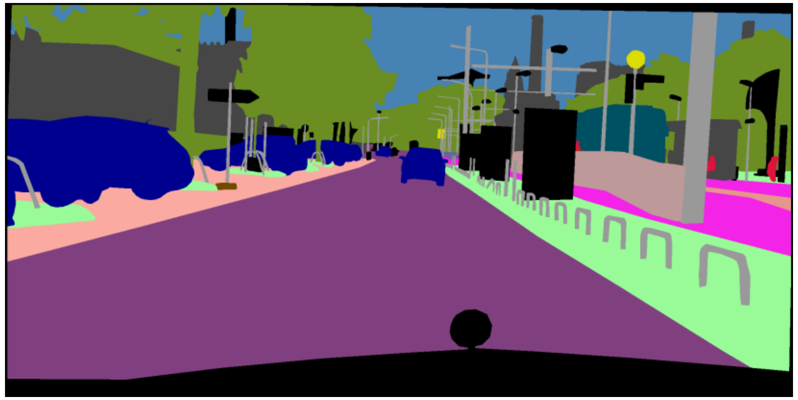
- Replace the global statistics of a with that of b

Domain Adaptation

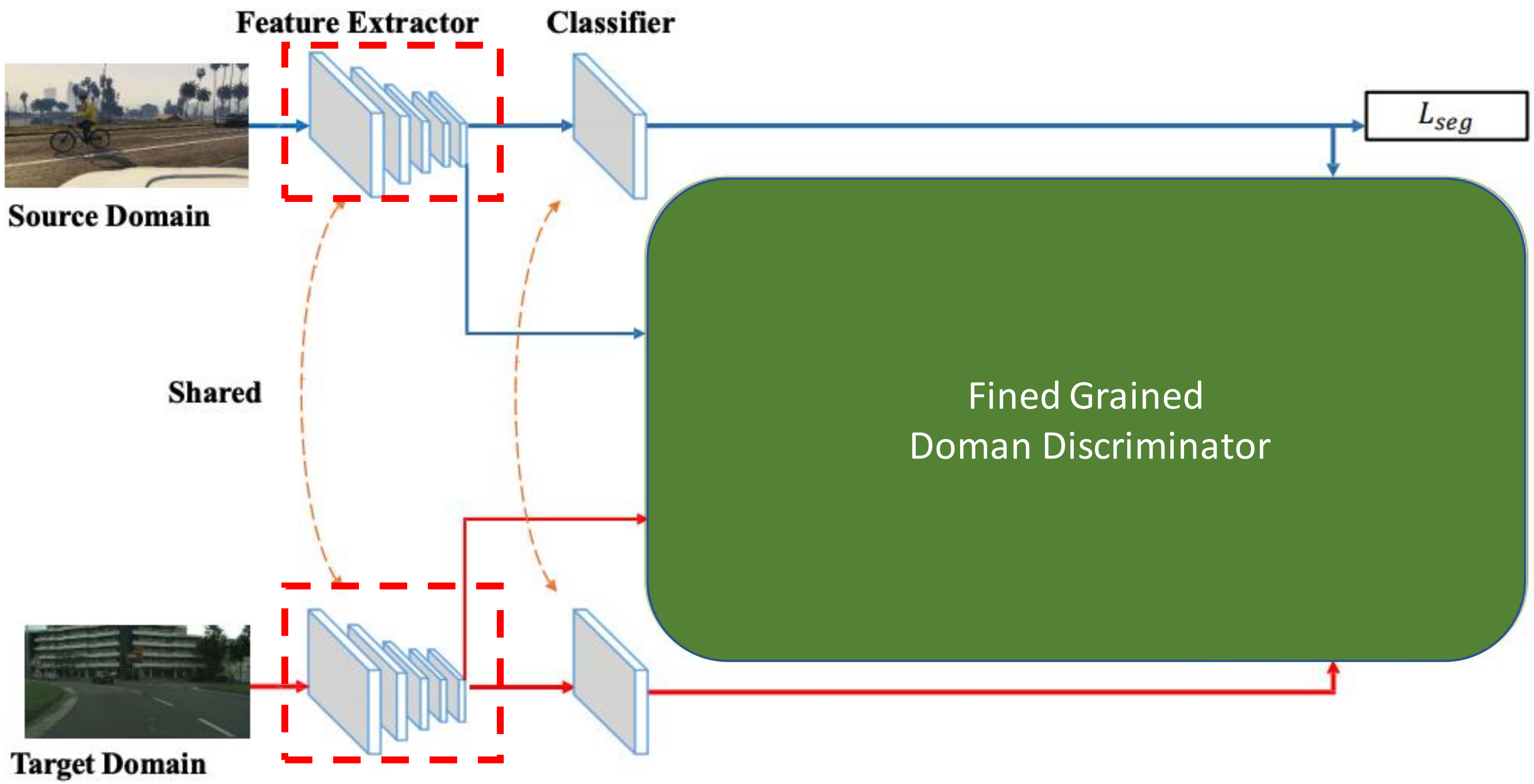
Supervised training on source domain and unsupervised on target domain

Source: GTAV

Target: Cityscapes

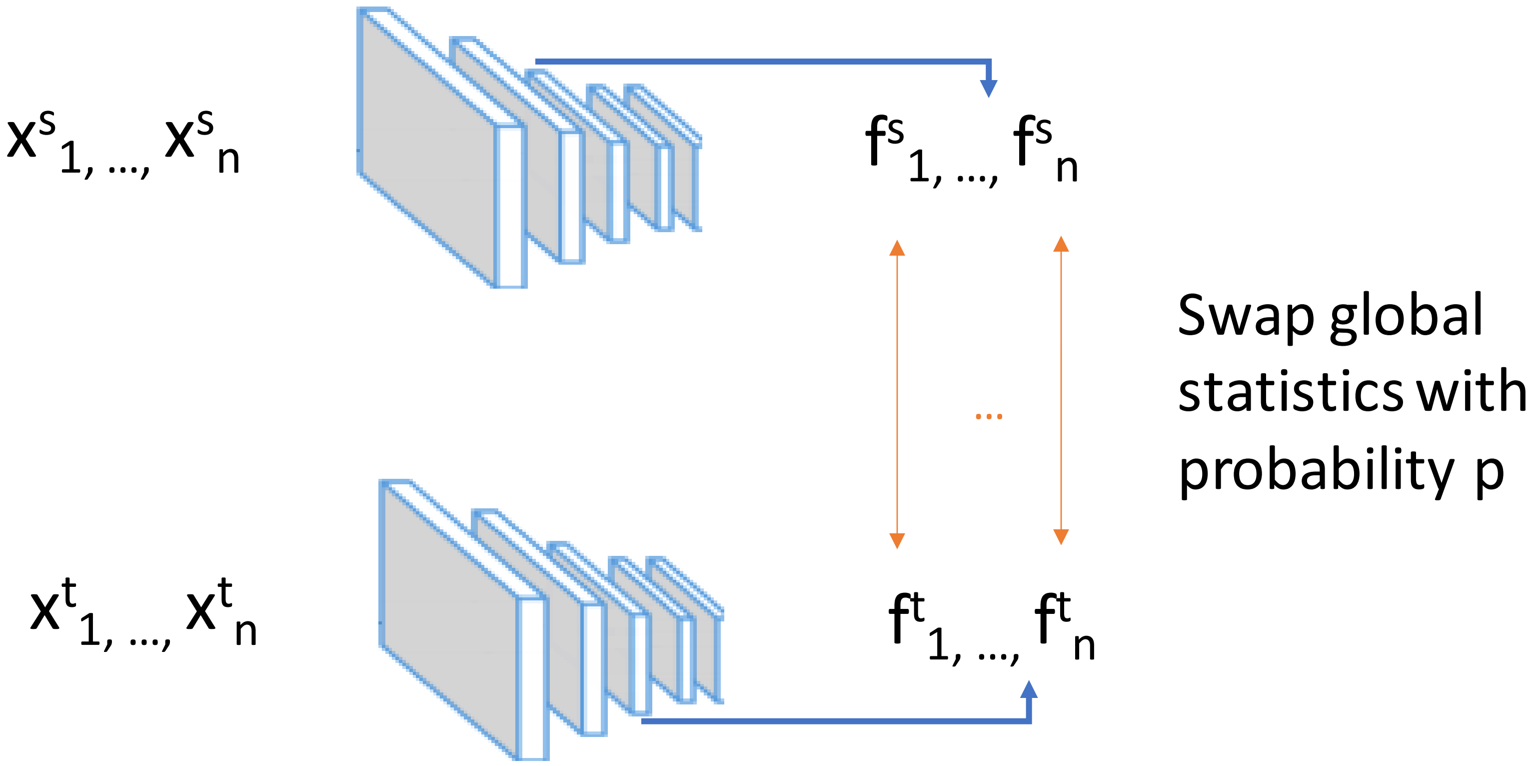


Unsupervised Domain Adaptation



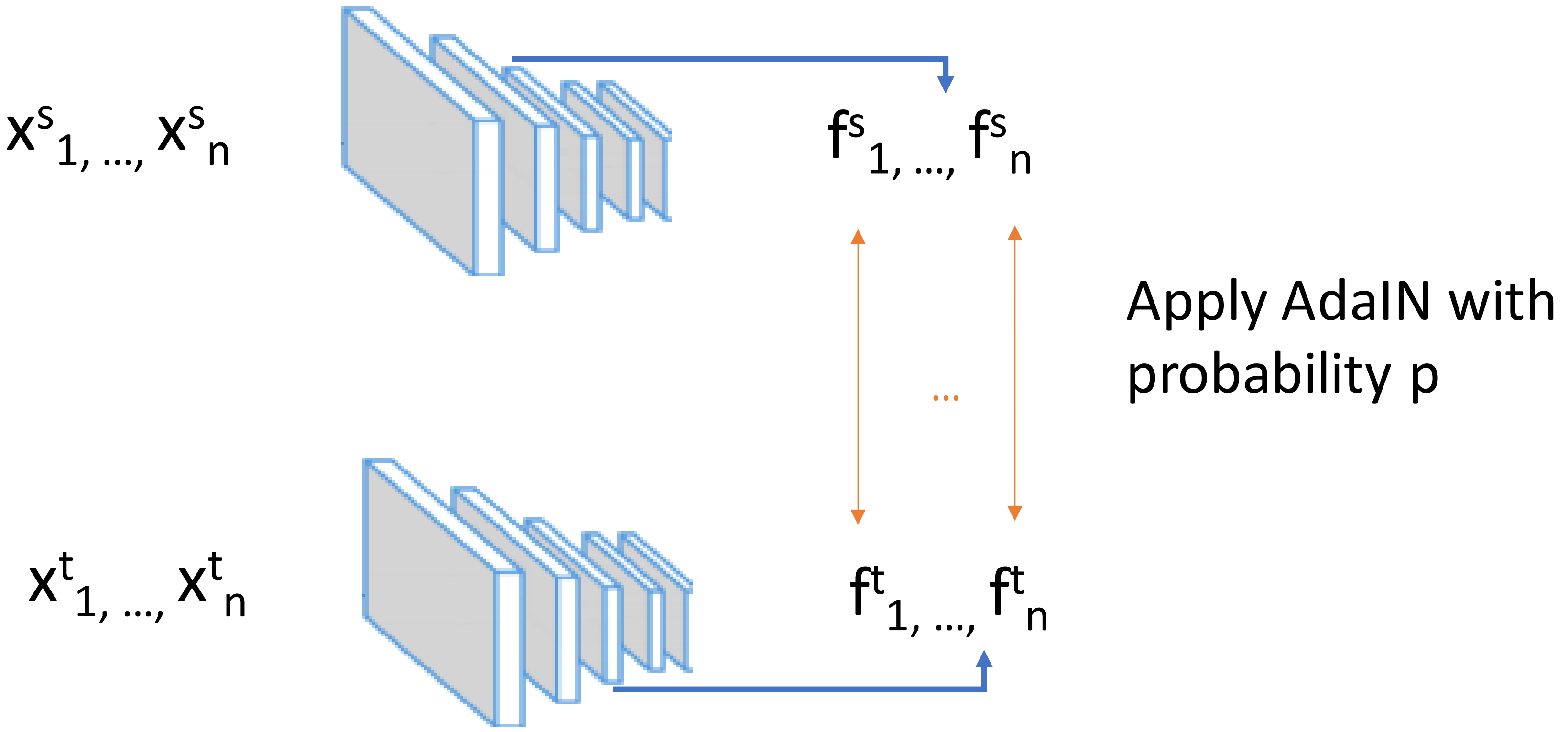
Permuted AdaIN: Reducing the Bias Towards Global Statistics in Image Classification

O. Nuriel, S. Benaim, L. Wolf. Submitted to CVPR 2021.

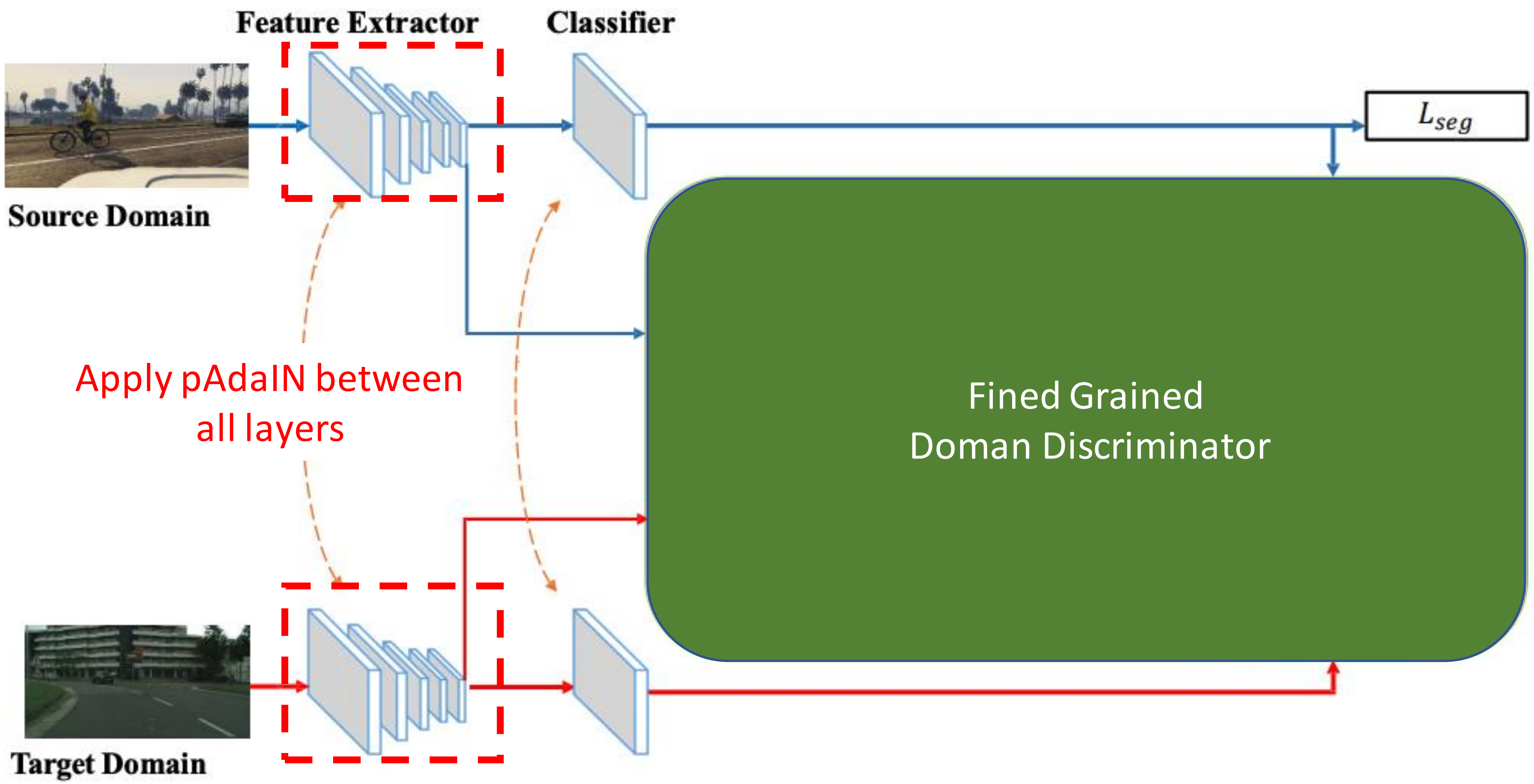


Permuted AdaIN: Reducing the Bias Towards Global Statistics in Image Classification

O. Nuriel, S. Benaim, L. Wolf. Submitted to CVPR 2021.



Unsupervised Domain Adaptation

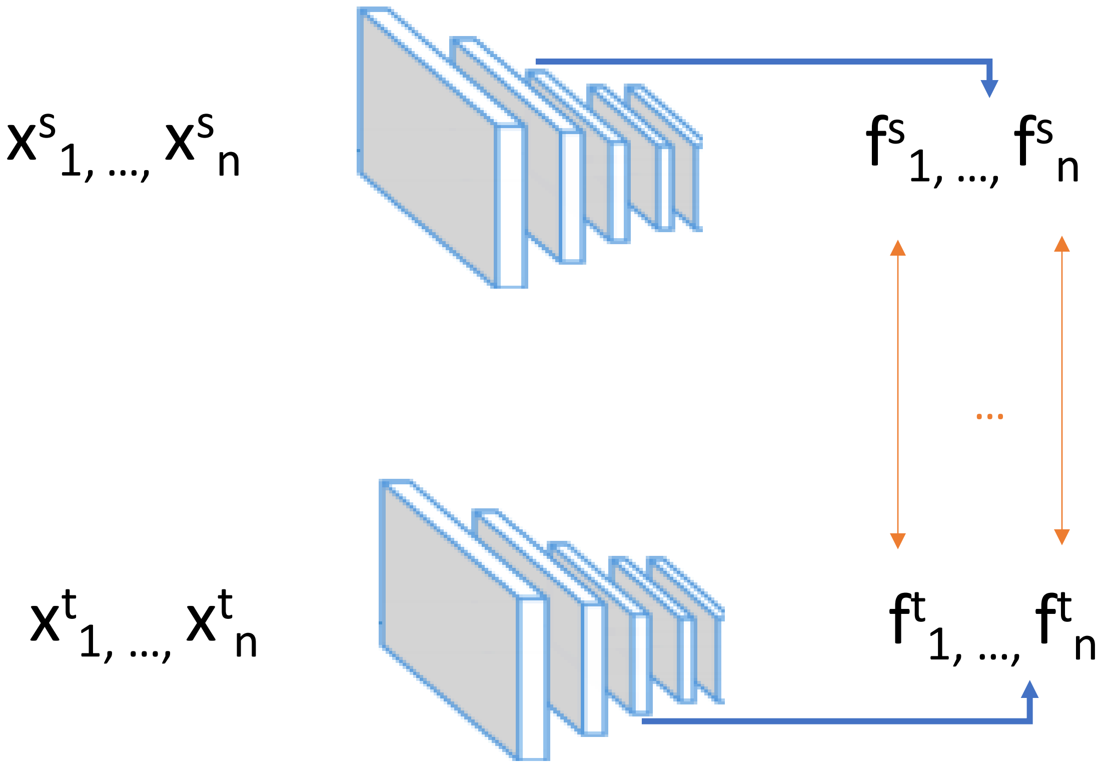


Unsupervised Domain Adaptation

GTVA to Cityscapes

AdaptSegNet [35]	86.5	36.0	79.9	23.4	23.3	23.9	35.2	14.8	83.4	33.3	75.6	58.5	27.6	73.7	32.5	35.4	3.9	30.1	28.1	42.4
SIBAN [28]	88.5	35.4	79.5	26.3	24.3	28.5	32.5	18.3	81.2	40.0	76.5	58.1	25.8	82.6	30.3	34.4	3.4	21.6	21.5	42.6
CLAN [29]	87.0	27.1	79.6	27.3	23.3	28.3	**35.5**	24.2	83.6	27.4	74.2	58.6	28.0	76.2	33.1	36.7	6.7	31.9	31.4	43.2
AdaptPatch [36]	92.3	51.9	82.1	29.2	25.1	24.5	33.8	**33.0**	82.4	32.8	82.2	58.6	27.2	84.3	33.4	46.3	2.2	29.5	32.3	46.5
ADVENT [38]	89.4	33.1	81.0	26.6	26.8	27.2	33.5	24.7	83.9	36.7	78.8	58.7	30.5	84.8	38.5	44.5	1.7	31.6	32.4	45.5
FADA [40]	92.5	47.5	85.1	37.6	**32.8**	**33.4**	33.8	18.4	85.3	37.7	83.5	63.2	**39.7**	87.5	32.9	47.8	1.6	34.9	**39.5**	49.2
FADA [40] + pAdaIN	**93.3**	**55.7**	**85.6**	**38.3**	29.6	31.2	34.2	17.8	**86.2**	**41.0**	**88.8**	**65.1**	37.1	**87.6**	**45.9**	**55.1**	15.1	**39.4**	31.1	**51.5**

Domain Adaptation



Swap global statistics with probability p

Image Classification

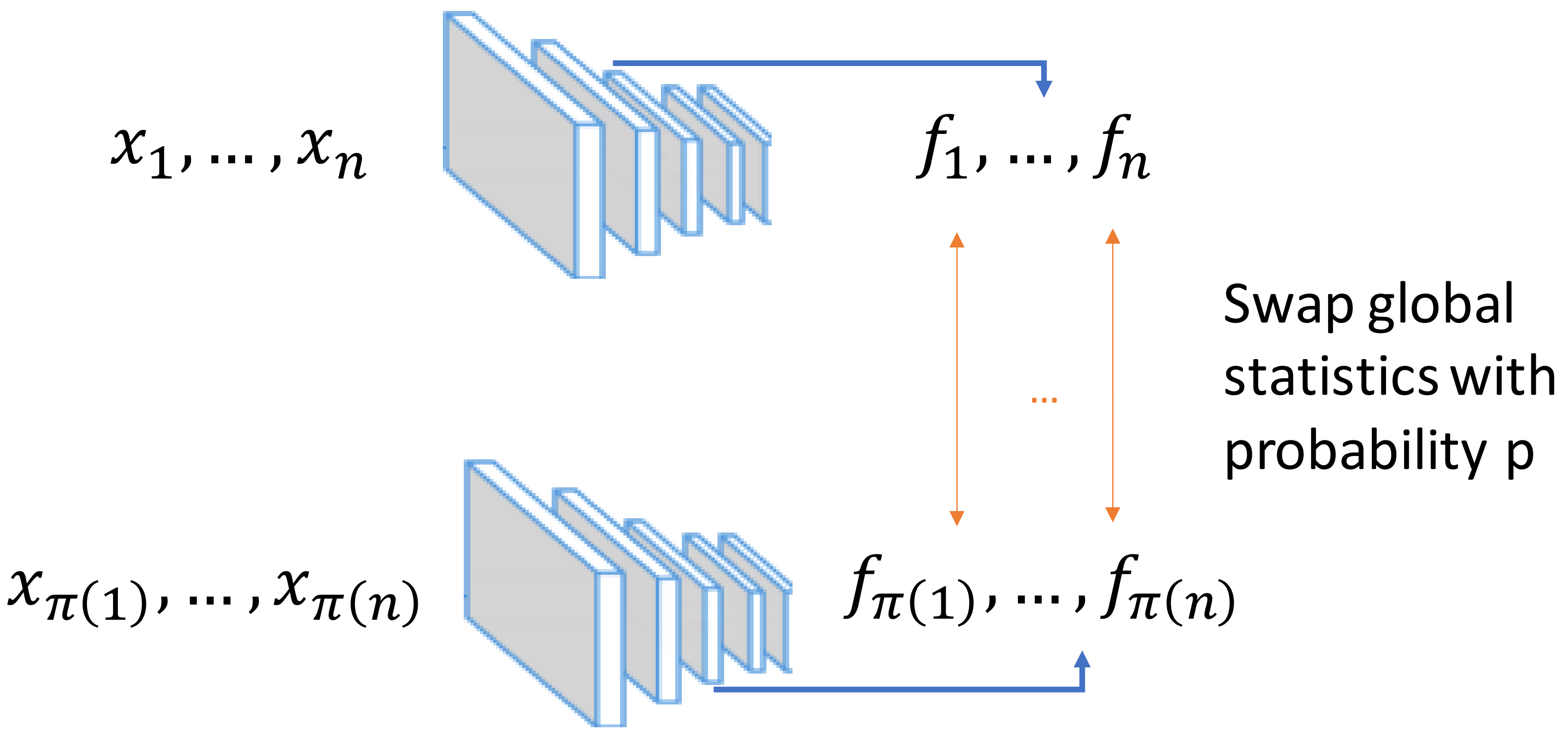


Image Classification

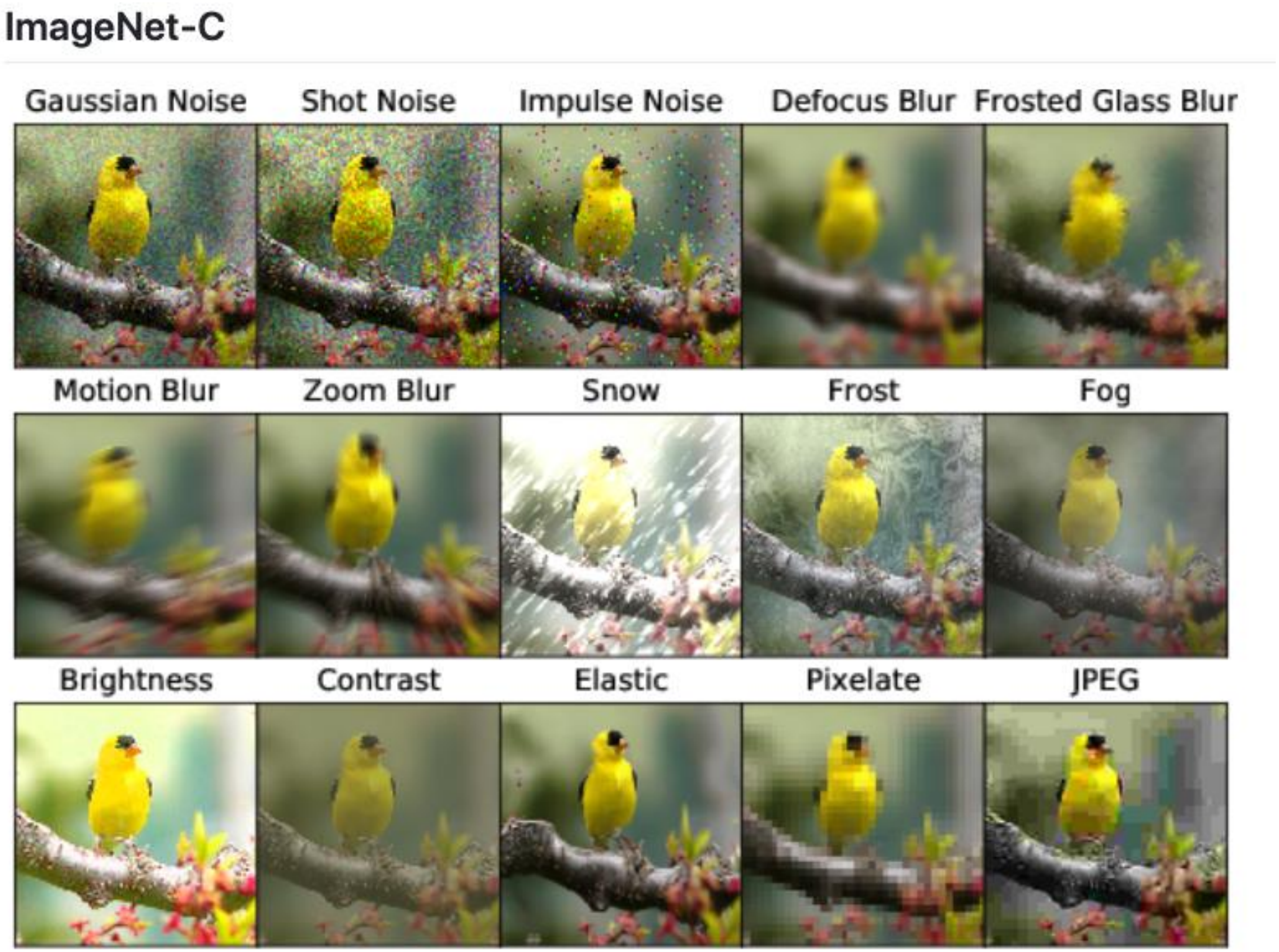
ImageNet

Method	Architecture	Top-1 Accuracy	Top-5 Accuracy
Baseline	ResNet50	77.1	93.63
pAdaIN	ResNet50	**77.7**	**93.93**
Baseline	ResNet101	78.13	93.71
pAdaIN	ResNet101	**78.8**	**94.35**
Baseline	ResNet152	78.31	94.06
pAdaIN	ResNet152	**79.13**	**94.64**

Cifar100

Method	Architecture	CIFAR 100
Baseline	PyramidNet	83.49
pAdaIN	PyramidNet	**84.17**
Baseline	ResNet18	76.13
pAdaIN	ResNet18	**77.82**
Baseline	ResNet50	78.22
pAdaIN	ResNet50	**79.03**

Robustness Towards Corruption



Robustness Towards Corruption

CIFAR100-C

	Baseline	Cutout [8]	Mixup [43]	CutMix [43]	Auto-Augment [7]	Adversarial Training [30]	Augmix [18]	pAdaIN+ Augmix
DenseNet-BC	59.3	59.6	55.4	59.2	53.9	55.2	38.9	**37.5**
ResNext-29	53.4	54.6	51.4	54.1	51.3	54.4	34.4	**31.6**

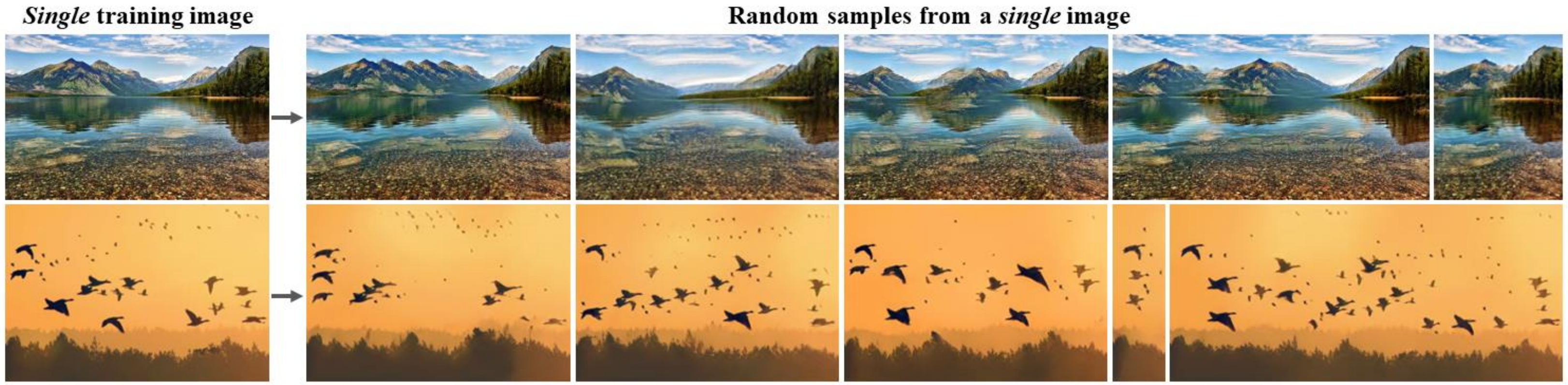
Category Wise Breakdown

Dataset	Network	Architecture	E	mCE	Noise			Blur				Weather				Digital			
					Gauss.	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG
INet-C	Baseline	ResNet50	22.9	76.7	80	82	83	75	89	78	80	78	75	66	57	71	85	77	77
INet-C	pAdaIN	ResNet50	**22.3**	**72.8**	**78**	**79**	**81**	**70**	**87**	**74**	**76**	**74**	**71**	**64**	**55**	**65**	**82**	**66**	**71**
C100-C	Augmix [18]	DenseNet-BC	24.2	38.9	60	51	41	27	55	31	29	36	39	35	28	37	33	39	41
C100-C	Augmix+pAdaIN	DenseNet-BC	**22.2**	**37.5**	**58**	**49**	**40**	**26**	**54**	**30**	**28**	**35**	**38**	**33**	**25**	**36**	**32**	**37**	**40**
C100-C	Augmix [18]	ResNext-29	21.0	34.4	**56**	**48**	32	23	**49**	27	25	32	35	32	24	32	30	34	37
C100-C	Augmix+pAdaIN	ResNext-29	**17.3**	**31.6**	58	**48**	**24**	**20**	54	**23**	**21**	**28**	**30**	**25**	**19**	**27**	**27**	**33**	**36**

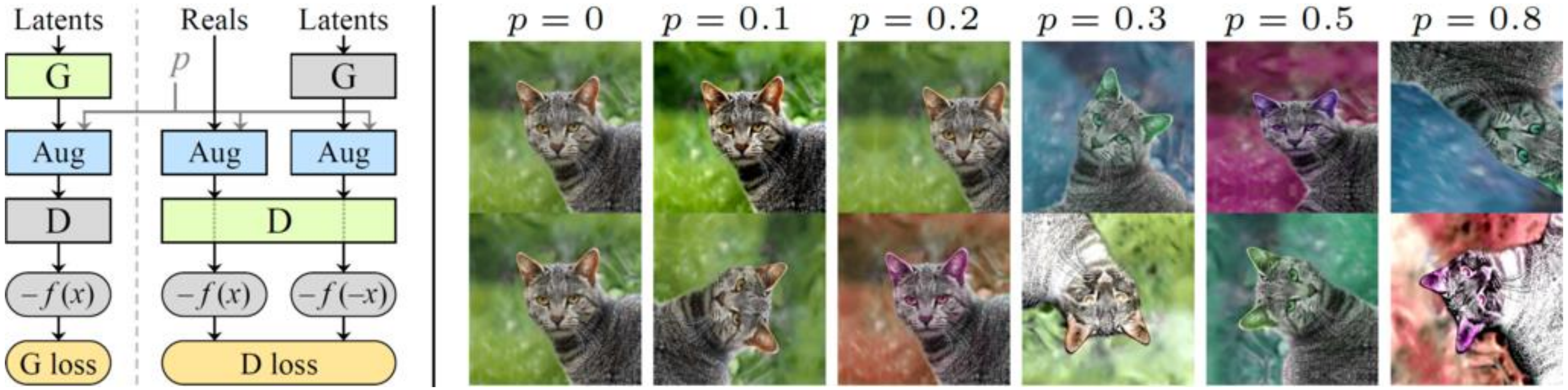
Part II: Few shot generation

Unconditional Few Shot Generation

SinGAN[1]

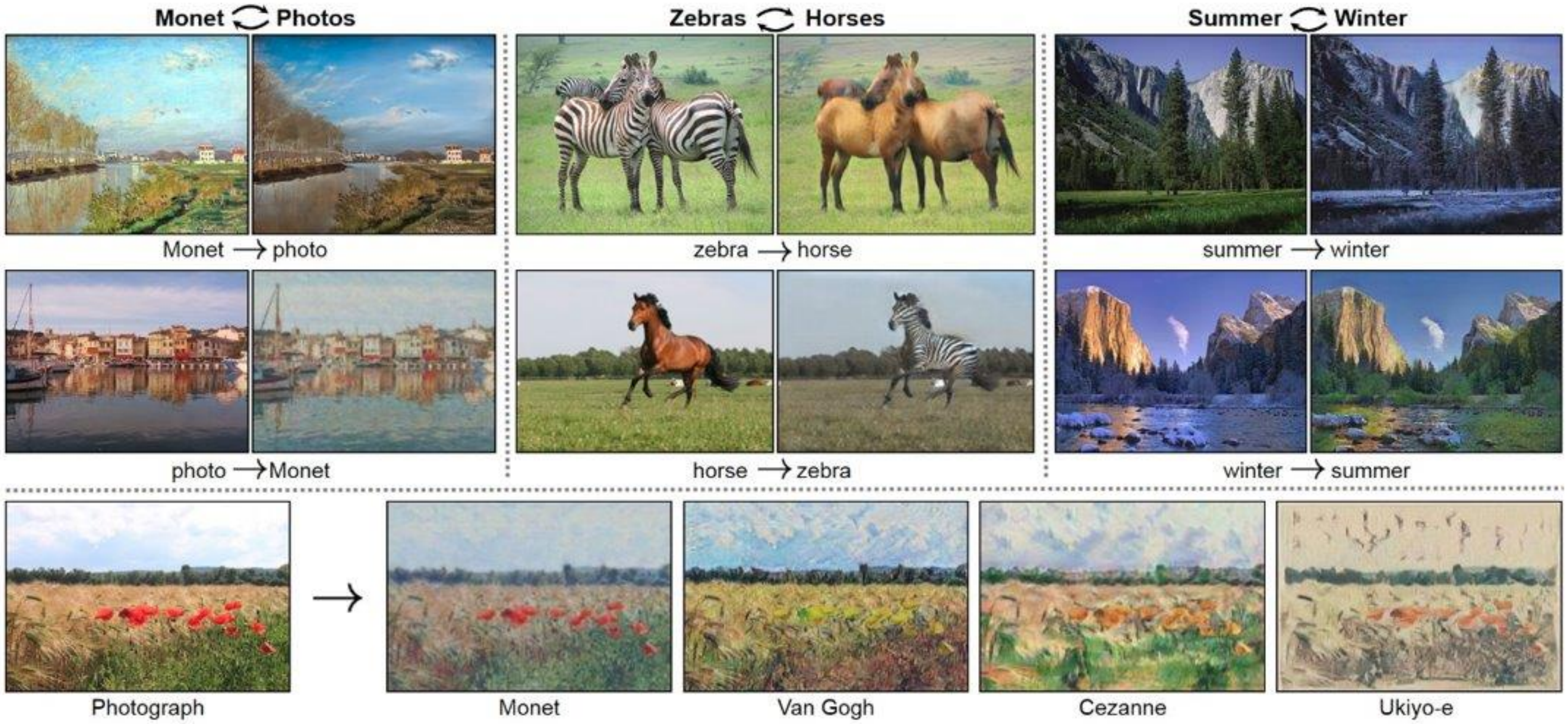


StyleGAN2-ada[2]



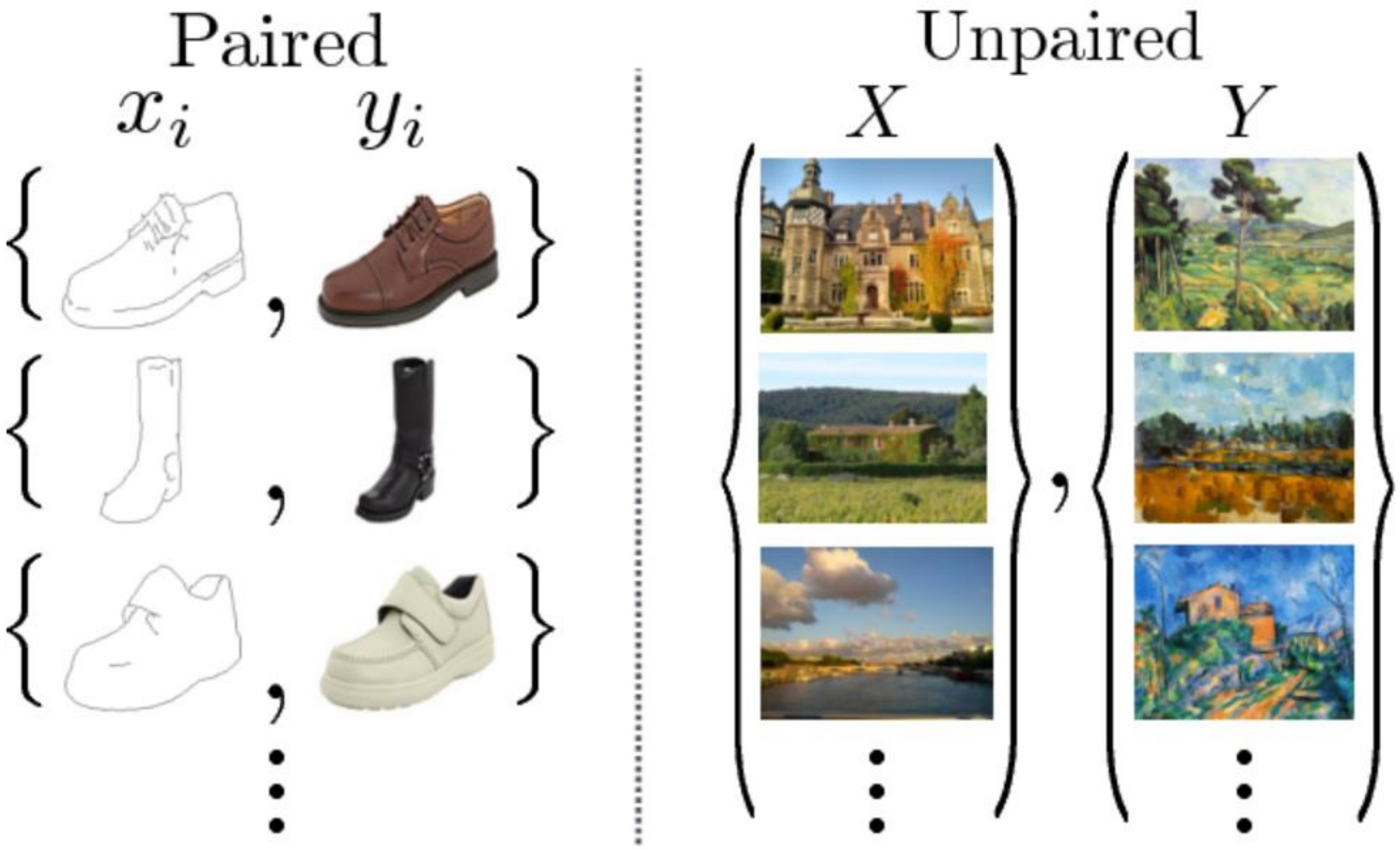
1. SinGAN: Learning a Generative Model from a Single Natural Image. ICCV 2019. Shaham et al.,
2. Training Generative Adversarial Networks with Limited Data. NeurIPS 2020. Karras et al.,

Image to Image Translation



CycleGAN, Zhu et al., ICCV 2017

Typical Training Setting



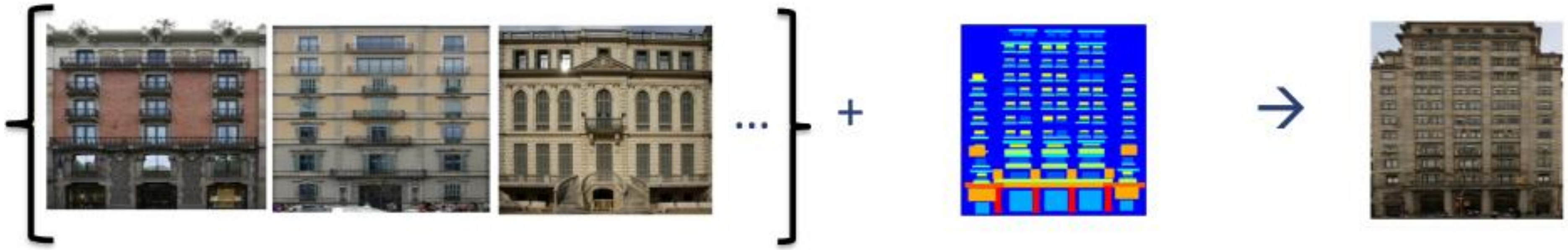
One-Shot Unsupervised Cross Domain Translation

S. Benaim, L. Wolf. NeurIPS 2018.

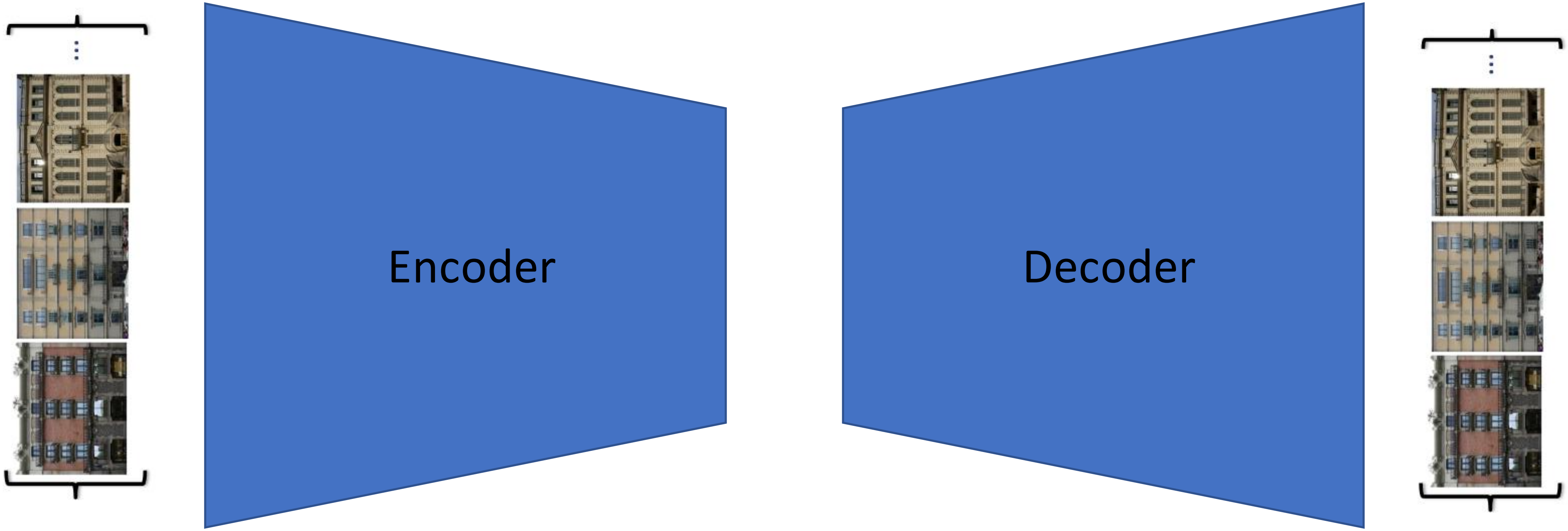
Domain B:
Many unmatched samples

Domain A:
A single sample x

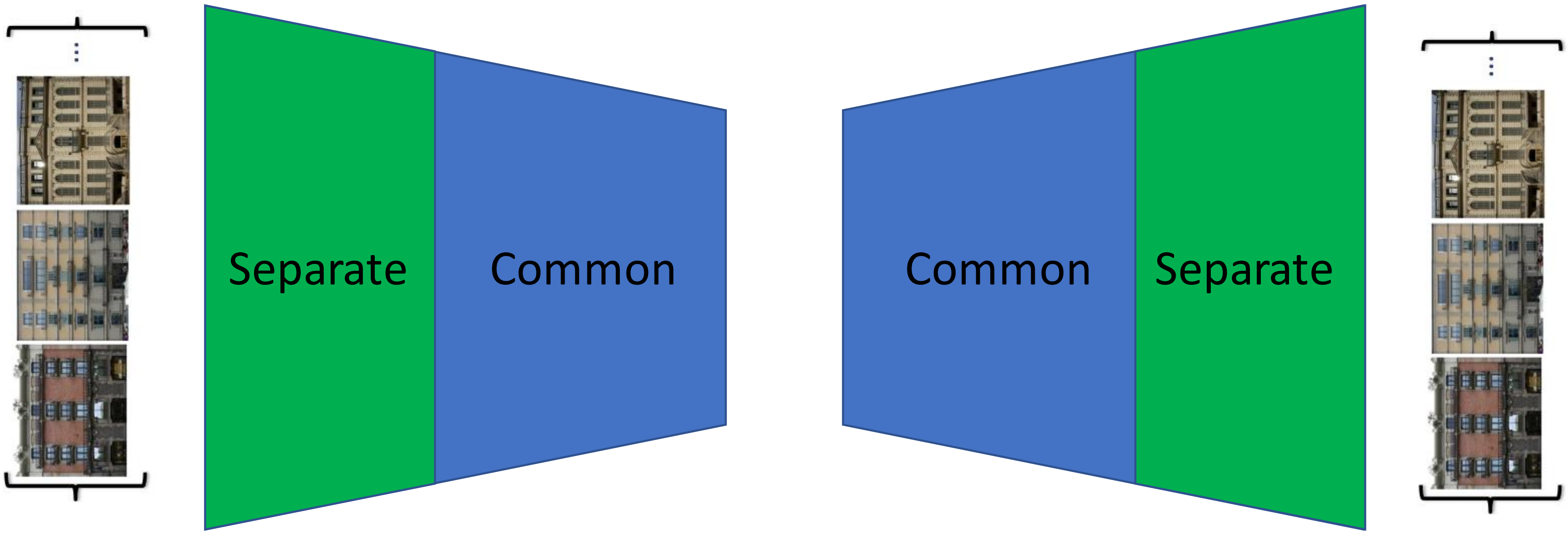
Analogue of
x in B



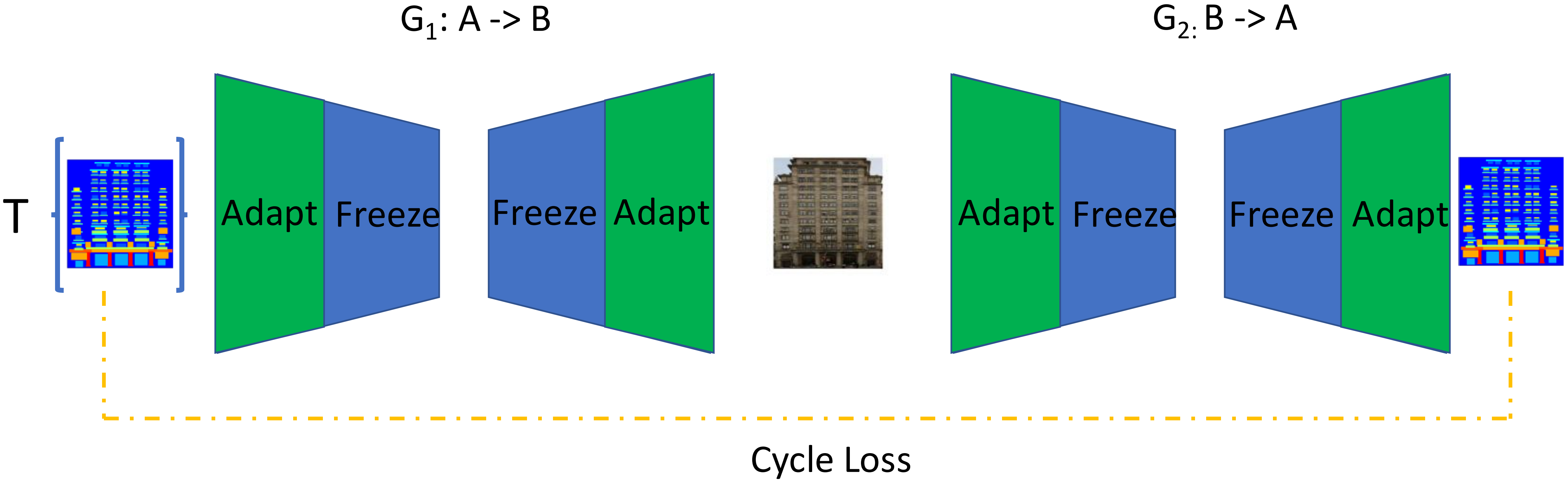
Phase I: Auto-Encoder for Domain B



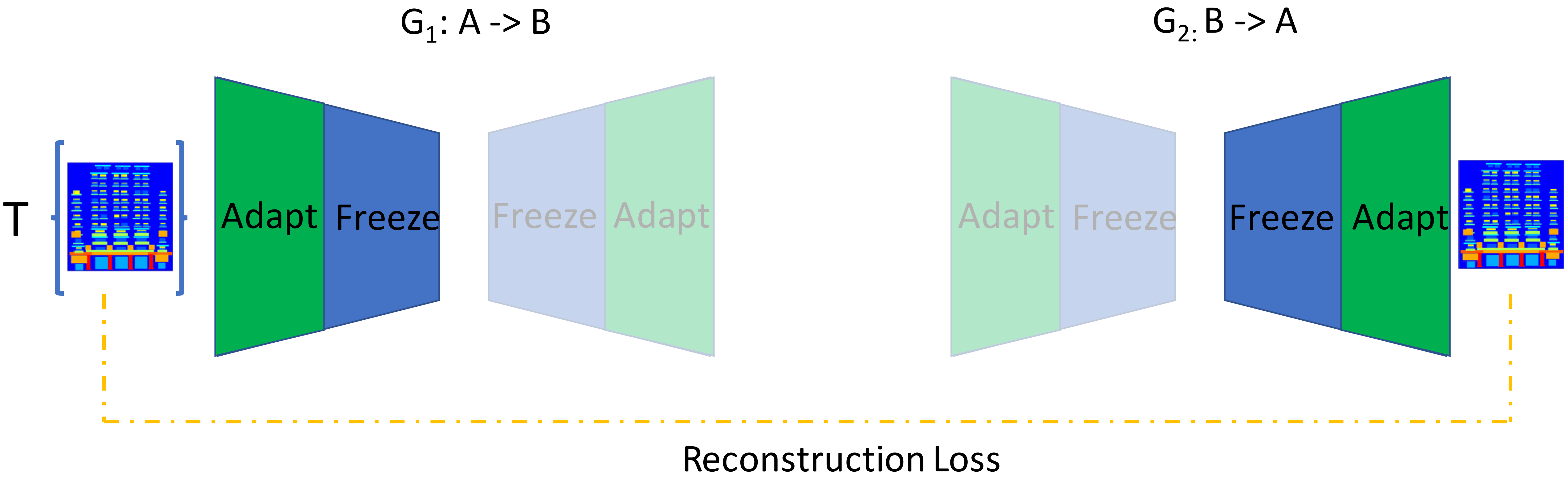
Phase II: Shared Latent Space Assumption



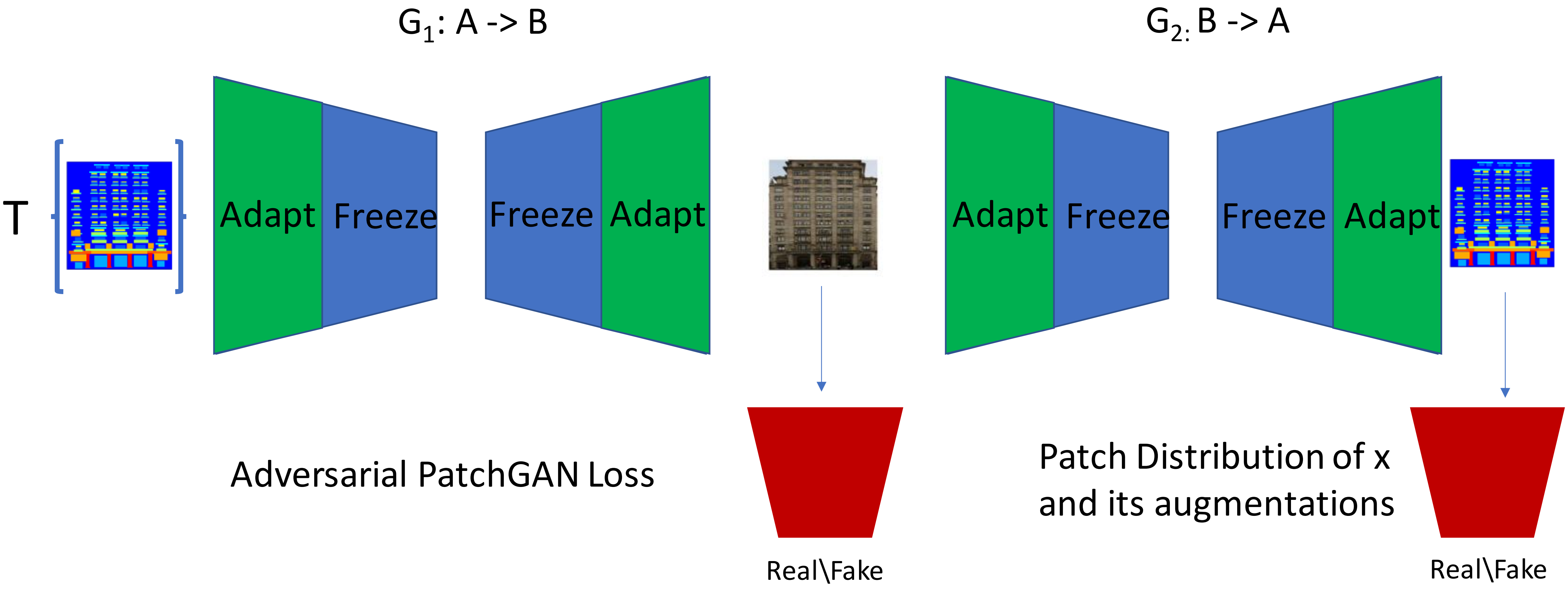
Phase II: Adapt Outer Layers



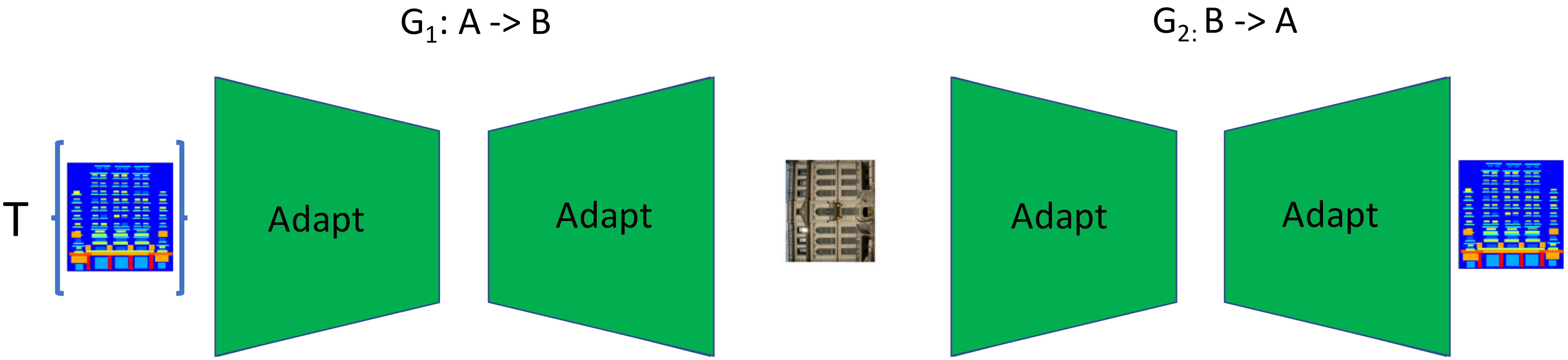
Phase II: Adapt Outer Layers



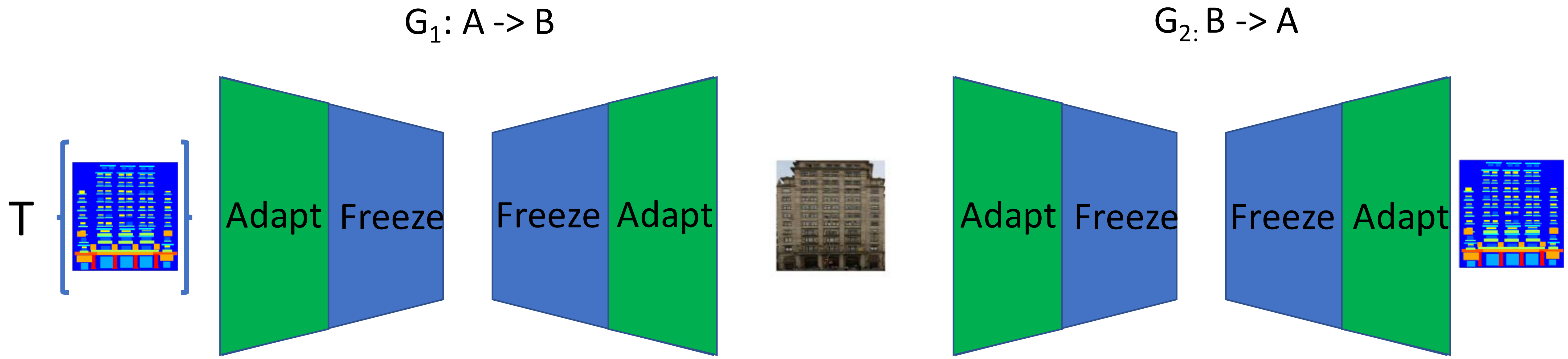
Phase II: Adapt Outer Layers



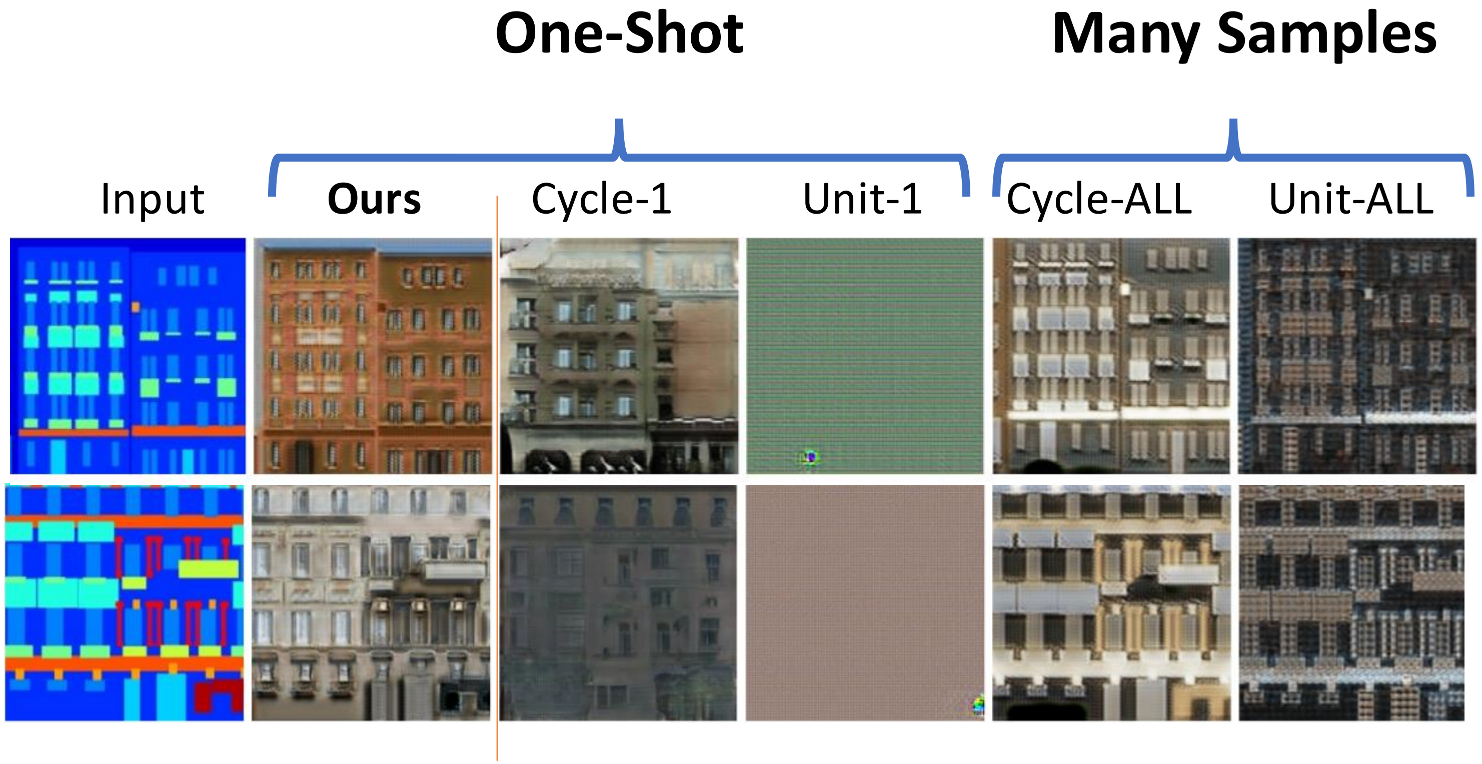
Adapt All Layers: Overfitting



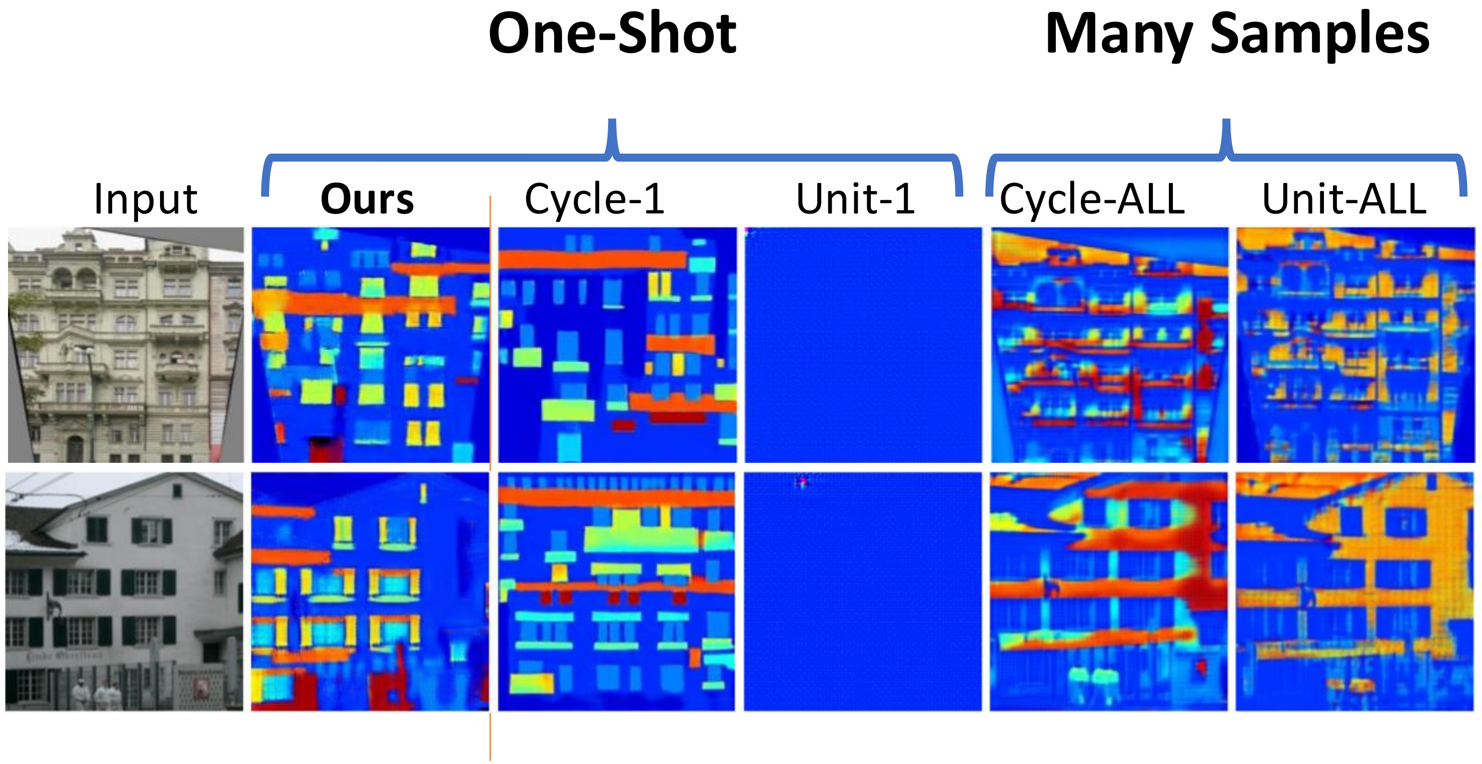
No Underfitting (Common Space Assumption)



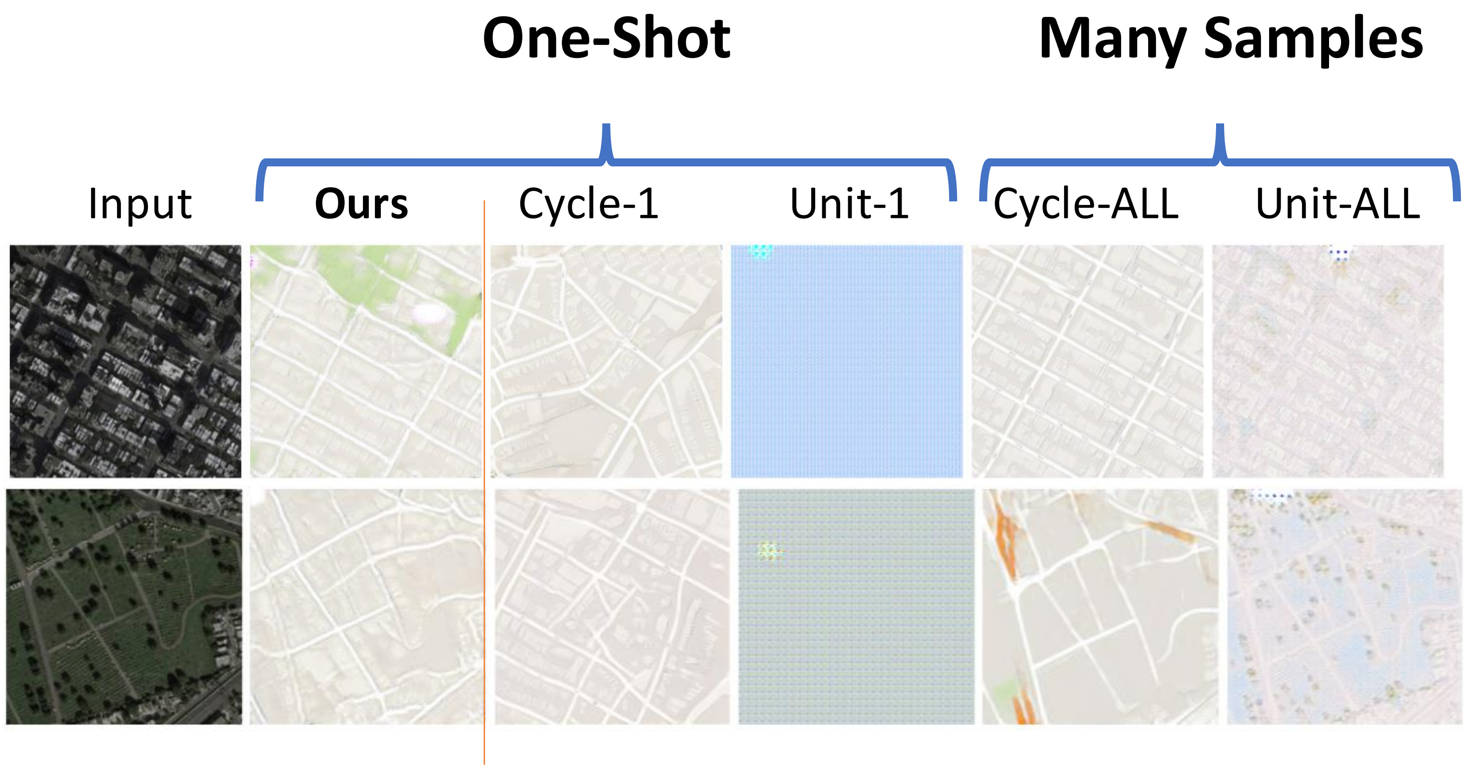
Segmentation to Facade



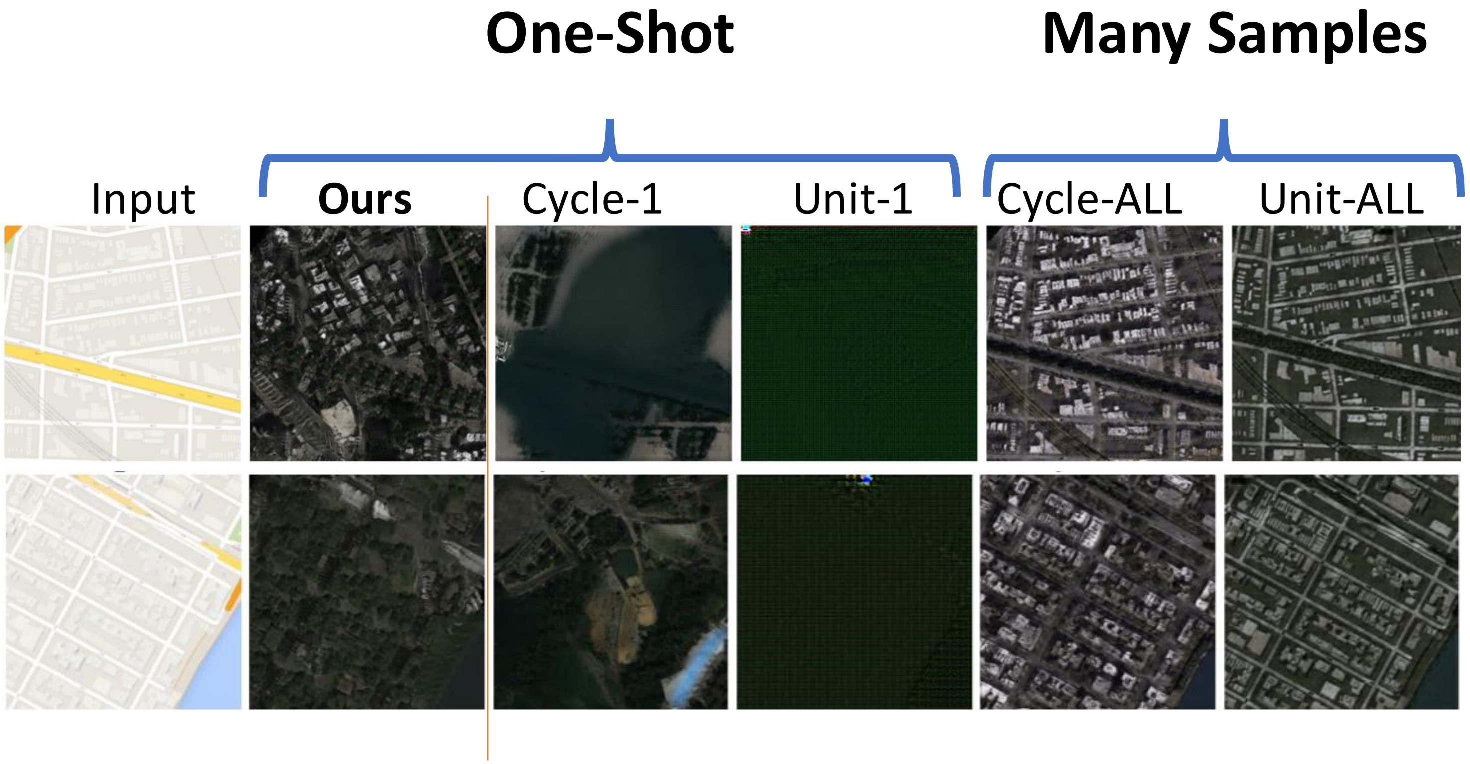
Facade to Segmentation



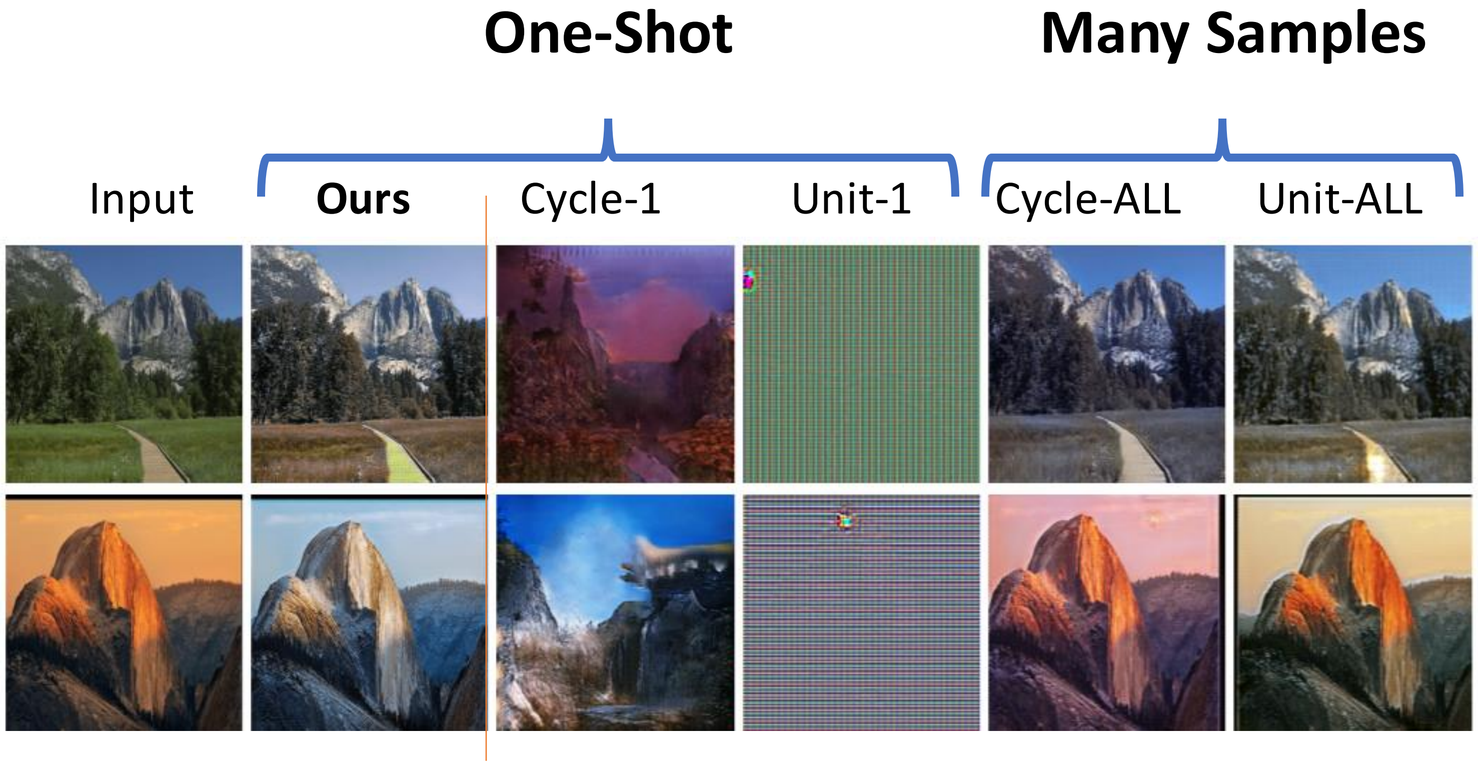
Aerial View to Maps



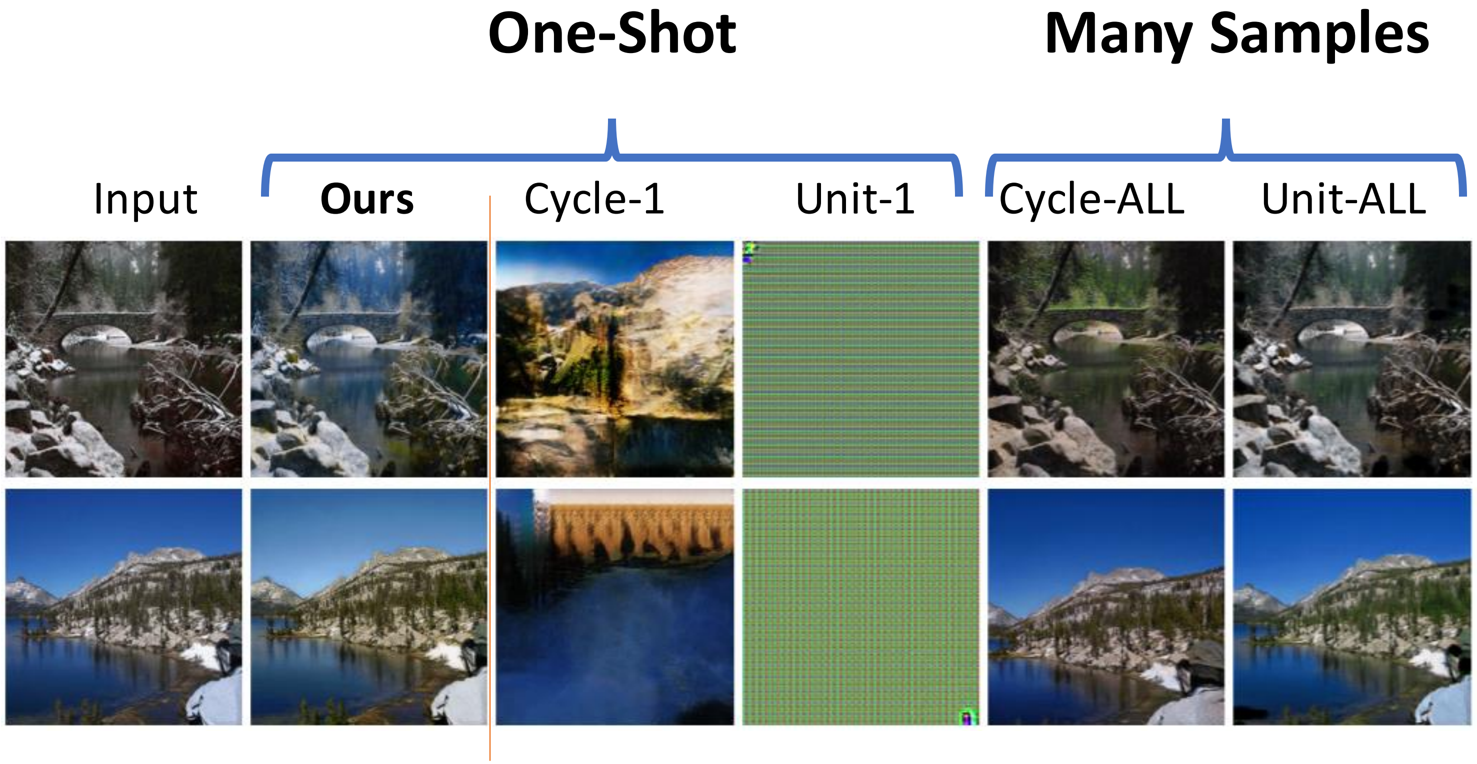
Maps to Aerial View



Summer to Winter



Winter to Summer



Structural-analogy from a Single Image Pair

S. Benaim*, R. Mokady*, A. Bermano, D Cohen-Or, L. Wolf. CGF 2020. (*Equal contribution)

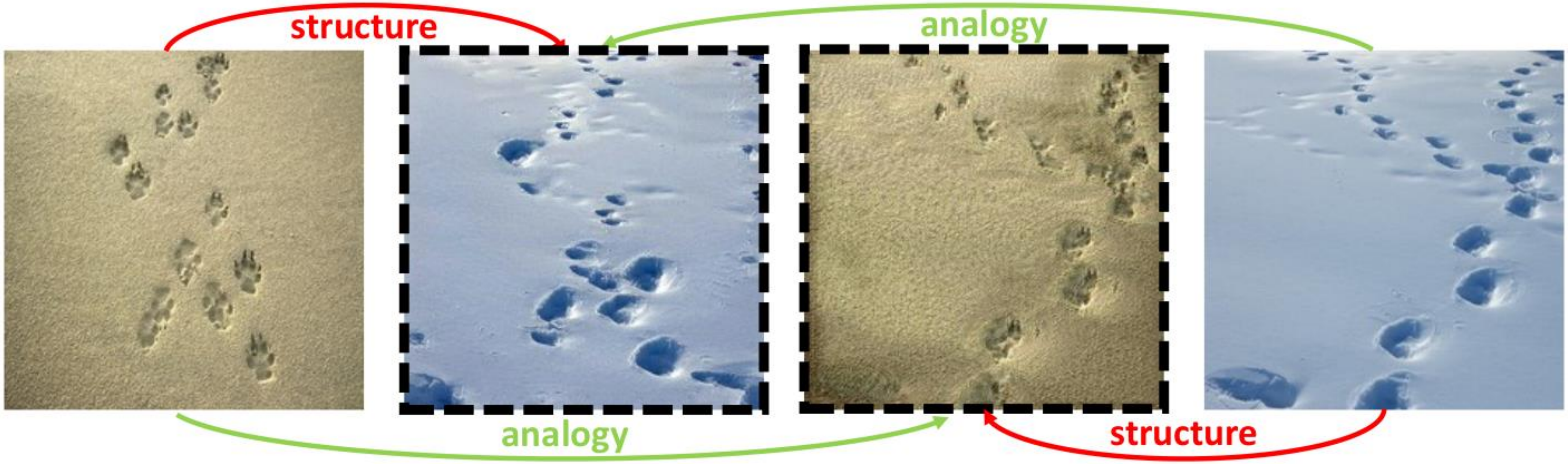
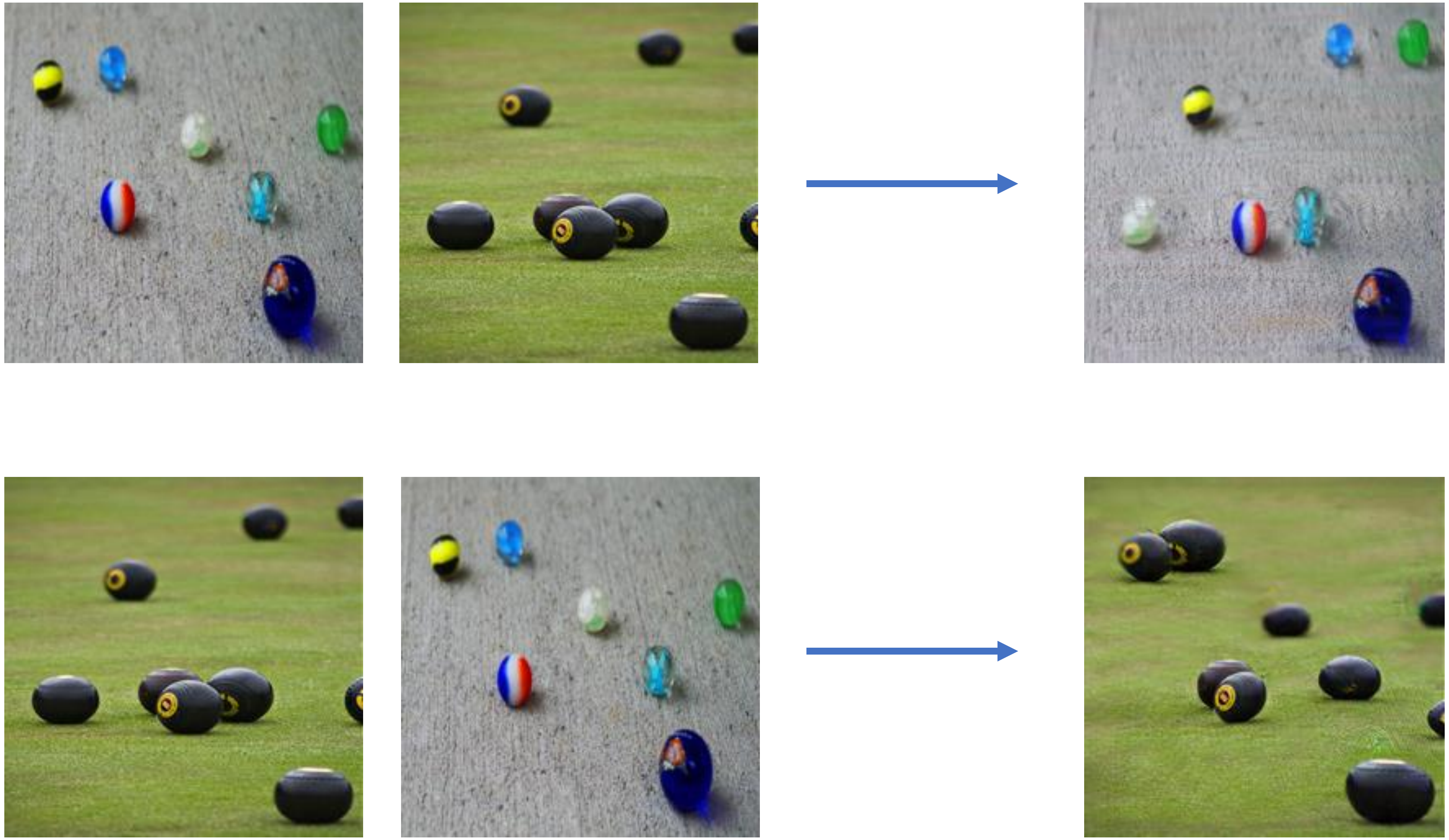


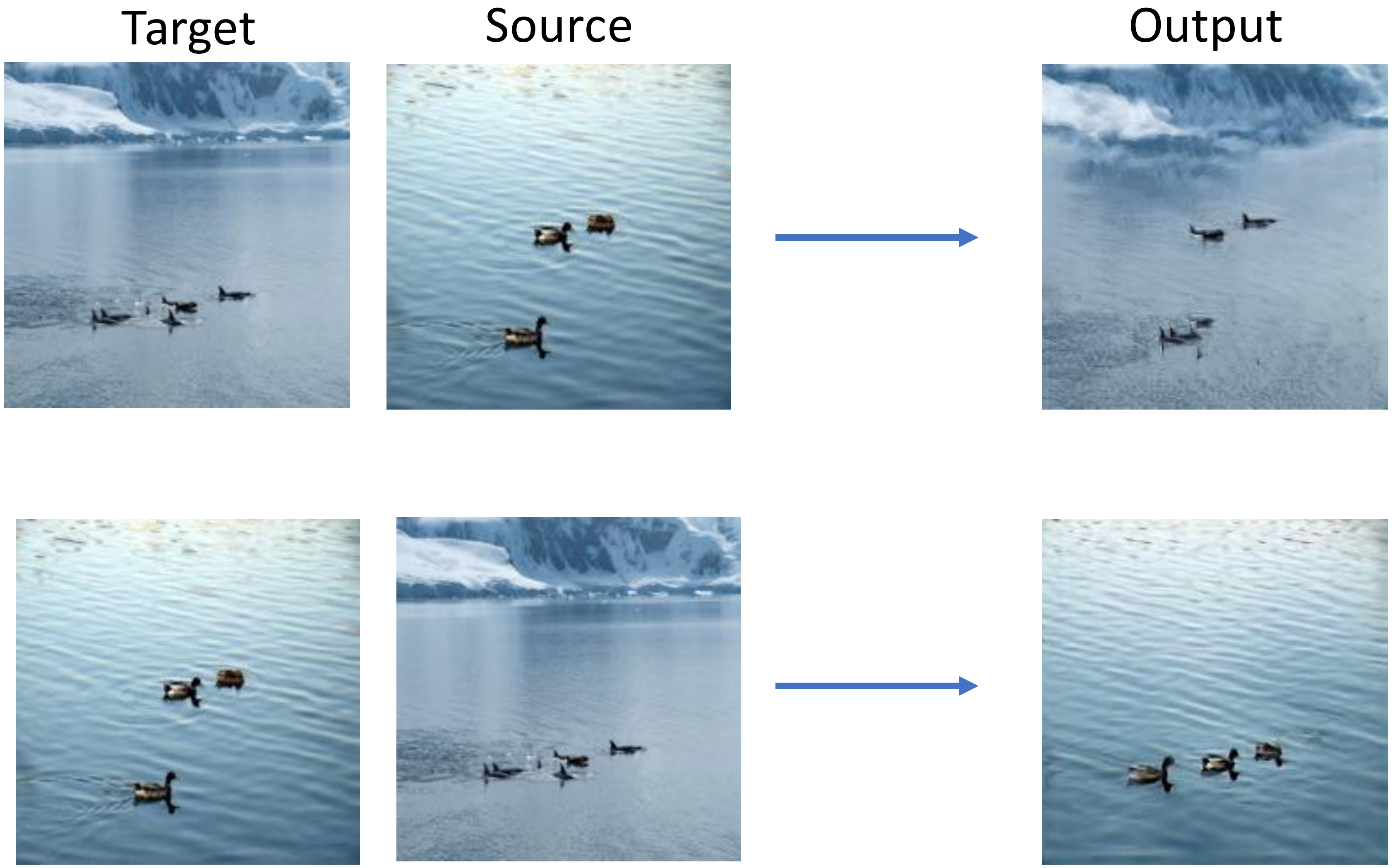
Fig. 1. Our method takes two images as input (left and right), and generates images that consist of features from one image, spatially structured analogically to the other.

Structural Analogy

Target Source Output

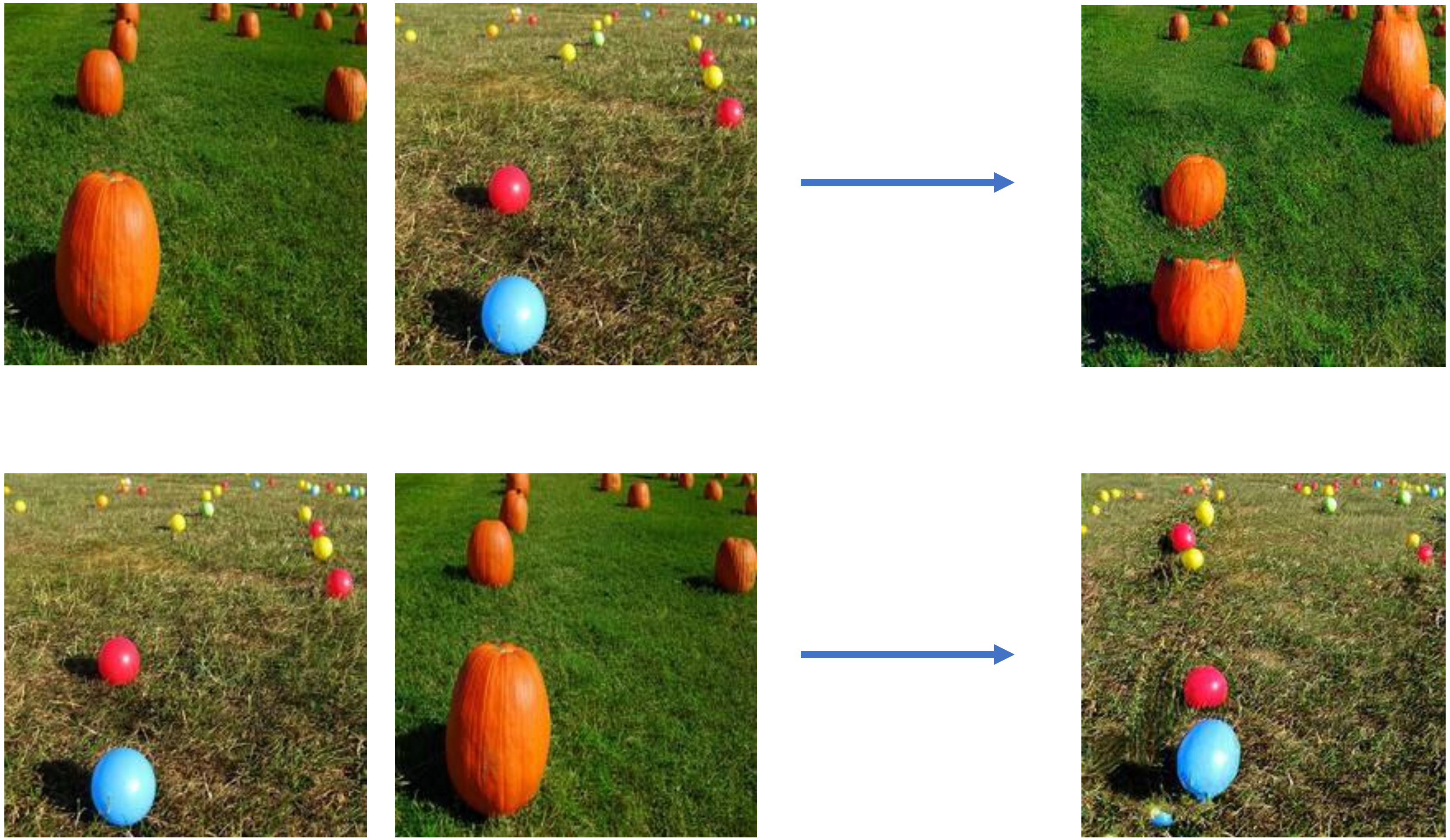


Structural Analogy

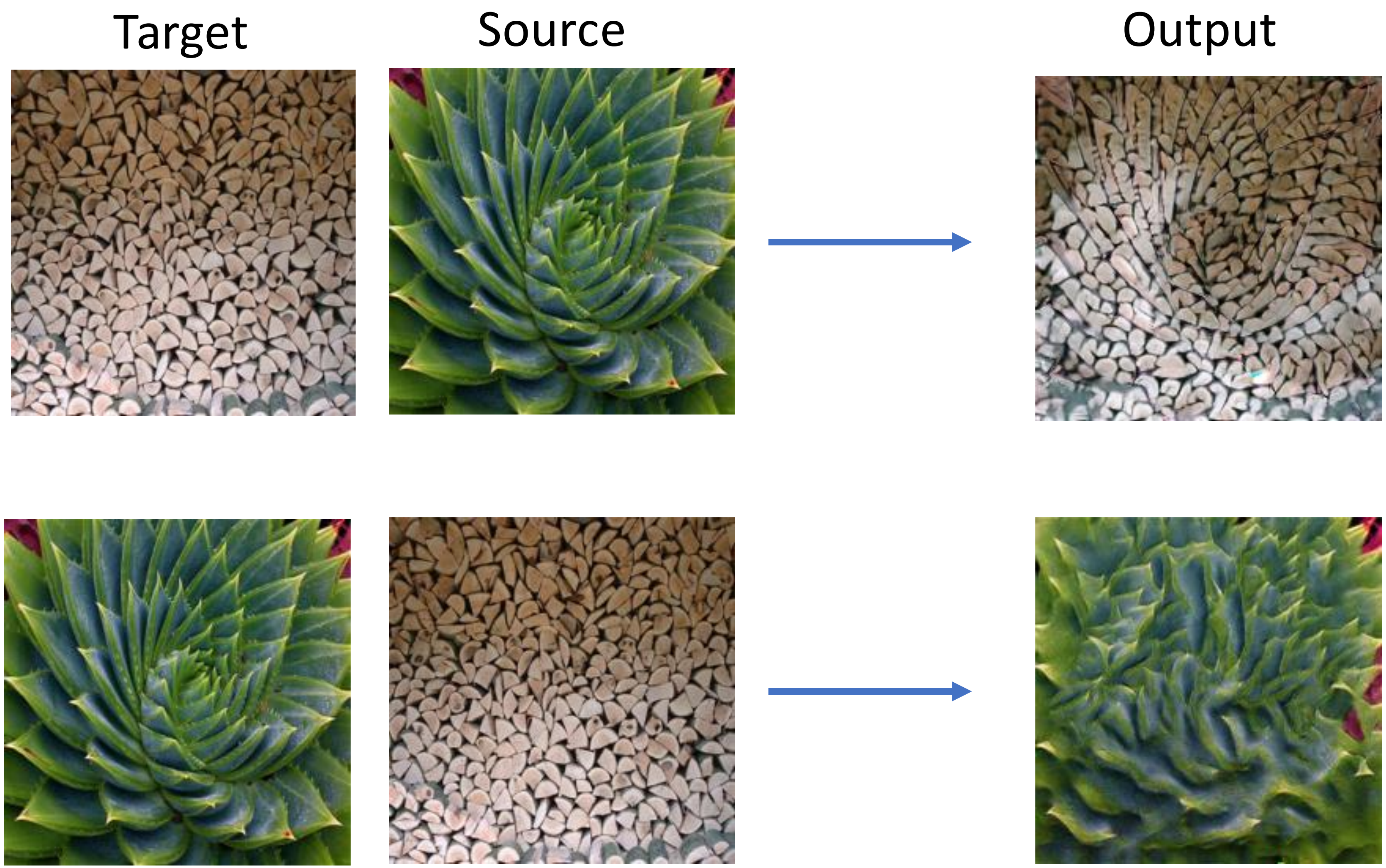


Structural Analogy

Target | Source | Output



Structural Analogy



Style Transfer

Style

Content

Result

Deep Image Analogy

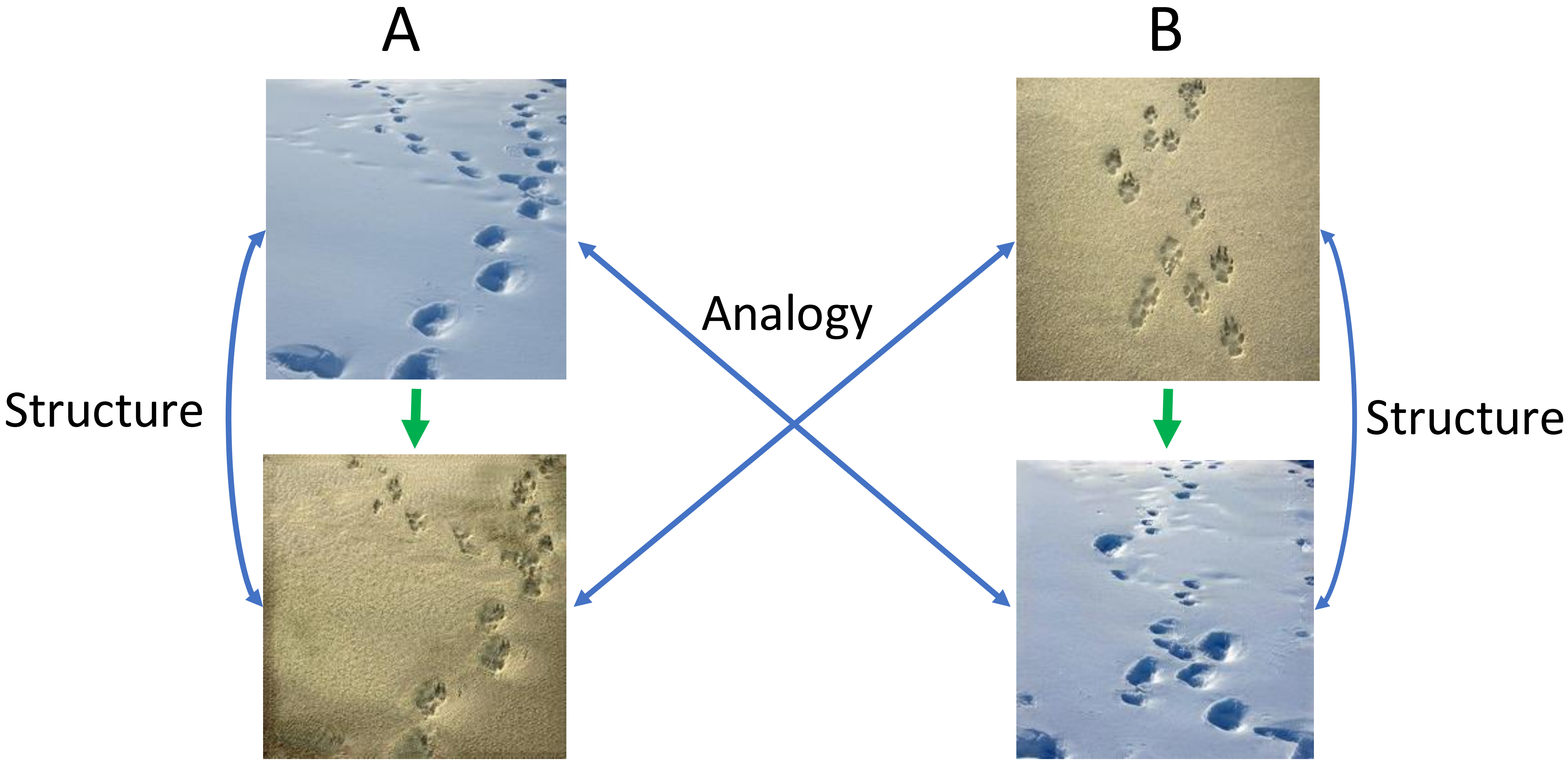
Style

Content

Result

Cannot Change Object Shape

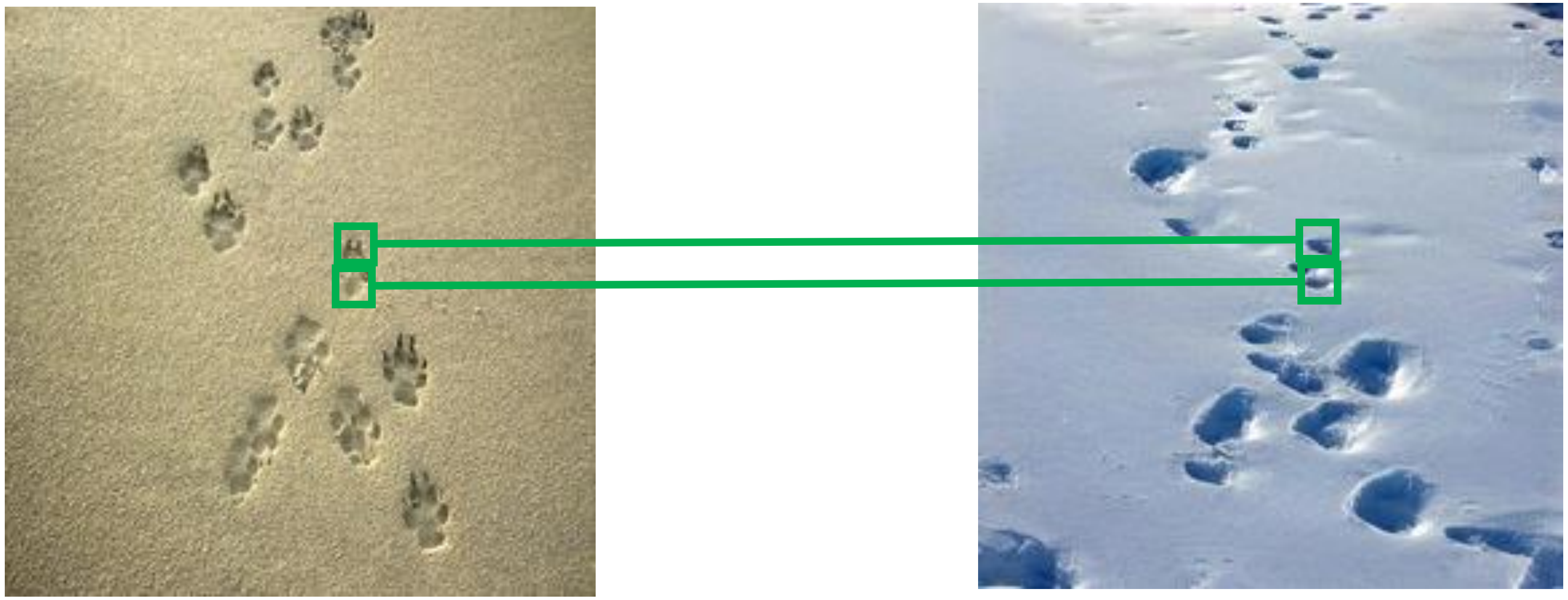
Structural Analogy



Motivation

Motivation

Motivation



Proposed Hierarchical Approach

Coarsest scale:
Large Patches

Finest scale:
Small Patches

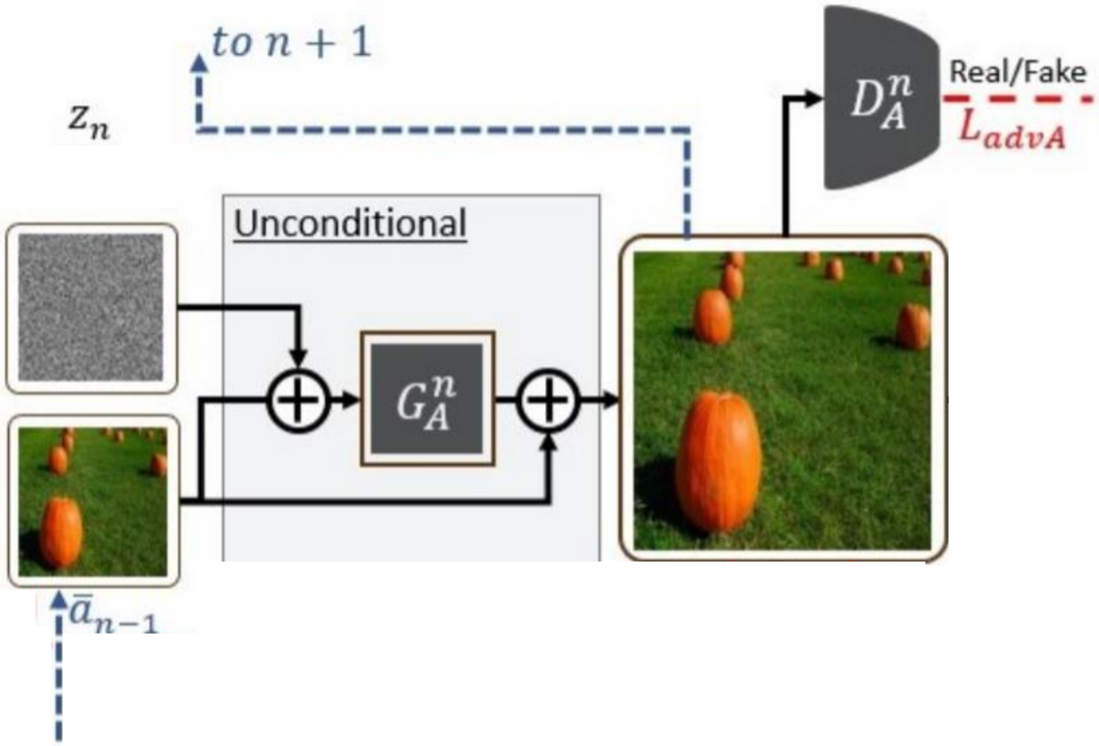
$\bar{a}^0$(Unconditional)
$\overline{ab}^0$(Conditional)

$\bar{a}^N$(Unconditional)
$\overline{ab}^N$(Conditional)

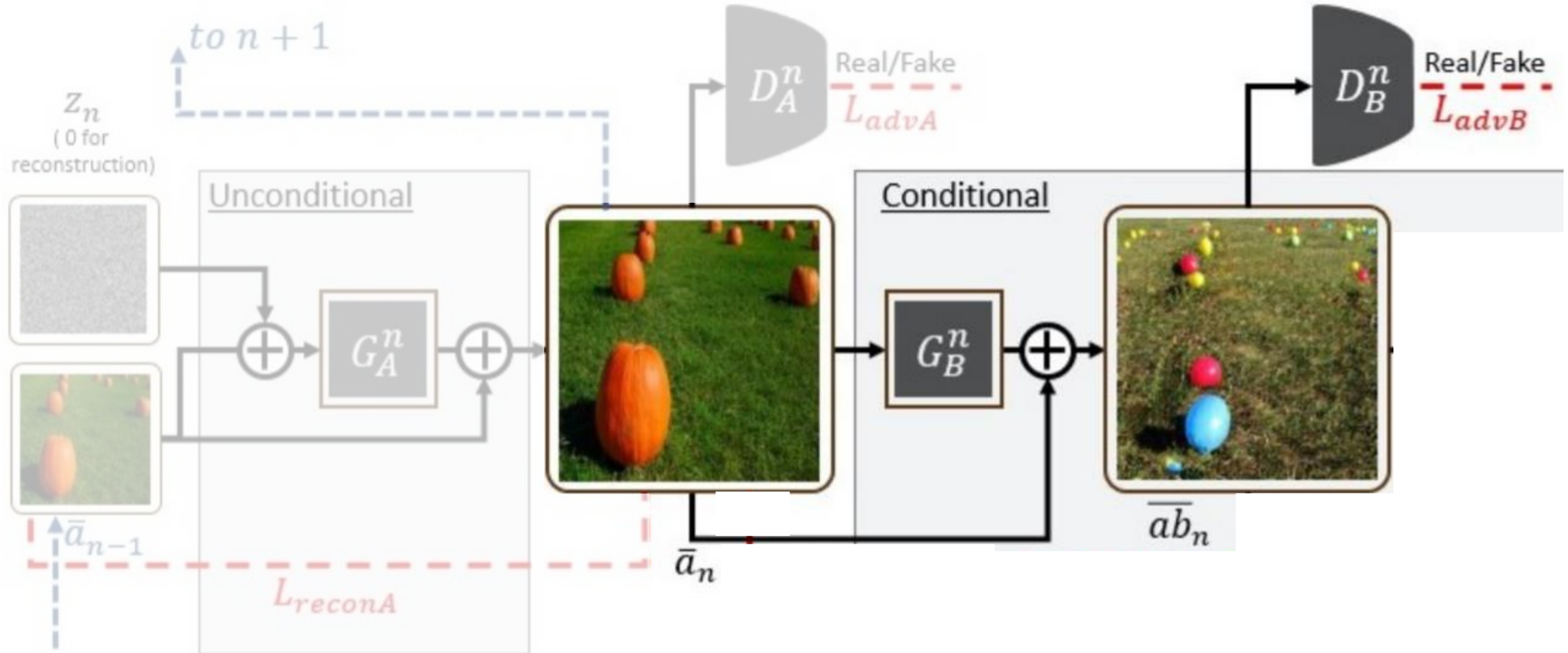
LEVEL $= 0$

LEVEL $= N$

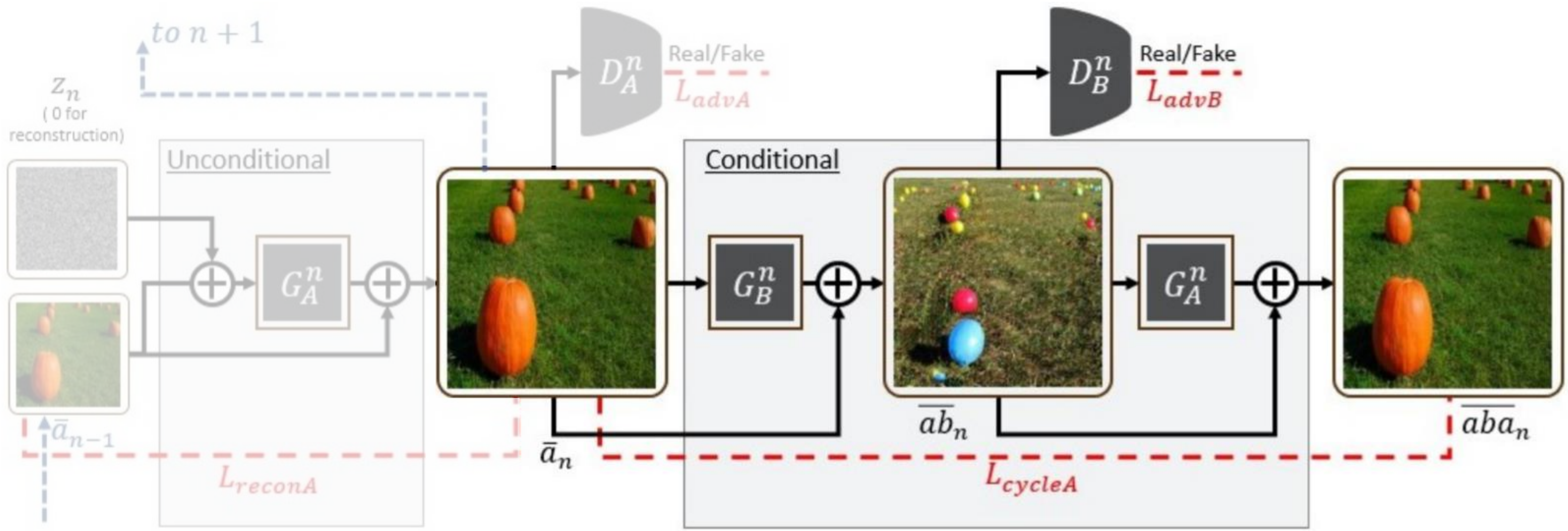
Unconditional Generation



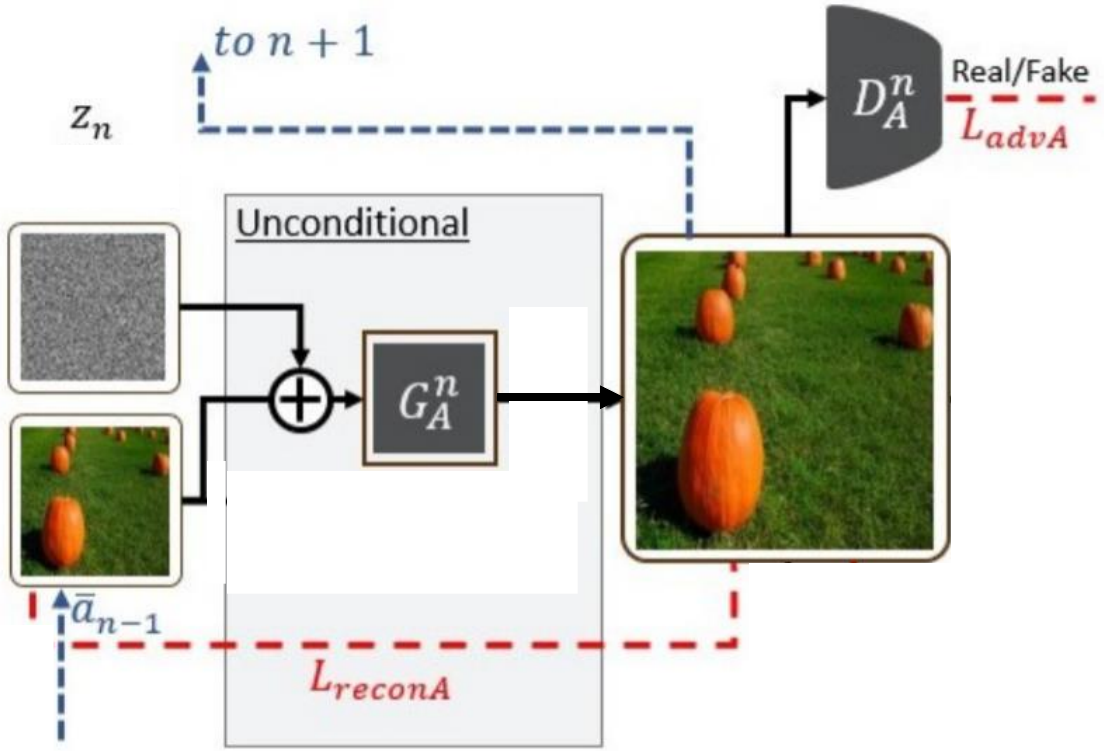
Conditional Generation



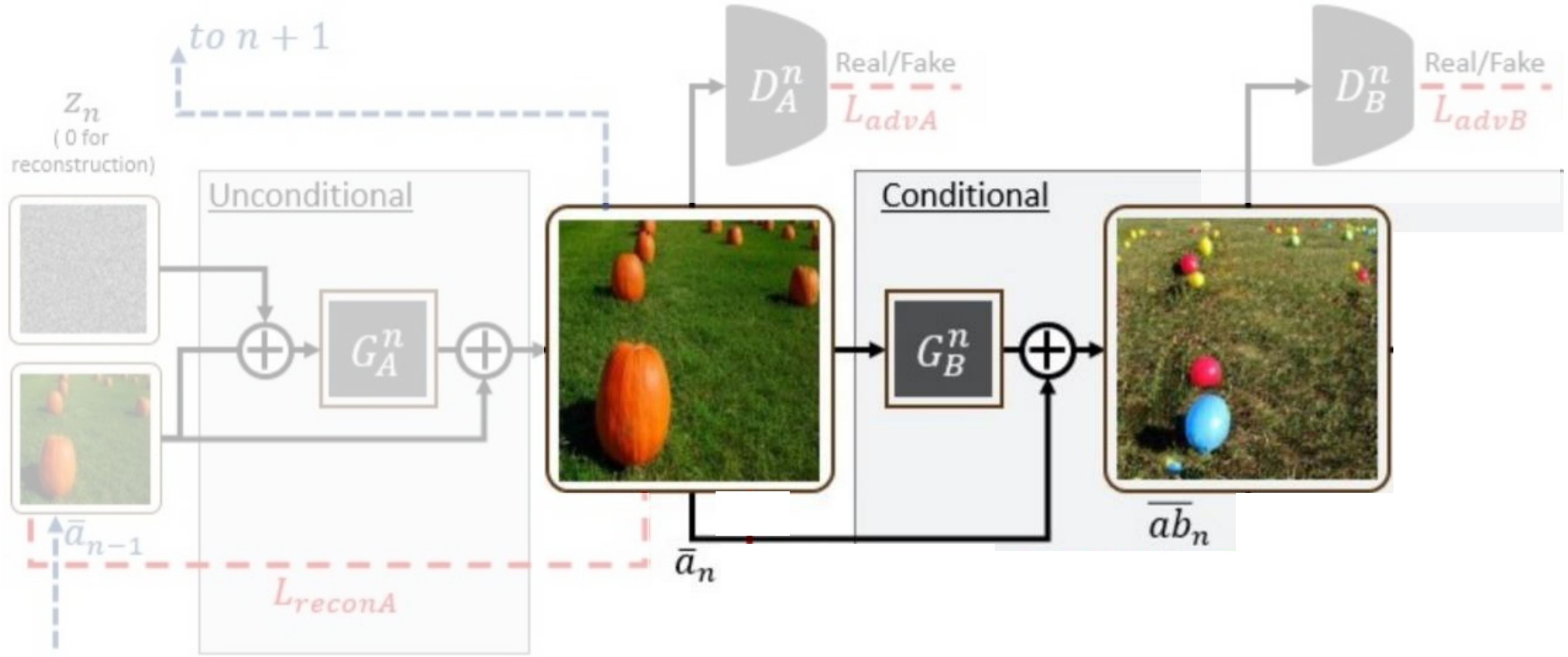
Conditional Generation



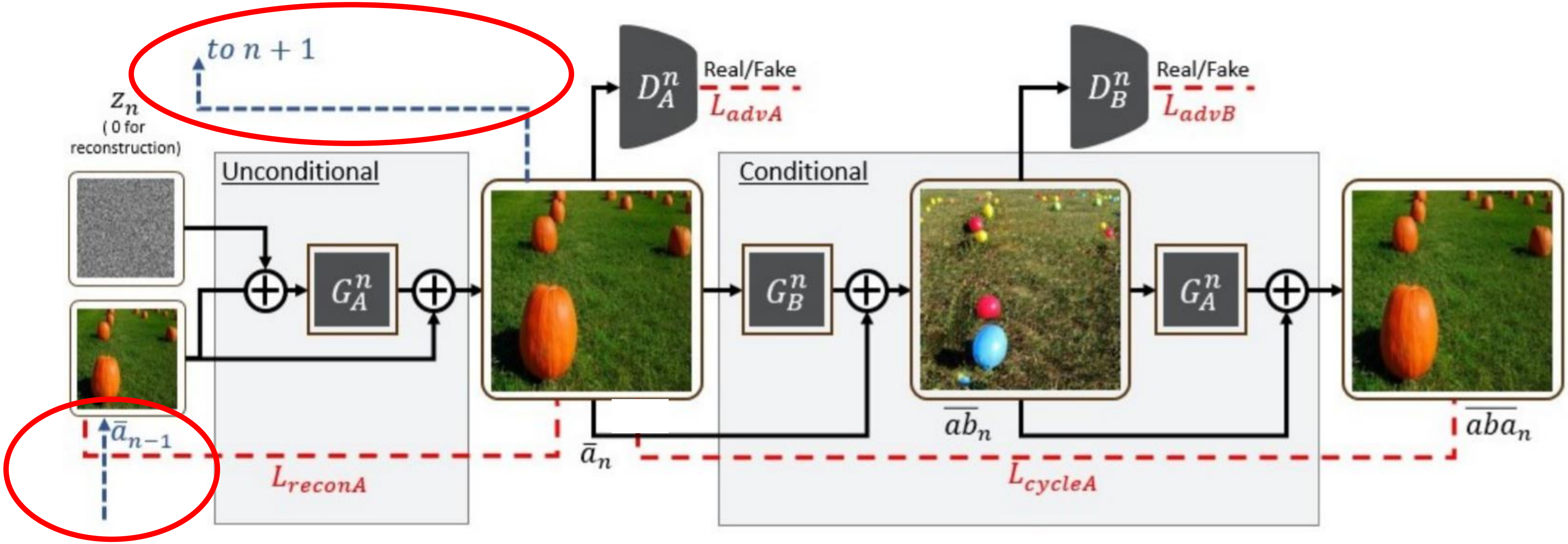
Coarse and Mid Scales: Residual Training

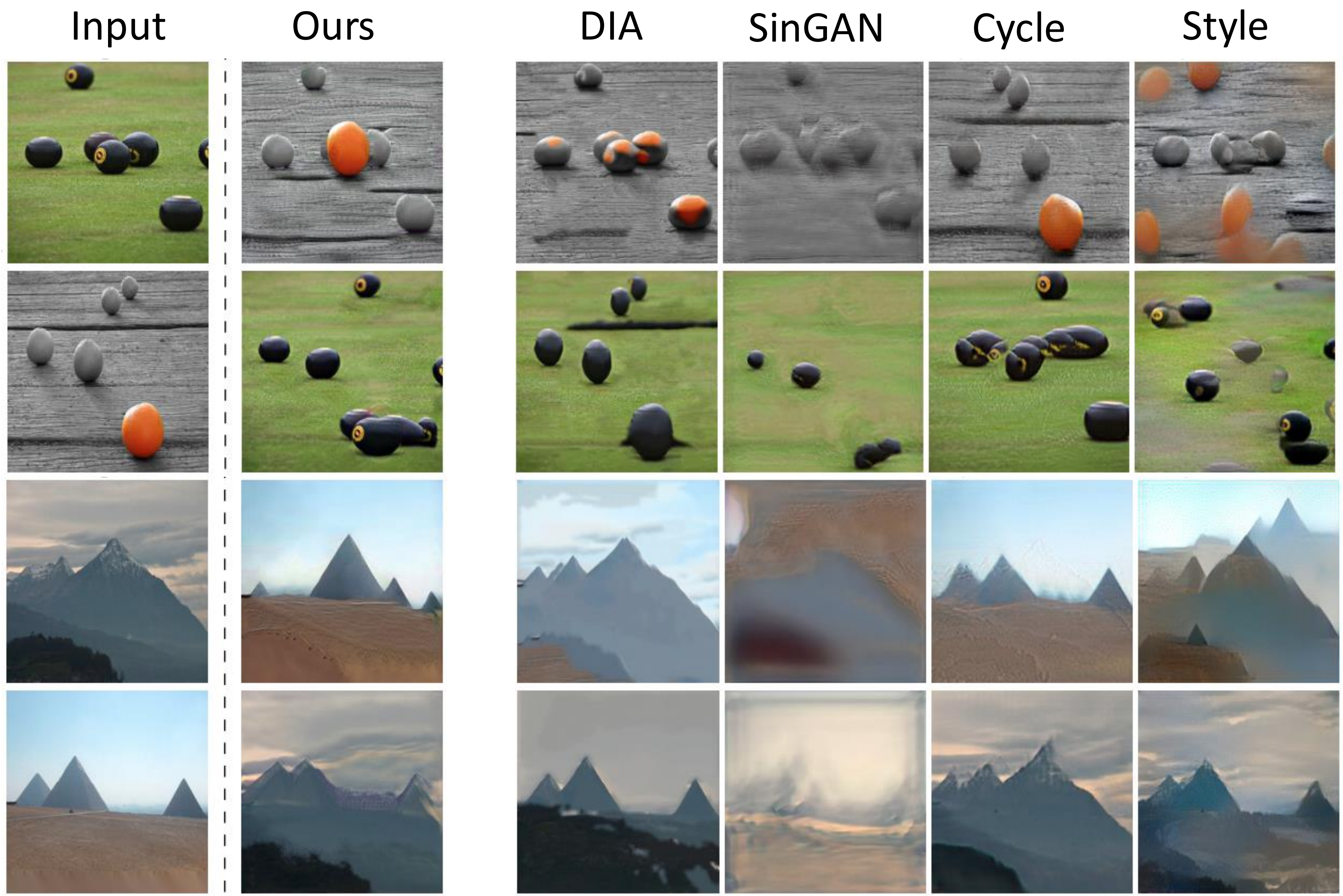


Coarse and Mid Scales: Residual Training



Indirect Interaction Between Scales

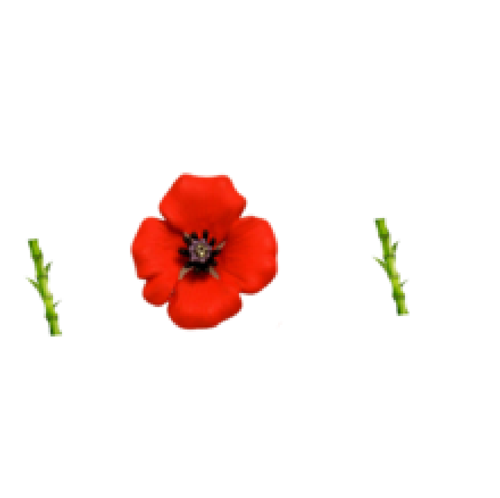



Input
Ours
DIA
SinGAN
Cycle
Style

Multiple Class Types

Input

Output

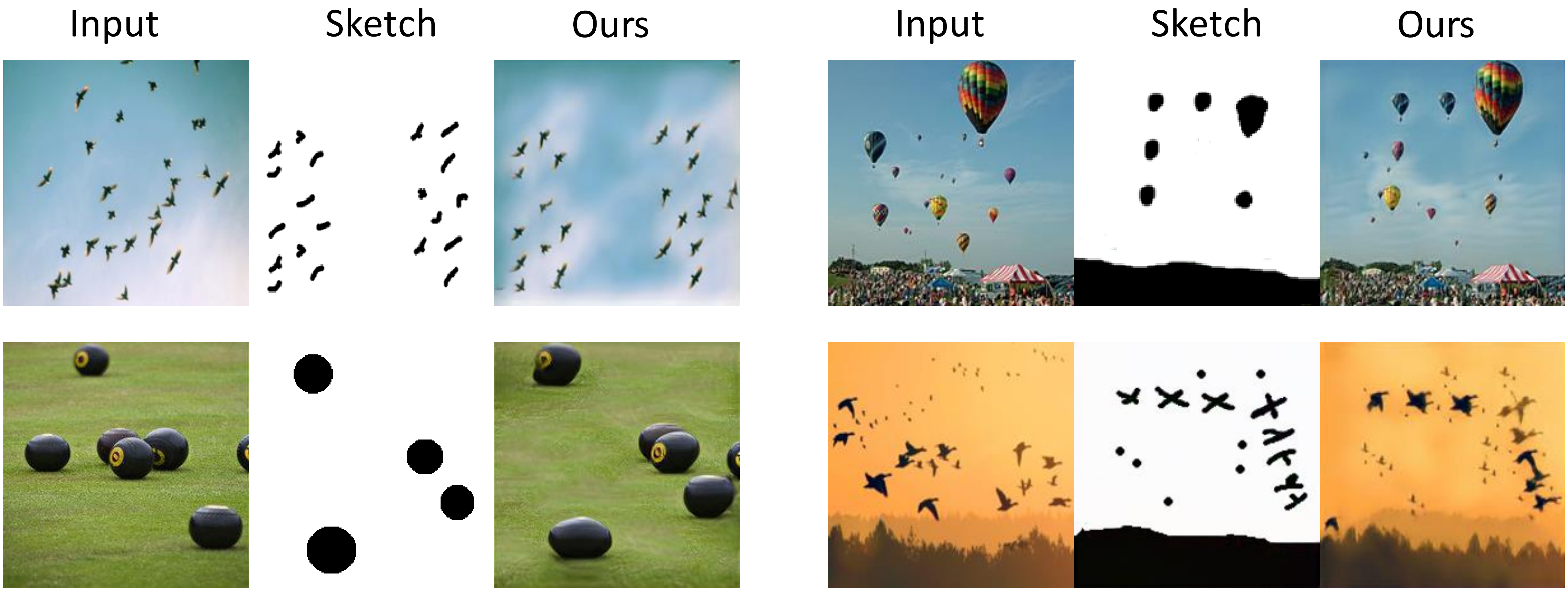


Paired Generation

A ~~Un~~conditional

B ~~Un~~conditional

Paint to Image

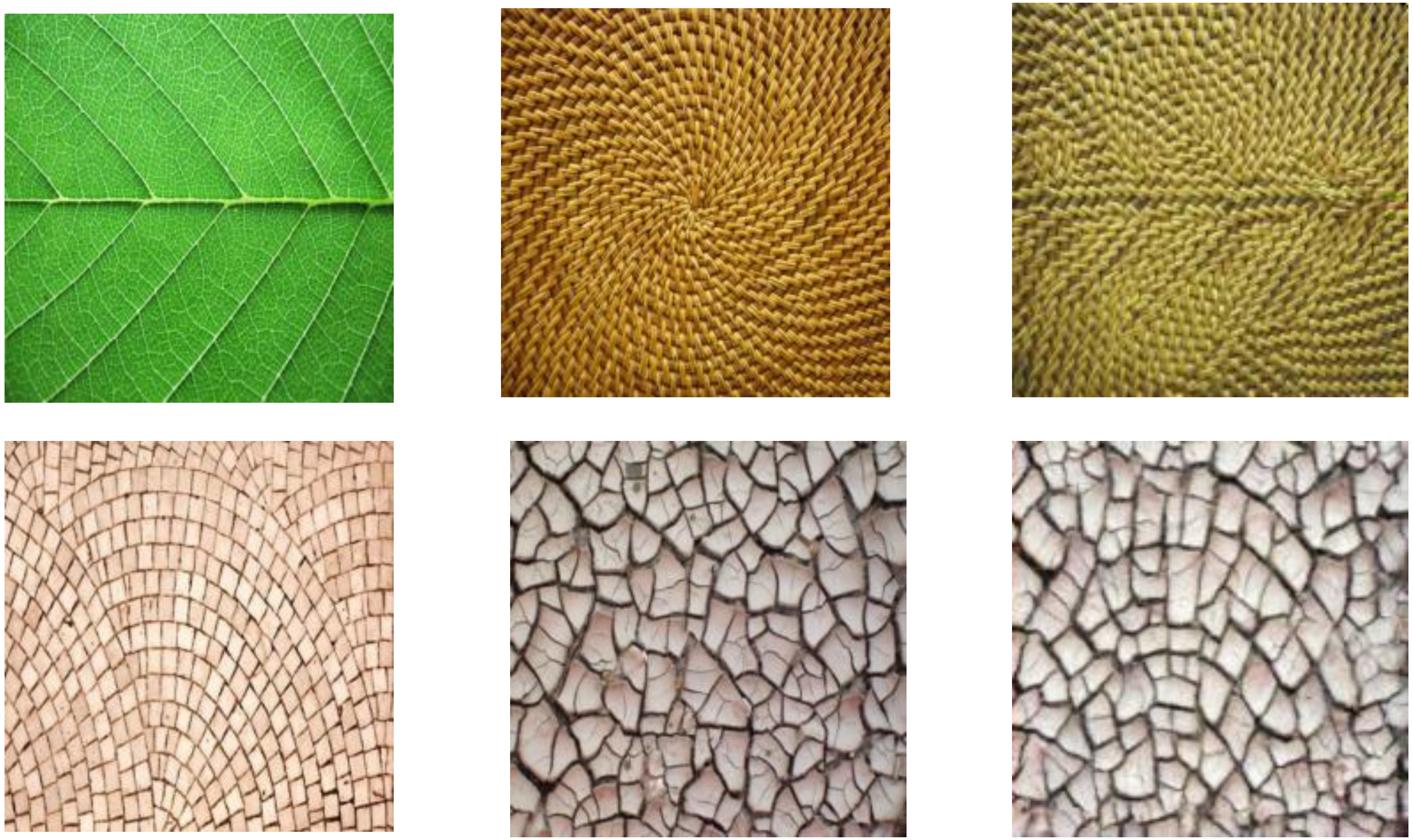


Text Transfer

Content | **Style** | **Ours**

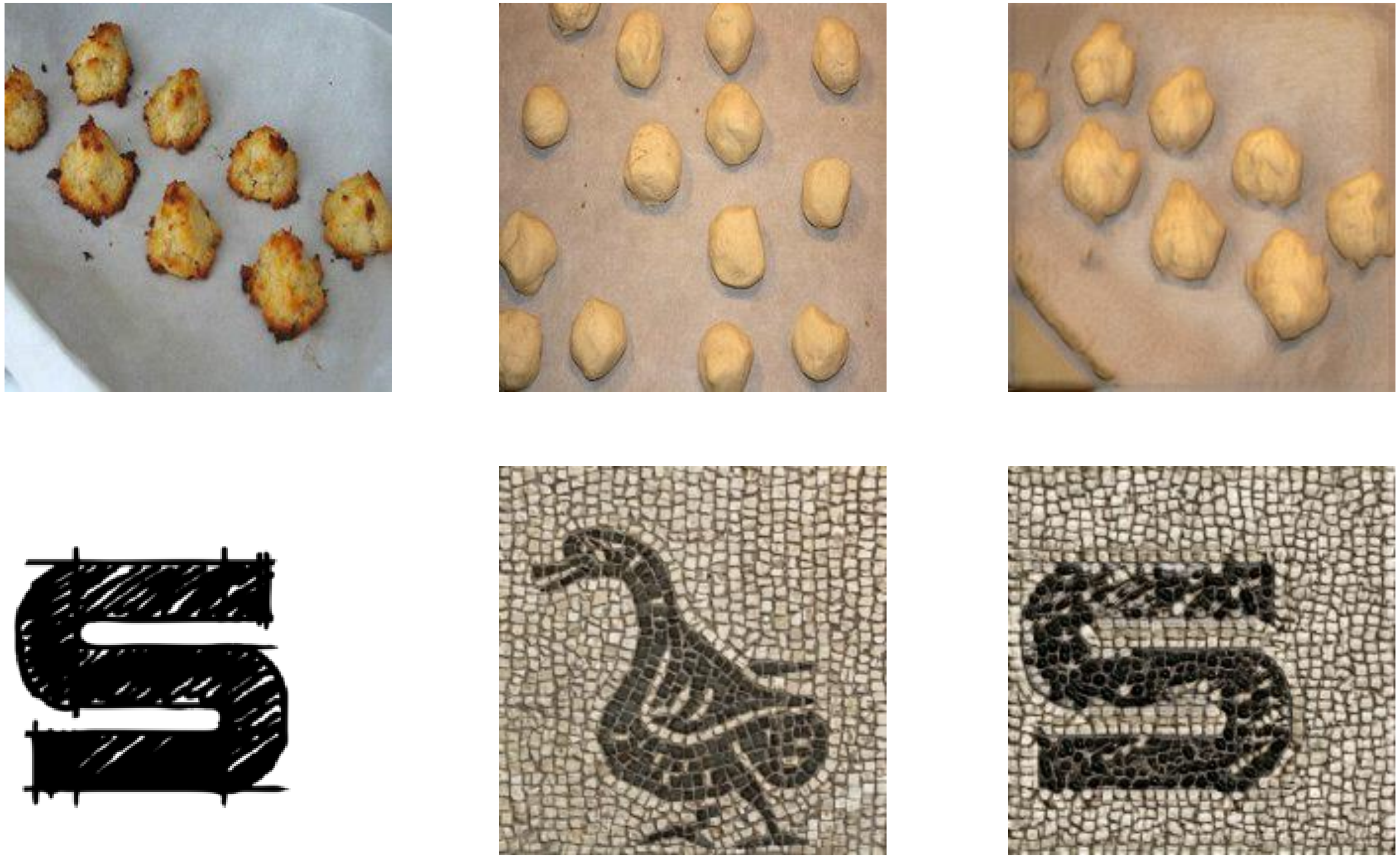
Texture Transfer

Content **Texture** **Ours**



Style Transfer

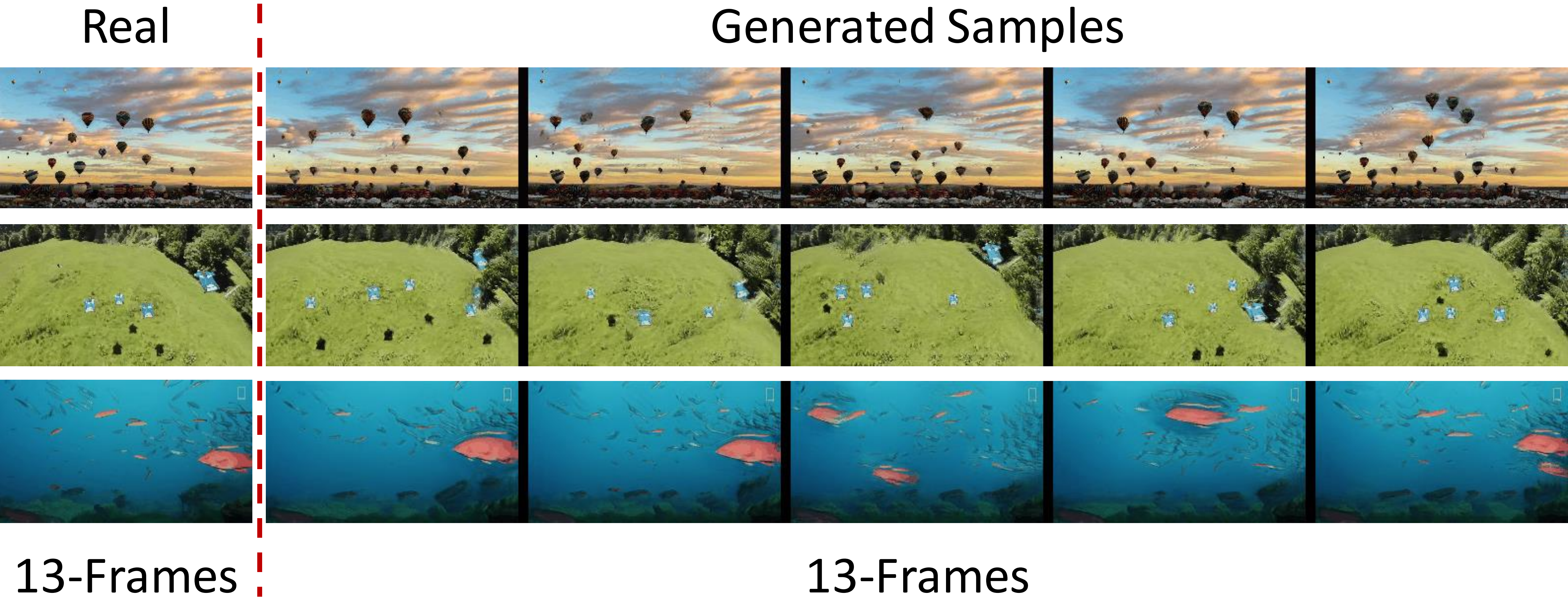
Content **Style** **Ours**



Video Generation

Hierarchical Patch VAE-GAN:
Generating Diverse Videos from a Single Sample

S. Gur*, S. Benaim*, L. Wolf. NeurIPS 2020 (*Equal contribution)



Extending 2D to 3D

Real

Ours

Real

SinGAN [1] + 3D Convolution

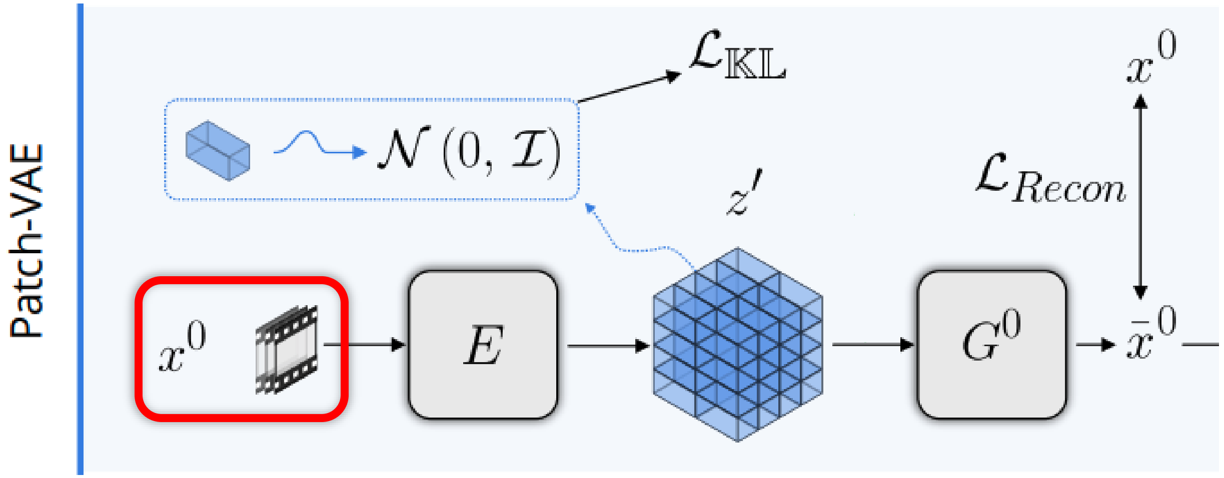
Real

ConSinGAN [2] + 3D Convolution

[1] “SinGAN: Learning a Generative Model from a Single Natural Image”, Shaham et al., ICCV 2019
[2] “Improved Techniques for Training Single-Image GANs”, Hinz et al., arXiv 2020

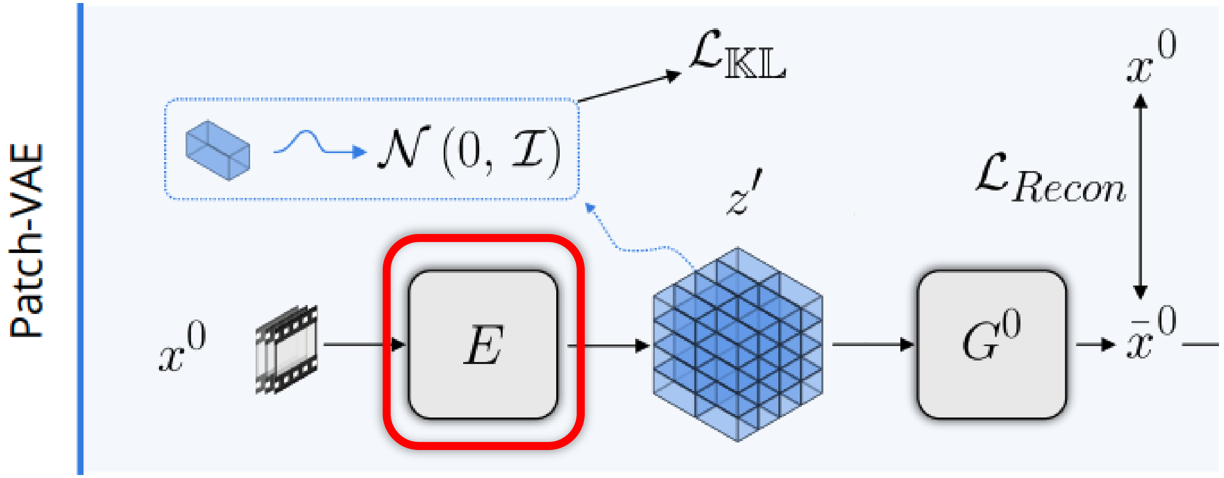
Proposed Approach: Patch VAE

Input video - x^0



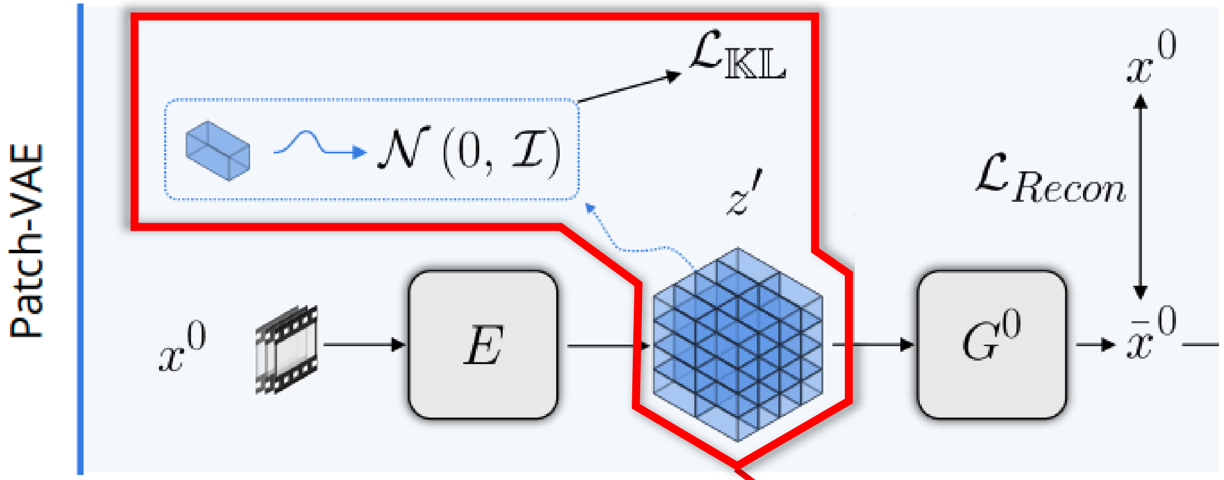
Proposed Approach: Patch VAE

Encoder – $\mathrm{E}(x^0)$

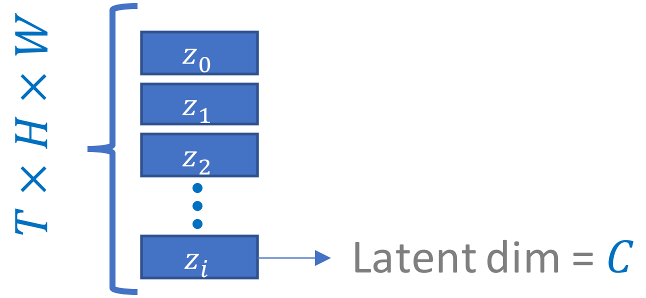


Proposed Approach: Patch VAE

KL Loss – z'

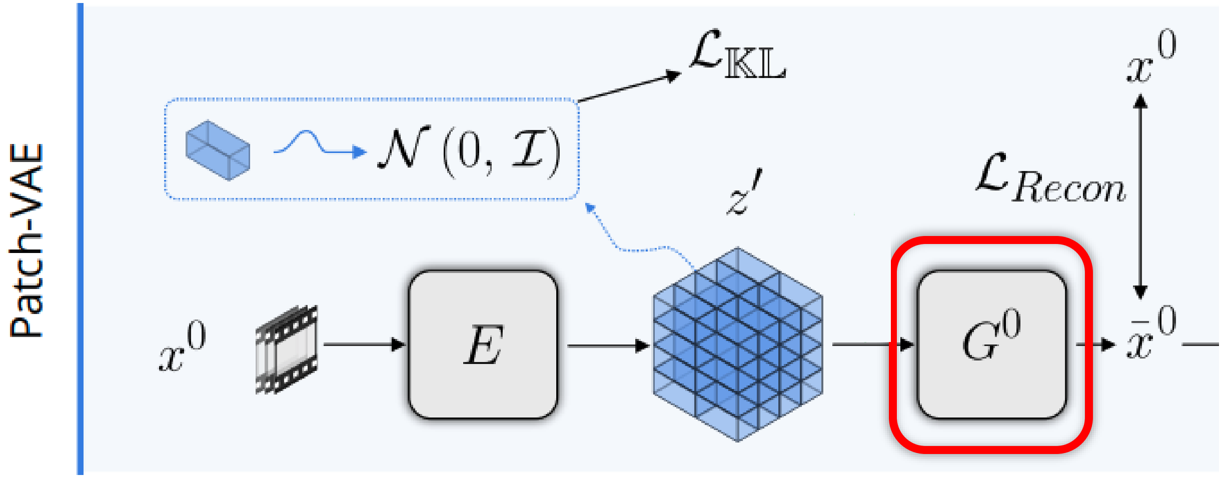


Each feature z_i , $i = [1 \ldots K], K = T \times H \times W$, in the latent space is associated with a patch ω_i



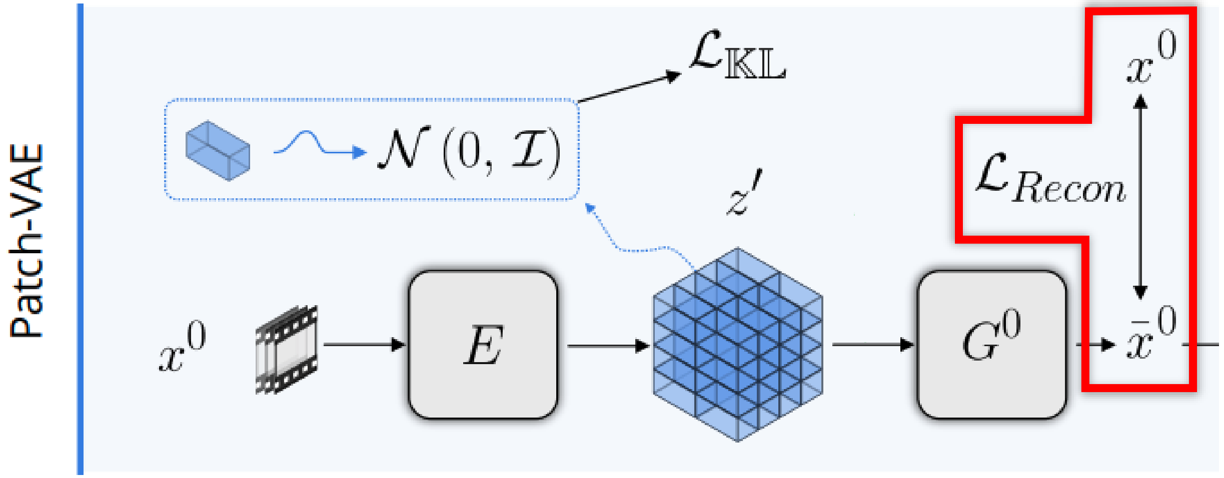
Proposed Approach: Patch VAE

Decoder - $\bar{x}^0$



Proposed Approach: Patch VAE

Reconstruction loss



Proposed Approach: Hierarchical Patch VAE

Coarsest scale:
Low resolution
and frame rate

x^0 (Real)
$\bar{x}^0$ (Generated)

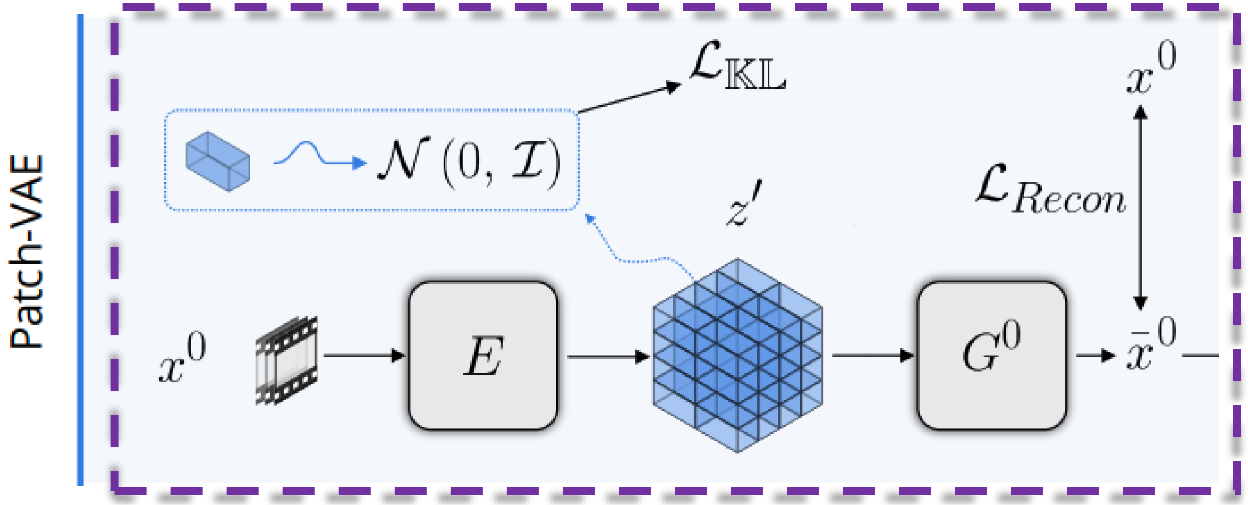
LEVEL $= 0$

Finest scale:
High resolution
and frame rate

x^N (Real)
$\bar{x}^N$ (Generated)

LEVEL $= N$

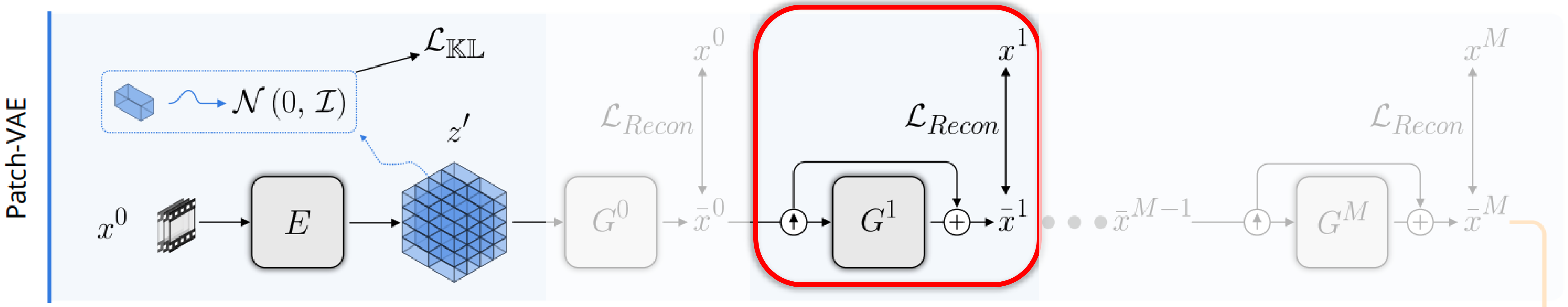
Proposed Approach: Hierarchical Patch VAE



LEVEL $= 0$

Proposed Approach: Hierarchical Patch VAE

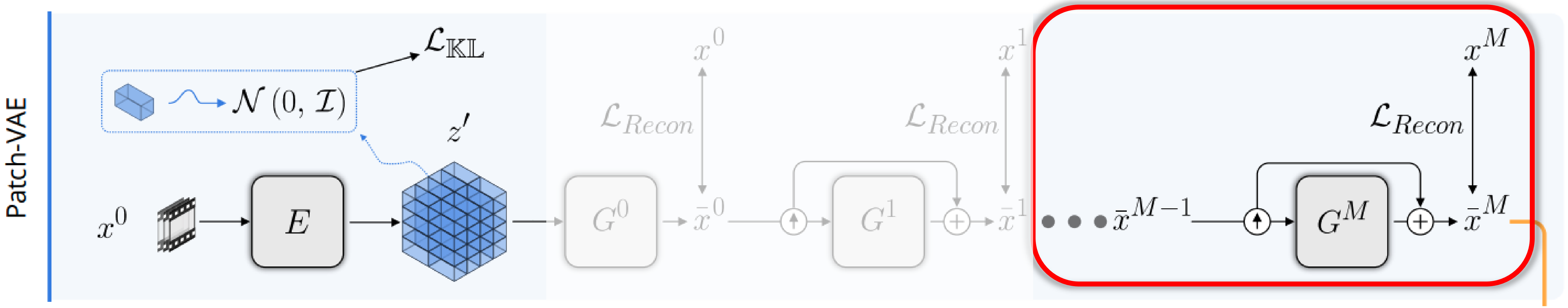
Up-sampling block - $\bar{x}^1$



LEVEL = 1

Proposed Approach: Hierarchical Patch VAE

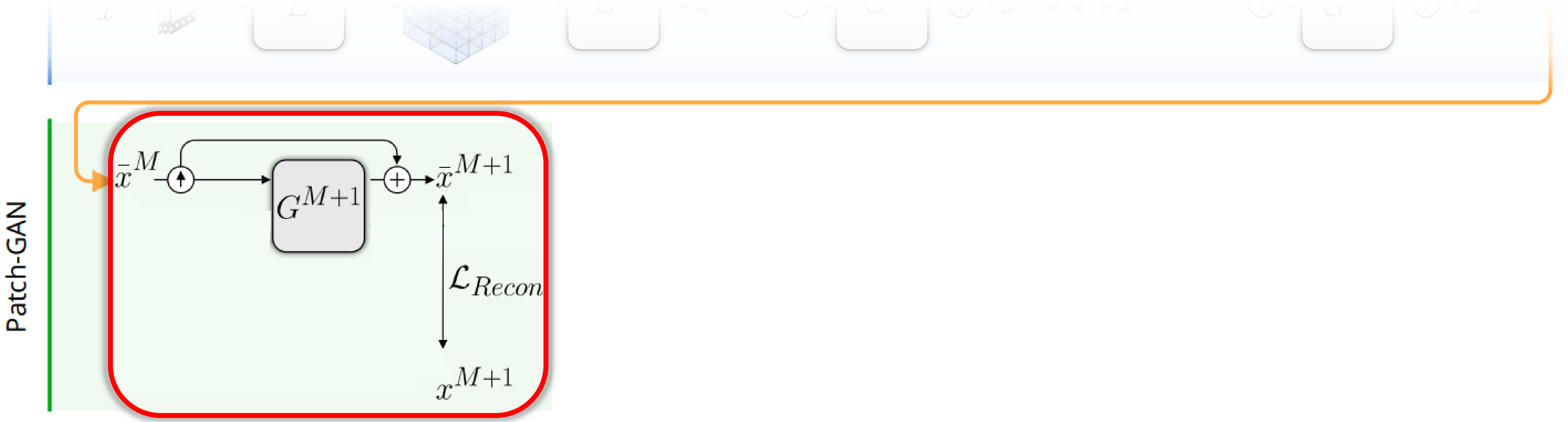
Hierarchical up-sampling up to $\bar{x}^M$



LEVEL $\leq M$

Proposed Approach: Hierarchical Patch VAE GAN

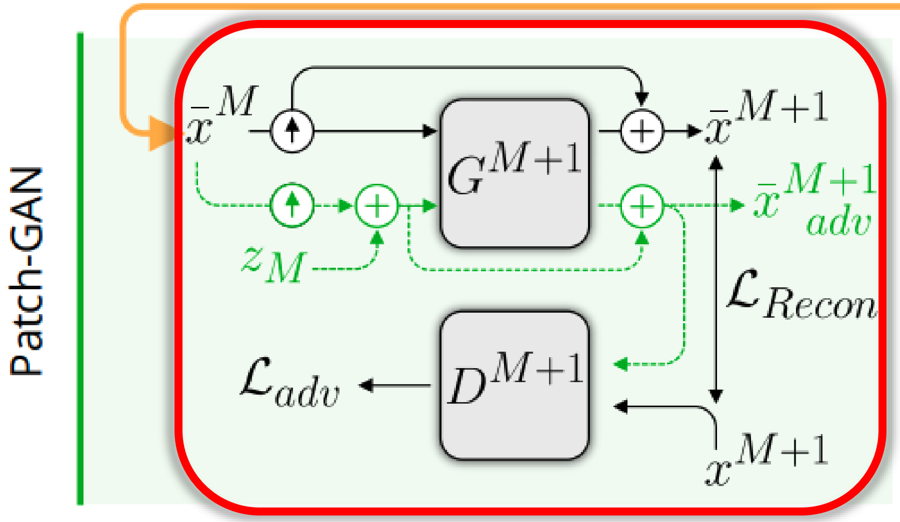
Up-sampling block $\bar{x}^{M+1}$



LEVEL $= M + 1$

Proposed Approach: Hierarchical Patch VAE GAN

Adversarial training

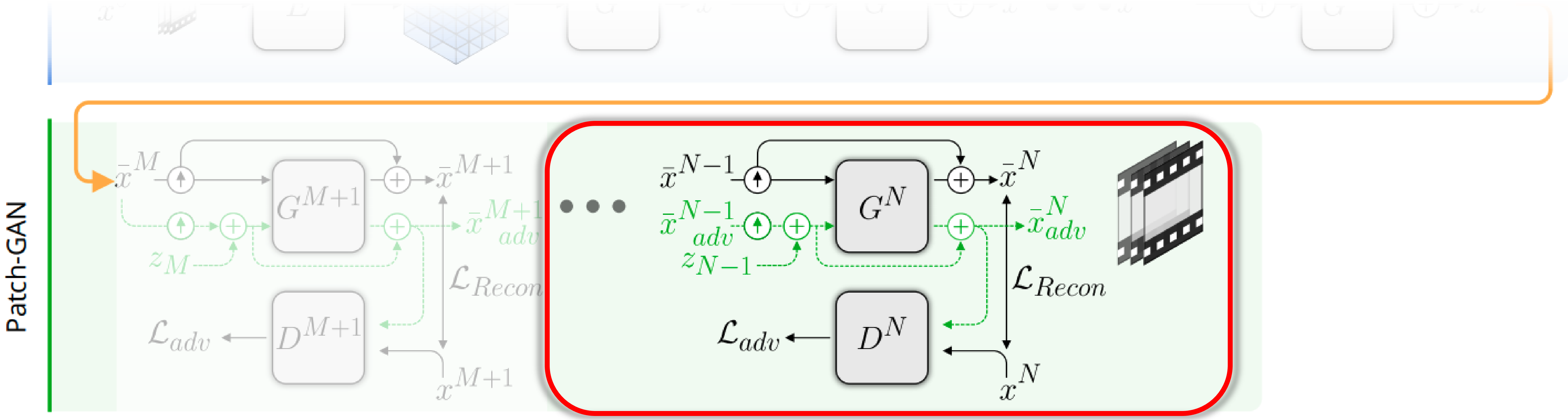


Added noise z_M

LEVEL $= M + 1$

Proposed Approach: Hierarchical Patch VAE GAN

Hierarchical up-sampling up to final resolution $\bar{x}^N$



$M + 1 <$ LEVEL $\leq N$

Effect of Number of patch-VAE levels

Training Video

9 Levels Total

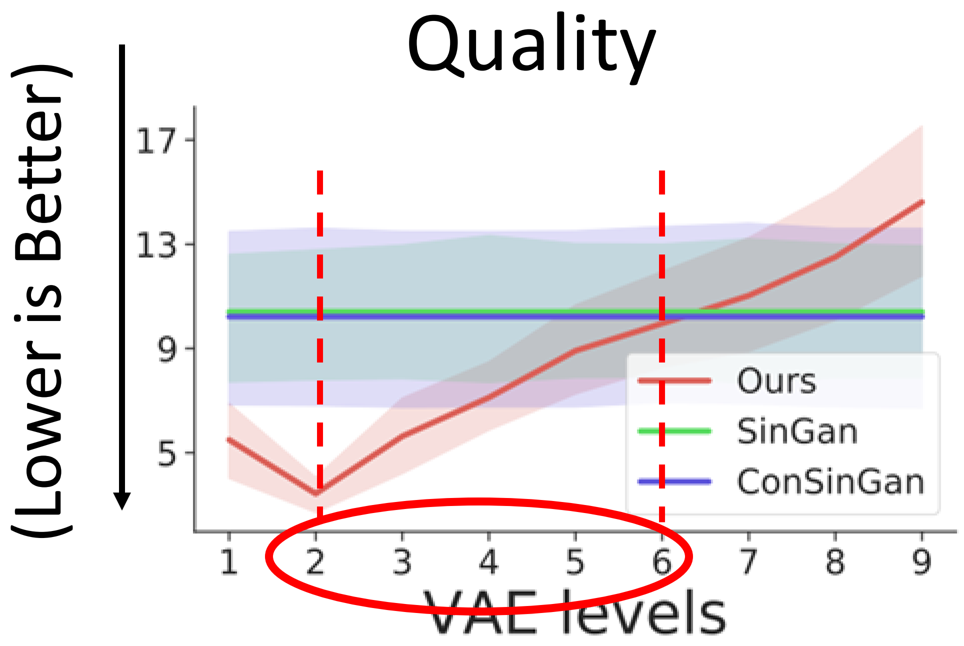
1 p-VAE – 8 p-GAN

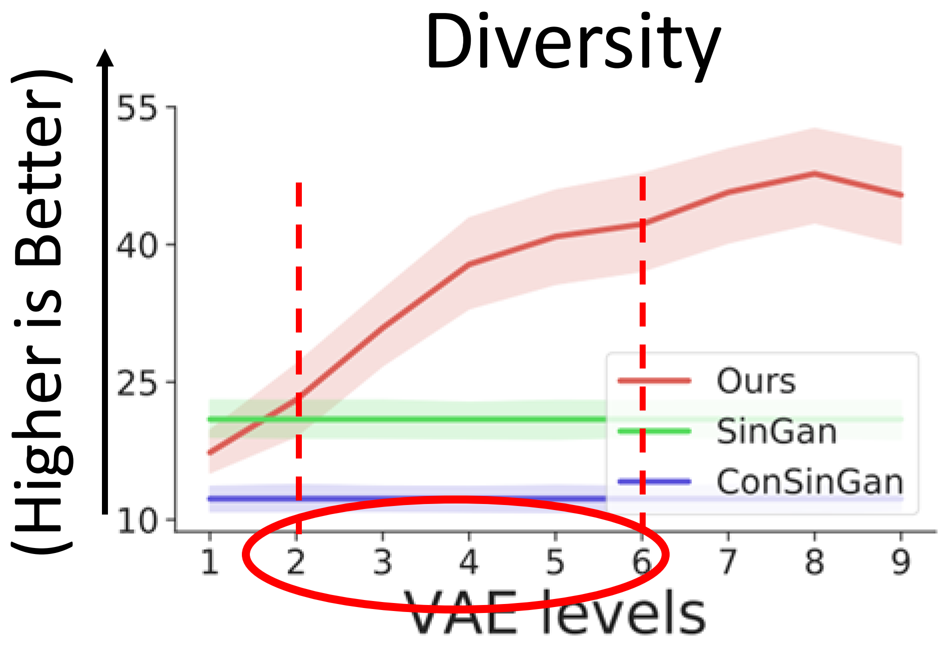
8 p-VAE – 1 p-GAN

3 p-VAE – 6 p-GAN

Effect of Number of patch-VAE levels

Total of 9 layers





Conclusion (Disentanglement)

- Supervision level: supervised, semi-supervised and unsupervised.
- Semi-supervised generation -> good representation for downstream tasks.
- Unsupervised disentanglement of “global” statistics vs content using permuted AdaIN (applied on top of every convolutional layer) -> good for domain adaptation and many image classification tasks.
- Next: “semi-supervised” and “unsupervised” disentanglement for more complex tasks: e.g decompose illumination from a scene or decompose time-dependent from static factors in video.

Conclusion (Few shot generation)

- Image to Image Translation:
  - Weight sharing (shared latent space assumption)
  - Transformations (strong inductive bias)
  - Matching patches (dense similarity measure)
  - Next: Few shot image understanding: anomaly detection, retrieval?

- Video generation:
  - Patch VAE for coarse scales (large variety) and Patch GAN for fine scales (high fidelity)
  - Next: Temporal super resolution, temporal inpainting, etc

Papers (In order of appearance)

- **S. Benaim**, M. Khaitov, T. Galanti, L. Wolf. Domain Intersection and Domain Difference. In **ICCV, 2019.**
- R. Mokady, **S. Benaim**, L. Wolf, A. Bermano. Mask Based Unsupervised Content Transfer. In **ICLR, 2020.**
- O. Nuriel, **S. Benaim**, L. Wolf. Permuted AdaIN: Reducing the Bias Towards Global Statistics in Image Classification. ArXiv, 2020 (In submission to CVPR 2021).
- **S. Benaim**, L. Wolf. One-Shot Unsupervised Cross Domain Translation. In **NeurIPS, 2018.**
- **S. Benaim***, R. Mokady*, A. Bermano, D. Cohen-Or, Lior Wolf. Structural-analogy from a Single Image Pair. In **Computer Graphics Forum, 2020**.
- S. Gur*, **S. Benaim***, Lior Wolf. Hierarchical Patch VAE-GAN: Generating Diverse Videos from a Single Sample. In **NeurIPS, 2020.**

Thank You! Questions?

Unsupervised Domain Adaptation

GTVA to Cityscapes

Generalization

Method	Road	SW	Build	Wall	Fence	Pole	TL	TS	Veg.	Terrain	Sky	PR	Rider	Car	Truck	Bus	Train	Motor	Bike	mIoU
Source only	57.9	17.4	71.5	19.3	18.3	25.39	32.5	16.8	82.3	28.2	78.0	55.3	31.3	71.6	19.1	26.8	9.2	26.3	13.7	37.0
Source only + pAdaIN	57.2	20.2	71.6	28.3	19.1	26.1	33.6	13.0	82.1	29.0	69.5	56.7	33.0	67.5	27.8	35.1	**17.6**	33.7	14.5	38.7

Domain Adaptation

SVHN to MNIST

Selective Adaptation

Adapt All Layers

Selective Adaptation

Adapt All Layers

Domain Adaptation

Domain B (Target)

No Labels

Domain A (Source)

With Labels

Unsupervised Domain Adaptation

Domain B (Target)

No Labels

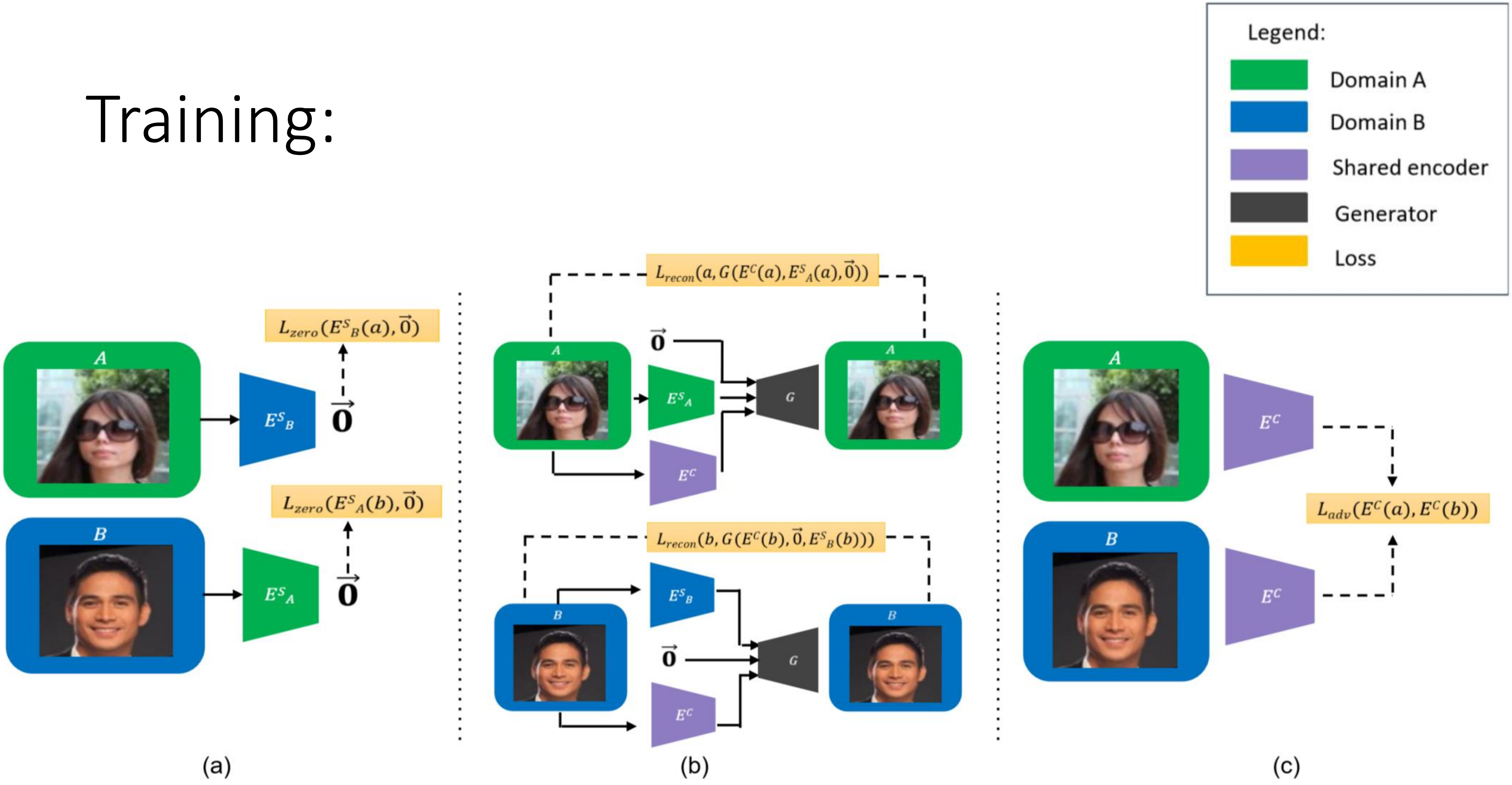
Domain A (Source)

No Labels

Unsupervised Domain Adaptation

- Given an MNIST digit a, we randomly sample an SVHN digit b and consider the translation to SVHN as $G(E_c(a), 0, E_A^s(b))$.
- Marginalize over samples in b.
- Achieve **SOTA:** MNIST to SVHN: 61.0%, Reverse: 41.0%

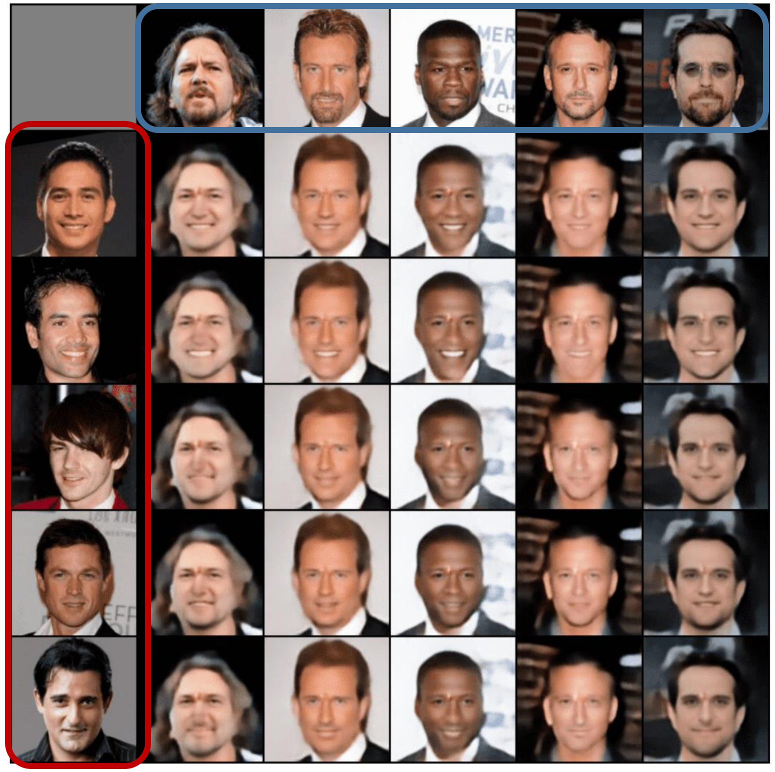
Training:



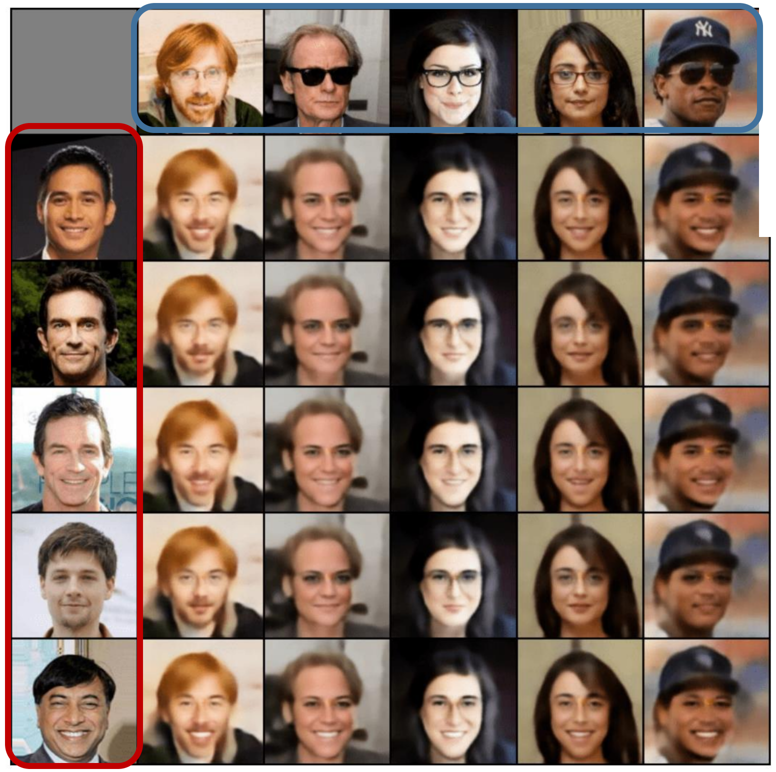
Legend:
Domain A
Domain B
Shared encoder
Generator
Loss
$L_{zero}(E^S{}_B(a), \vec{0})$
$L_{zero}(E^S{}_A(b), \vec{0})$
$L_{recon}(a, G(E^C(a), E^S{}_A(a), \vec{0}))$
$L_{recon}(b, G(E^C(b), \vec{0}, E^S{}_B(b)))$
$L_{adv}(E^C(a), E^C(b))$
A
B
$E^S{}_B$
$E^S{}_A$
E^C
G
$\vec{0}$
(a)
(b)
(c)

Results

Beard to Smile

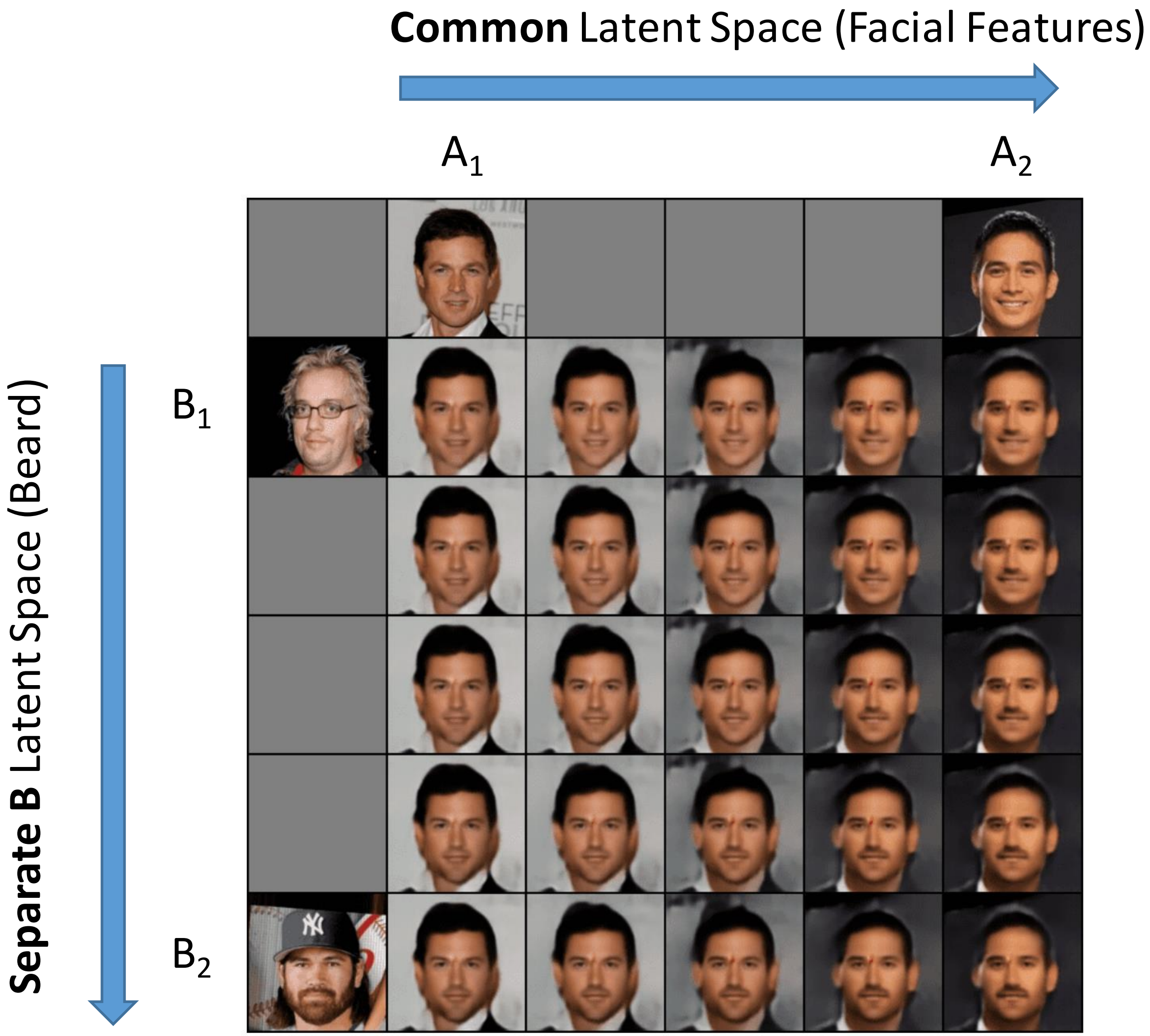


Glasses to Smile

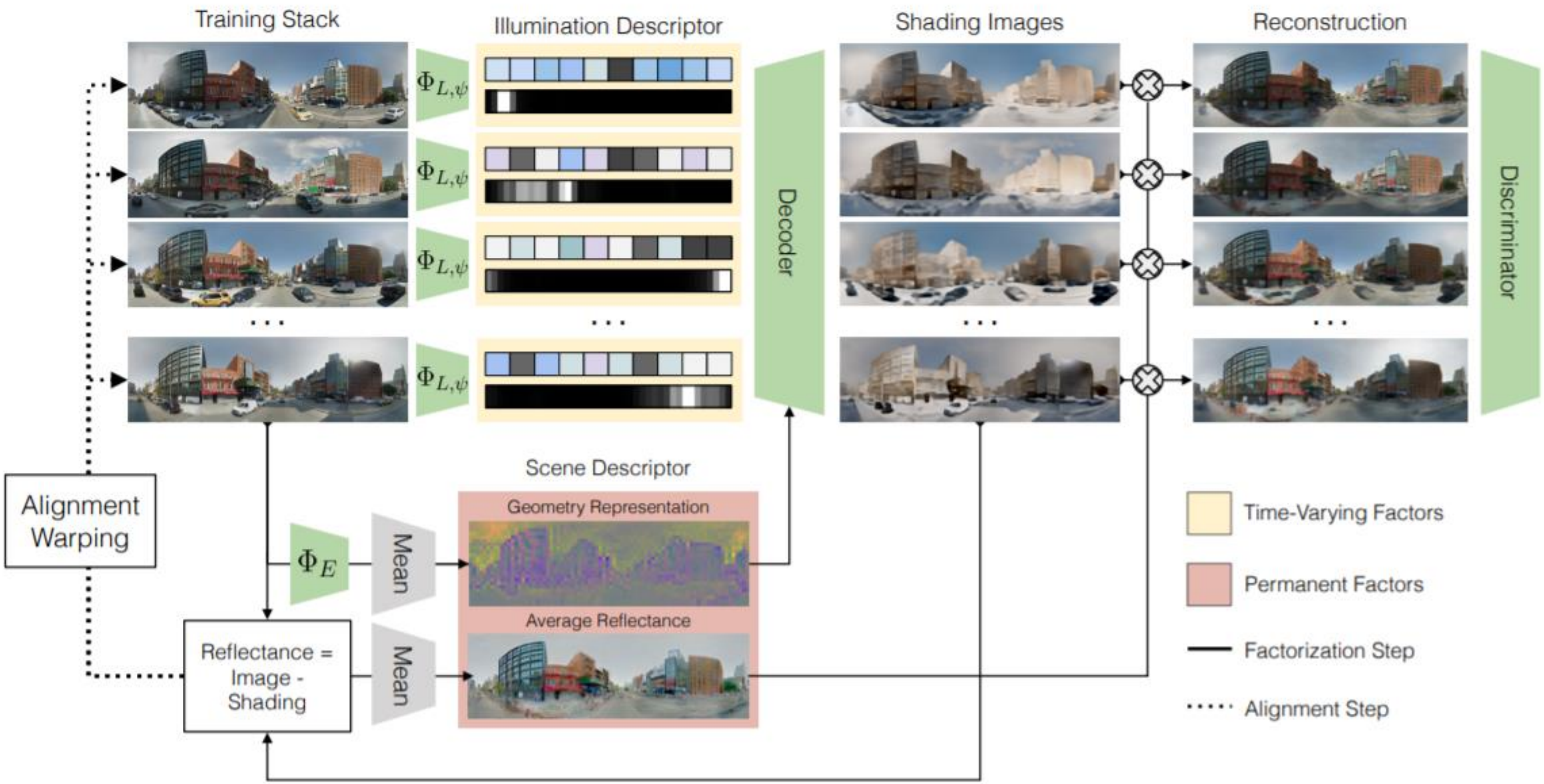


Glasses ∩ Smile

Interpolations



Fully supervised (example)



Learning to Factorize and Relight a City. Liu et al., ECCV 2020

Numerical Results: Pretrained Classifier

	Smile To Glasses	Glasses To Smile	Facial Hair To Smile	Smile To Facial Hair	Facial Hair To Glasses	Glasses To Facial Hair
Fader networks [15]	76.8%	97.3%	95.4%	84.2%	77.8 %	85.2%
Guided content transfer [20]	45.8%	92.7%	85.6%	85.1%	38.6%	82.2%
MUNIT [12]	7.3%	9.2%	9.3%	8.4%	7.3%	8.5%
DRIT [16]	8.5%	6.3%	6.3%	10.3%	8.6%	10.1%
Ours	91.8%	99.3%	93.7%	87.1%	93.1%	97.2%

Table 1. We pretrain a classifier to distinguish between samples in A (e.g. images of persons with glasses) and samples in B (e.g. images of persons with smile). We then sample $a \in A$, $b \in B$ from the test samples and check the membership of the generated image $G(E^c(b), E^s_A(a), 0))$ in A. Similarly, in the reverse direction, we check the membership of $G(E^c(a), 0, E^s_B(b))$ in B.

Numerical Results: User Study

- Q1: Is the specific attribute of A (e.g smile) removed?
- Q2: Is the guided image b specific attribute (e.g glasses) added?
- Q3: Is the identify of a's image preserved?

	Smile To Glasses	Glasses To Smile	Facial Hair To Smile	Smile To Facial Hair	Facial Hair To Glasses	Glasses To Facial Hair
Question (1) ours	4.74 ±0.13	4.30 ±0.21	4.26 ±0.20	4.30 ±0.15	4.18 ±0.17	4.50 ±0.18
Question (2) ours	3.92 ±0.16	4.45 ±0.12	4.03 ±0.15	3.34 ±0.17	3.85 ±0.20	3.95 ±0.22
Question (3) ours	3.95 ±0.23	3.20 ±0.24	3.24 ±0.25	3.22 ±0.27	3.49 ±0.22	3.39 ±0.23
Question (1) for [20]	3.67 ±0.17	4.16 ±0.18	3.39 ±0.19	3.34 ±0.13	4.24 ±0.12	3.15 ±0.15
Question (2) for [20]	1.87 ±0.35	4.42 ±0.22	3.00 ±0.32	2.67 ±0.33	2.20 ±0.42	3.30 ±0.22
Question (3) for [20]	3.95 ±0.15	2.93 ±0.22	3.37 ±0.25	3.40 ±0.27	3.43 ±0.28	3.75 ±0.20

Table 2. Given 20 randomly selected images $a \in A$ and $b \in B$, we consider the generated image $G(E^c(a), 0, E^s_B(b)))$ and ask if (1) a's separate part is removed (2) b's separate part is added (3) a's common part is preserved (similarly in the reverse direction). Mean opinion scores in the range of 1 to 5 are reported, where higher is better.

Minimality

- Potentially Infinitely many solutions preserving distance correlations

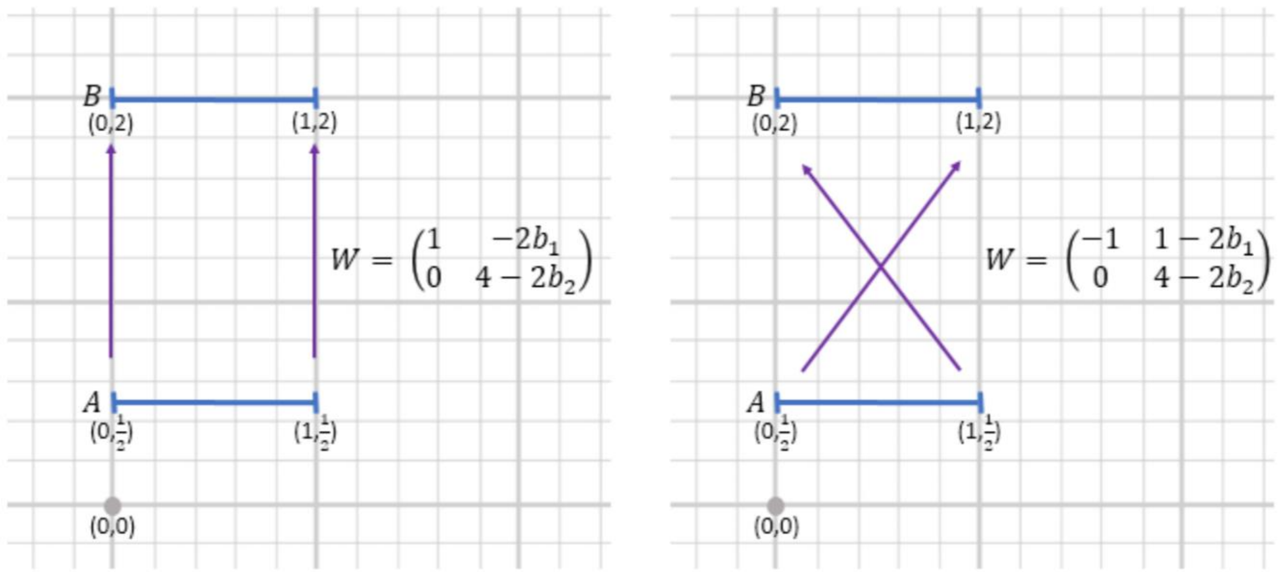


Figure 1: An illustrative example where the two domains are line segments in $\mathbb{R}^2$. There are infinitely many mappings that preserve the uniform distribution on the two segments. However, only two stand out as "semantic". These are exactly the two mappings that can be captured by a neural network 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.

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%