Contrastive Learning for Unpaired Image-to-Image Translation

Taesung Park[1], Alexei A. Efros[1], Richard Zhang[2], Jun-Yan Zhu[2]

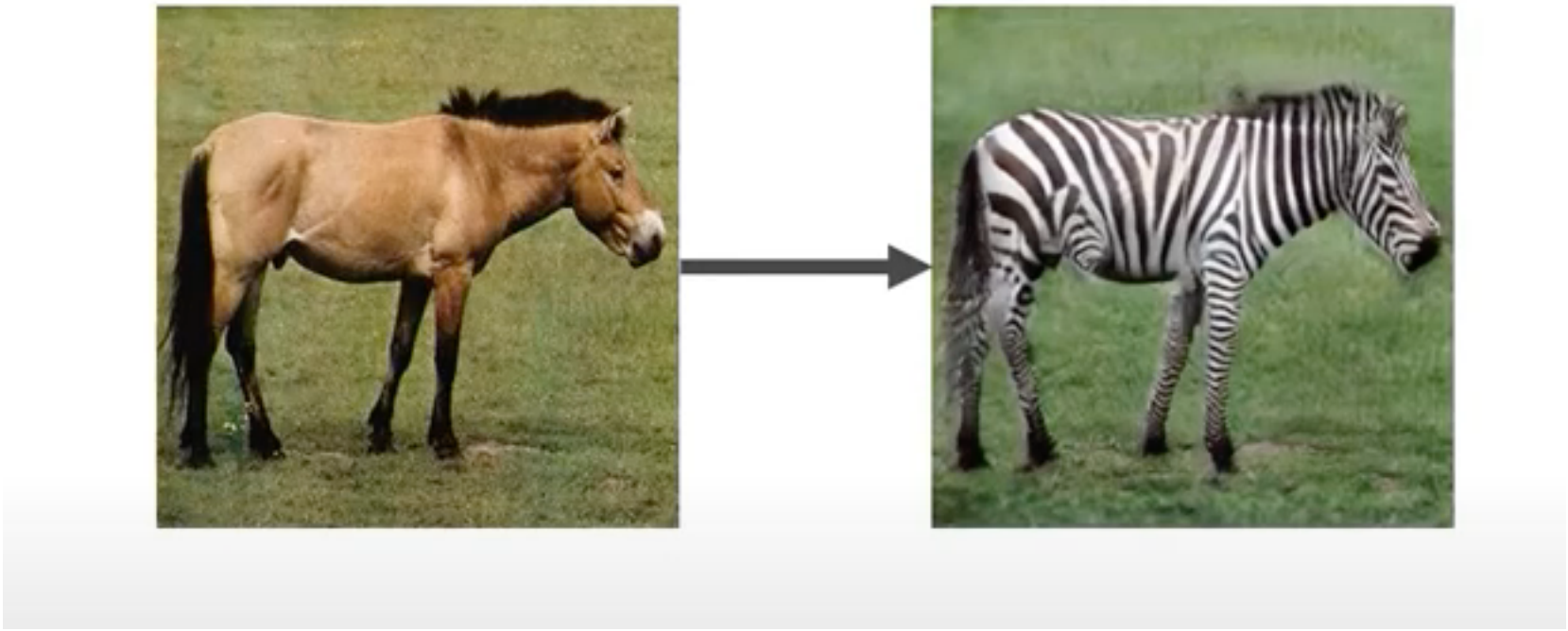
[1]University of California, Berkeley

[2]Adobe Research

ECCV 2020

Presented by Huidong Xie

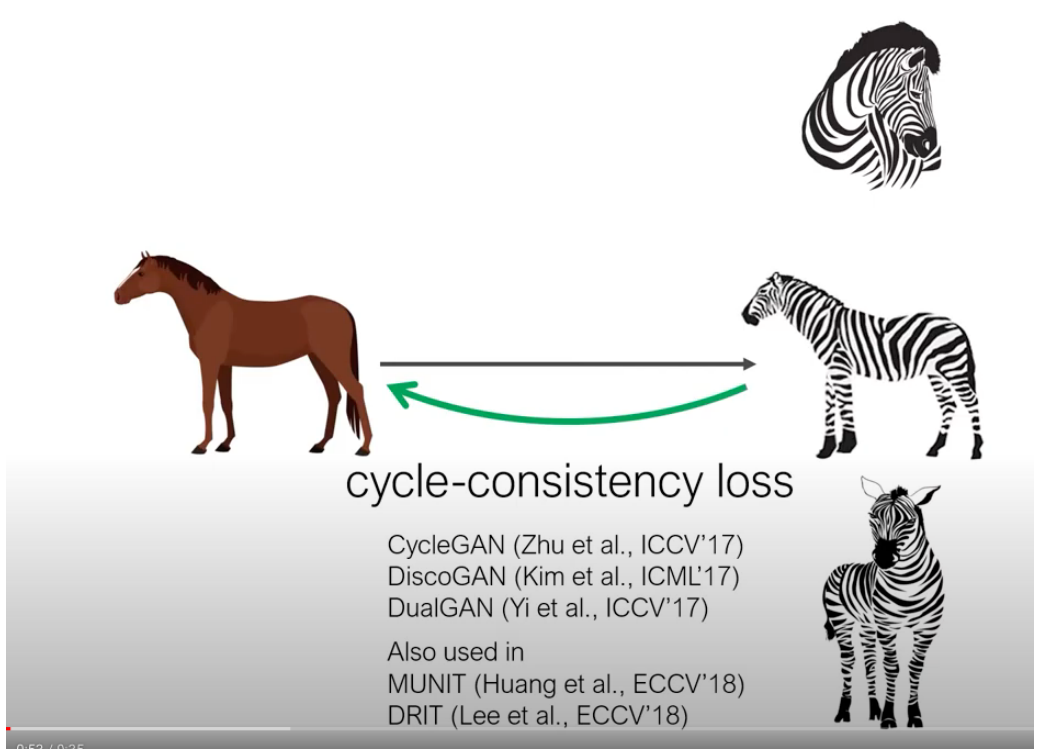
Background

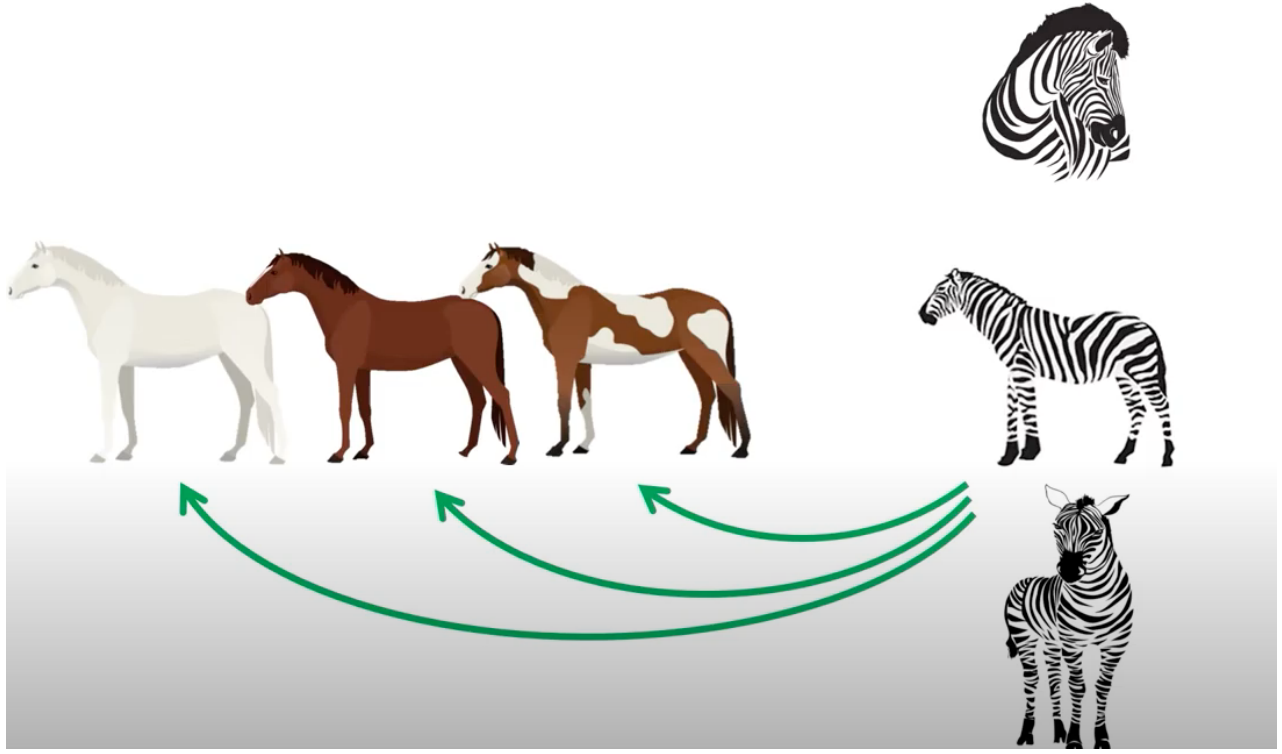


Translate image from domain X to a target domain Y
Not possible to implement supervised learning.

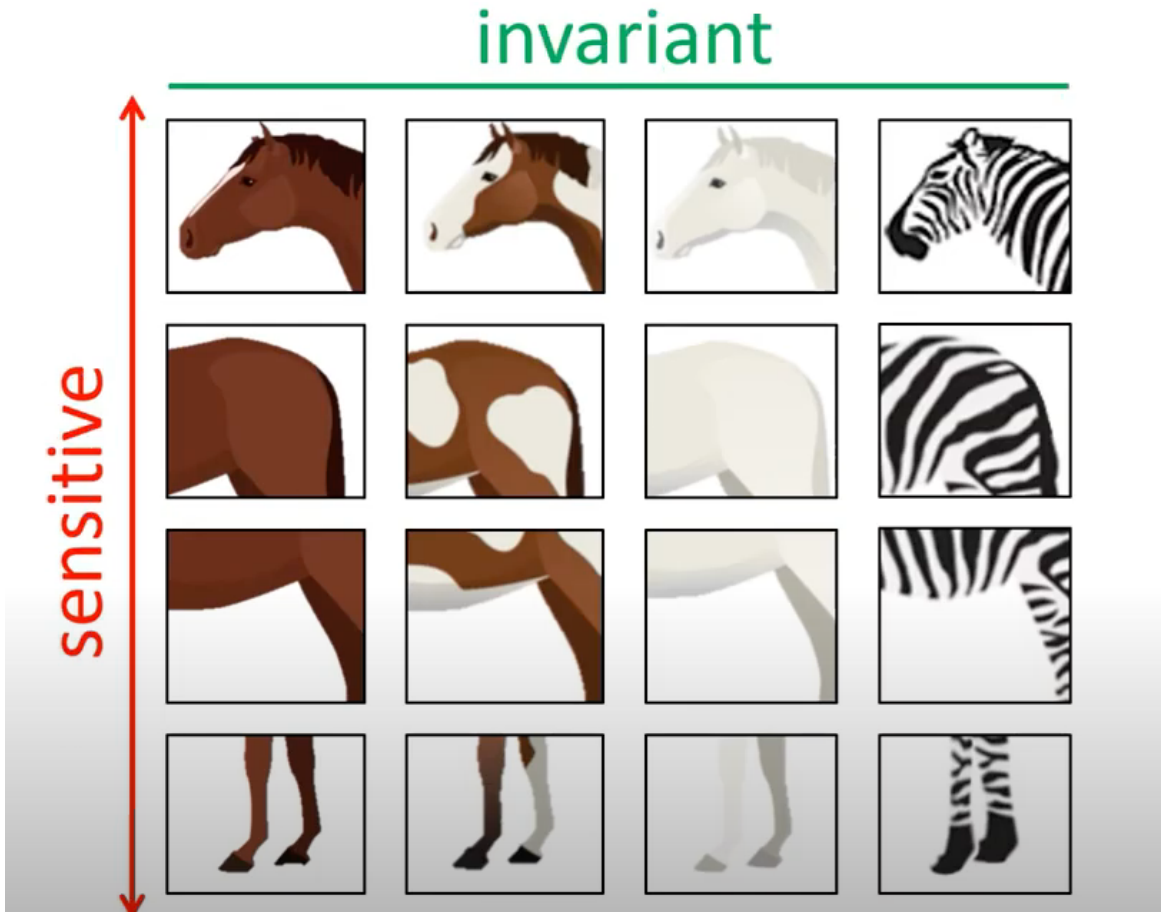
Background

- Other methods such as cycle-gan are sometime too restrictive.





Background



Network

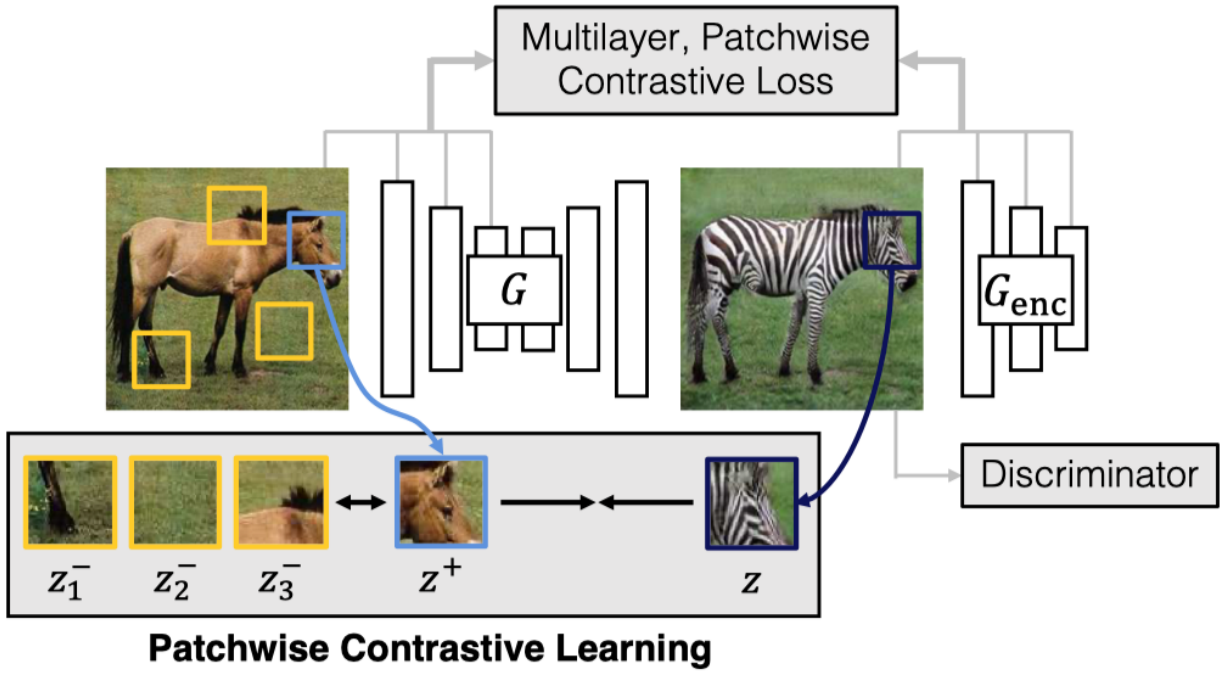


Fig. 1: **Patchwise Contrastive Learning for one-sided translation.** A generated **output patch** should appear closer to its **corresponding input patch**, in comparison to other **random patches**. We use a multilayer, patchwise contrastive loss, which maximizes *mutual information* between corresponding input and output patches. This enables one-sided translation in the unpaired setting.

Network

There is no fixed similarity metric, such as The L1 or perceptual loss

When comparing data from two different domains, these fixed metric may not be appropriate.

The proposed method is one-sided.
There is no inverse mapping needed.
Therefore, the training is faster.
Memory burden is reduced.

(N+1)-way classification problem is set up.

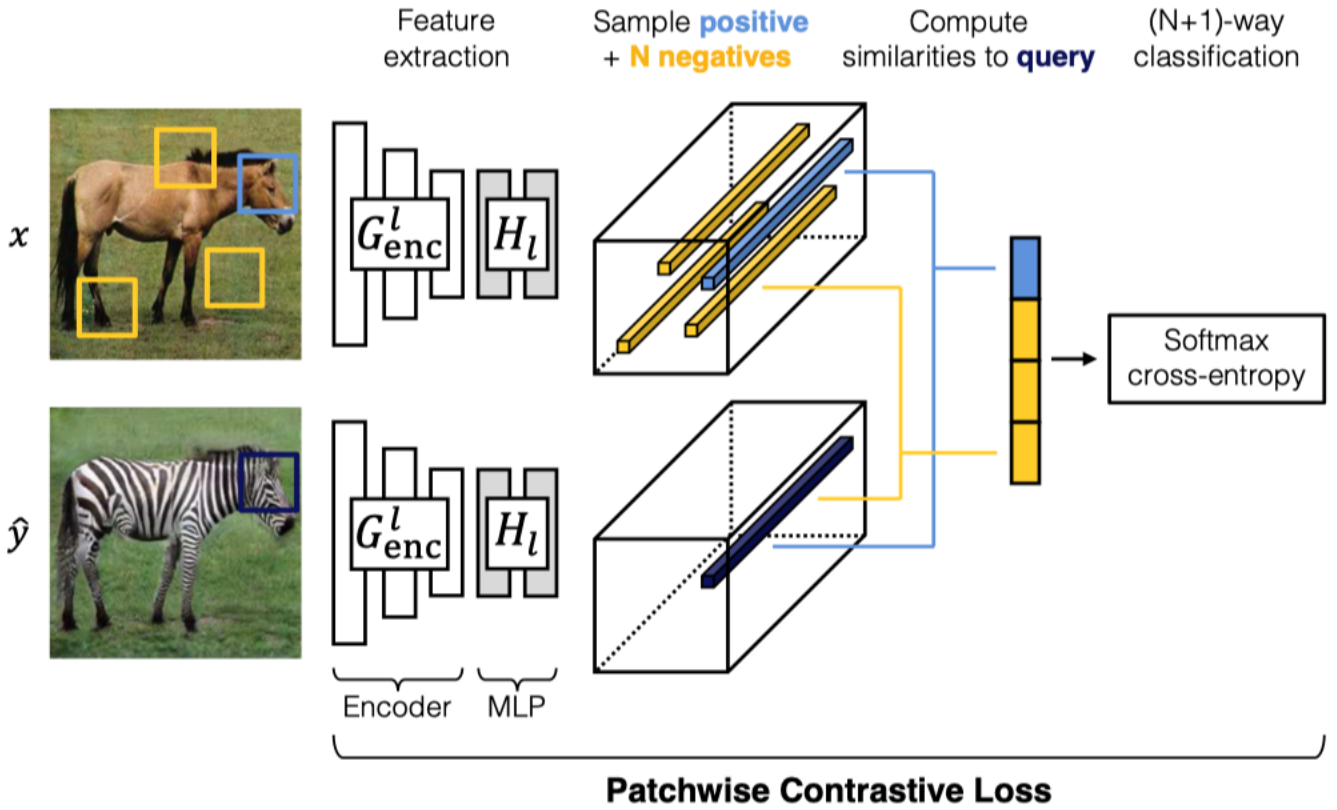


Fig. 2: **Patchwise Contrastive Loss.** Both images, $\boldsymbol{x}$ and $\hat{\boldsymbol{y}}$, are encoded into feature tensor. We sample a **query** patch from the output $\hat{\boldsymbol{y}}$ and compare it to the input patch at the same location. We set up an (N+1)-way classification problem, where N negative patches are sampled from the same input image at different locations. We reuse the encoder part G_{enc} of our generator and add a two-layer MLP network. This network learns to project both the input and output patch to a shared embedding space.

Loss function

$$\mathcal{L}_{\text{GAN}}(G, D, X, Y) + \lambda_X \mathcal{L}_{\text{PatchNCE}}(G, H, X) + \lambda_Y \mathcal{L}_{\text{PatchNCE}}(G, H, Y). \quad (5)$$

The original GAN loss

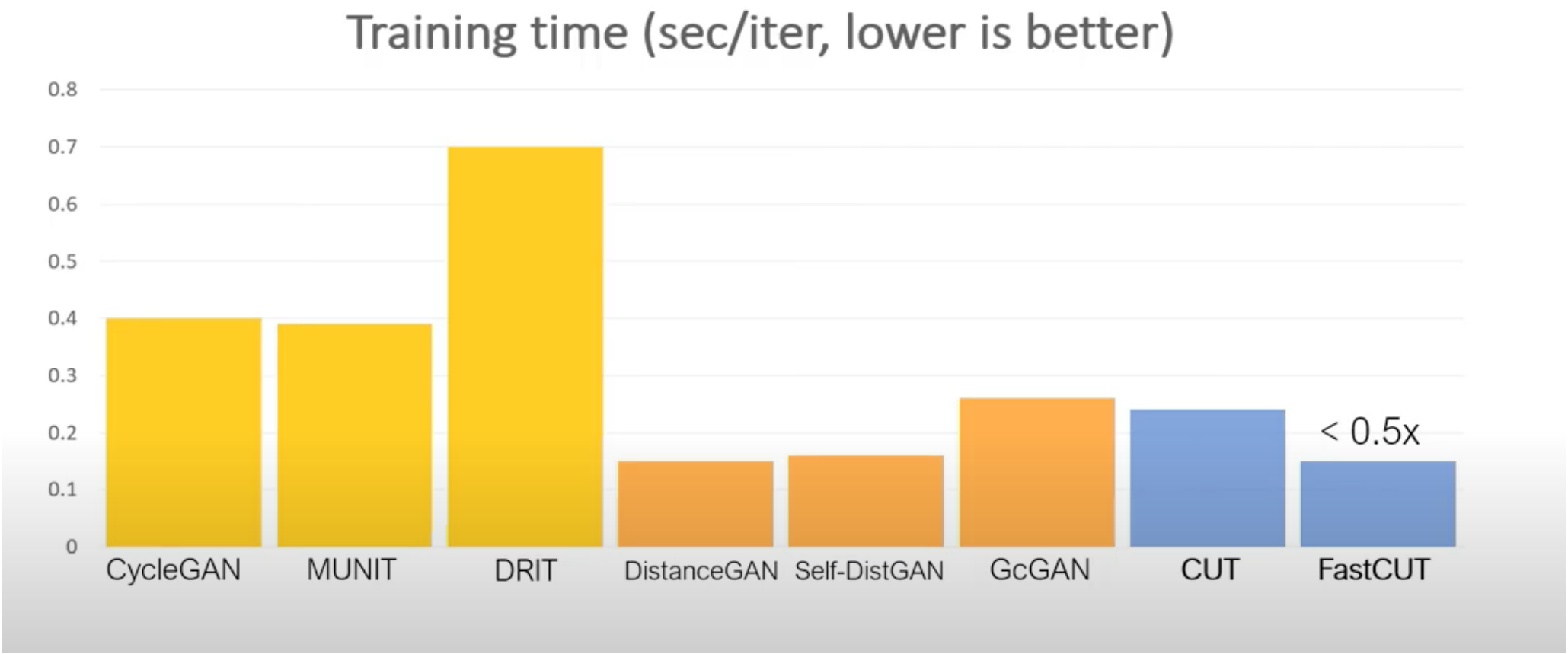
Contrastive loss term

Identity loss term

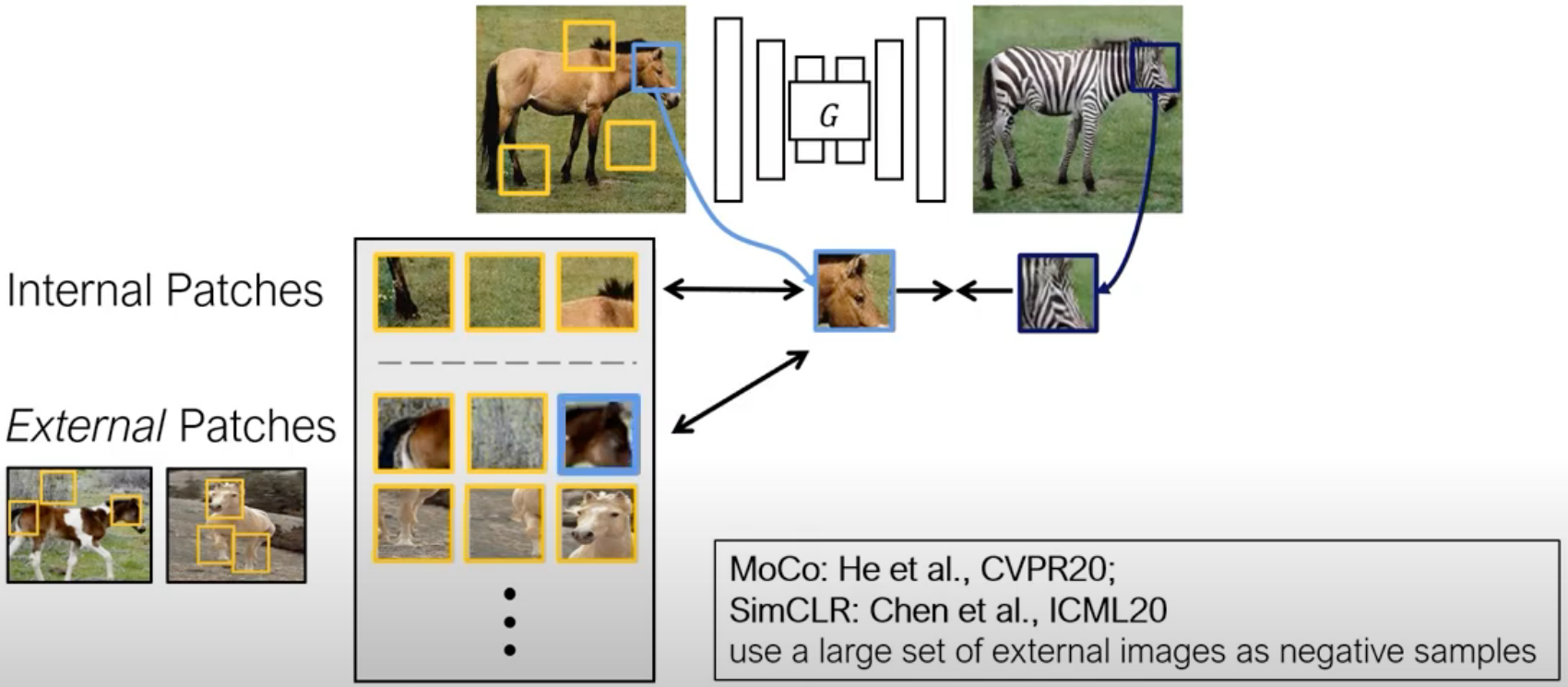
X contains images from the input domain, and Y contains images from the output (target) domain

When $\lambda_X = \lambda_Y = 1$, they called it CUT network, when $\lambda_X = 10, \lambda_Y = 0$, they called it FastCUT
FastCUT does not have identity loss term, and it is faster than CUT.
FastCUT performs worse than CUT.

Training time

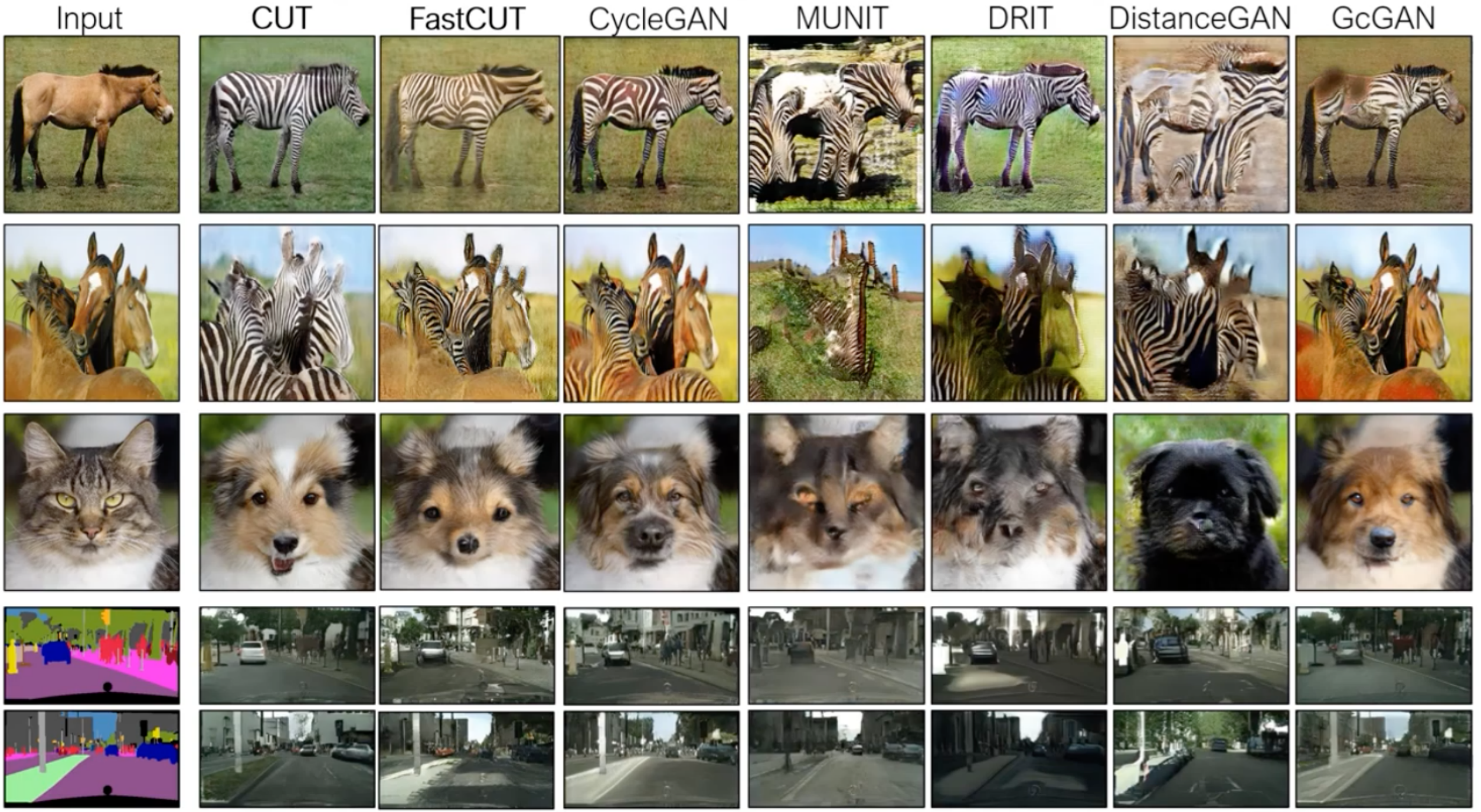


Internal and External patches

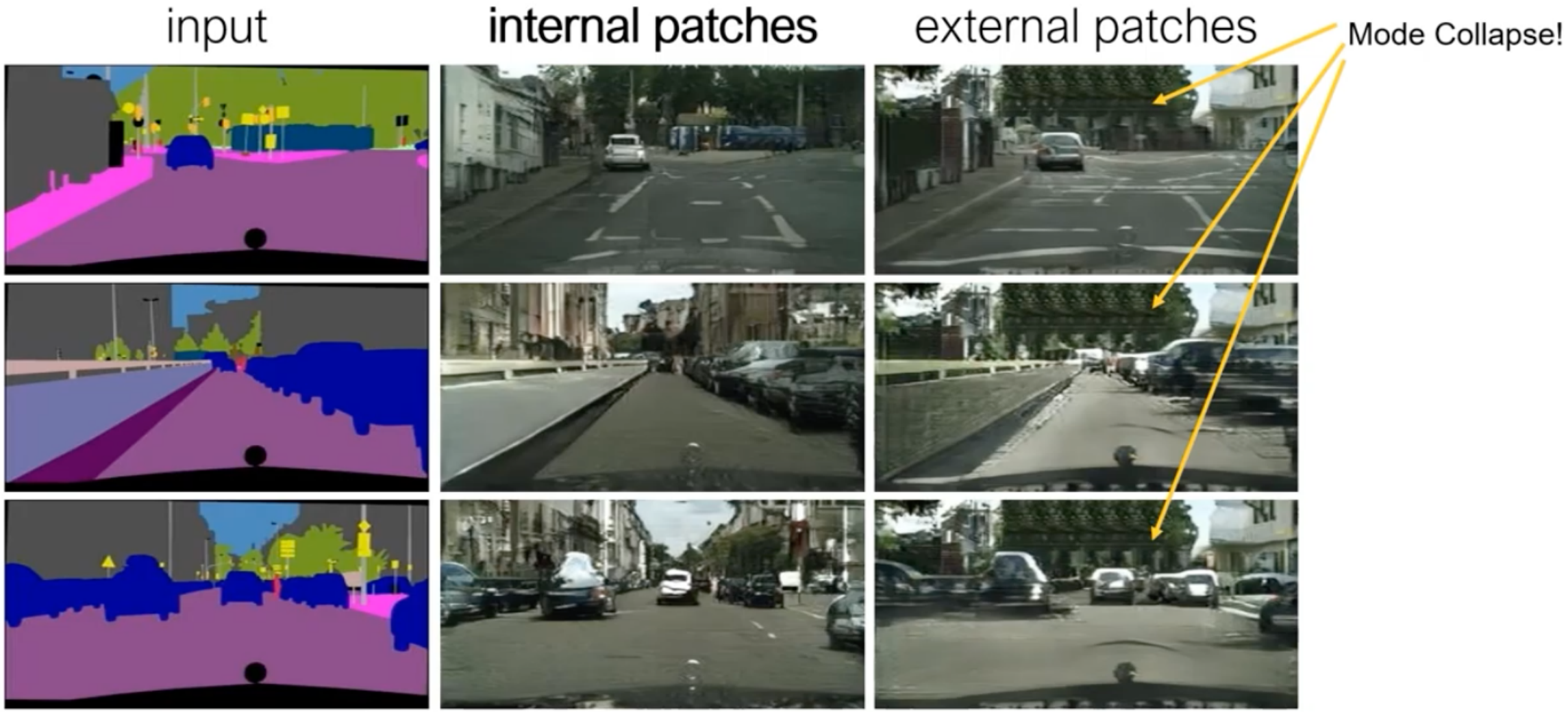


Unexpectedly, when external patches were included, performance became worse.

They said, it may because the external patches are too easy to distinguish. And there may be some false negative cases.


Input
CUT
FastCUT
CycleGAN
MUNIT
DRIT
DistanceGAN
GcGAN

When external patches are used



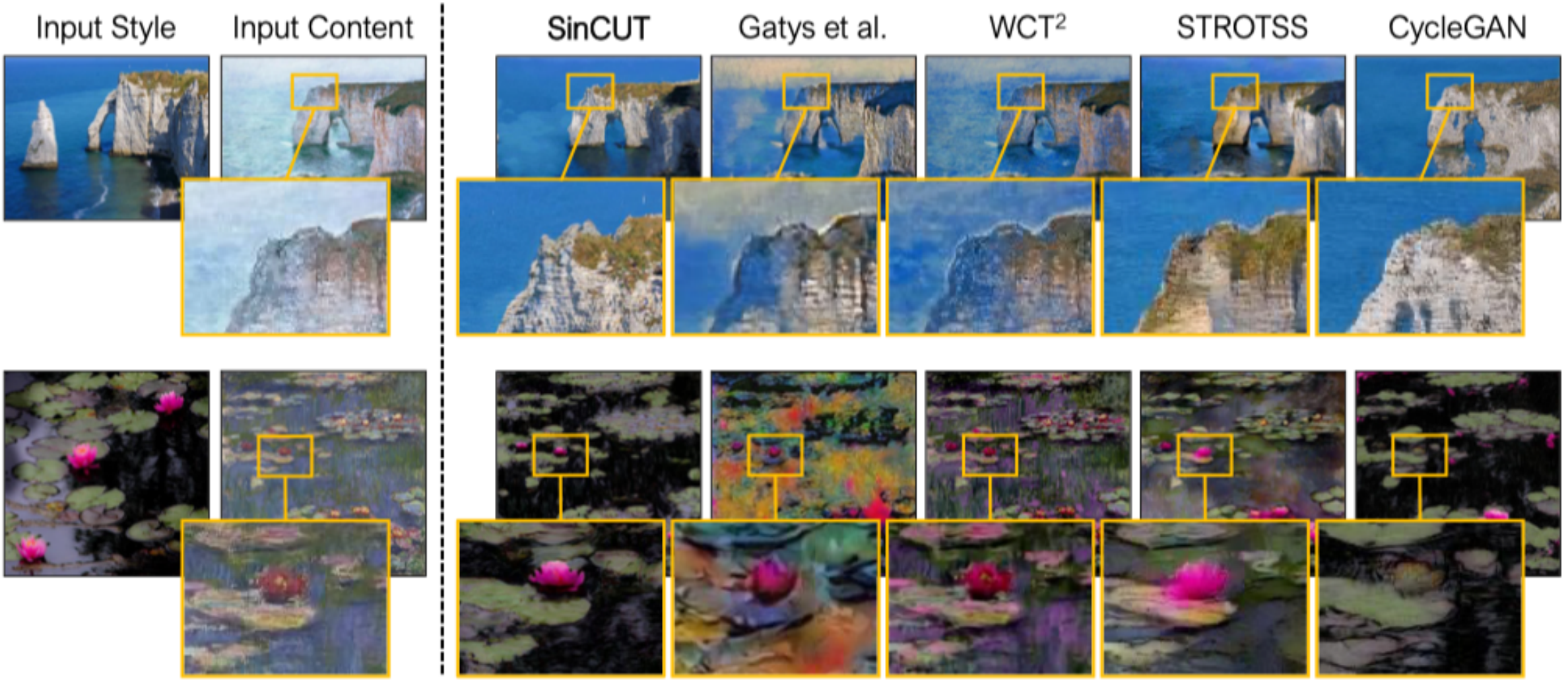
Quantitative measurements

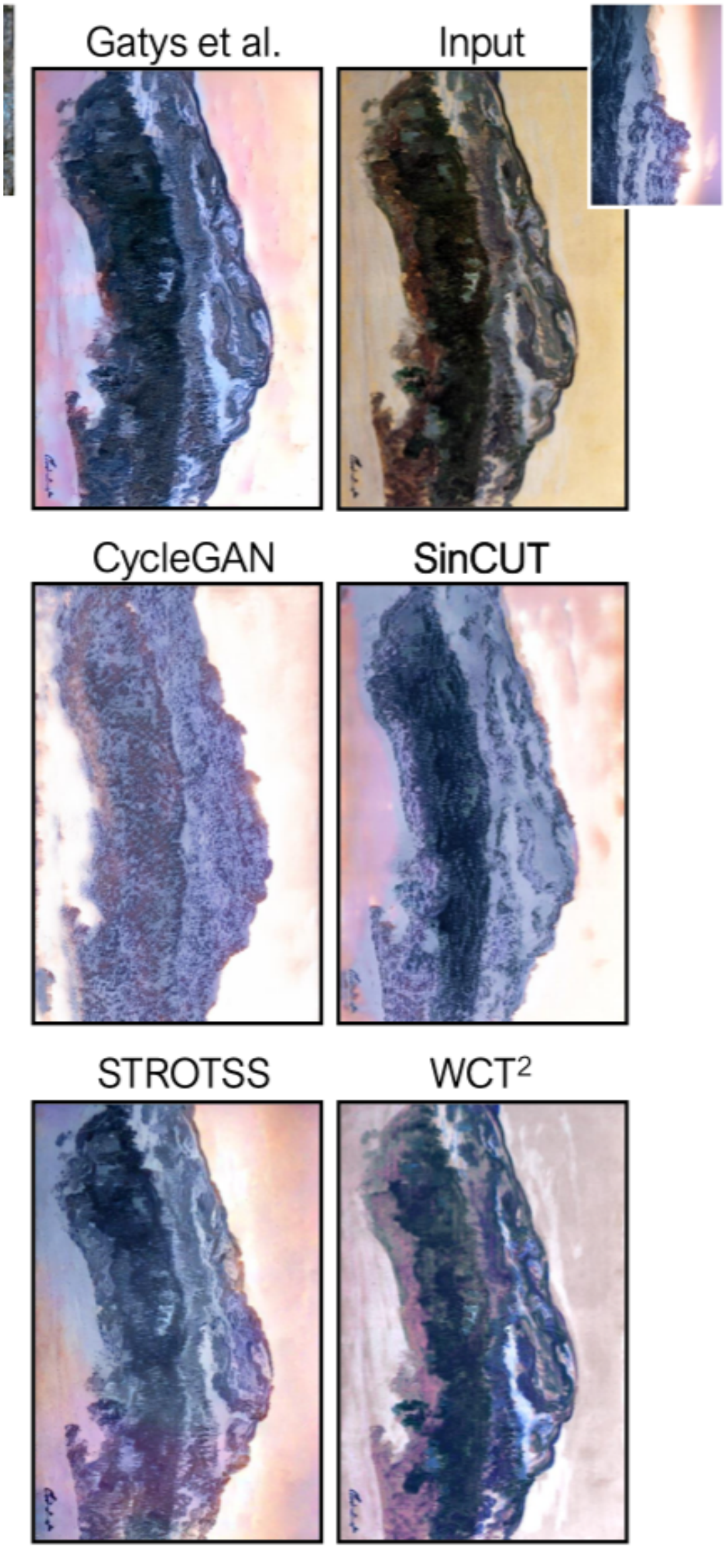
Method	Cityscapes				Cat→Dog	Horse→Zebra		
	mAP↑	pixAcc↑	classAcc↑	FID↓	FID↓	FID↓	sec/iter↓	Mem(GB)↓
CycleGAN [88]	20.4	55.9	25.4	76.3	85.9	77.2	0.40	4.81
MUNIT [43]	16.9	56.5	22.5	91.4	104.4	133.8	0.39	3.84
DRIT [40]	17.0	58.7	22.2	155.3	123.4	140.0	0.70	4.85
Distance [3]	8.4	42.2	12.6	81.8	155.3	72.0	**0.15**	2.72
SelfDistance [3]	15.3	56.9	20.6	78.8	144.4	80.8	0.16	2.72
GCGAN [17]	21.2	63.2	26.6	105.2	96.6	86.7	0.26	2.67
CUT	**24.7**	**68.8**	**30.7**	**56.4**	**76.2**	**45.5**	0.24	3.33
FastCUT	19.1	59.9	24.3	68.8	94.0	73.4	**0.15**	**2.25**

High-res single image translation

- Since CUT and FastCUT only rely on internal patches, they can be trained on a single image dataset.
- To prevent overfitting, training images were scaled.
- They called this network, SinCUT.

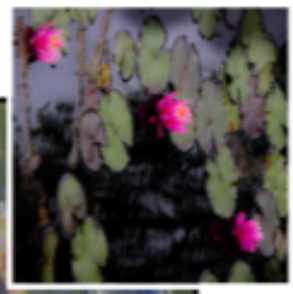
Translating painting to photo



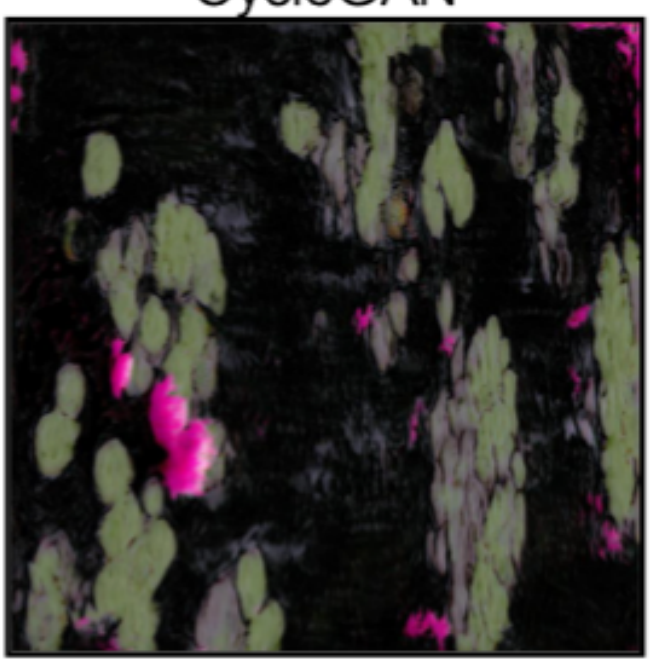

Gatys et al.
Input
CycleGAN
SinCUT
STROTSS
WCT2

Gatys et al.

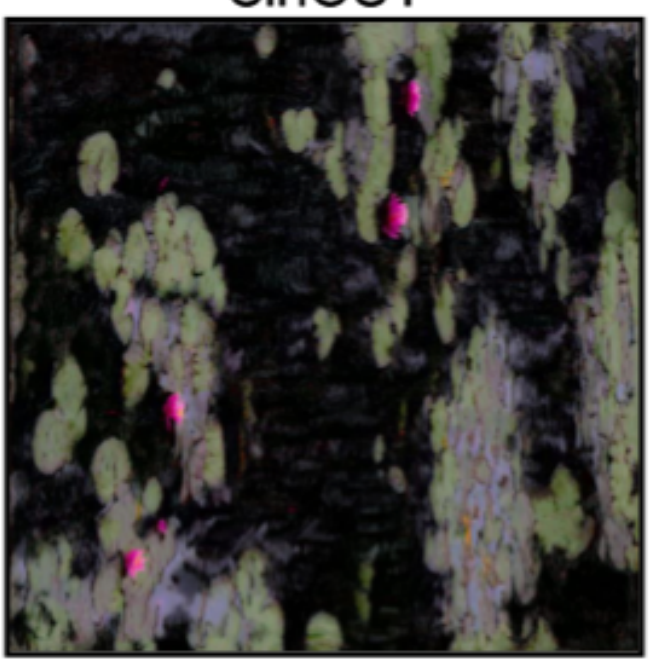
Input



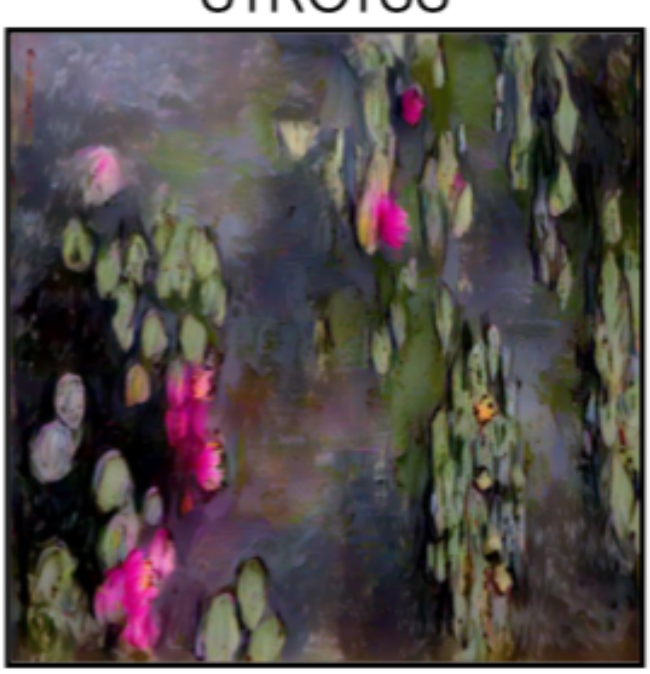
CycleGAN



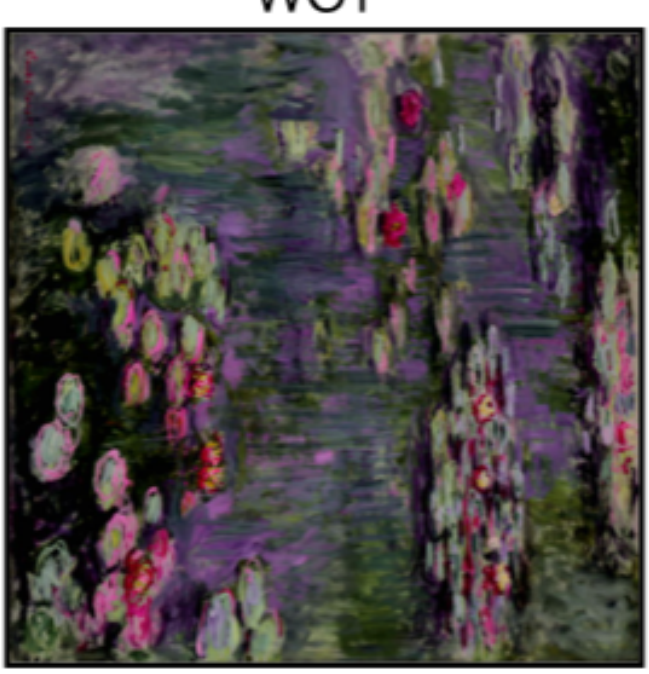
SinCUT



STROTSS



WCT^2



Input

Gatys et al.

SinCUT

CycleGAN

WCT²

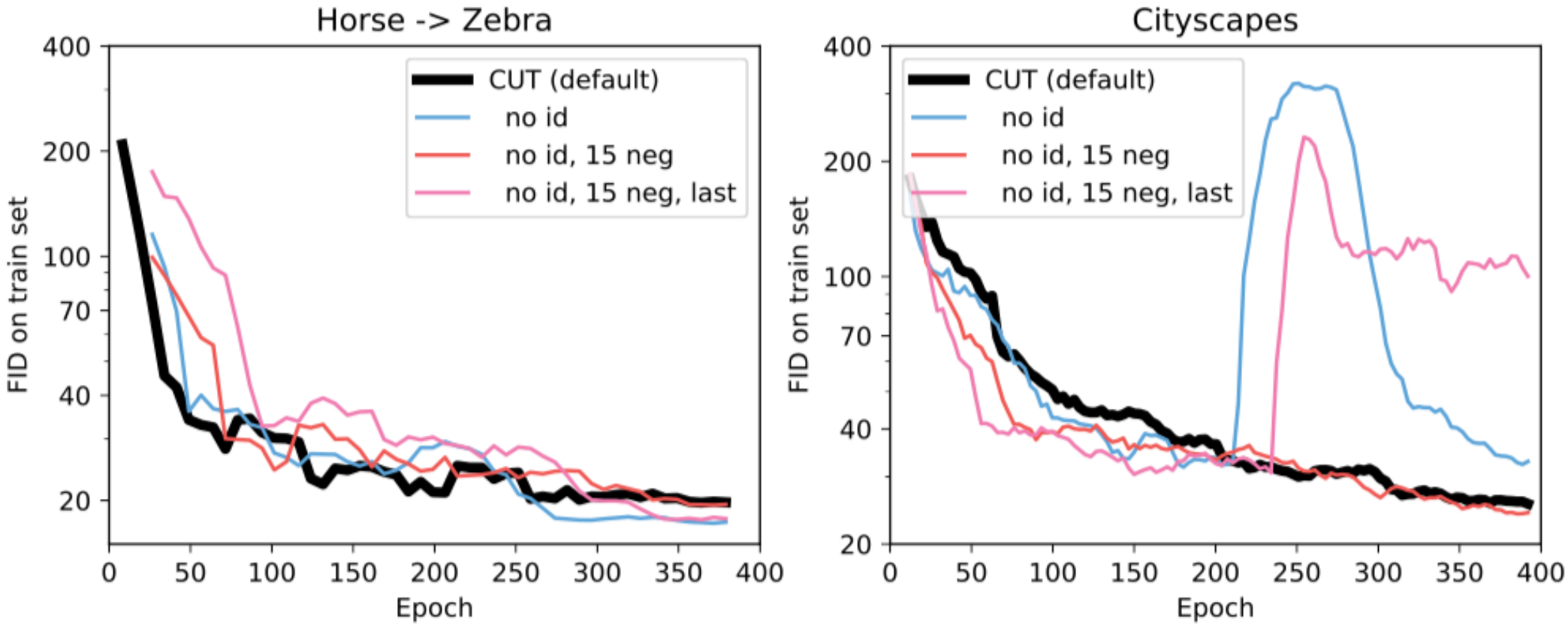
STROTSS

Ablation analysis

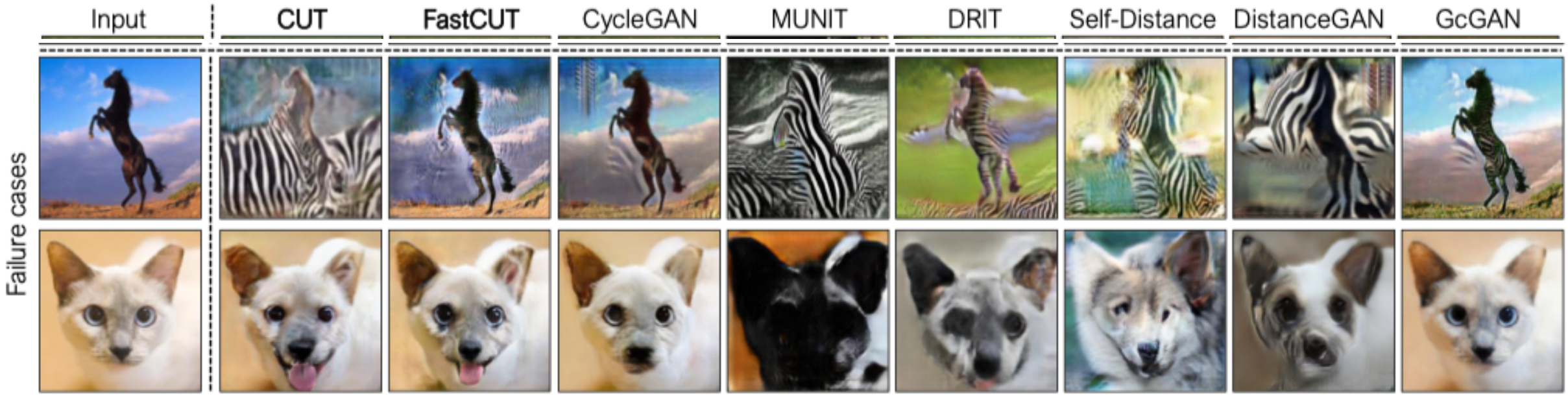
Method	Training settings					Testing datasets		
	Id	Negs	Layers	Int	Ext	Horse→Zebra	Cityscapes	
						FID↓	FID↓	mAP↑
CUT (default)	✓	255	All	✓	✗	45.5	**56.4**	**24.7**
no id	✗	255	All	✓	✗	39.3	68.5	22.0
no id, 15 neg	✗	15	All	✓	✗	44.1	59.7	23.1
no id, 15 neg, last	✗	15	Last	✓	✗	**38.1**	114.1	16.0
last	✓	255	Last	✓	✗	441.7	141.1	14.9
int and ext	✓	255	All	✓	✓	56.4	64.4	20.0
ext only	✓	255	All	✗	✓	53.0	110.3	16.5
ext only, last	✓	255	Last	✗	✓	60.1	389.1	5.6

1. Internal patches are more effective than external patches
2. Using multiple layers of encoder improves performance.
3. No identity loss and less negatives still perform strongly.
4. Reducing number of layers or using external patches hurts performance.

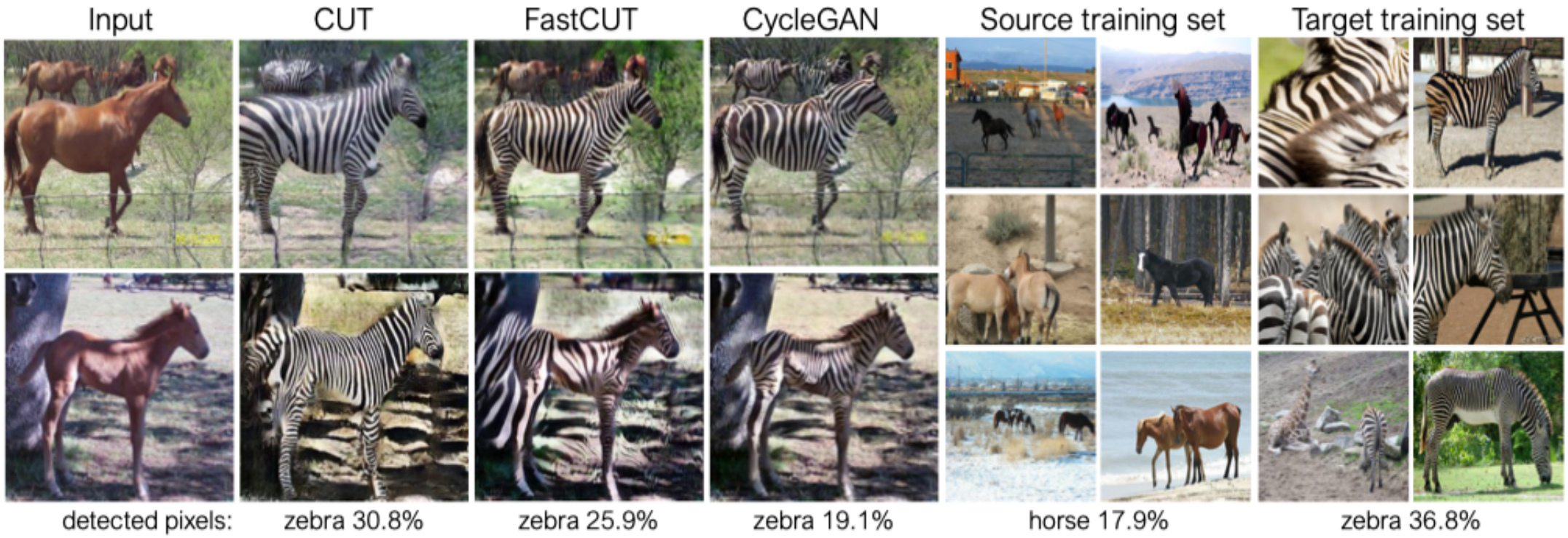
Ablation analysis



Failure cases



Number of pixels mismatch between X and Y



The CUT method has the flexibility to enlarge the horses.