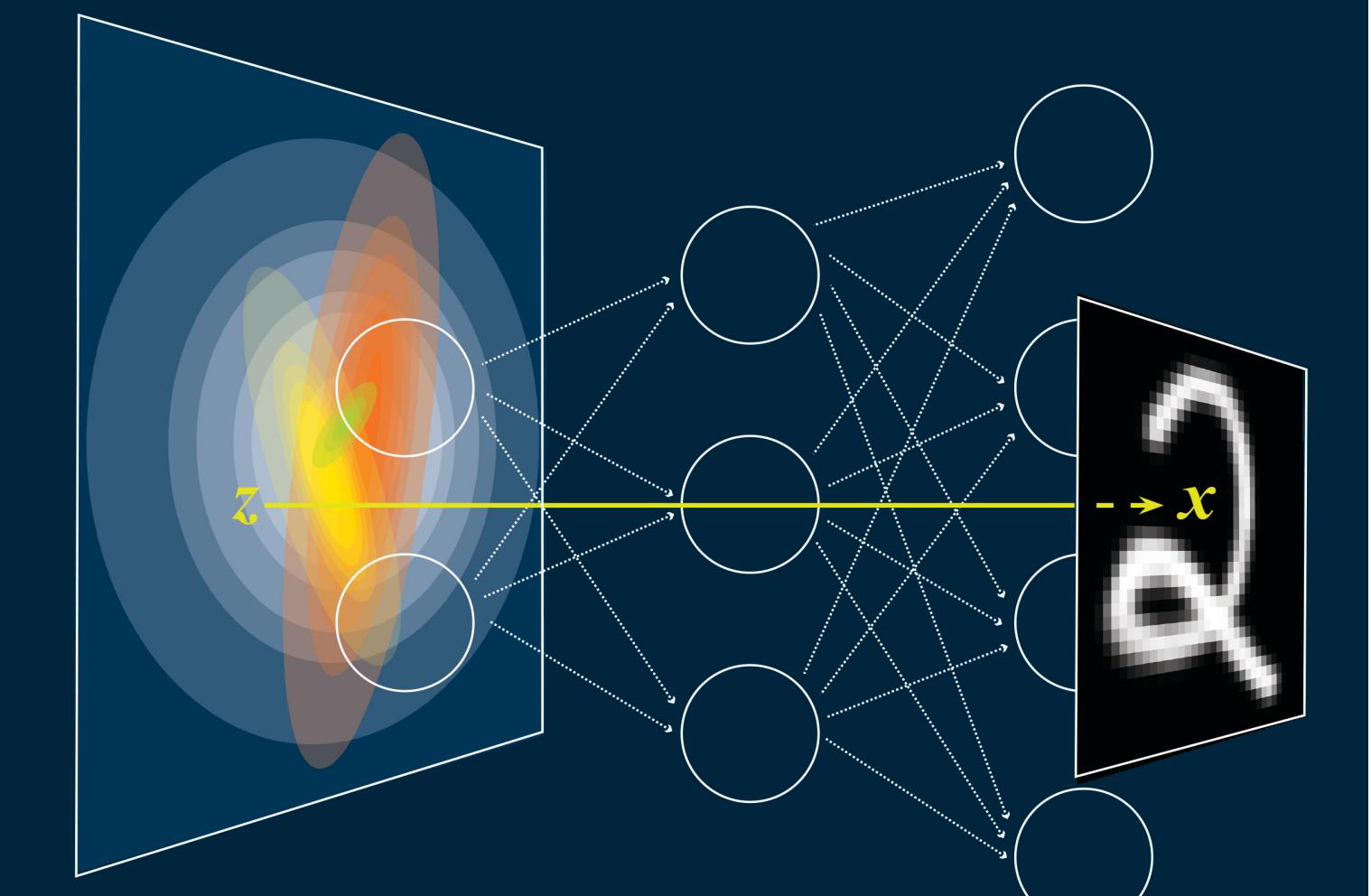


Style and Content, Texture Synthesis

Jun-Yan Zhu

16-726, Spring 2022



Probabilistic Machine Learning

Advanced Topics

Kevin P. Murphy

IV Generation 767

21	Generative models: an overview	769
22	Variational autoencoders	783
23	Auto-regressive models	827
24	Normalizing Flows	835
25	Energy-based models	855
26	Denoising diffusion models	875
27	Generative adversarial networks	883

Collection Style Transfer



Photograph ©Alexei Efros



Monet



Van Gogh



Cezanne



Ukiyo-e

Style and Content Separation

A

Classification

A	B	C	D	E
<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>
A	B	C	D	E
<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>
A	B	C	D	E
B	C	A	E	D

Domain Adaptation

B

Extrapolation

A	B	C	D	E
<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>
A	B	C	D	E
<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>
A	B	C	D	E
?	?	C	D	E

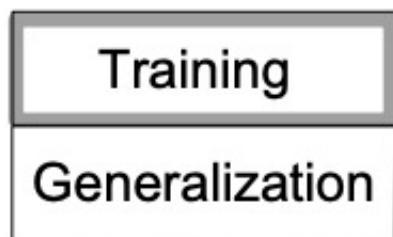
Paired Image-to-Image Translation

C

Translation

A	B	C	D	E	?	?	?
<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>			
A	B	C	D	E			
<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>			
A	B	C	D	E	?	?	?
?				?	F	G	H

Unpaired Image-to-Image Translation



Separating Style and Content
[Tenenbaum and Freeman 1996]

$$y_k^{sc} = \sum_{i=1}^I \sum_{j=1}^J w_{ijk} a_i^s b_j^c.$$

Style and Content

Adversarial loss

$$\mathbb{E}_x \log(1 - D_Y(G(x))) + \mathbb{E}_y \log D_Y(y)$$



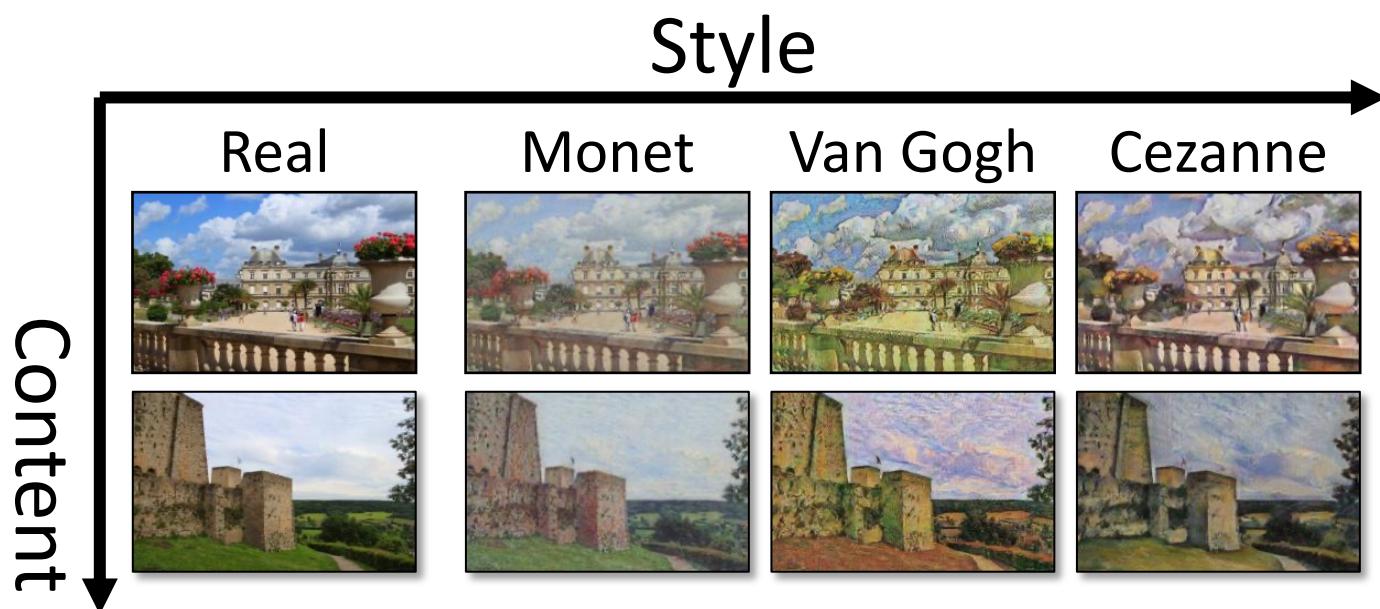
$p(x) \rightarrow p(y)$ change style

Cycle-consistency loss

$$\mathbb{E}_x \|F(G(x)) - x\|_1$$

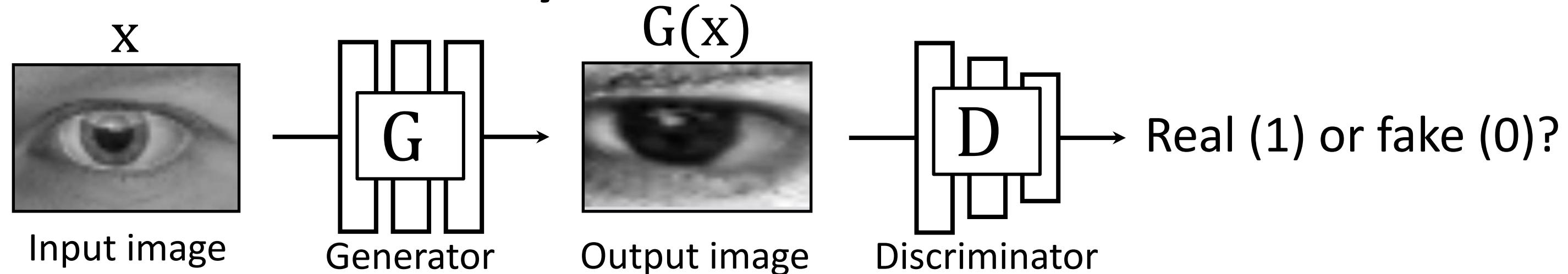


Bidirectional: preserve content



Separating Style and Content
[Tenenbaum and Freeman 1996]

Style and Content

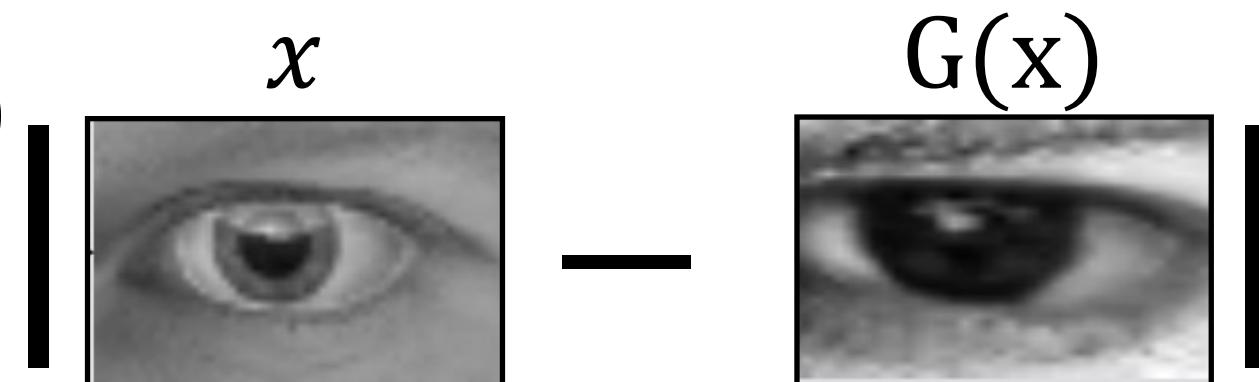


Adversarial loss (change style)

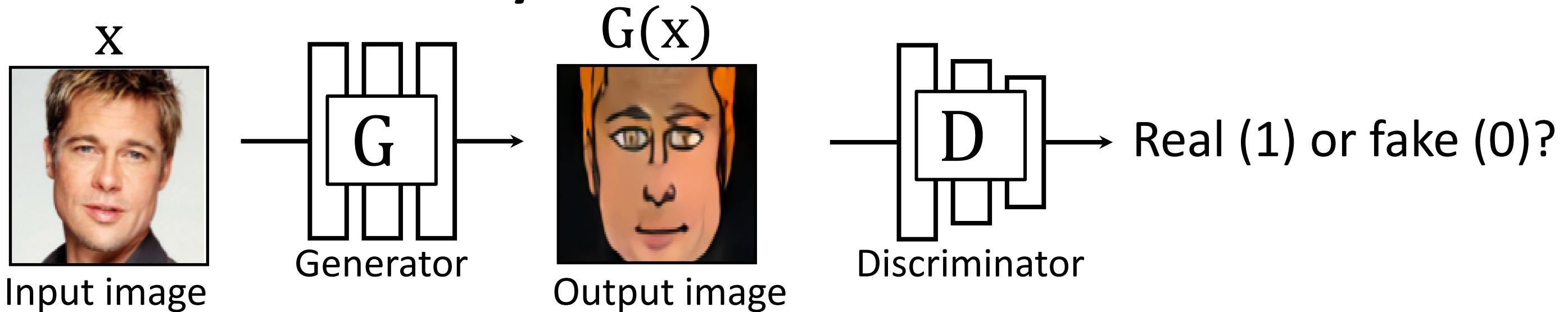
$$\mathbb{E}_x \log(1 - D_Y(G(x))) + \mathbb{E}_y \log D_Y(y)$$

L1 loss (preserve content in pixel space)

$$\mathbb{E}_x \|G(x) - x\|_1$$



Style and Content



Adversarial loss (change style)

$$\mathbb{E}_x \log(1 - D_Y(G(x))) + \mathbb{E}_y \log D_Y(y)$$

Feature loss (Preserve content in feature space)

$$\mathbb{E}_x \|F(G(x)) - F(x)\|$$

$$|F(\text{Input}) - F(\text{Output})|$$

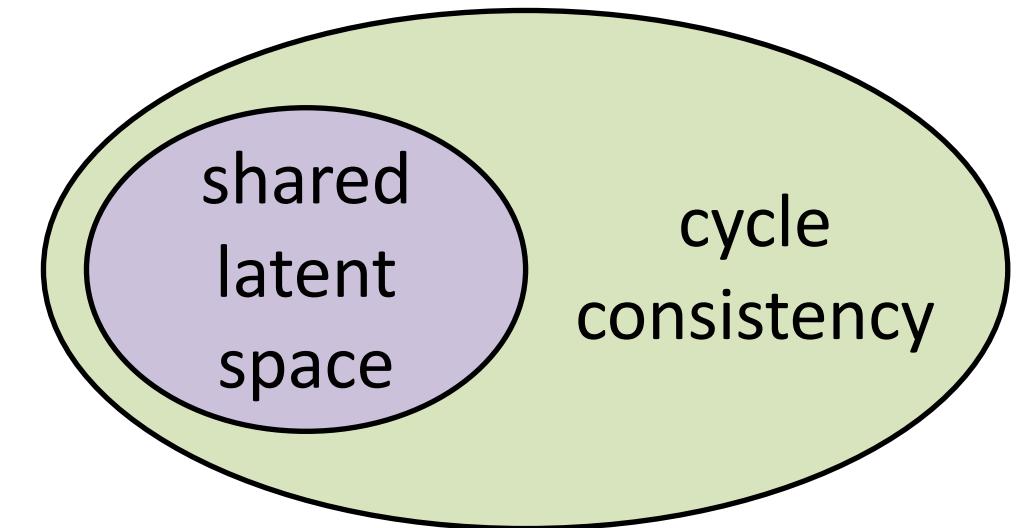
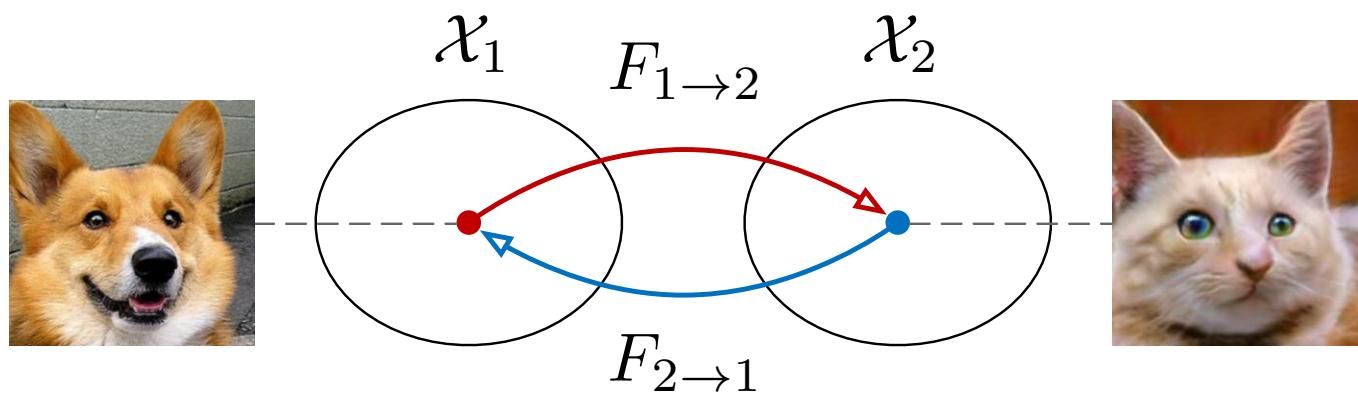
DTN [Taigman et al., 2017]

Style and Content

- Style: domain-specific features
(horse vs. zebra)
- Content: features shared across two domains

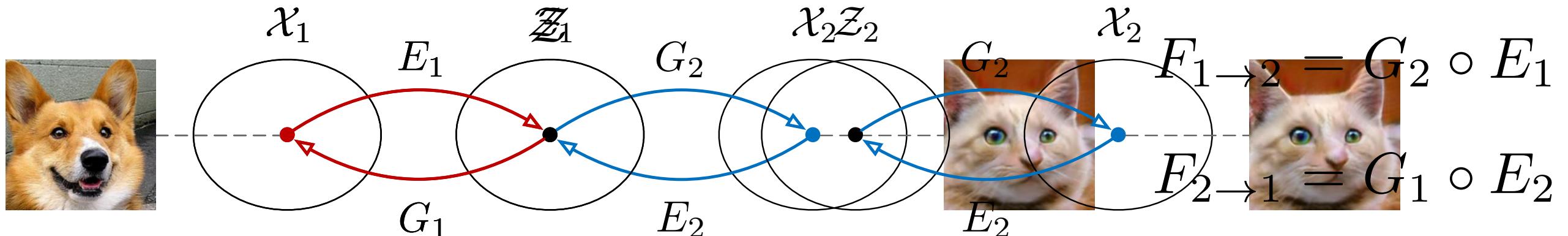
CycleGAN and UNIT

- CycleGAN (**cycle consistency**)



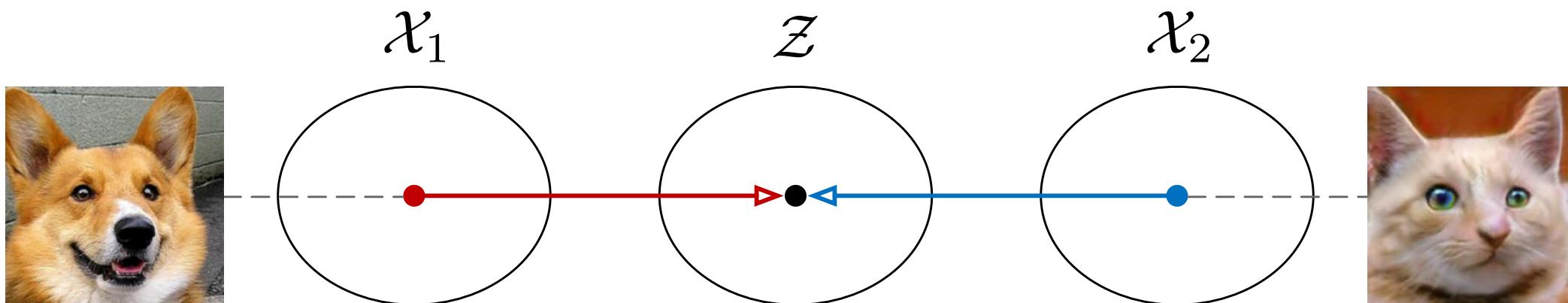
- UNIT (**shared latent space**) [Liu et al. 2017]

shared latent space \Rightarrow cycle consistency



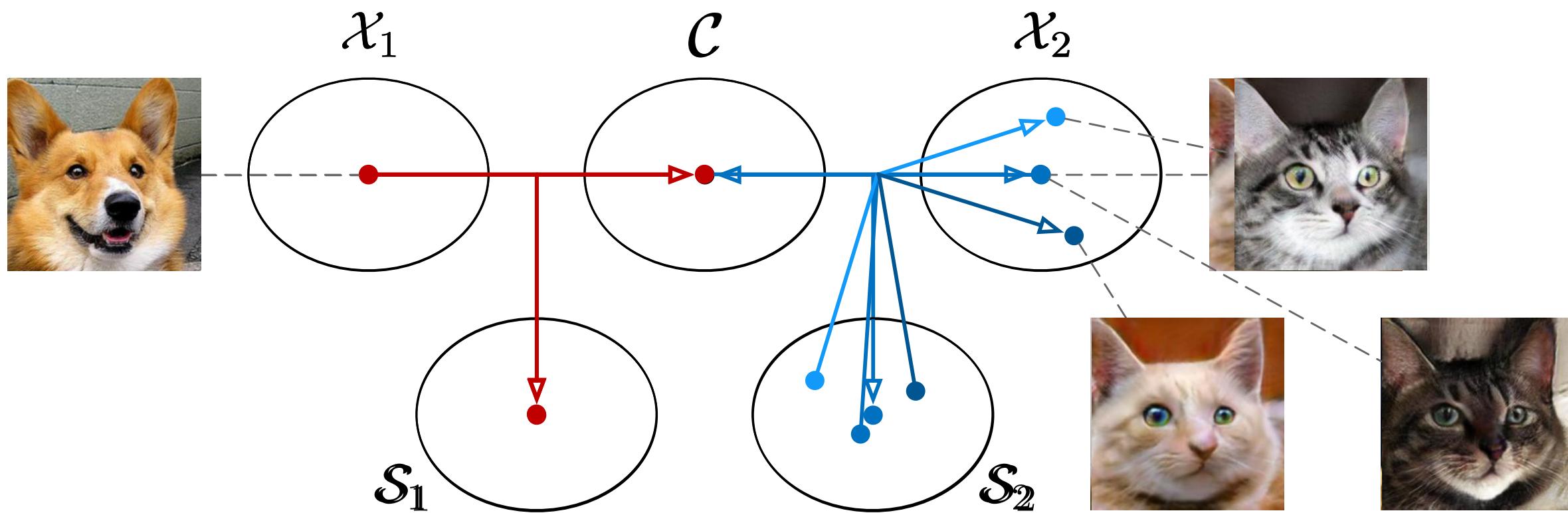
Disentangling the Latent Space

- UNIT
 - A single **shared, domain-invariant** latent space \mathcal{Z}



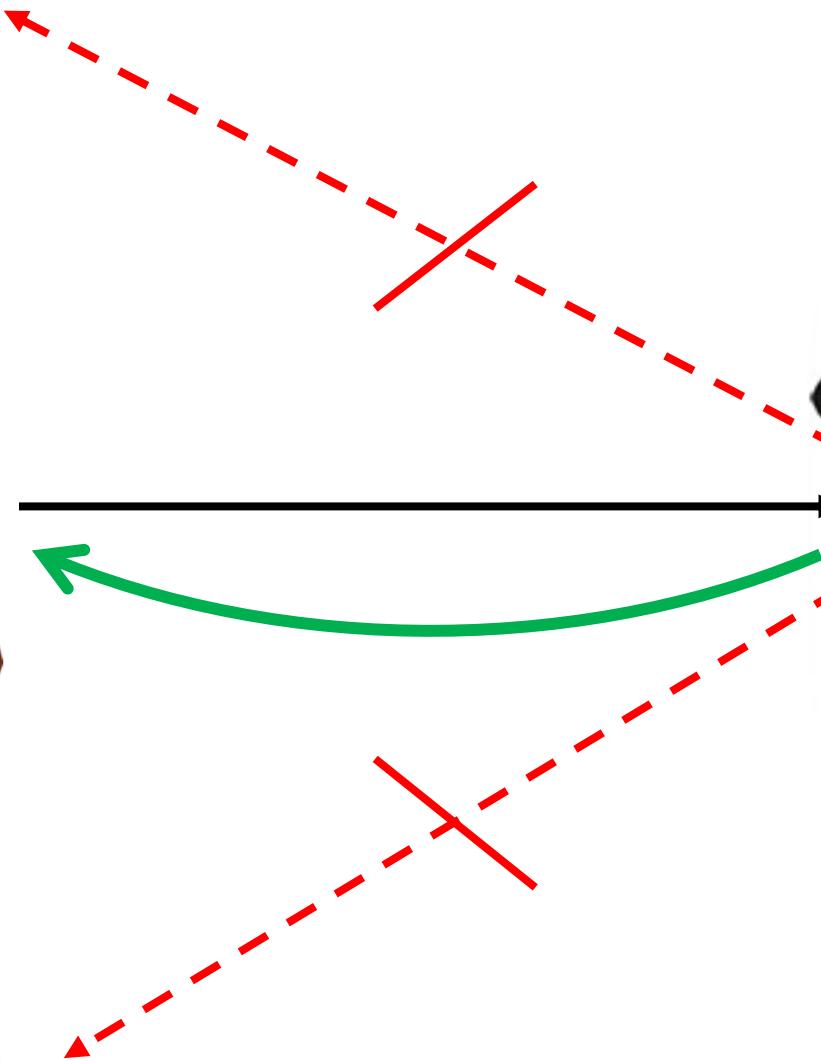
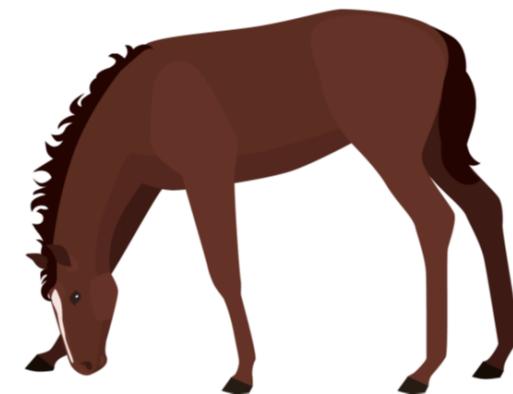
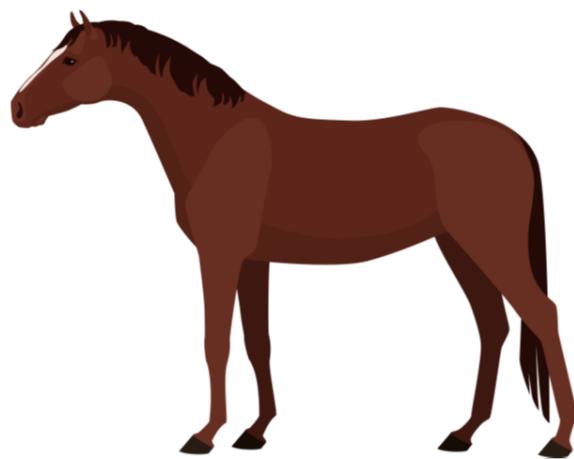
Disentangling the Latent Space

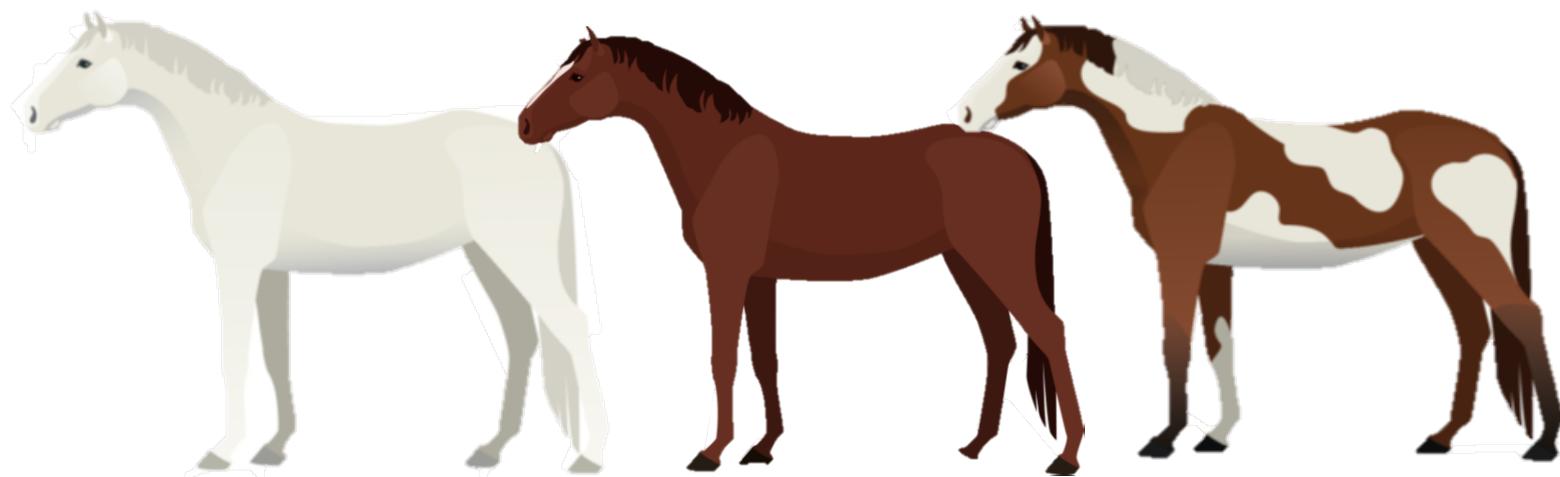
- Multimodal UNIT (MUNIT)
 - A **content** space \mathcal{C} that is **shared, domain-invariant**
 - Two **style** spaces $\mathcal{S}_1, \mathcal{S}_2$ that are **unshared, domain-specific**



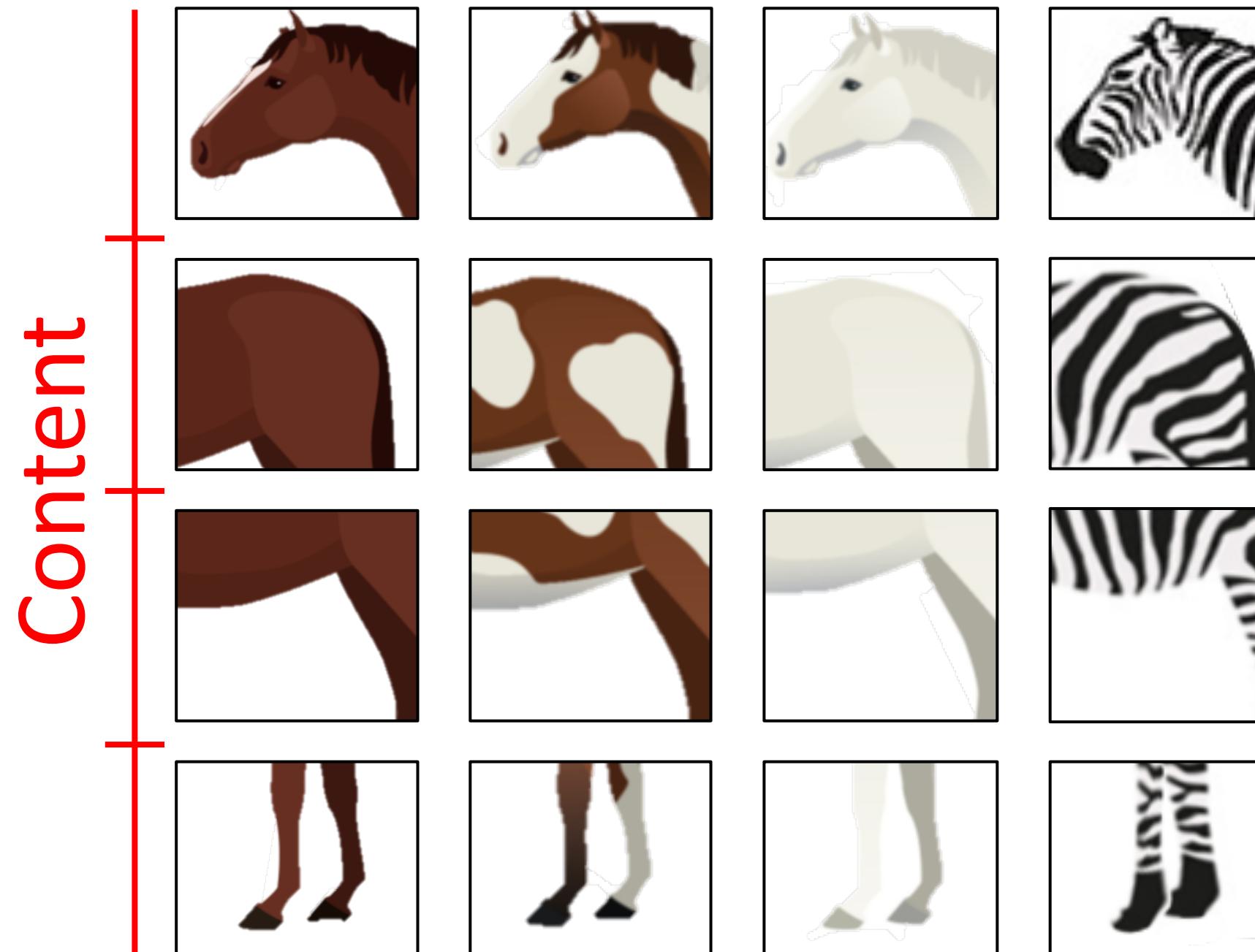
Style and Content

- Style: variations within the same domain
(different colors, textures, etc.)
- Content: features shared across two domains





Style

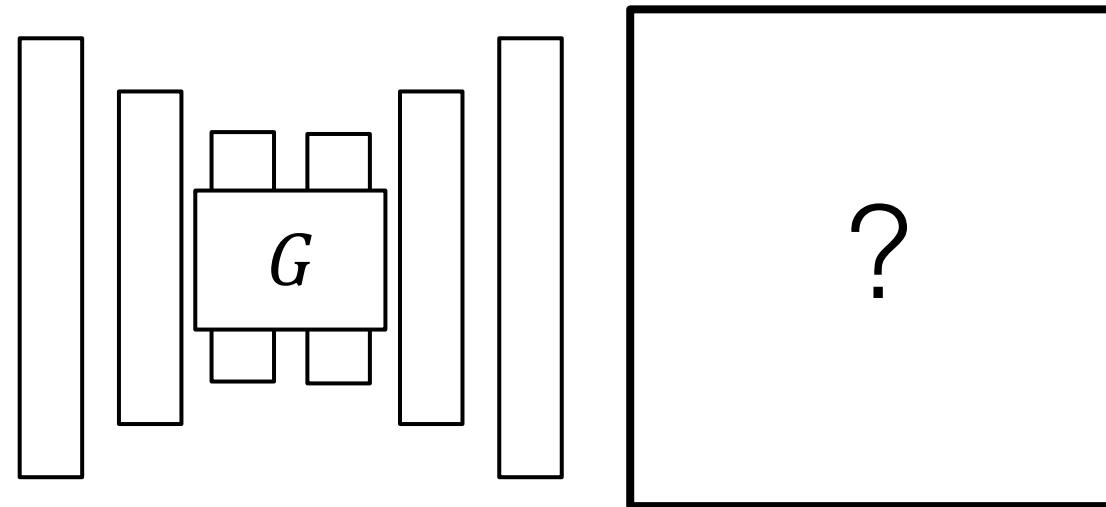


What makes for a good output?

Input (horse)

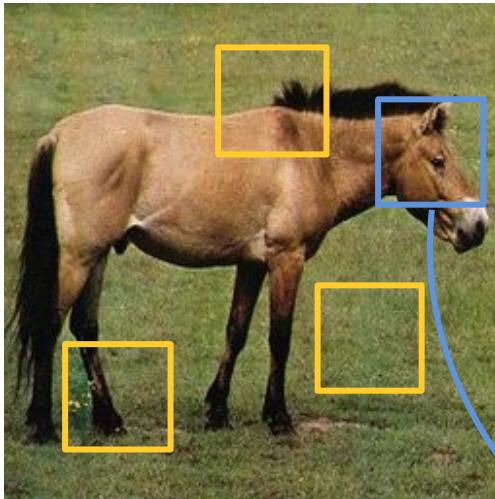


Output (zebra)

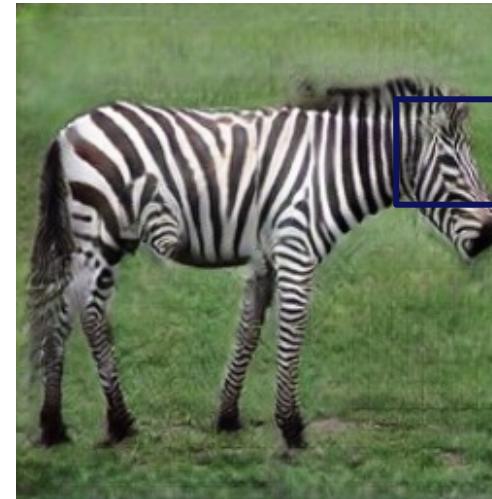


Retaining input content

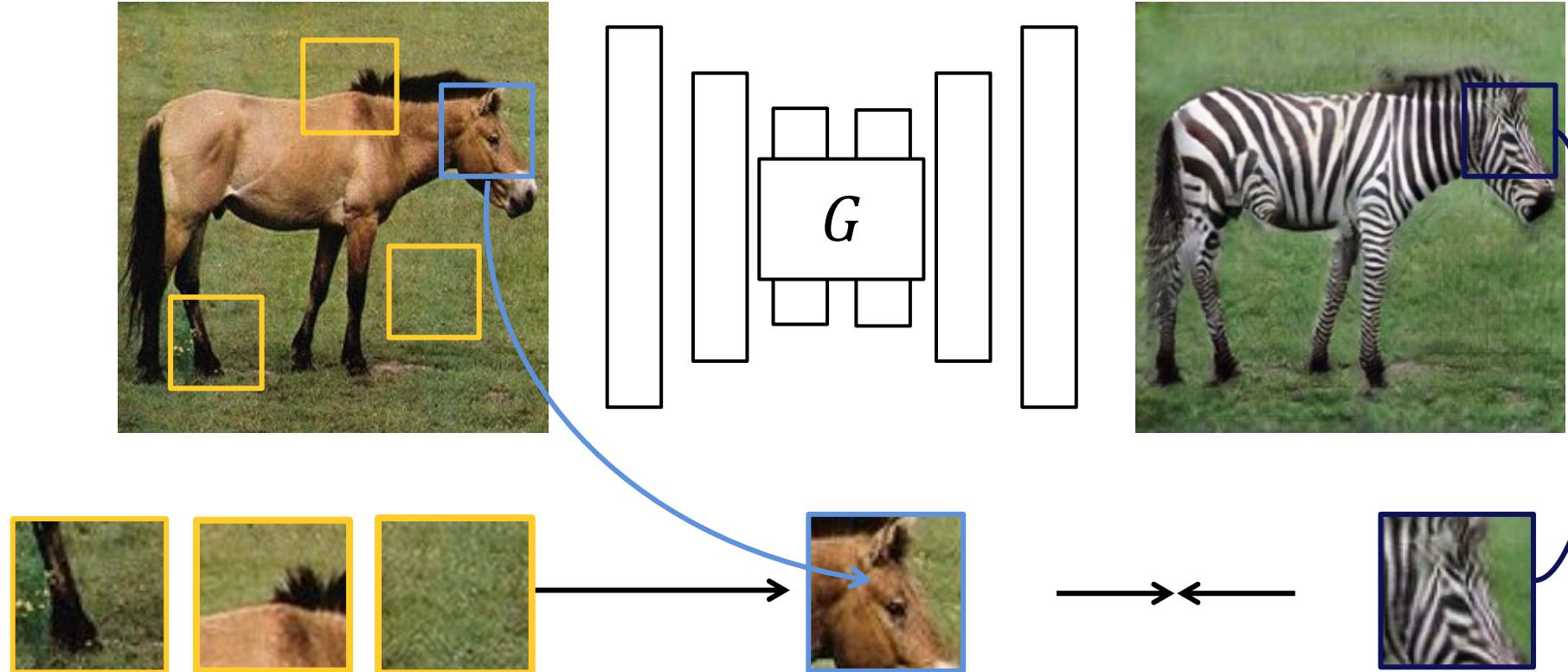
Input (horse)



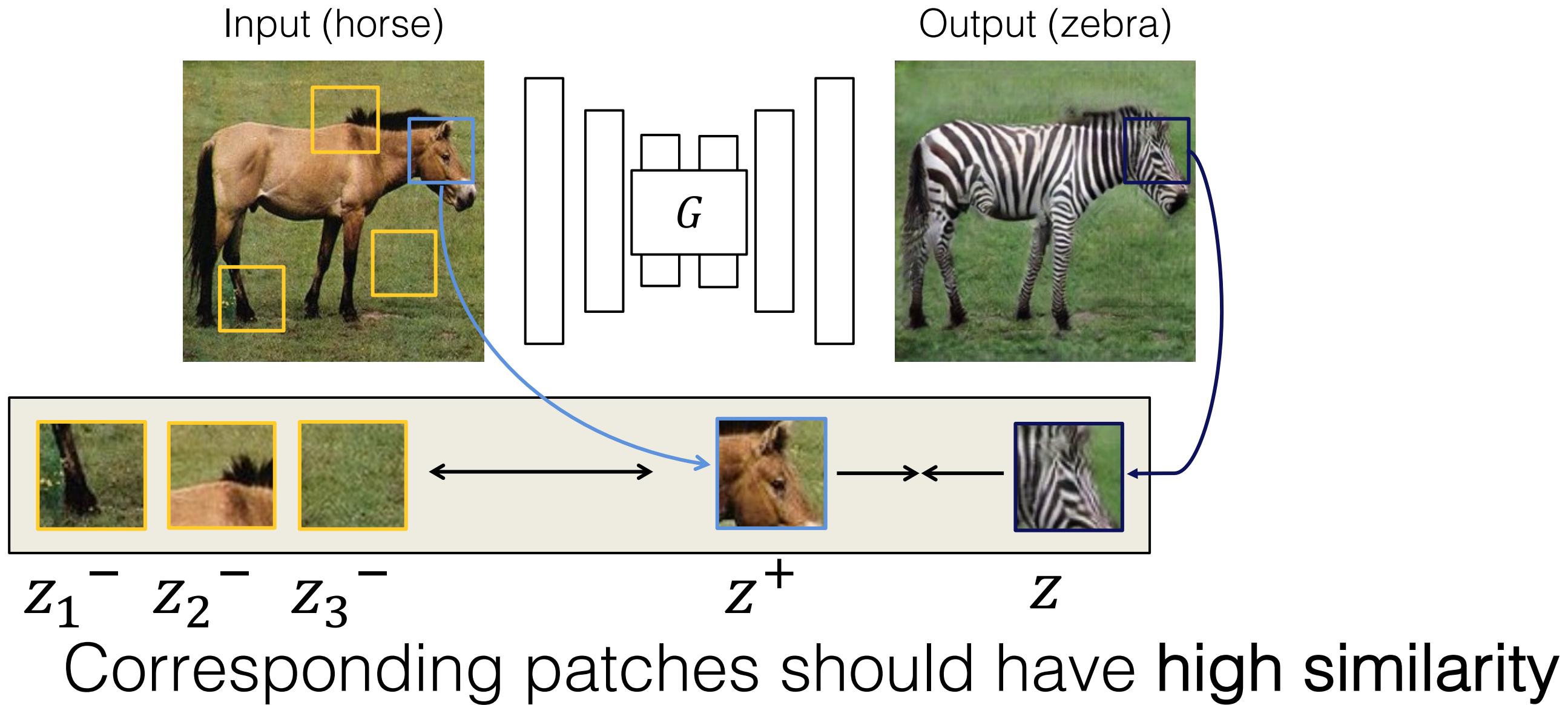
Output (zebra)



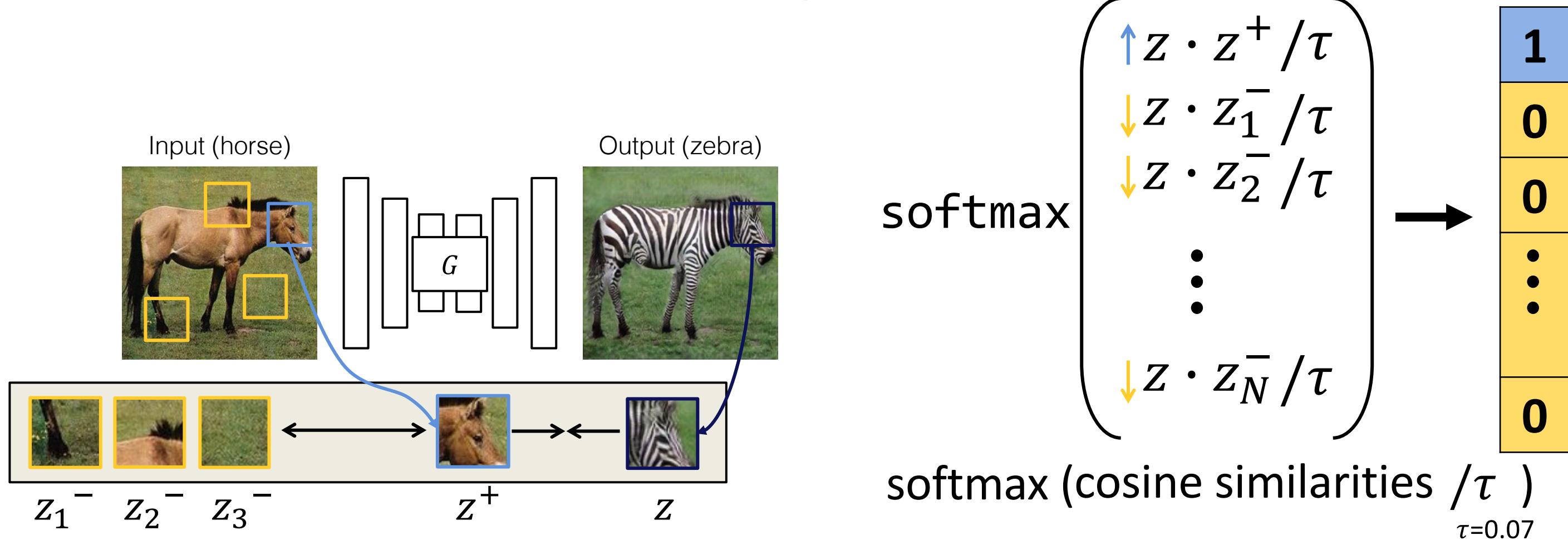
Discriminator



Retaining input content

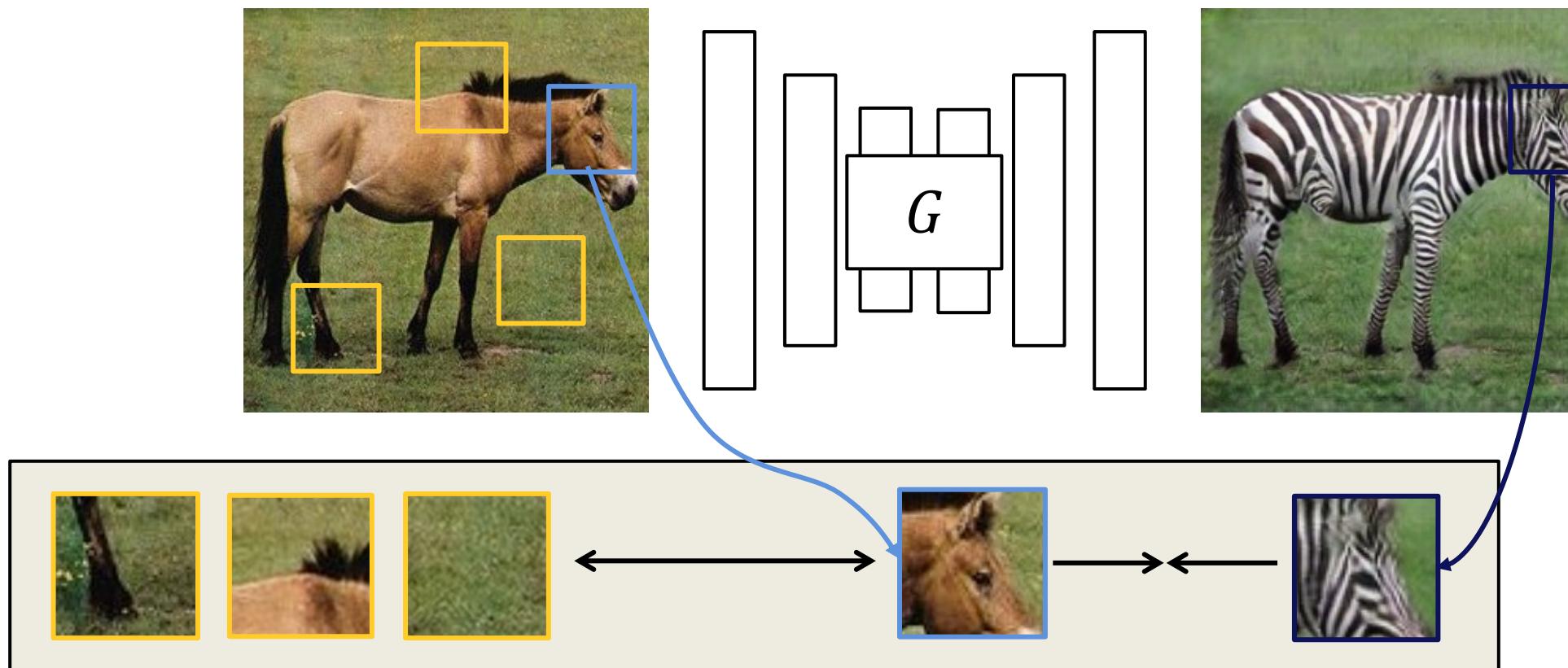


Patch-based Contrastive Loss

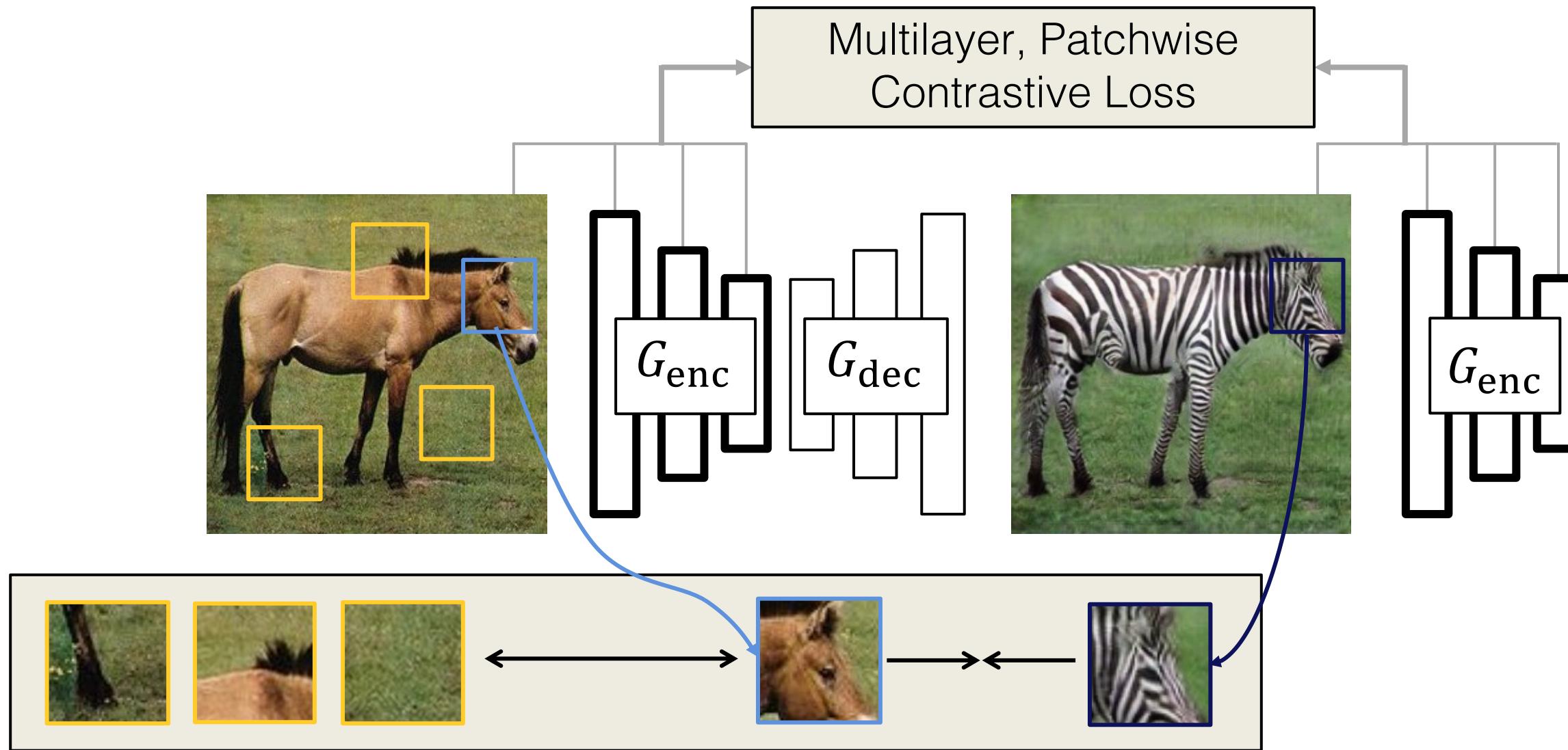


- InfoNCE loss (Gutmann et al., AISTATS18 , van den Oord et al., 2018) used in MoCo and SimCLR
- To produce positive pairs:
 - Handcrafted data augmentation (MoCo, SimCLR, etc.)
 - Input and synthesized image (ours)

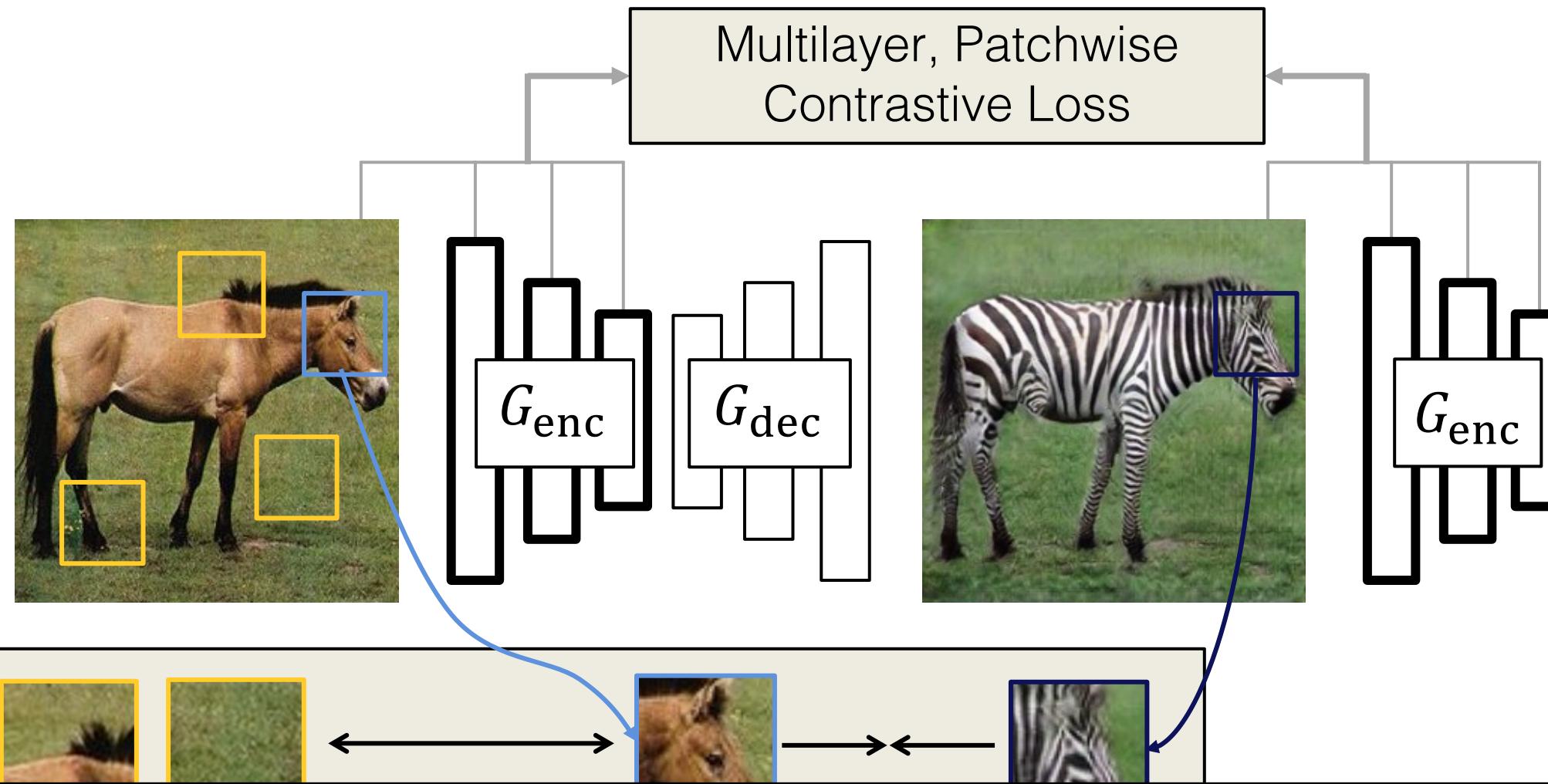
Patchwise contrastive loss



Patchwise contrastive loss

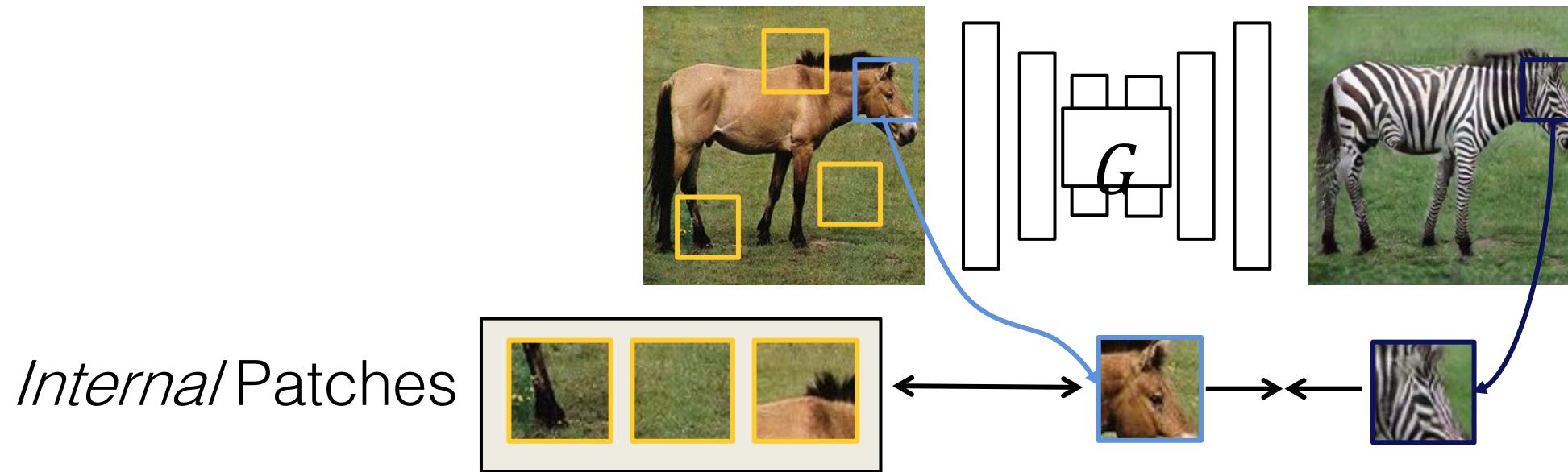


Patchwise contrastive loss

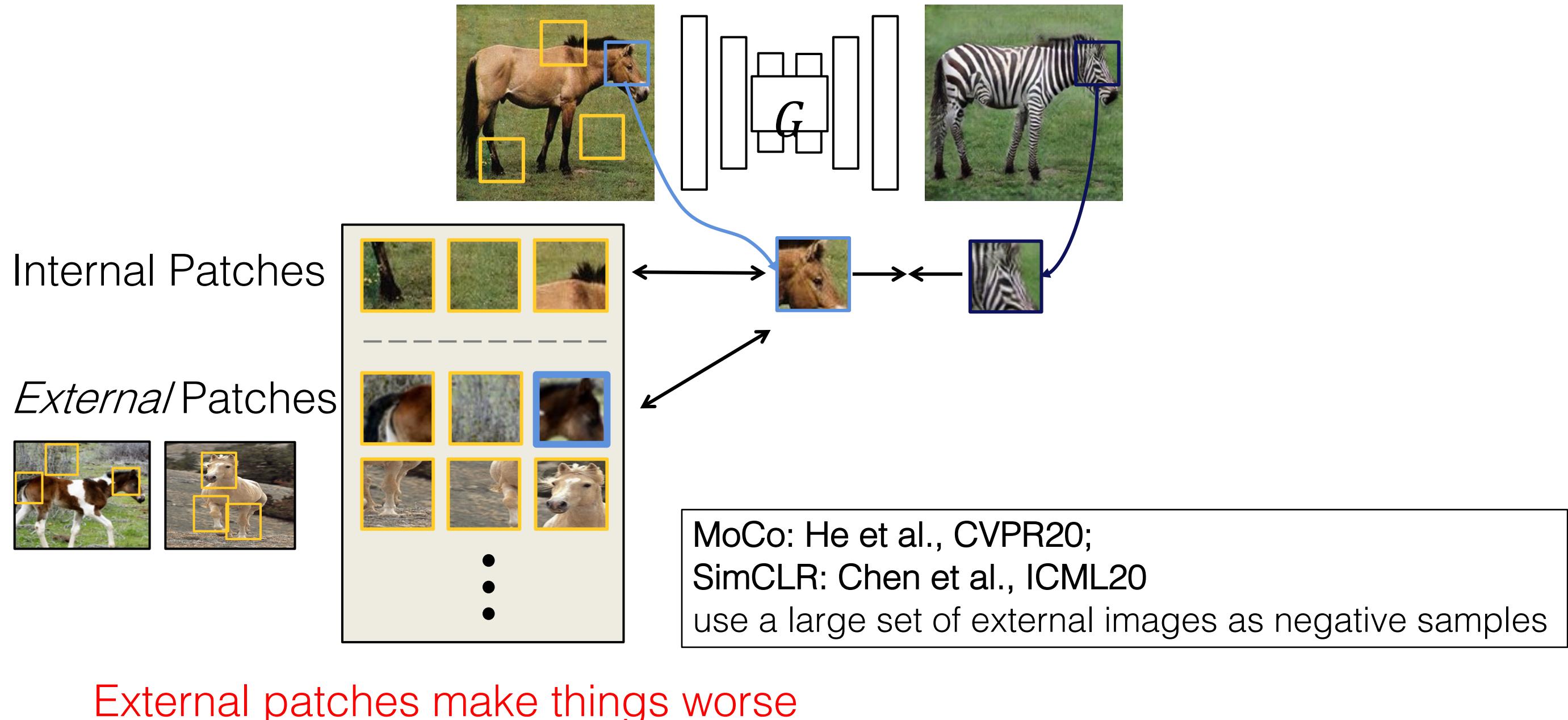


- + No fixed similarity metric (e.g., L1 or perceptual loss)
- + One-sided (no inverse mapping needed)

Internal vs External Patches

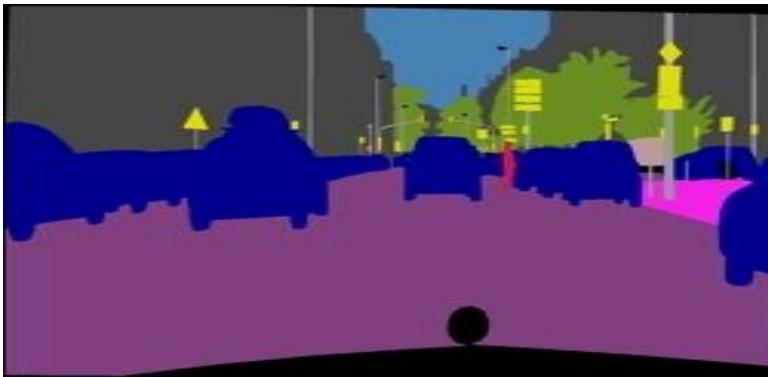
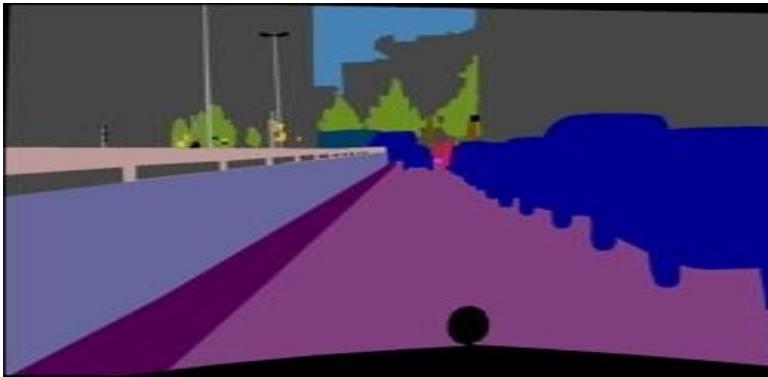


Internal vs External Patches



Internal vs External Patches

input



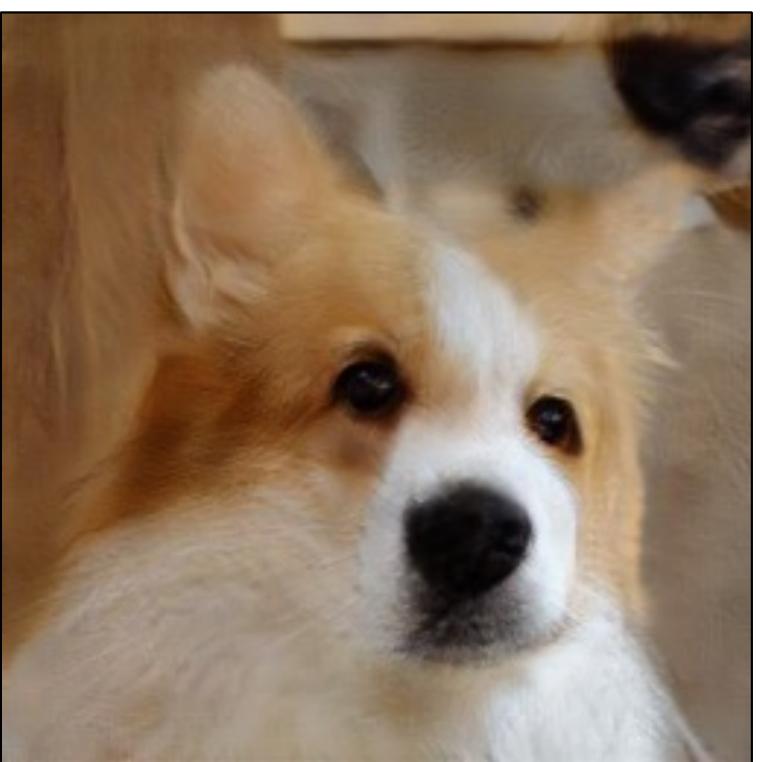
internal patches



external patches



Mode
Collapse!



Cat



Yosemite Summer



Apple



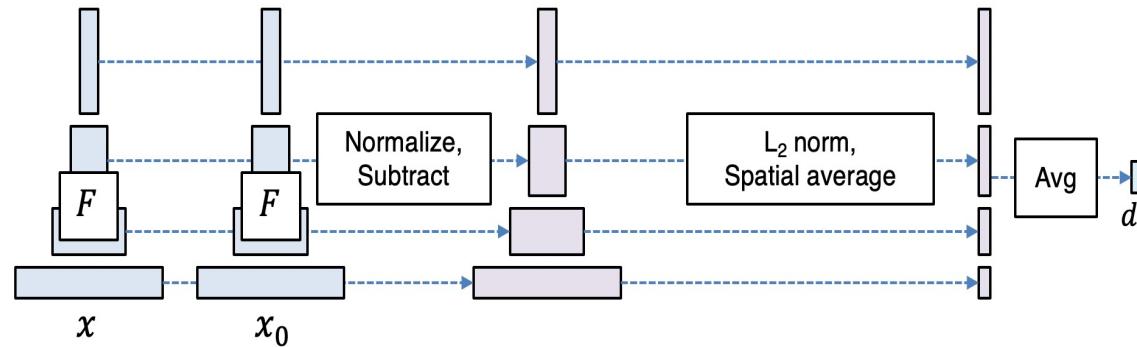
Paris



GTA

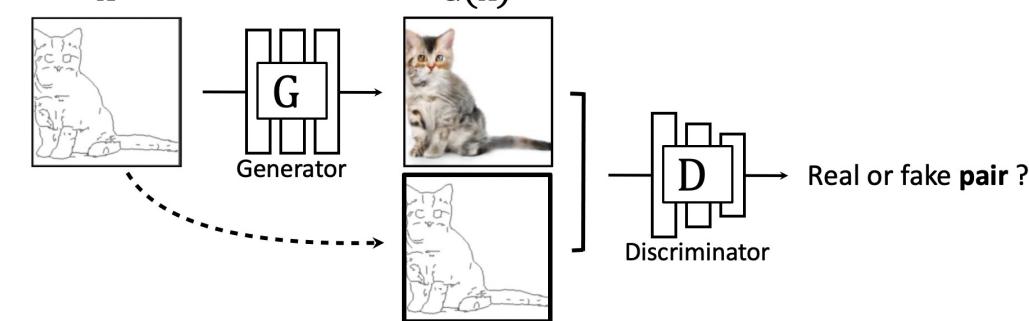
Review

Perceptual/Feature Loss



How well do “perceptual losses” describe perception?

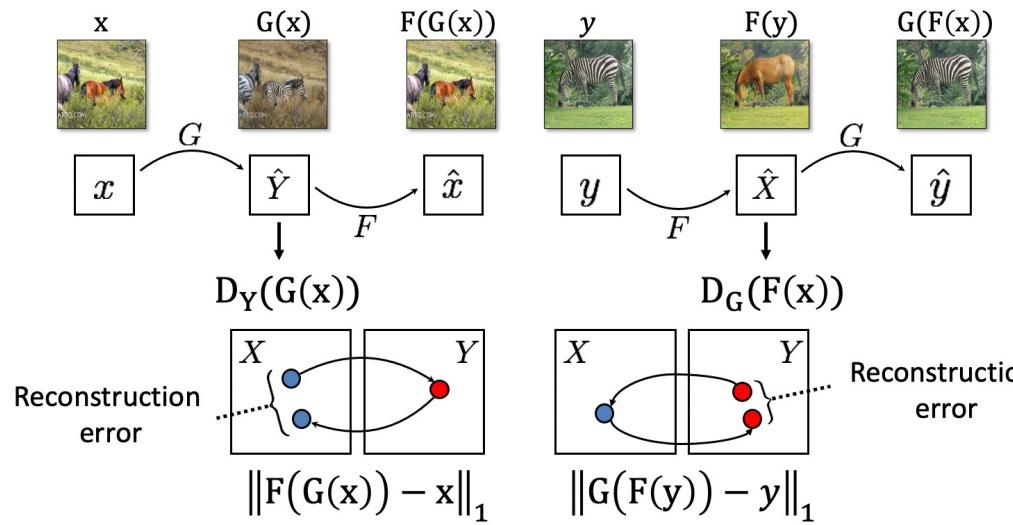
(Conditional) GAN Loss



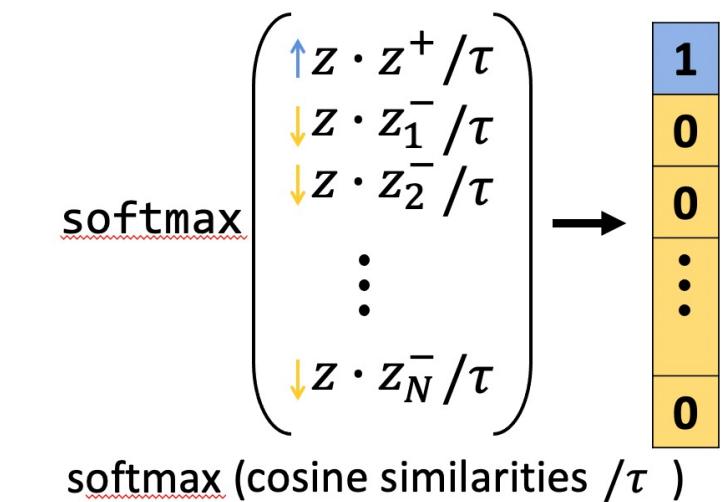
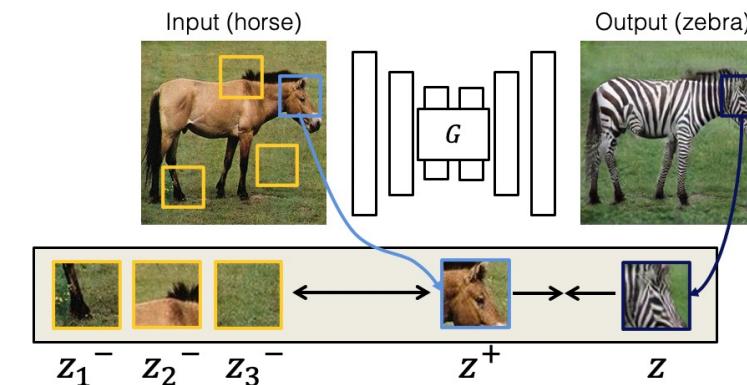
Learning objective

$$\min_G \max_D \mathbb{E}_x[\log(1 - D(\hat{x}, G(x)))] + \mathbb{E}_{x,y}[\log D(\hat{x}, y)]$$

Cycle-Consistency Loss



Patch-wise Contrastive Loss



Other loss functions: Style Loss [Gatys et al.], Contextual Loss [Mechrez et al.], Domain-specific Loss (e.g., face), 3D-aware Loss (for geometric data)



Style and Content, Texture Synthesis

Jun-Yan Zhu

16-726, Spring 2022

Texture

- Texture depicts spatially repeating patterns
- Many natural phenomena are textures



radishes



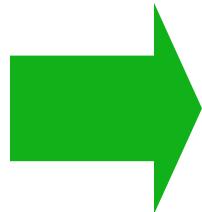
rocks



yogurt

Texture Synthesis

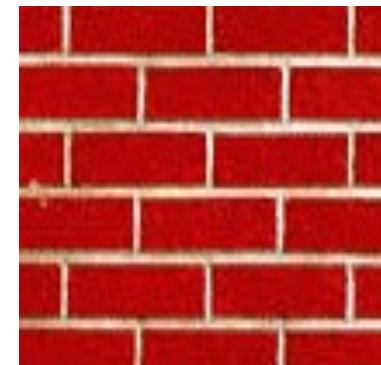
- Goal: create new samples of a given texture
- Applications: virtual environments, inpainting, texturing surfaces



Non-parametric Texture Synthesis

The Challenge

- Need to model the whole spectrum: from repeated to stochastic texture



repeated

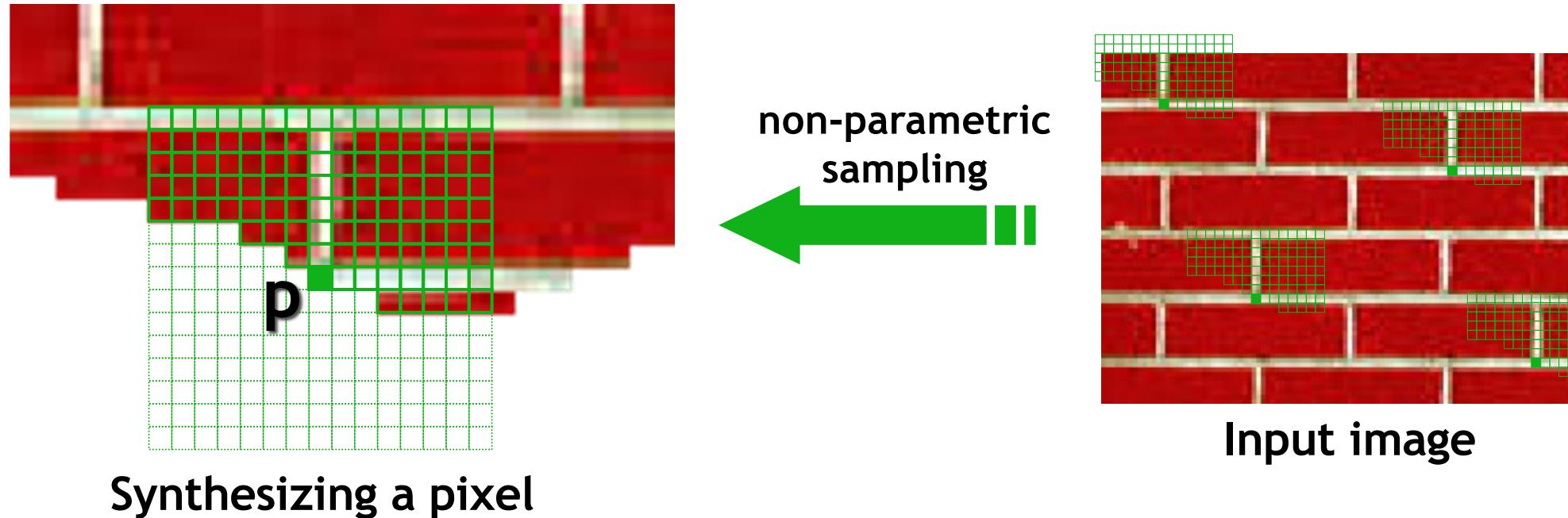


stochastic



Both?

Efros & Leung Algorithm

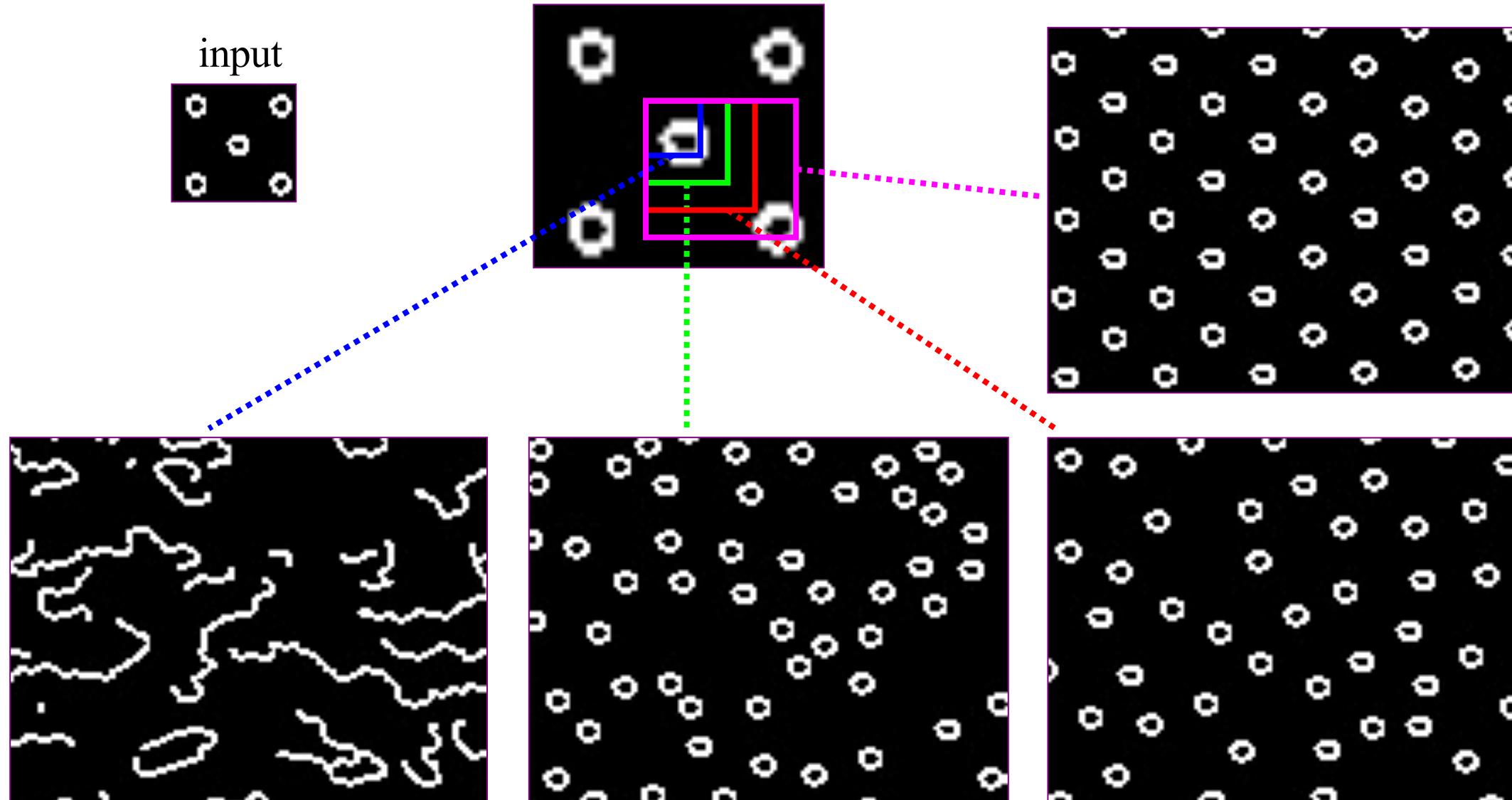


- Assuming Markov property, compute $P(p | N(p))$
 - Building explicit probability tables infeasible
 - Instead, we *search the input image* for all similar neighbourhoods — that's our pdf for p
 - To sample from this pdf, just pick one match at random

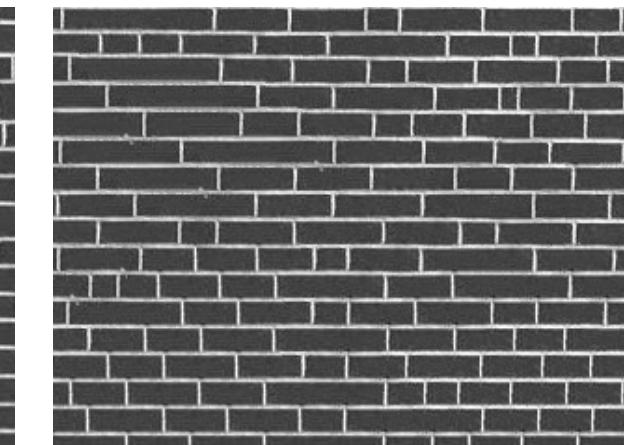
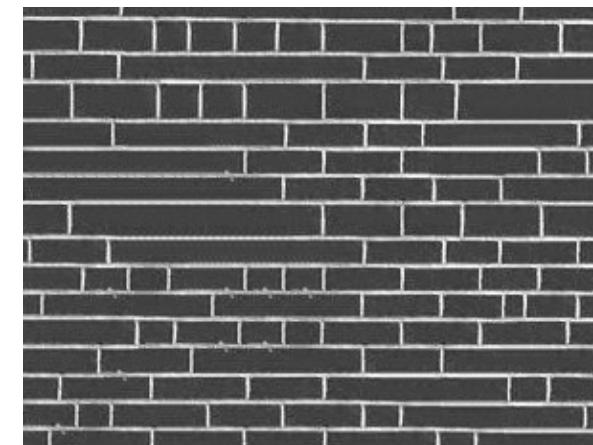
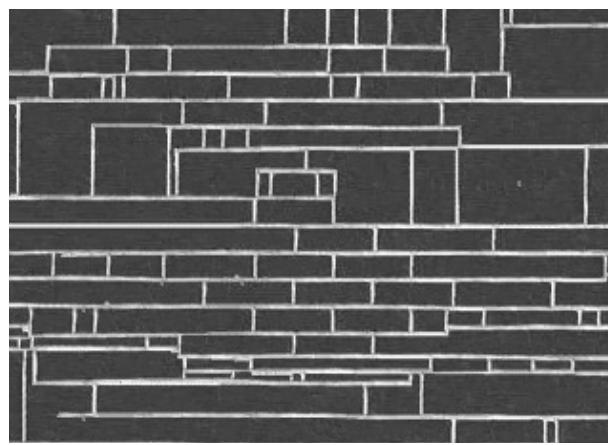
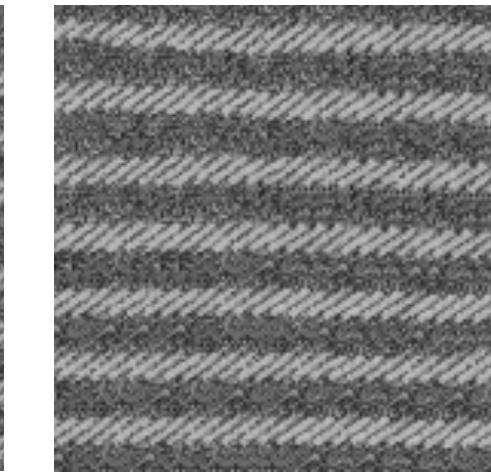
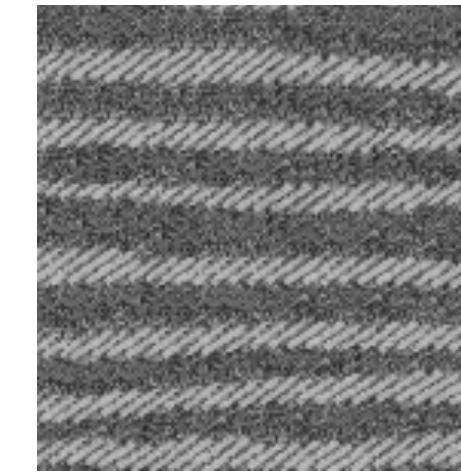
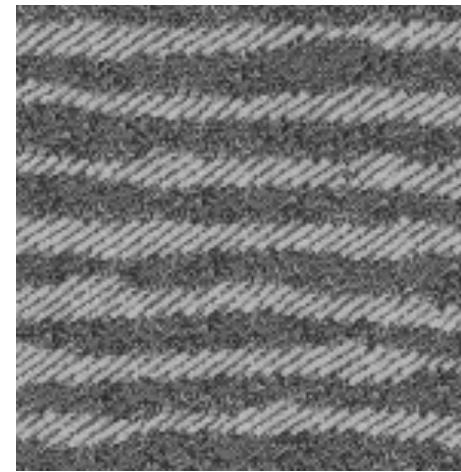
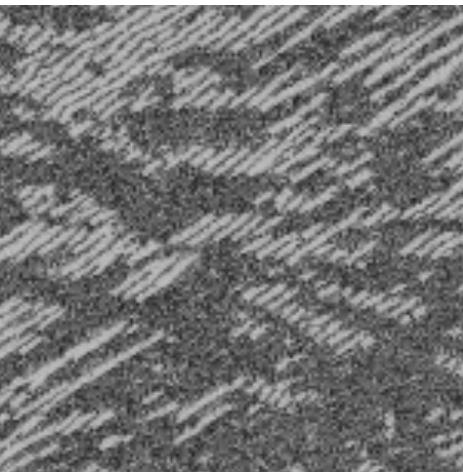
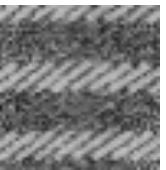
Some Details

- Growing is in “onion skin” order
 - Within each “layer”, pixels with most neighbors are synthesized first
 - If no close match can be found, the pixel is not synthesized until the end
- Using *Gaussian-weighted* SSD is very important
 - to make sure the new pixel agrees with its closest neighbors
 - Approximates reduction to a smaller neighborhood window if data is too sparse

Neighborhood Window



Varying Window Size

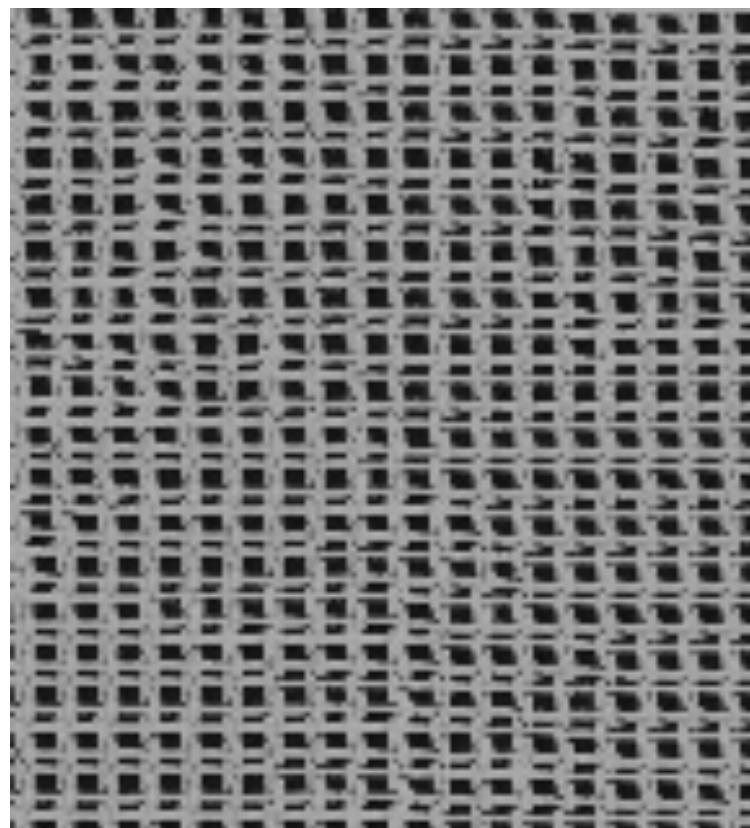
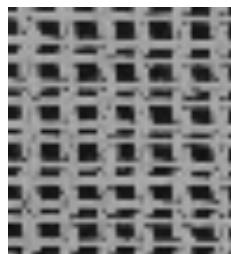


Increasing window size

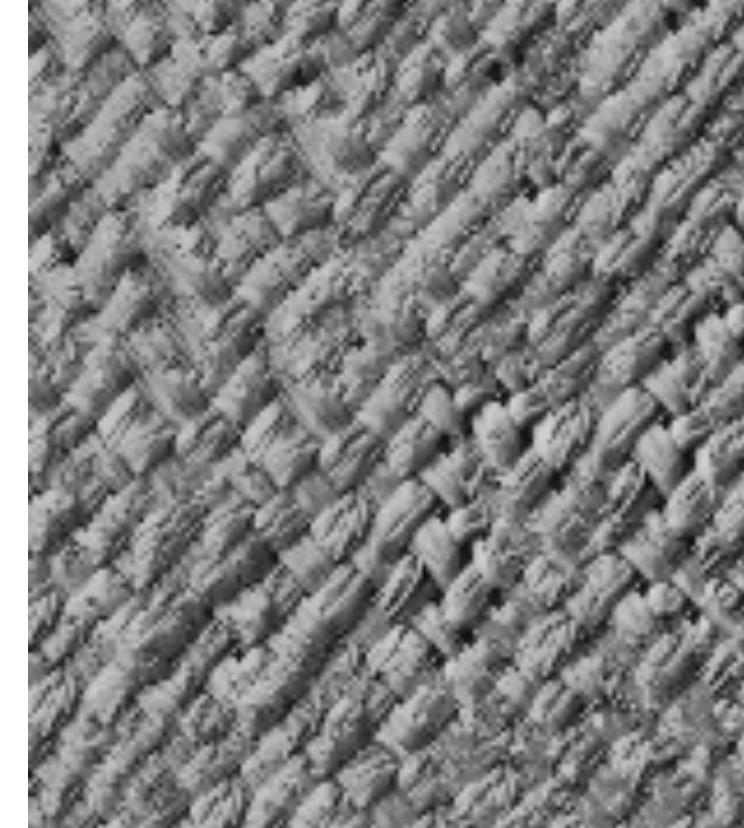
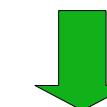
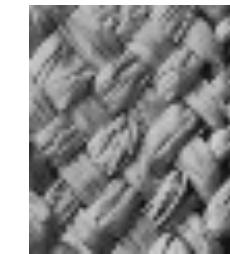


Synthesis Results

french canvas



rafia weave

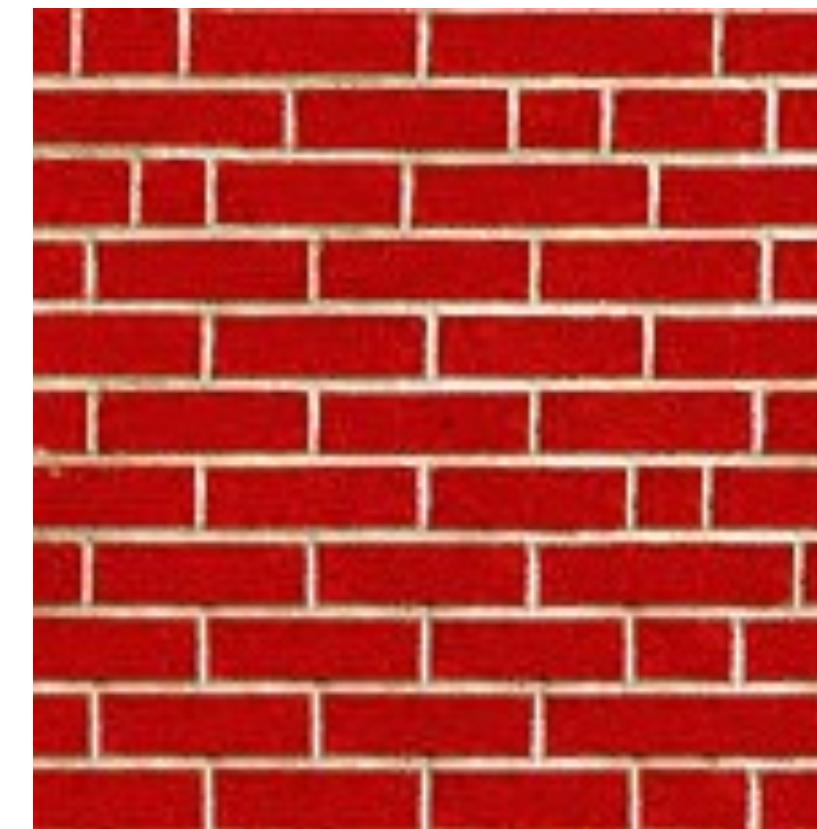
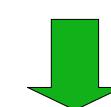
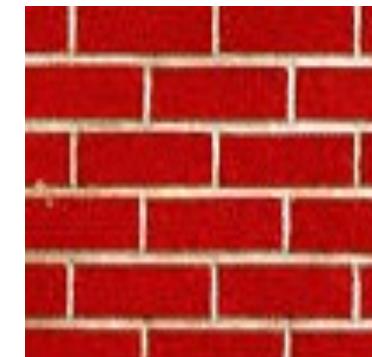


More Results

white bread



brick wall



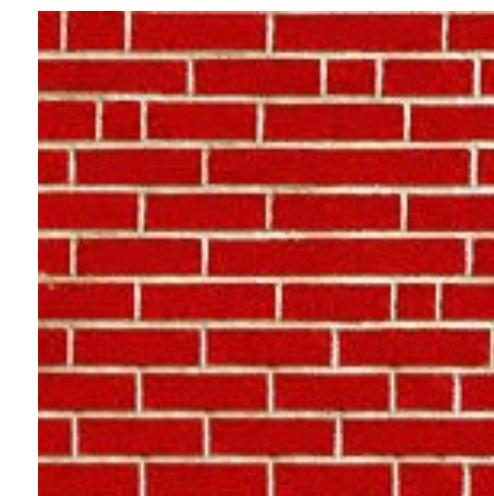
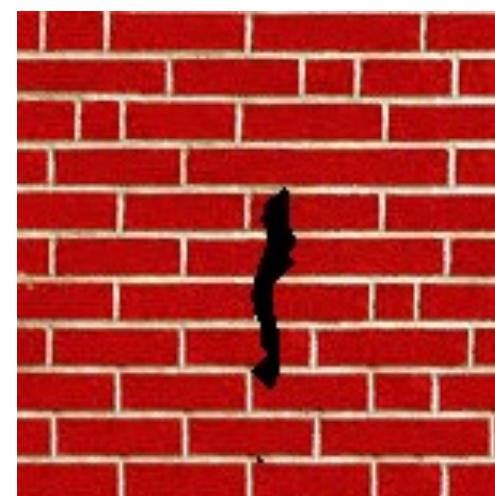
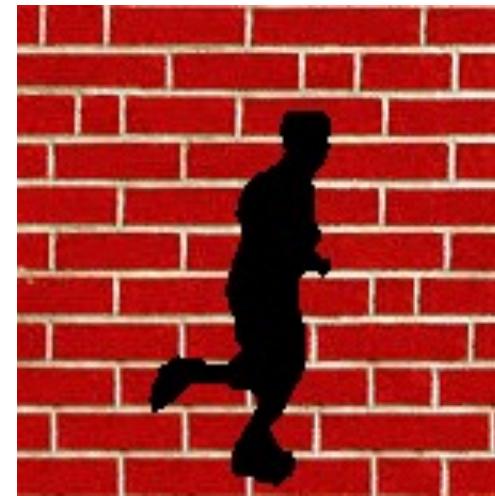
Homage to Shannon

uring in the unsensatior
r Dick Gephardt was fai
rful riff on the looming
nly asked, "What's your
tions?" A heartfelt sigh
story about the emergenc
es against Clinton. "Boy
g people about continuin
ardt began, patiently obs
, that the legal system h
e with this latest tanger

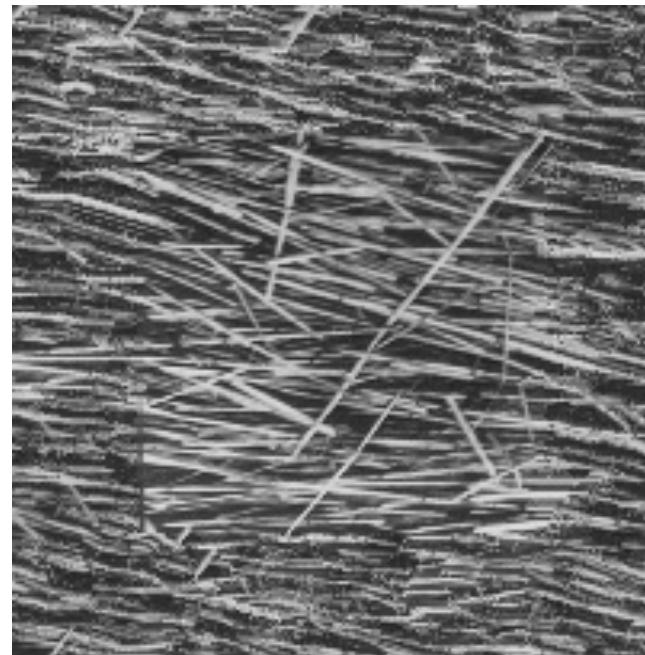
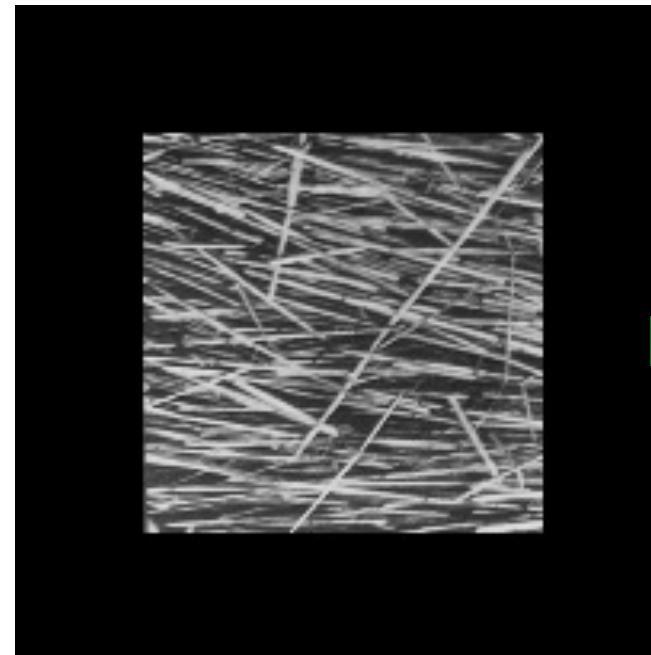
I b' s n o p i o o b' r i e h . A u g ' h i h t , t i m r a i thunne, t
e g l e p e r s o t o t t " l a t t i n u n i r t h t , f e o r i P u n t a
v a g ' a s " h e d t C u l , e i o r t a o r A i b
h r u . A , y f t e t , f i t a u s ? f i r t f o o e a r J e a r t h n
t i a e " t d i a b , b b i v e h u j f m r t e o p m t F U
t e f u n i t t , t u s l l t h r i s t d h p n r t i k
n t i s e u , w i o b n t u r n h r i s t d h p n r t i k
n i d r f P u e , c j d t g u l g l u n t o n u m u a b r y s
u t o n u c f u b e s , i t , f l g b l i t p a l e o b o s h i
j t h e n l y n A n f e r E l o a e u n o h , n B J V
d t h f p + d i b o o n t i n e o t i , h a m u u n y C r o b a b
x r , f a i f , J h o o e g a h m t t , s y o k e , i
t ' l b , s i h h s k a s h e k , y s h n , C " u - e t f l l ,
e e n s t h a f e c t d , t e i l , r h , C A r t s
u i t n t r p l t n i t , " t f f a f e c t d , t e i l , r h , C A r t s
s C o n h r u , t n l e g d t f f a f e c t d , t e i l , r h , C A r t s
s t h i n g e o r t t e j t y a b i n , A h m e ,
f h o t e , e t y e n , C i r y , - j i u r t , w e n n n r o p n y
d v i t e , f l o a e , w o e , K f f b b l i t r r n , " b t e i
t o e [p s] t h e l , t g e , f u i o u f u , H s v p n a r p a
t e n , " a r e n , l o u n t e , n t f u , C u t t t o , t h , p n i s , t , a
z m r i o p , p r t e , l u i t u r i , t w a j o b , u i t e , t e t l , i
B u n u p u n o r s y , p s a , h b , ' C " a i t i , a i w r , f b p
t , a i t i , a i w r , f b p

l h a i m , them . " W h e p h a r t f e l a r t i f e l i n o m i n e
f e l c k C l i r t i o o t o m a i m t h a r t f e l i n s , f a i t s a n e n t c
t h e r y o n s t w a r t f e l c k G e p h n t o o m i m e a t i o n l s i g a k
C h i o o u f i t C l i n u t C l l r i f f o n , h a t ' s y o d n , p a n t t l y :
o n s y o n t o n s t e h t w a s k e d , p a i m t s a h e l o o r i f f o n l
n s k o n e p l o o u r t f e a s l e i l A n s t C l i t , " W h e o n t o n g a l s
k C i r t i o o u i r t f e p e ó n g p r e a b e g a l f a r t f e n s t e m e m
t i e n s t e n e l t o r y d t t e l e m e p h n i s v e r d t w a s a g e m e n
f f o n s a r t i e n t o n t C l i n g p r e m e a s u r t f e a t i h , " B o u i s
n a l s f a r t f e l t s i g p e d r i l d t s k e a b o u n u t i e a b o u t i o o
s t f e a o n e w a s y o u a b o u n t h a r d t t h a t i n s f a i n , p e d ,
s a i n s , t h e m , p a b o u t w a s y a r f u t c o u i t l y d , l n A h
p l e e i n t h r i n g b o o r e m e a g a s f a b o n t i s y s t C l i n u t
o r y a b o u t c o n i n s t C l i p e o u i n s t C l o k e a g a t i f f o u t C
s t o m e z i n e m e n t l y a r d t b e o r a b o u l n , t h e n l y a s t C
c o n s f a i m e m e D i o n t o n t w a t c o u t l y o h g a n s a s f a n
i e n , p h r t f a u l , " W b a u t c o u t c o n g a g a l c o m i n g a
m i f m s t C l i y a b o n a l c o o u n t h a e m u n g a i t t f o u n
W h e l o o r y s t e n l o o n t i e p h , l n t l y o n , t h e o p l e g a t i c k c
n u l t a t i e z o n t l y a t i e D i o n t o m i w a l s f t b e g à e e n e r
n t h a h e a t ' s e n e n h i h m a s f a n , " i n t c h t h n r w a h o n s v

Hole Filling



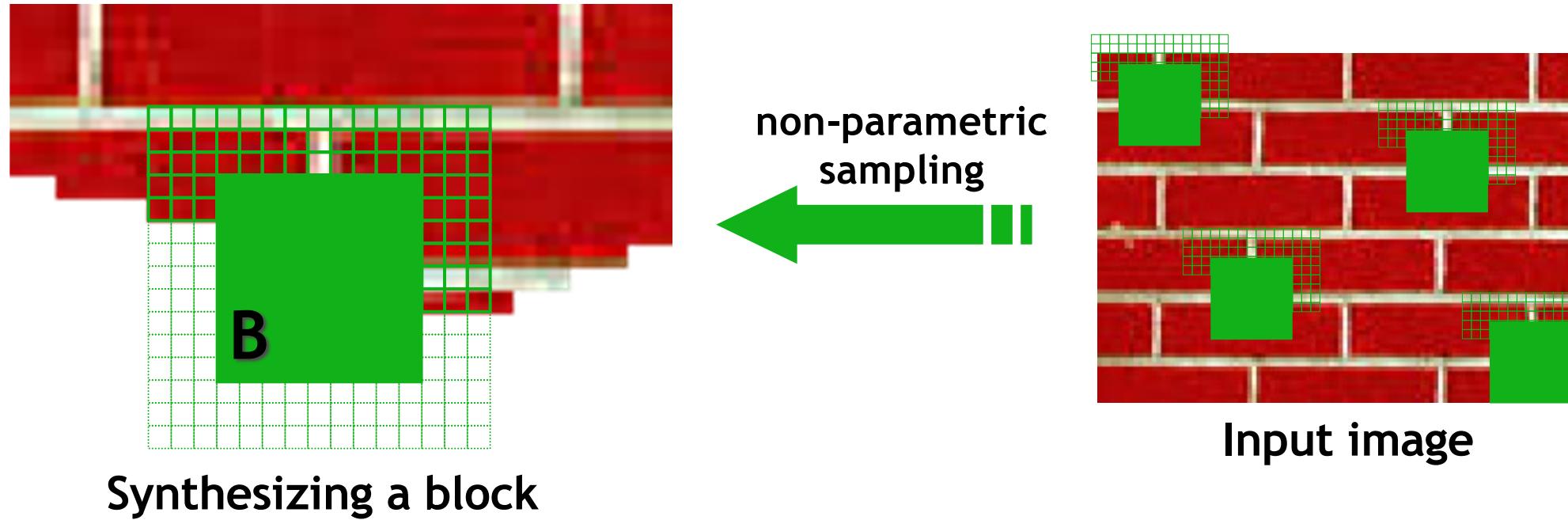
Extrapolation



Summary

- The Efros & Leung algorithm
 - + Very simple
 - + Surprisingly good results
 - + Synthesis is easier than analysis!
 - ...but very slow

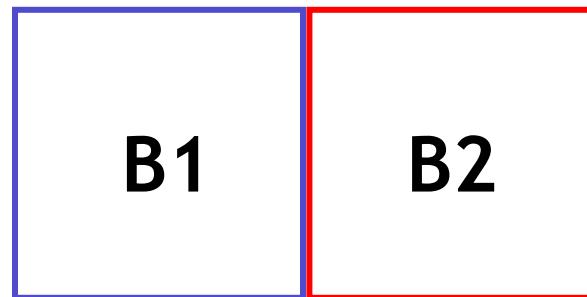
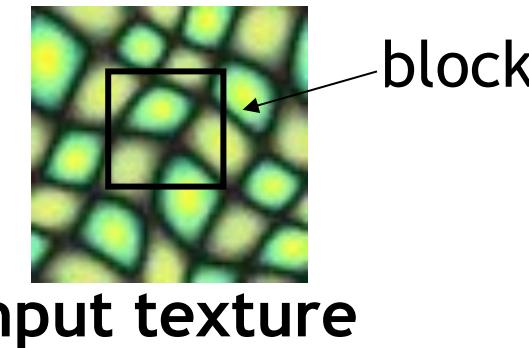
Image Quilting [Efros & Freeman]



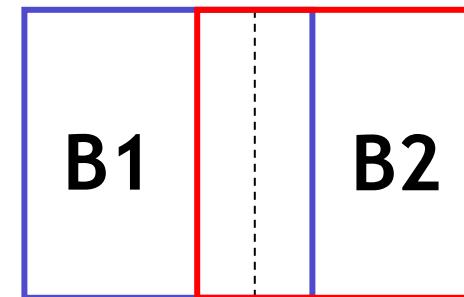
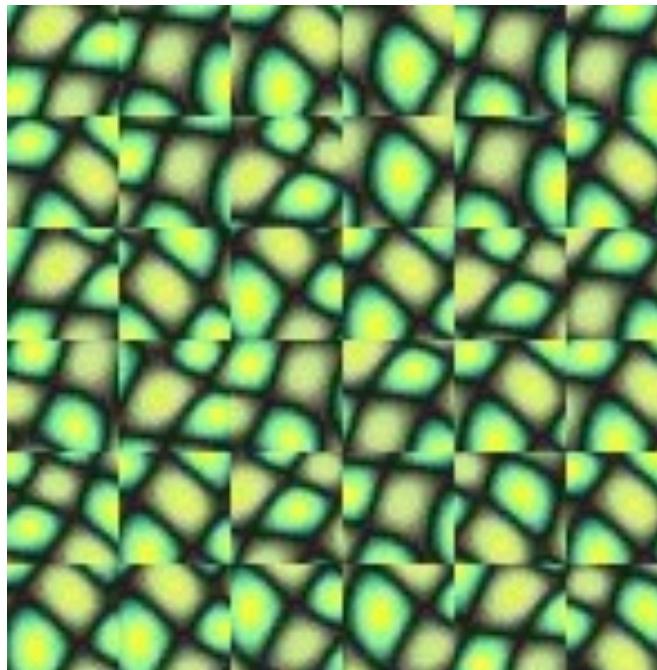
- Observation: neighbor pixels are highly correlated

Idea: unit of synthesis = block

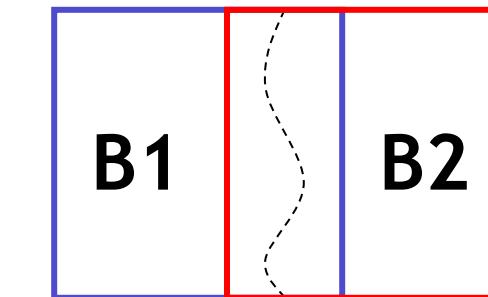
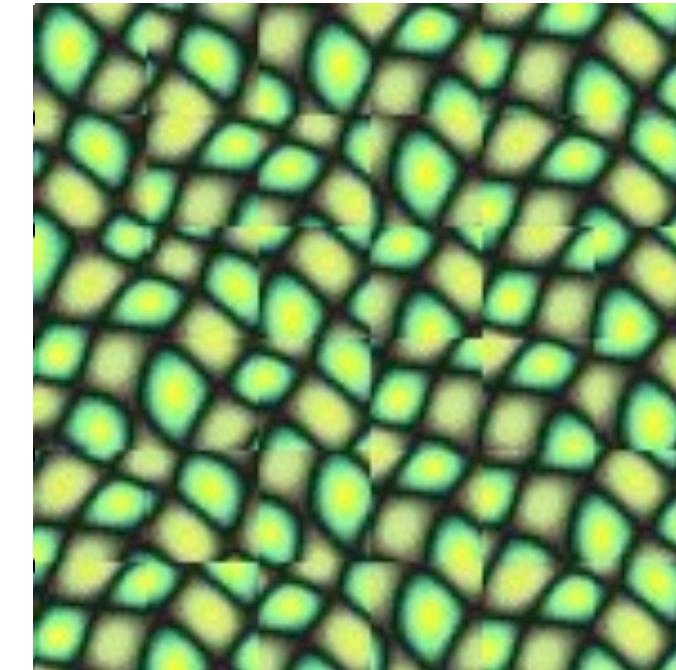
- Exactly the same but now we want $P(B | N(B))$
- Much faster: synthesize all pixels in a block at once
- Not the same as multi-scale!



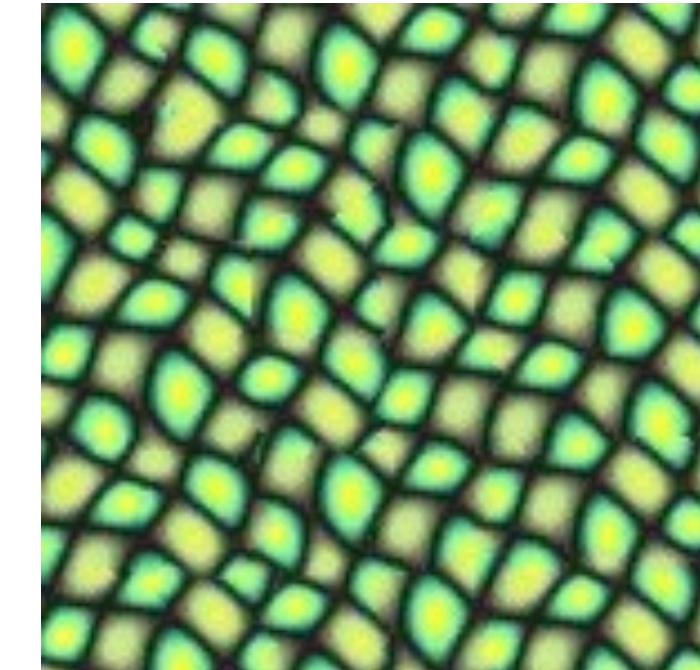
Random placement
of blocks



Neighboring blocks
constrained by overlap

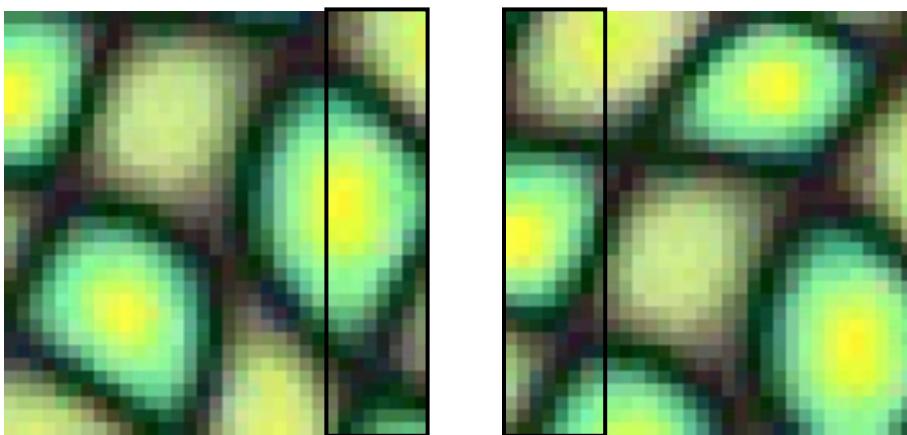


Minimal error
boundary cut

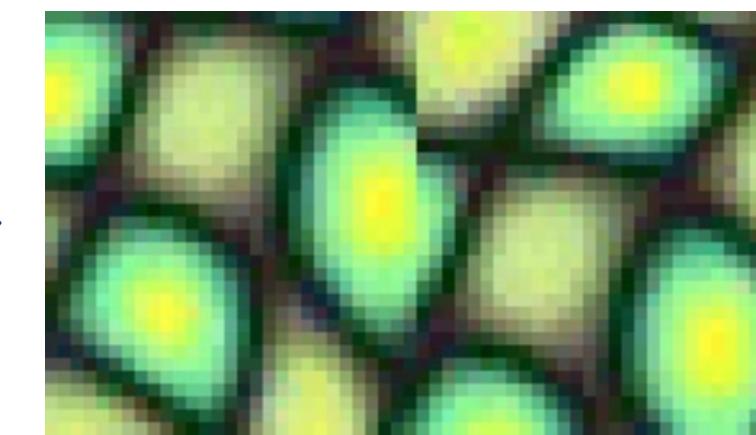


Minimal error boundary

overlapping blocks

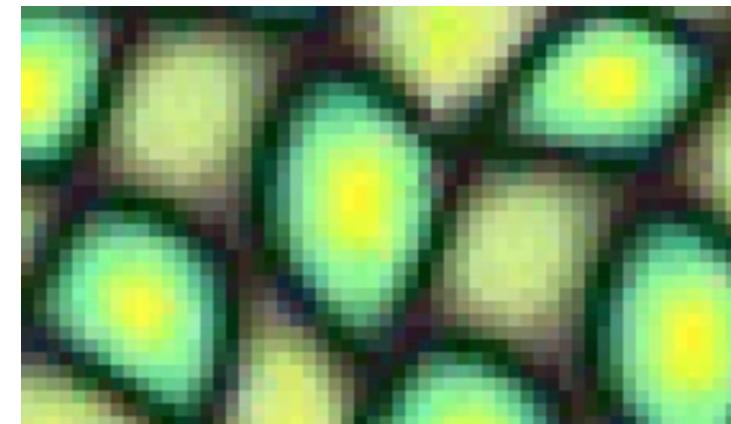


vertical boundary



$$\left(\begin{array}{c} \text{[Heatmap block]} \\ - \\ \text{[Heatmap block]} \end{array} \right)^2 = \text{[Binary mask with red border]}$$

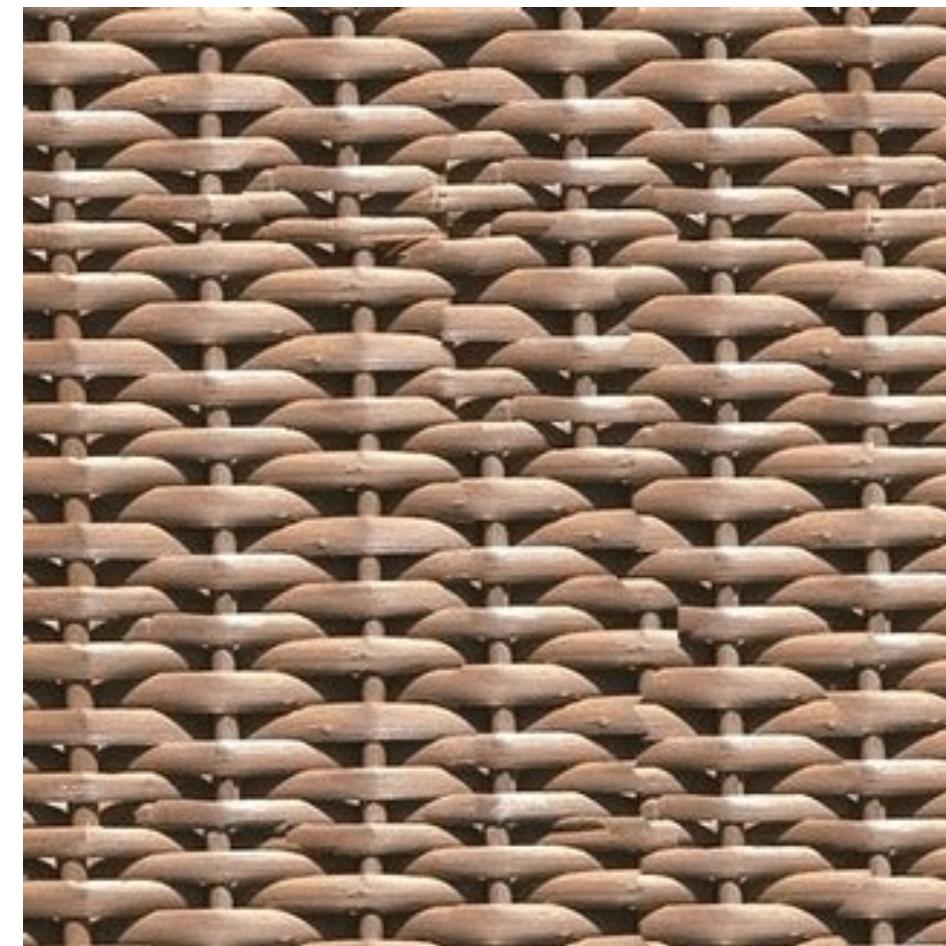
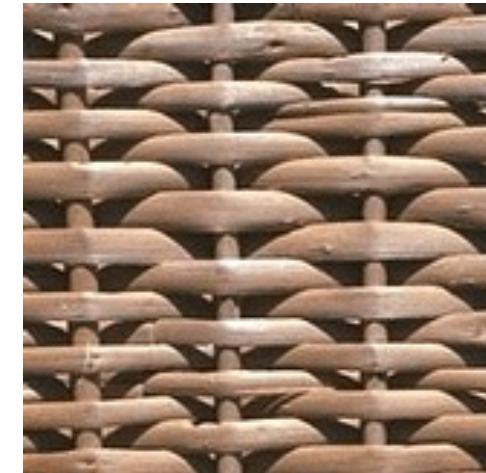
overlap error

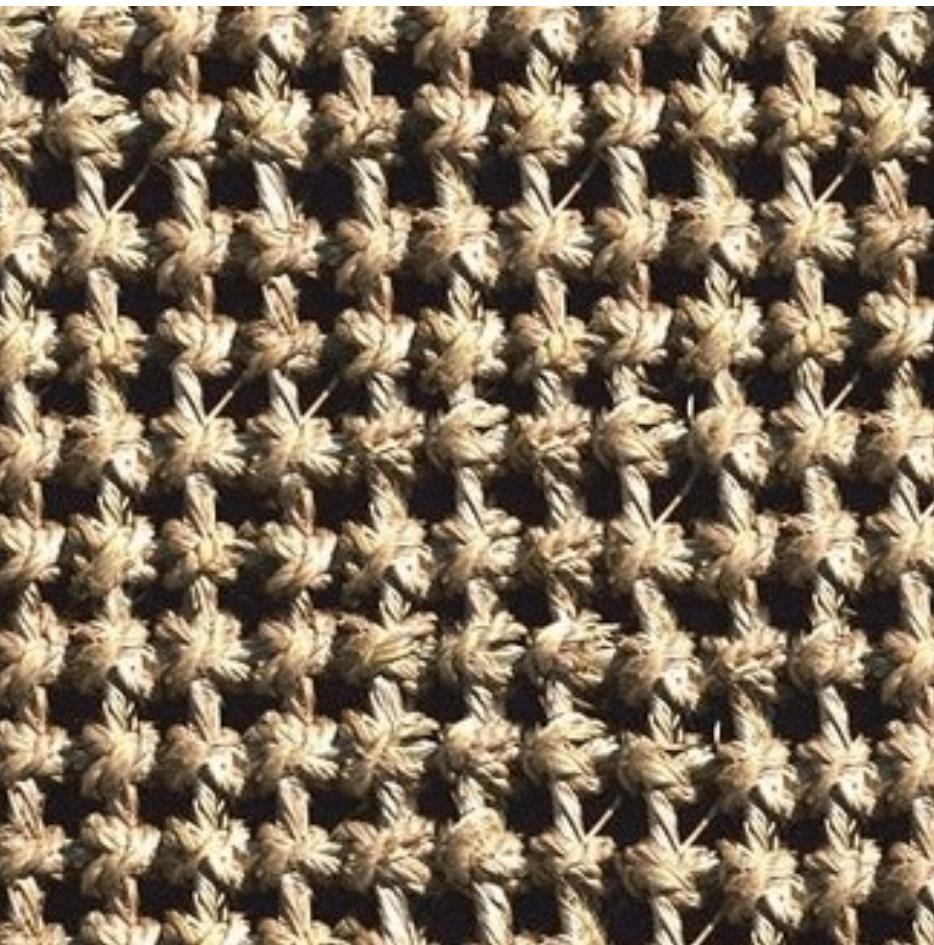


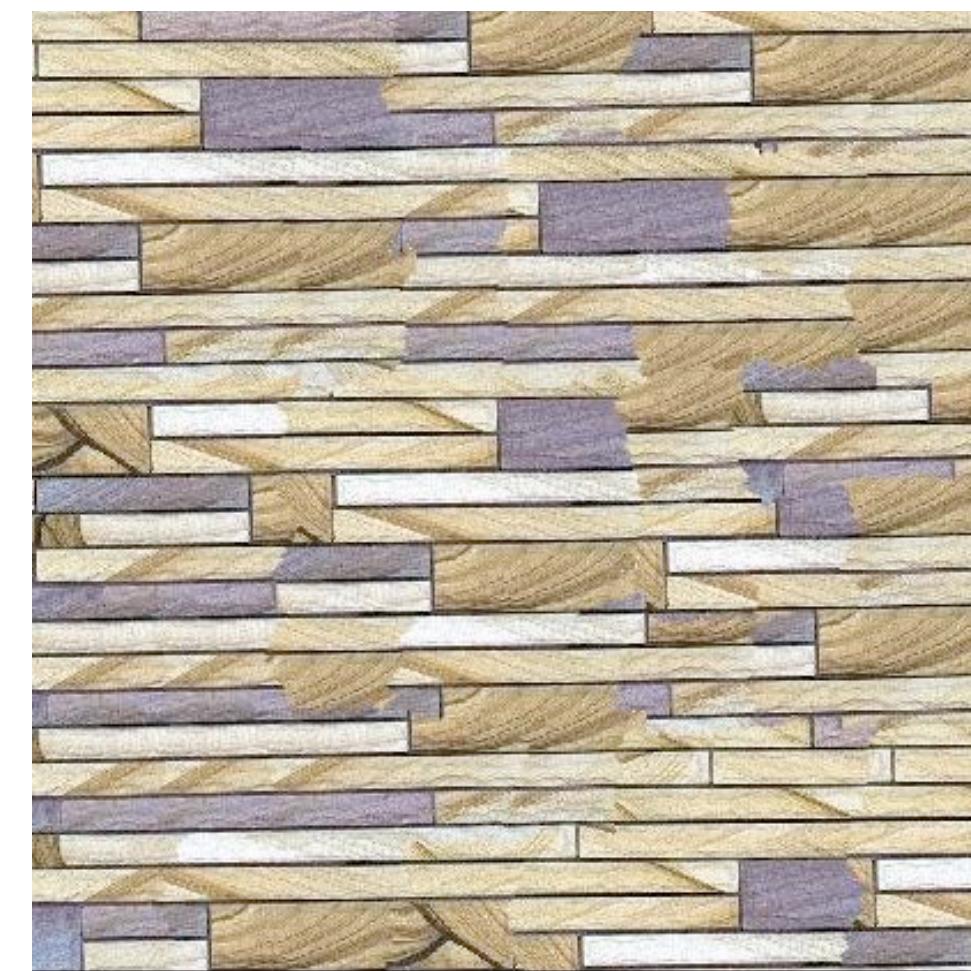
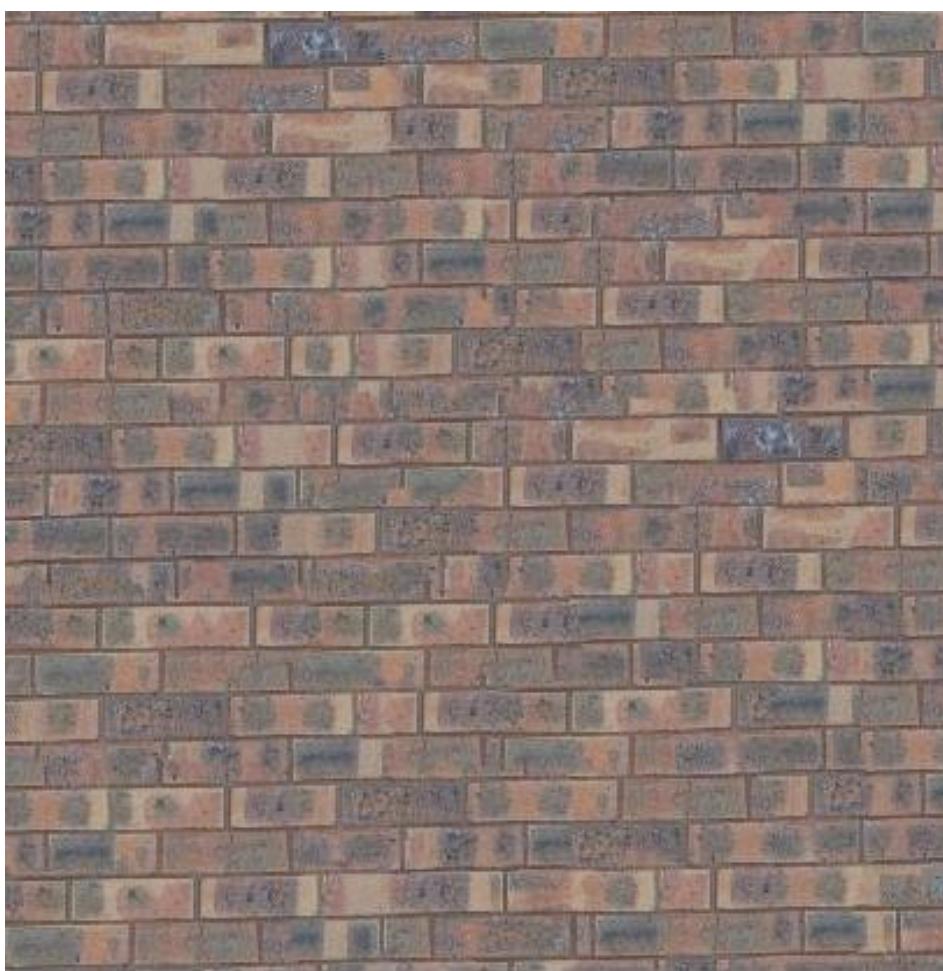
min. error boundary

Our Philosophy

- The “Corrupt Professor’s Algorithm”:
 - Plagiarize as much of the source image as you can
 - Then try to cover up the evidence
- Rationale:
 - Texture blocks are by definition correct samples of texture so problem only connecting them together

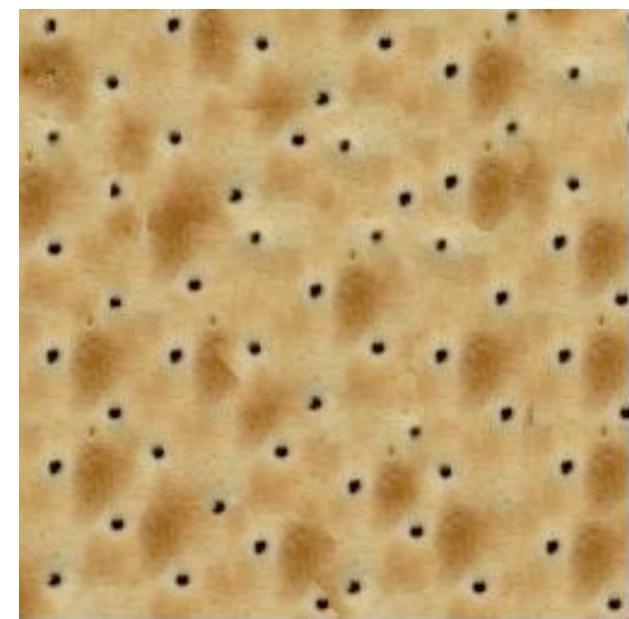
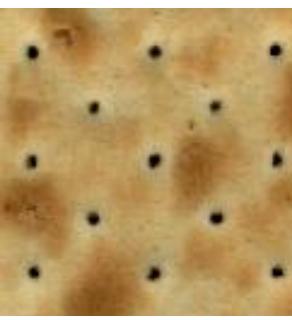






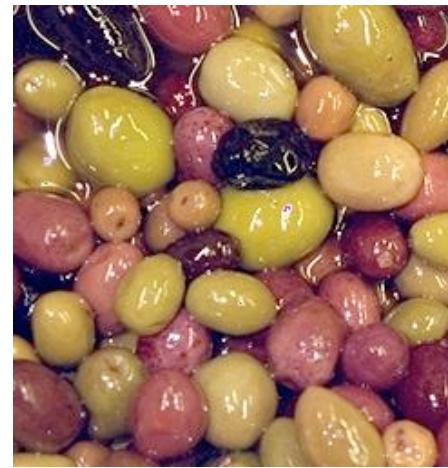


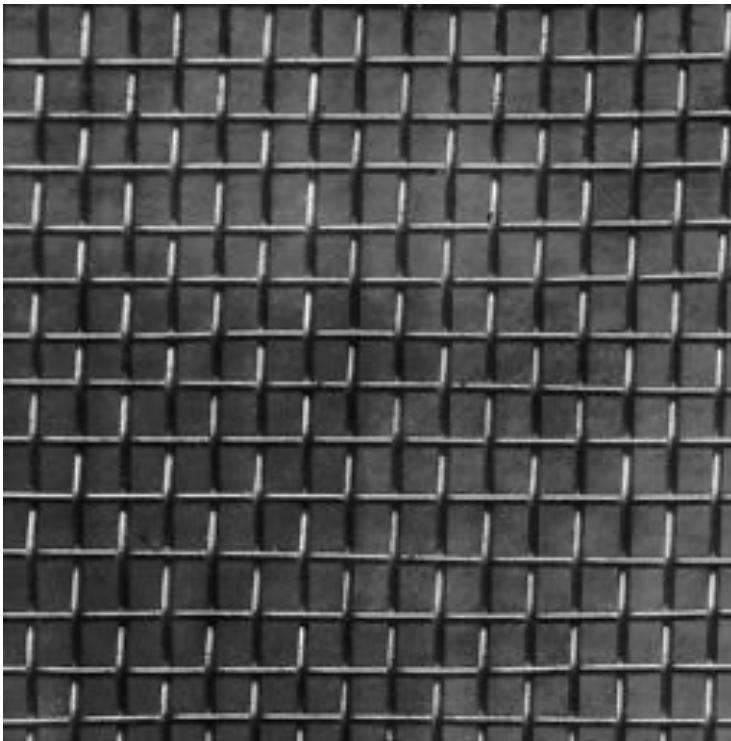




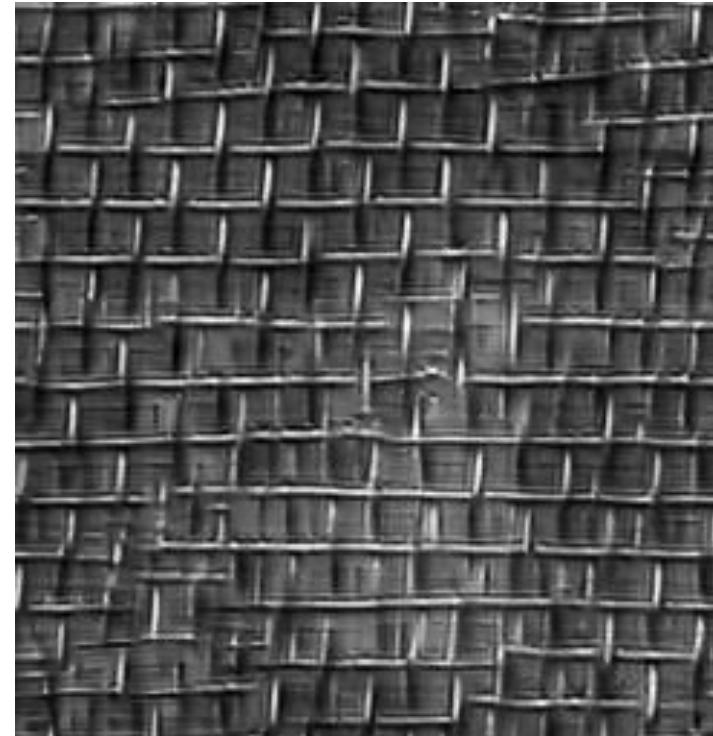


Failures (Chernobyl Harvest)

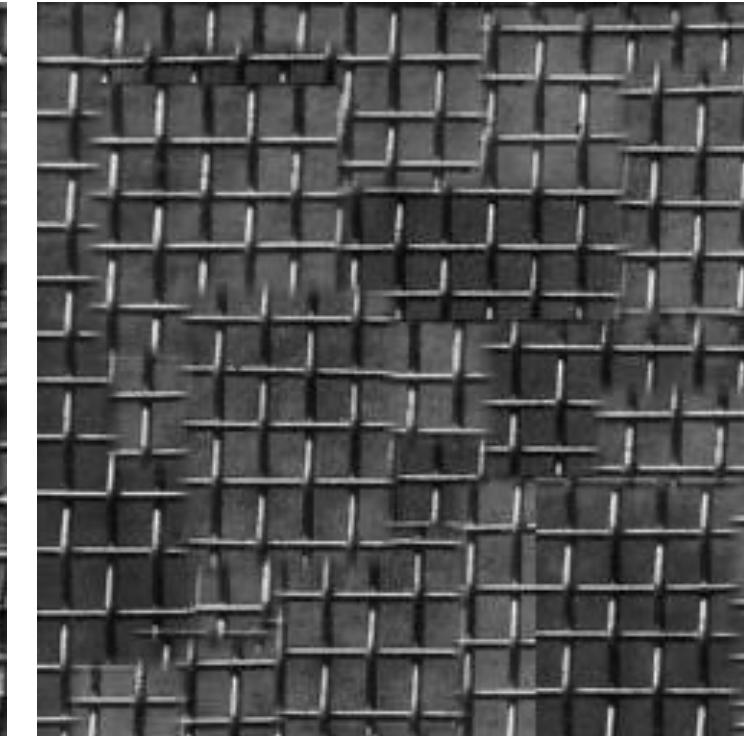




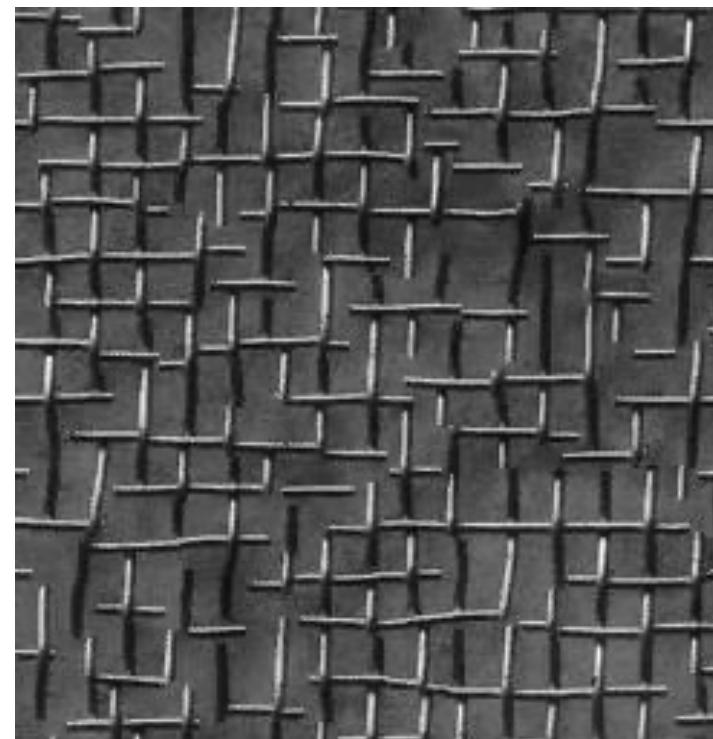
input image



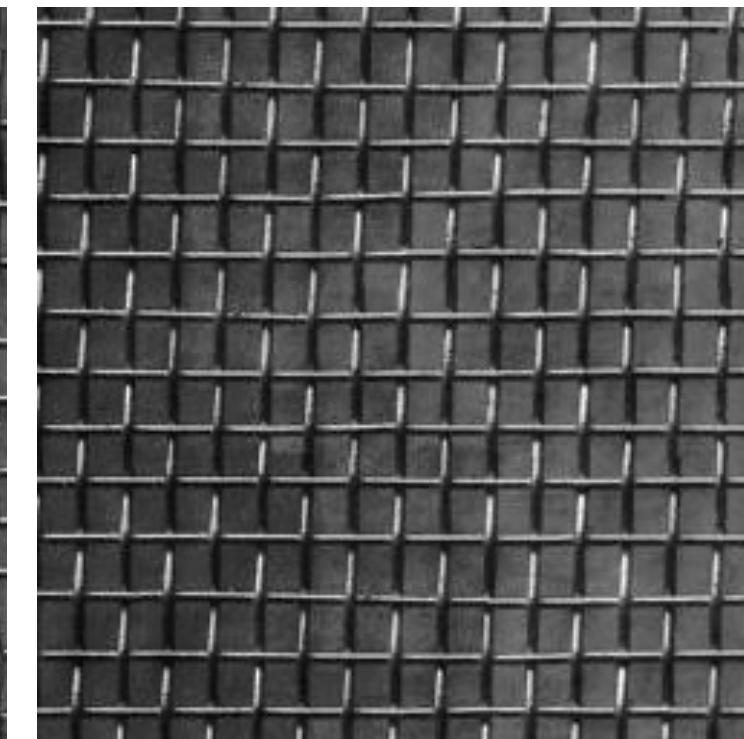
Portilla & Simoncelli



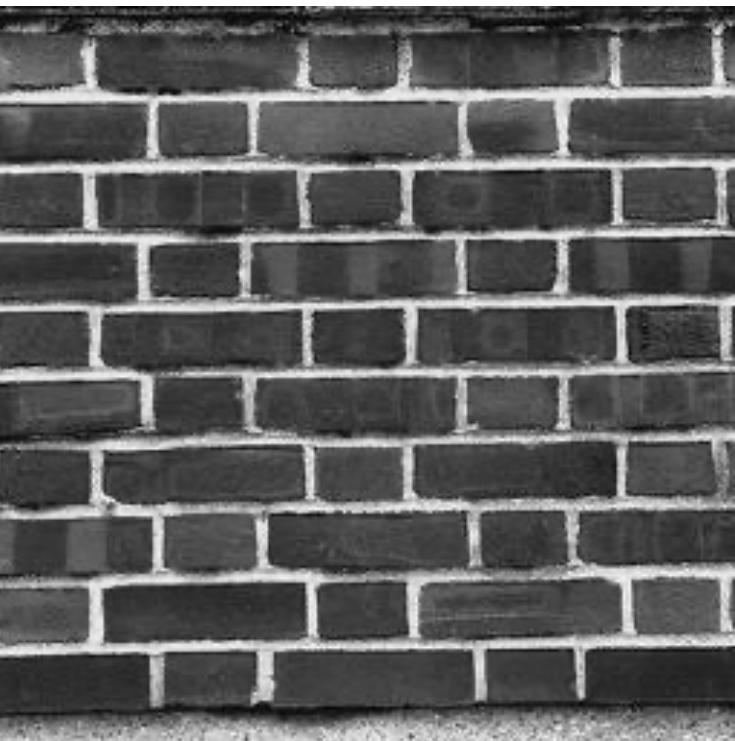
Xu, Guo & Shum



Wei & Levoy



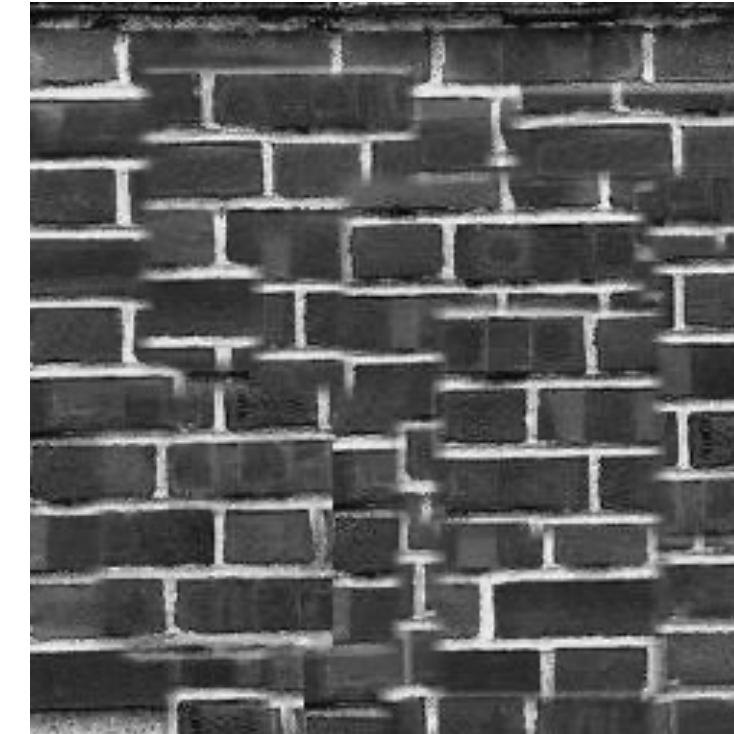
Efros and Freeman



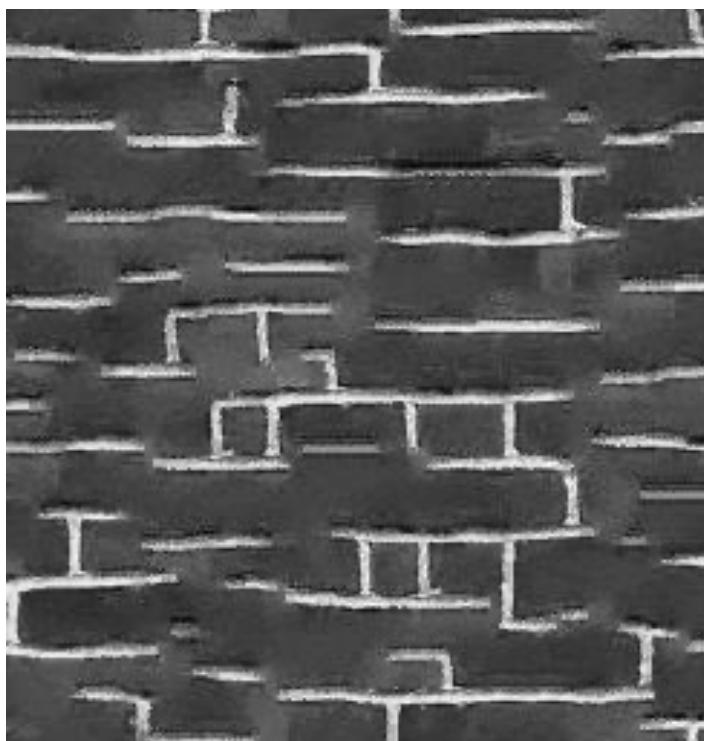
input image



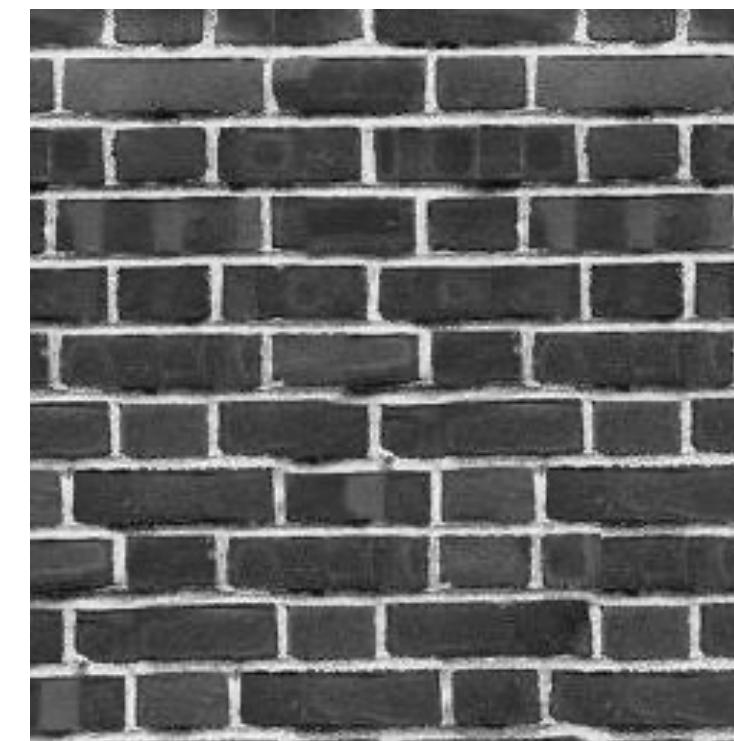
Portilla & Simoncelli



Xu, Guo & Shum



Wei & Levoy



Efros and Freeman

eru or a visual cortical neuron—the model describing the response of that neuron—*as a function of position*—is perhaps the most functional description of that neuron. We seek a single conceptual and mathematical framework to describe the wealth of simple-cell receptive fields neurophysiologically¹⁻³ and inferred especially if such a framework has the benefit of helping us to understand the function in a deeper way. Whereas no generic model of simple-cell receptive fields (e.g., Gaussians (DOG), difference of offset Gaussians (DOG), derivative of a Gaussian, higher derivatives of a Gaussian, etc.)—can be expected to fully describe the wealth of simple-cell receptive fields, we nonetheless believe that such a framework can help.

input image

but allows for greater generality than the simple-cell receptive field as a function of position. This allows for more complex receptive fields, including those that are not local or stationary. For example, it allows for fields that summate over time, space, or both. In addition, this model can incorporate multiple offset Gaussians, derivatives of multiple offset Gaussians, and higher derivatives of multiple offset Gaussians. This allows for a more detailed description of the receptive field, including non-local and non-stationary features.

eru or a visual cortical neuron—the model describing the response of that neuron—*as a function of position*—is perhaps the most functional description of that neuron. We seek a single conceptual and mathematical framework to describe the wealth of simple-cell receptive fields neurophysiologically¹⁻³ and inferred especially if such a framework has the benefit of helping us to understand the function in a deeper way. Whereas no generic model of simple-cell receptive fields (e.g., Gaussians (DOG), difference of offset Gaussians (DOG), derivative of a Gaussian, higher derivatives of a Gaussian, etc.)—can be expected to fully describe the wealth of simple-cell receptive fields, we nonetheless believe that such a framework can help.

Portilla & Simoncelli

eru or a visual cortical neuron—the model describing the response of that neuron—*as a function of position*—is perhaps the most functional description of that neuron. We seek a single conceptual and mathematical framework to describe the wealth of simple-cell receptive fields neurophysiologically¹⁻³ and inferred especially if such a framework has the benefit of helping us to understand the function in a deeper way. Whereas no generic model of simple-cell receptive fields (e.g., Gaussians (DOG), difference of offset Gaussians (DOG), derivative of a Gaussian, higher derivatives of a Gaussian, etc.)—can be expected to fully describe the wealth of simple-cell receptive fields, we nonetheless believe that such a framework can help.

Xu, Guo & Shum

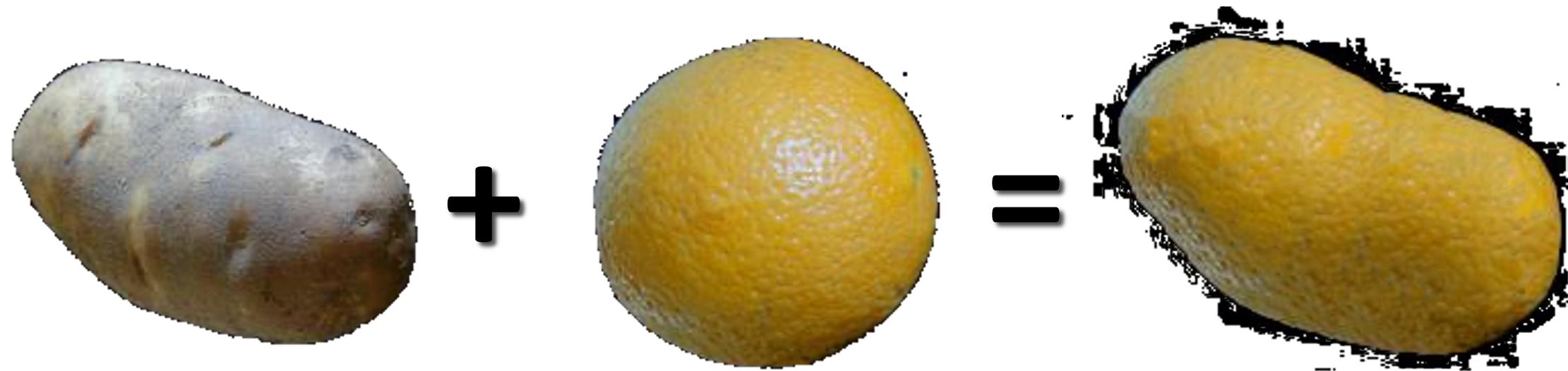
eru or a visual cortical neuron—the model describing the response of that neuron—*as a function of position*—is perhaps the most functional description of that neuron. We seek a single conceptual and mathematical framework to describe the wealth of simple-cell receptive fields neurophysiologically¹⁻³ and inferred especially if such a framework has the benefit of helping us to understand the function in a deeper way. Whereas no generic model of simple-cell receptive fields (e.g., Gaussians (DOG), difference of offset Gaussians (DOG), derivative of a Gaussian, higher derivatives of a Gaussian, etc.)—can be expected to fully describe the wealth of simple-cell receptive fields, we nonetheless believe that such a framework can help.

Wei & Levoy

Efros and Freeman

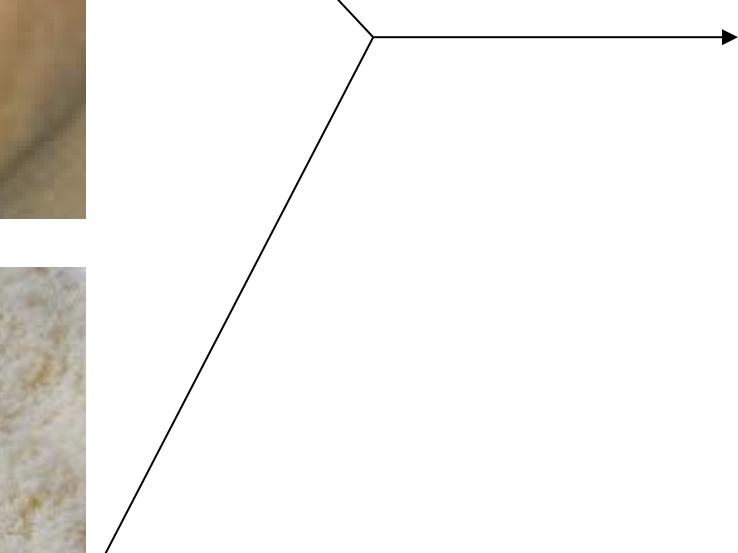
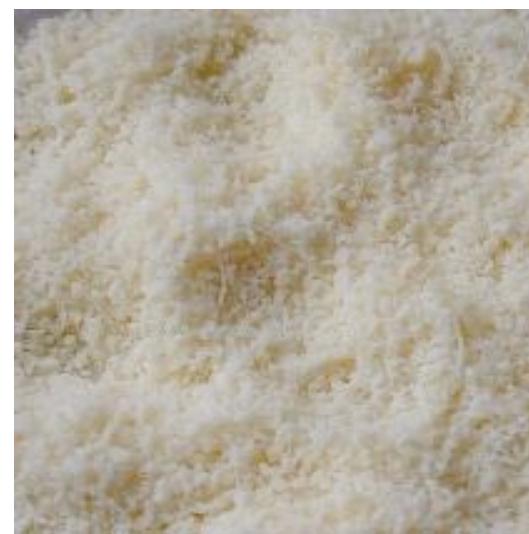
Application: Texture Transfer

- Try to explain one object with bits and pieces of another object:



Texture Transfer

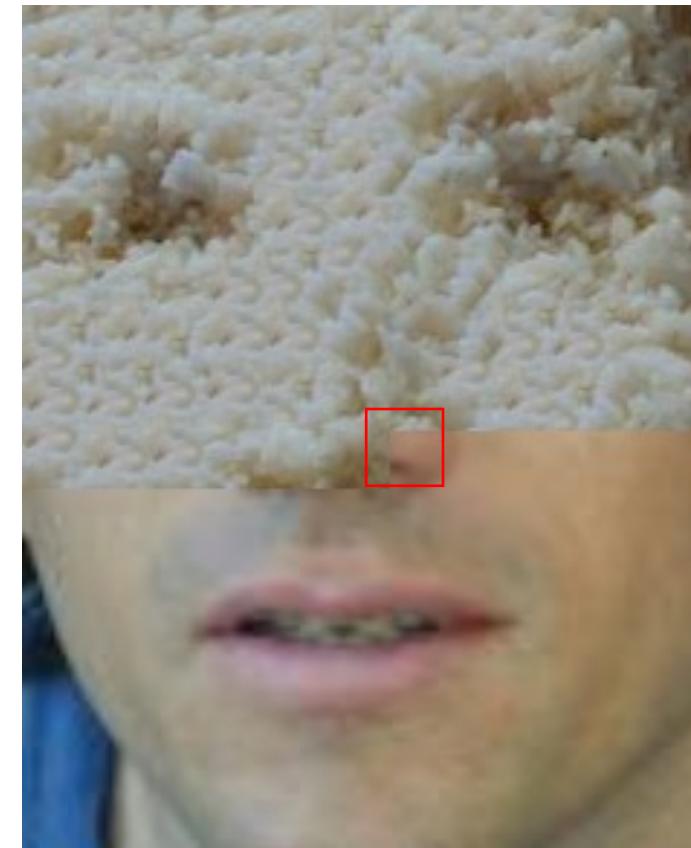
Constraint



Texture sample

Texture Transfer

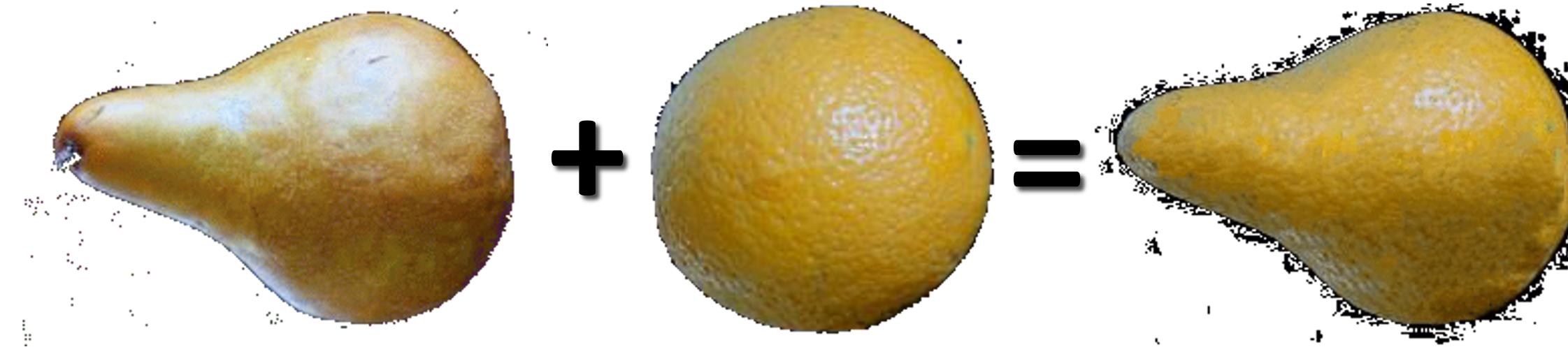
- Take the texture from one image and “paint” it onto another object



Same as texture synthesis, except an additional constraint:

1. Consistency of texture
2. Similarity to the image being “explained”

Texture Transfer



Texture Transfer

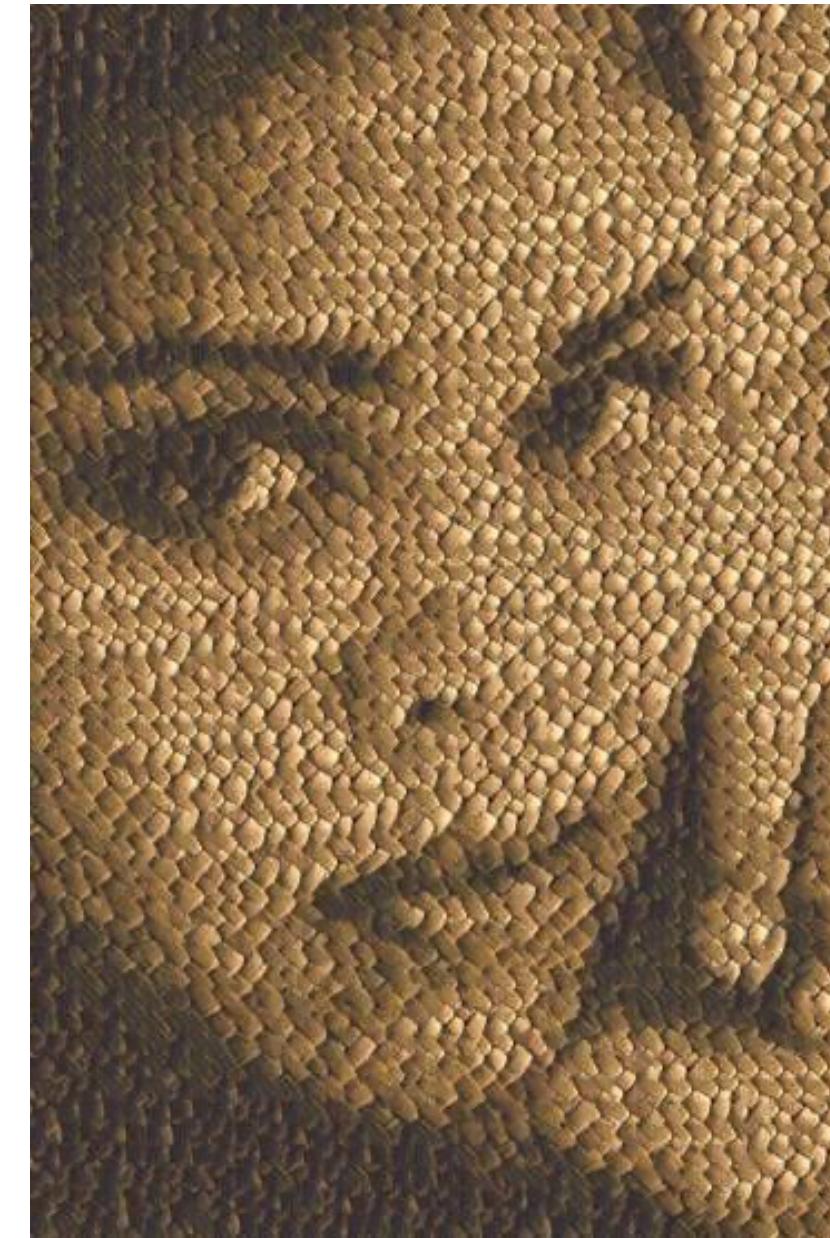
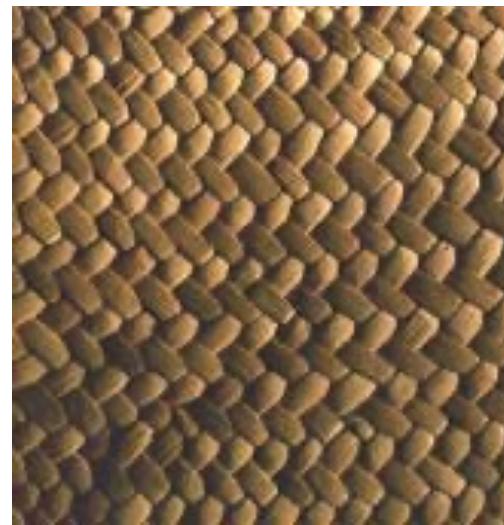


Image Analogies

Aaron Hertzmann^{1,2}

Chuck Jacobs²

Nuria Oliver²

Brian Curless³

David Salesin^{2,3}

¹**New York University**

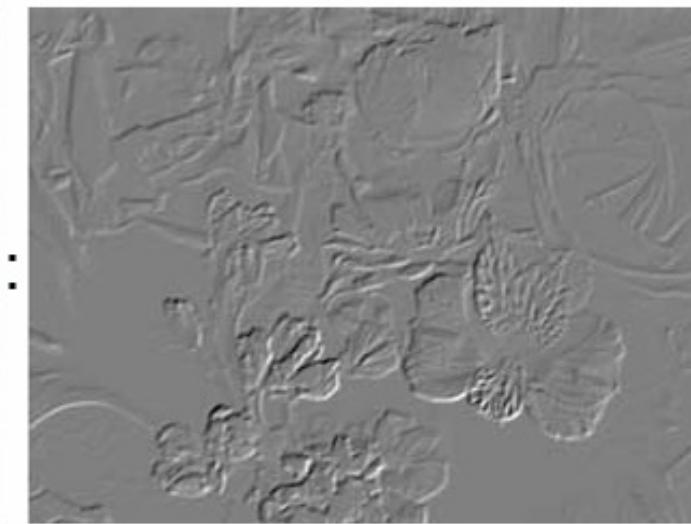
²**Microsoft Research**

³**University of Washington**

Edge Filter



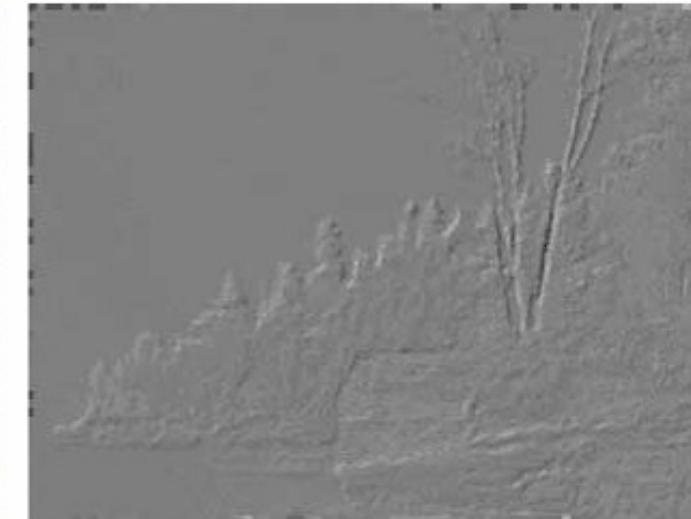
Unfiltered source (A)



Filtered source (A')



Unfiltered target (B)



Filtered target (B')

Artistic Filters



A



A'



B



B'

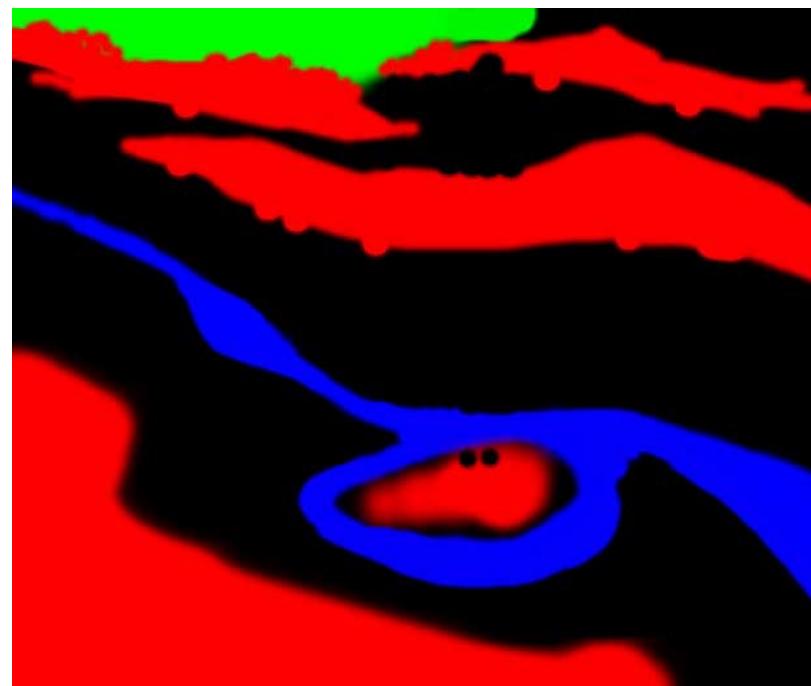
Texture-by-numbers



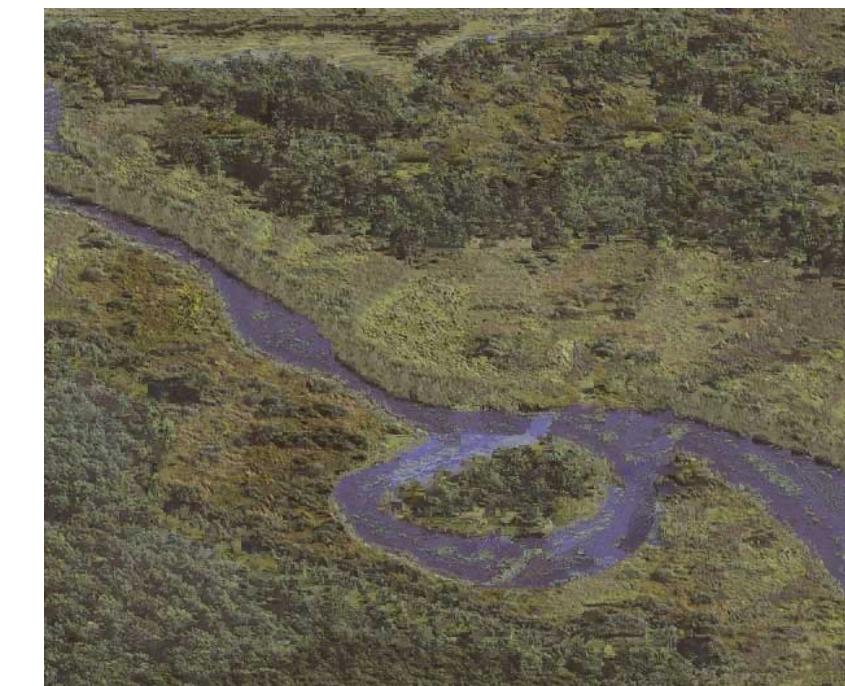
A



A'

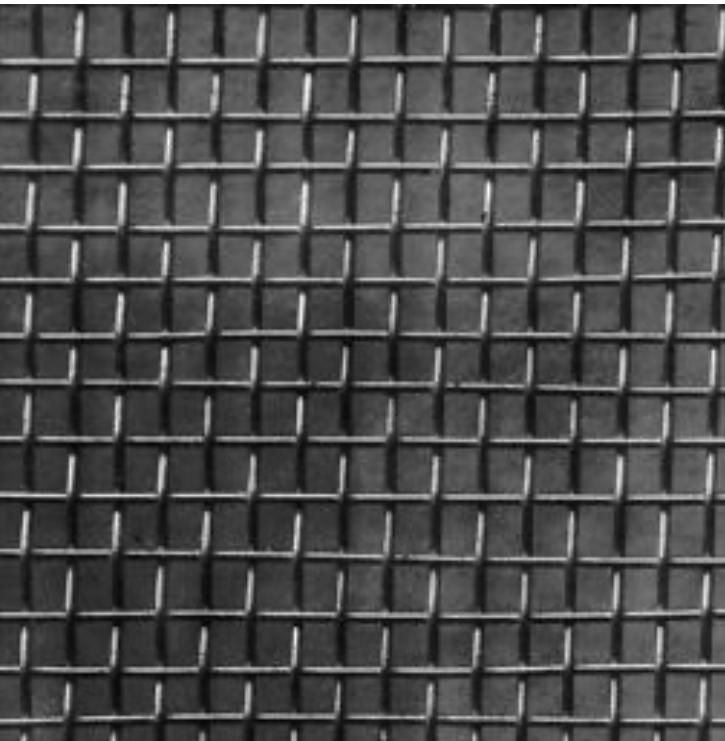


B

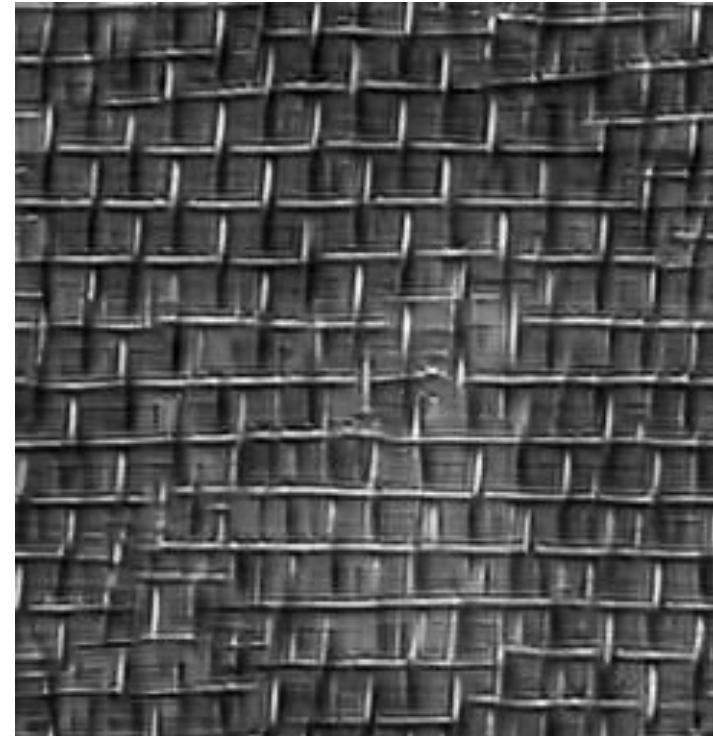


B'

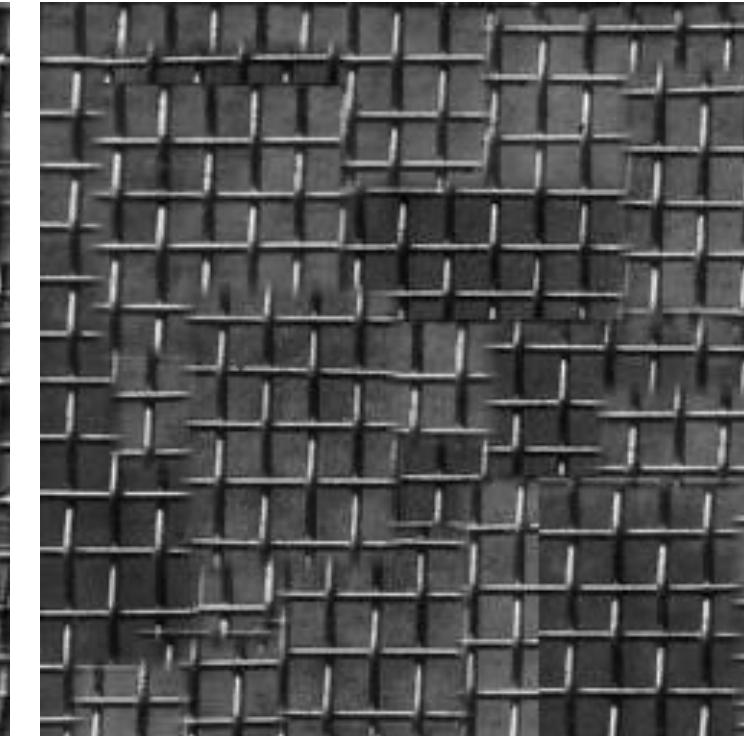
Parametric Texture Synthesis



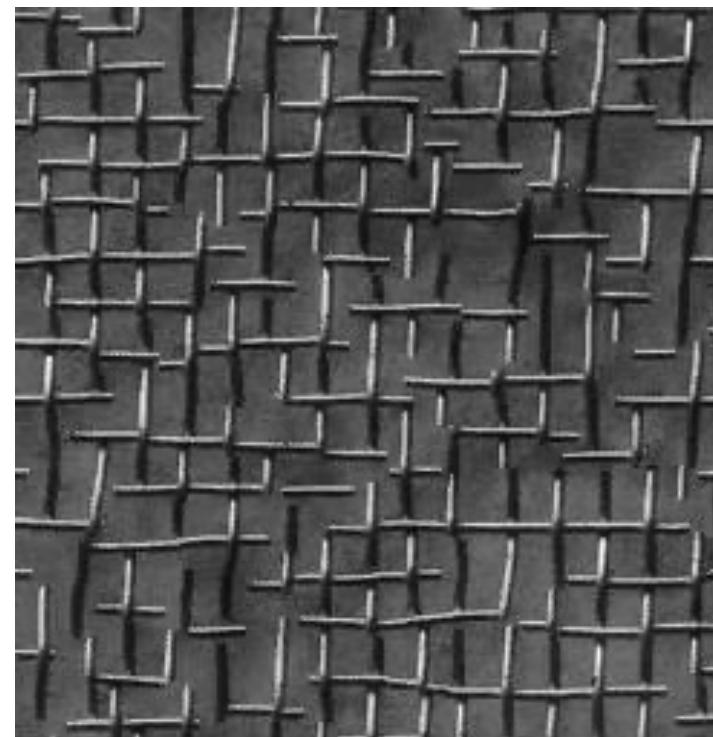
input image



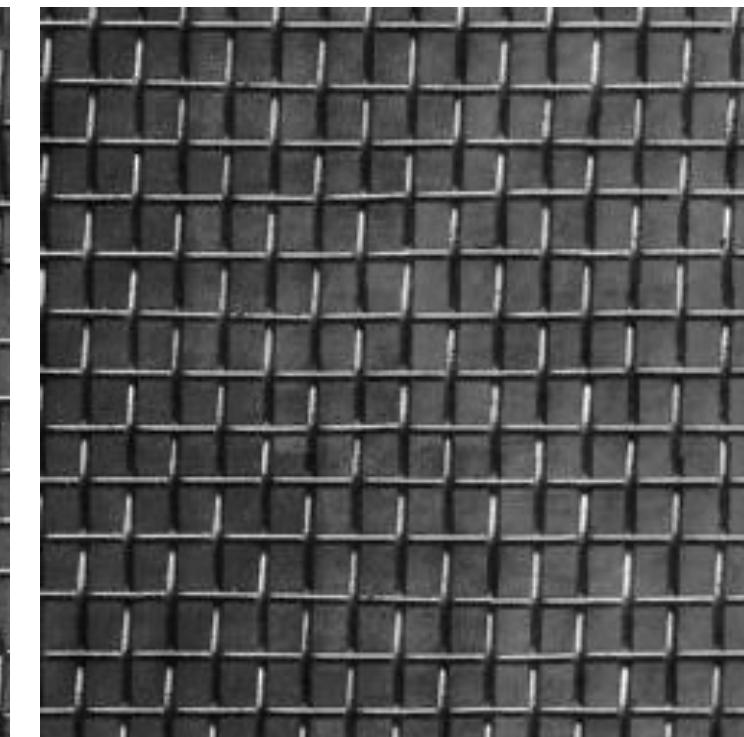
Portilla & Simoncelli



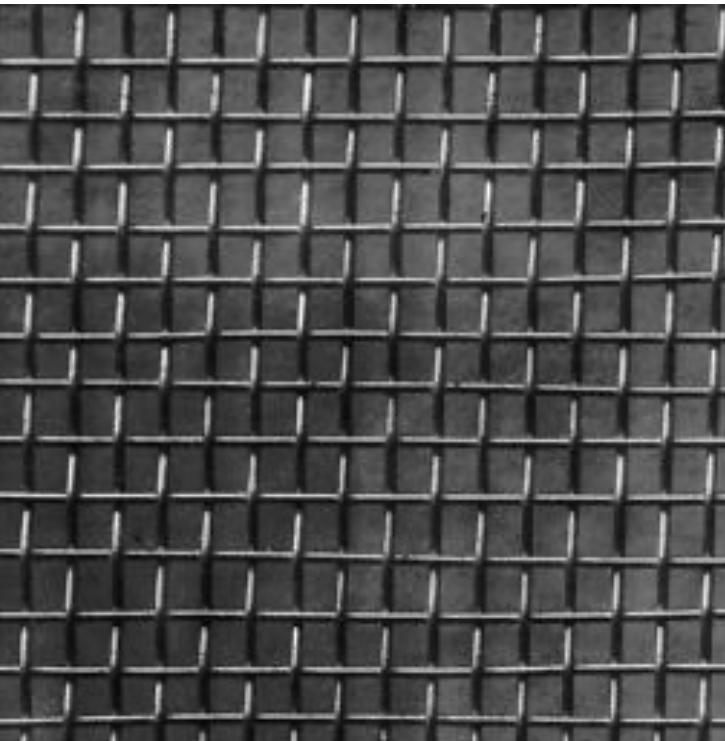
Xu, Guo & Shum



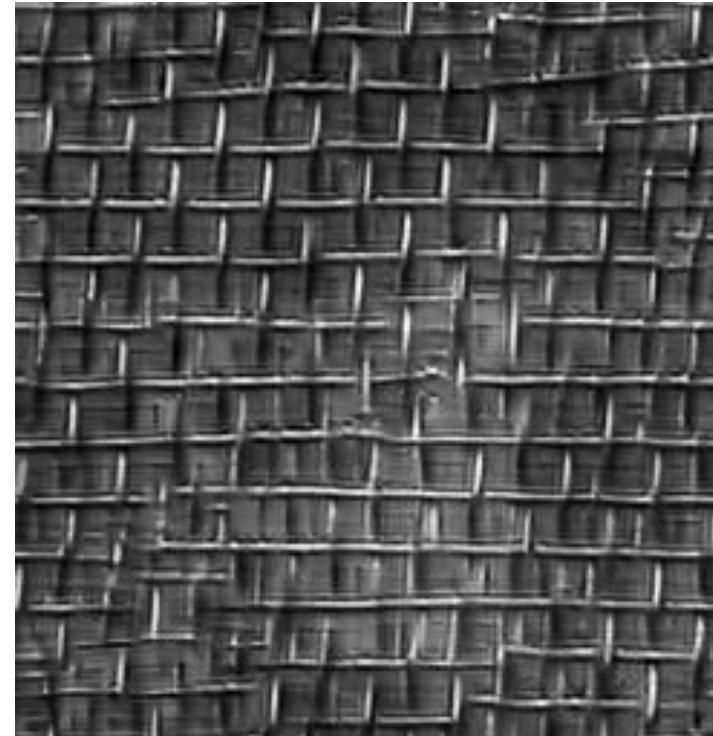
Wei & Levoy



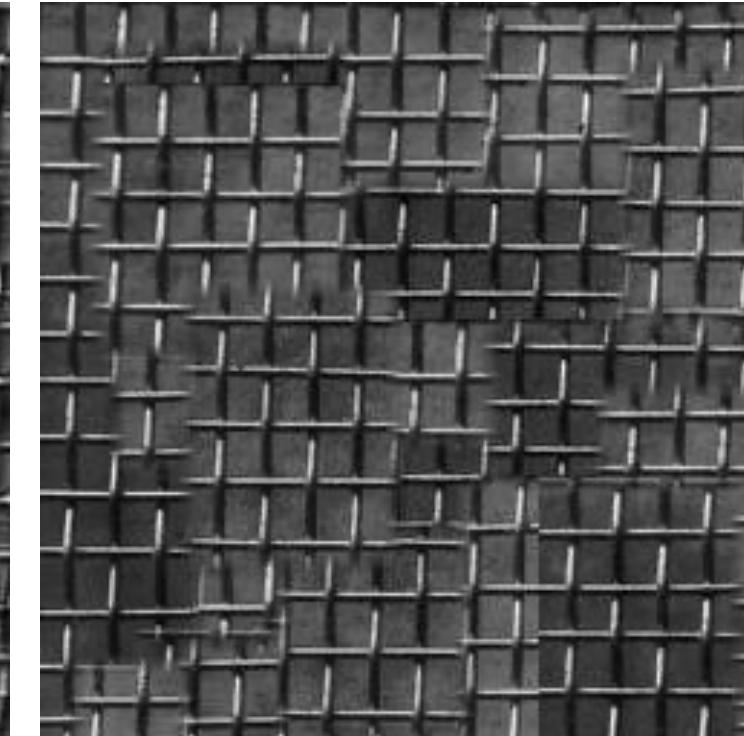
Efros and Freeman



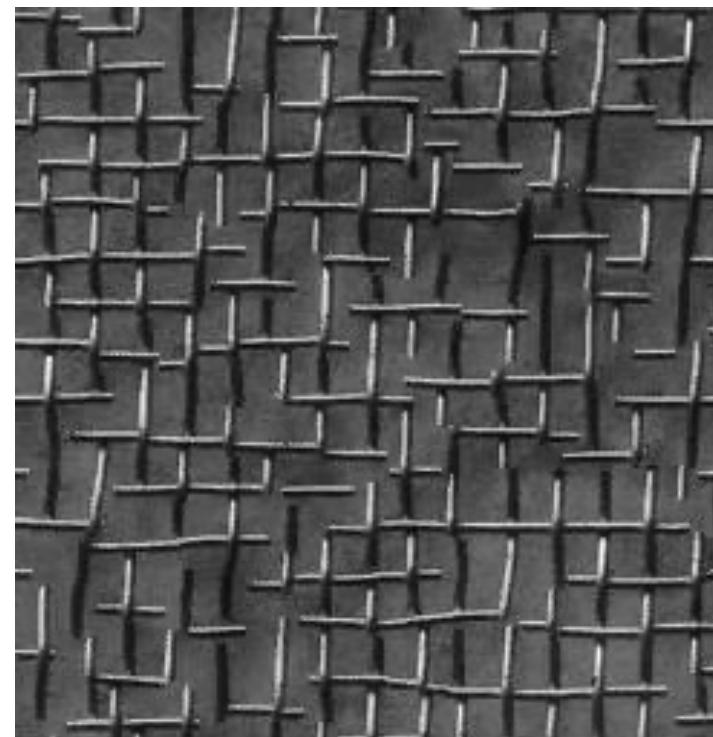
input image



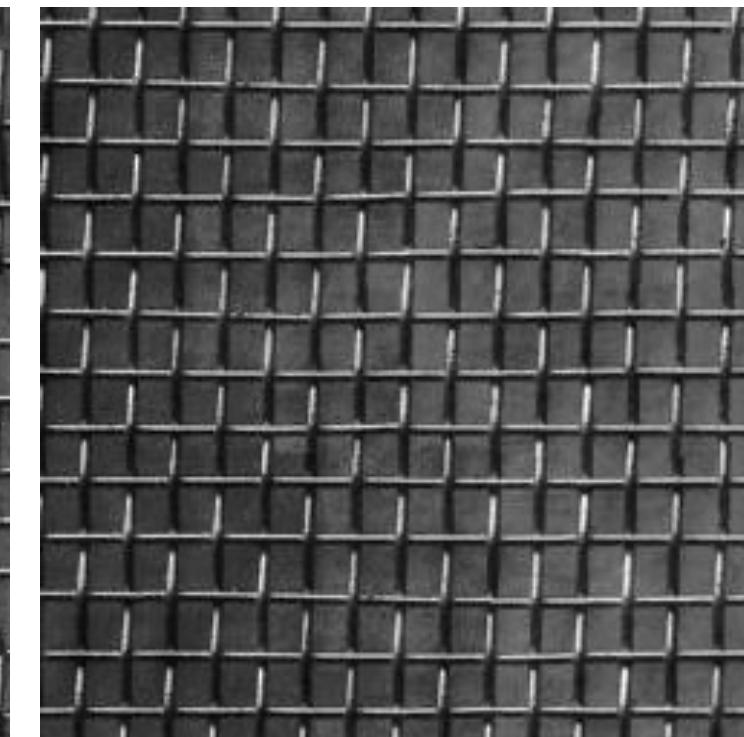
Portilla & Simoncelli



Xu, Guo & Shum

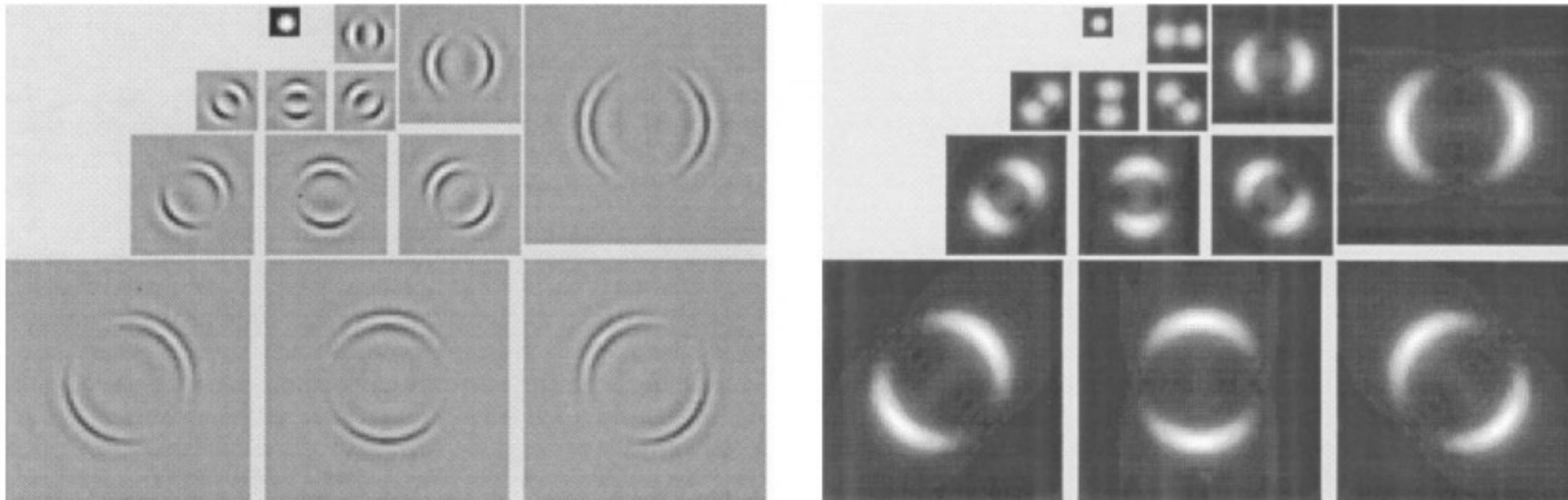


Wei & Levoy



Efros and Freeman

Parametric Texture Synthesis



Histogram and cross-channel correlation using wavelet basis

Statistics $\longrightarrow \mathcal{E}(\phi_j(y)) \approx \mathcal{E}(\phi_j(\hat{y}))$

Wavelet features

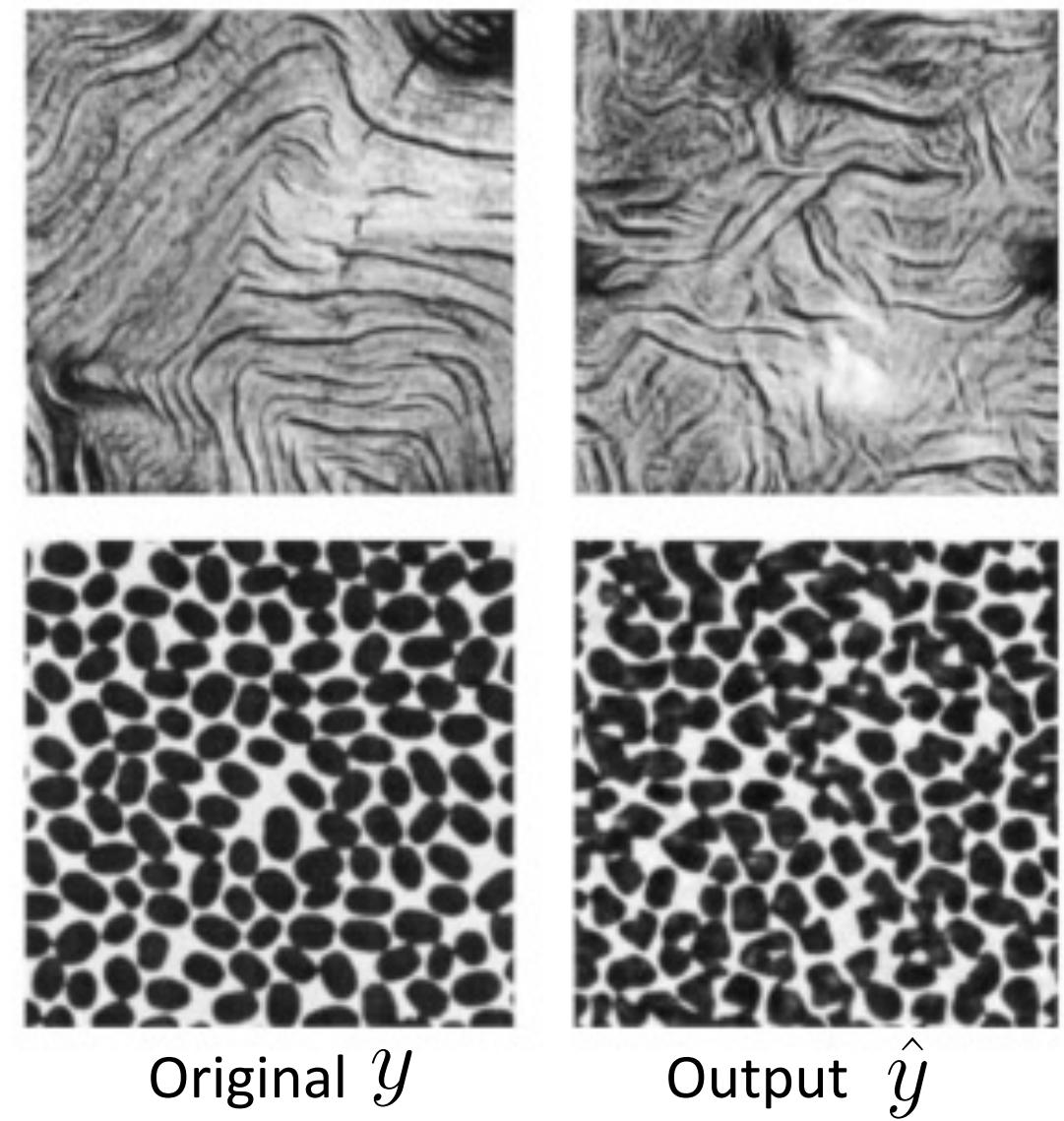
A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients
Portilla and Simoncelli, IJCV 1999

Parametric Texture Synthesis

Objective function

Given input texture y , feature descriptor ϕ ,
and statistics summary function \mathcal{E}
We aim to optimize the output image \hat{y}

$$\hat{y}^* = \arg \min_{\hat{y}} \|\mathcal{E}(\phi_j(\hat{y})) - \mathcal{E}(\phi_j(y))\|$$



Deep Learning Version

Gram matrix:

- Cross Correlation of CNN features
- Invariant to the feature locations

$$V = [v_1, v_2, \dots, v_n]$$

$$G_{ij} = \langle v_i, v_j \rangle \quad G = V^\top V$$

$$\text{Gram}^{(j)}(x) = \frac{1}{C_j H_j W_j} \sum_{h=1}^{H_j} \sum_{w=1}^{W_j} \phi_j(x)_{h,w,c} \phi_j(x)_{h,w,c'}.$$

h, w: pixel locations index

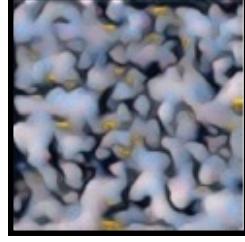
c: channel index

H, W: height and width of feature map

C: the number of total channels

Style Reconstruction (Style Loss)

$$|\text{Gram}(\hat{y}) - \text{Gram}(y)|$$


optimized output 
style image

Gram = Gram Matrix of a deep network's features (e.g., ImageNet classifier)

Style Loss

$$\arg \min_{\hat{y}} \sum_j^M \lambda_j ||\text{Gram}^{(j)}(\hat{y}) - \text{Gram}^{(j)}(y)||^2$$

weight
 \downarrow
 M
 j

(j)-th layer

Portilla & Simoncelli

original



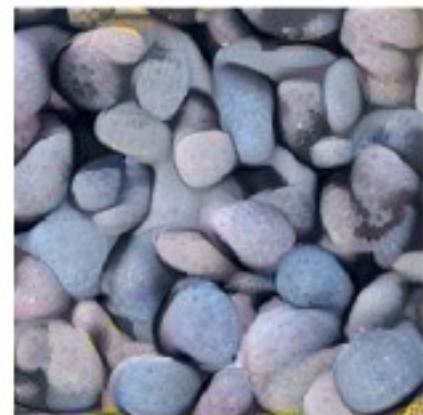
pool4



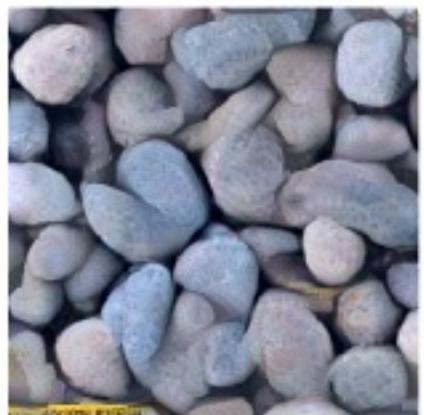
pool3



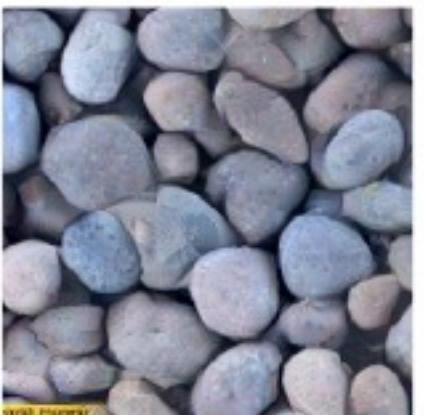
A ~1k parameters



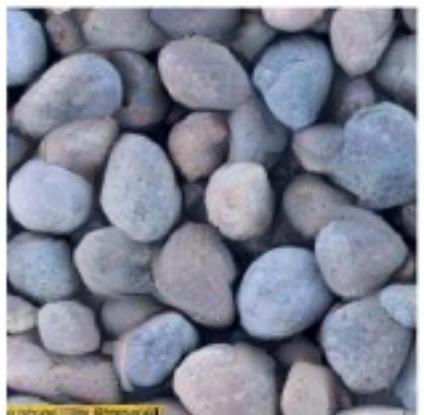
~10k parameters



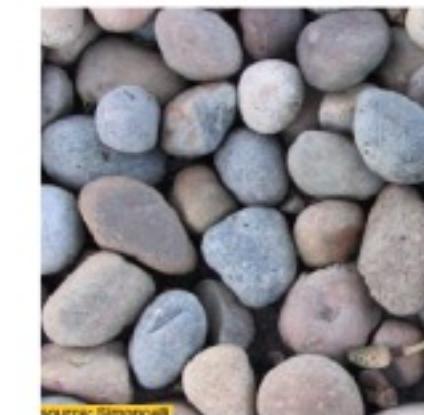
~177k parameters



~852k parameters

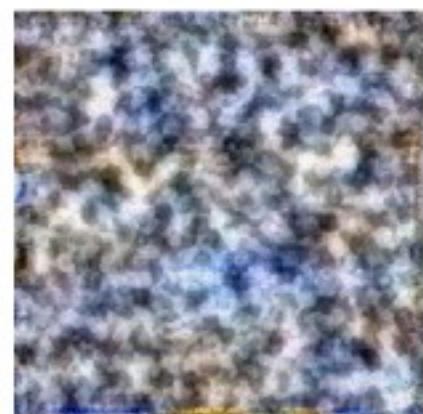


original

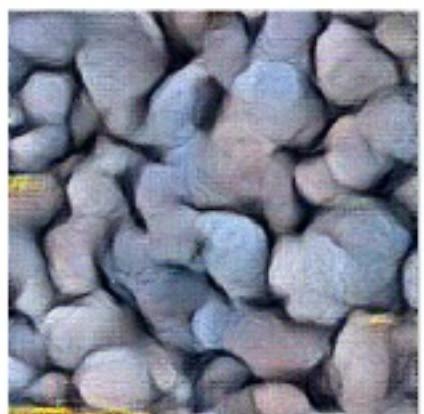


Number of parameters

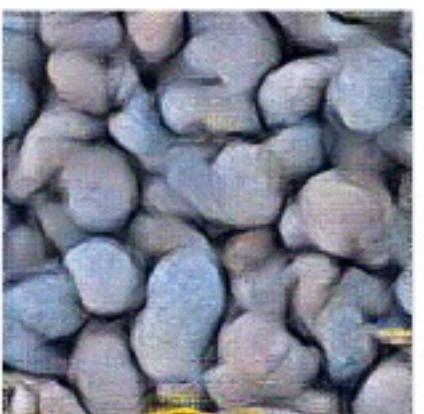
B conv1



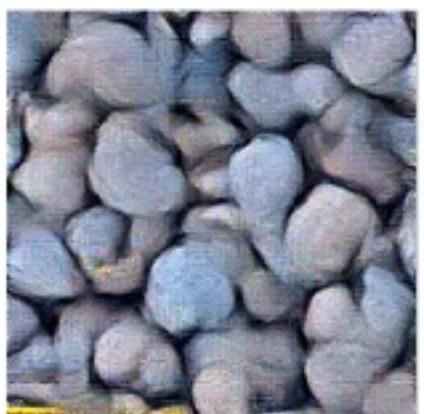
conv2



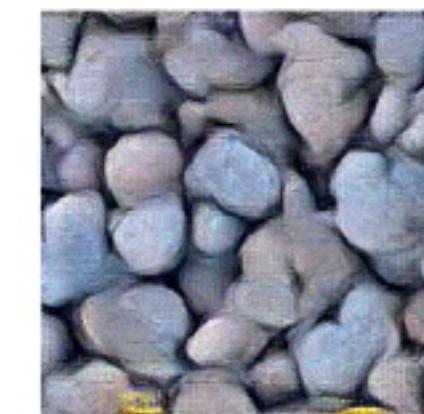
conv3



conv4



conv5

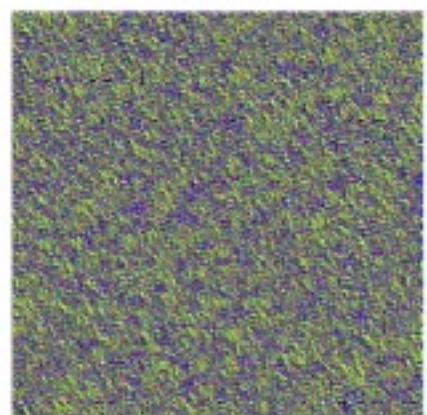


Different layers

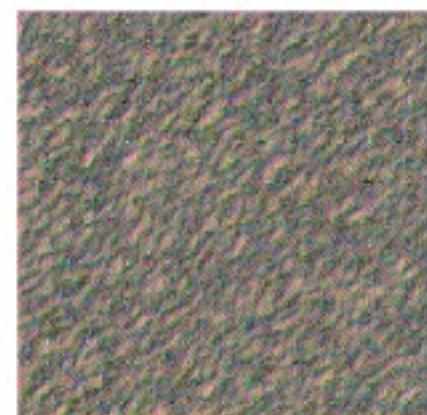
C conv1_1



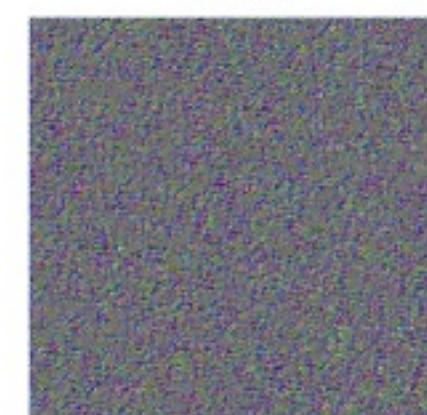
pool1



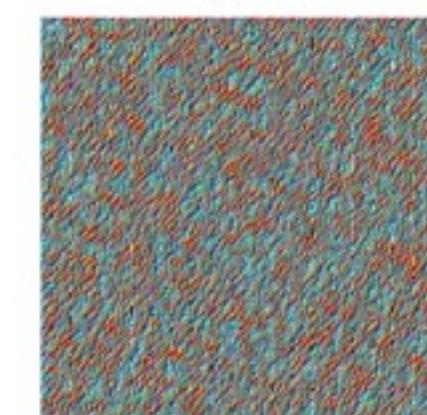
pool2



pool3



pool4



The same network architecture with random weights

Neural Style Transfer



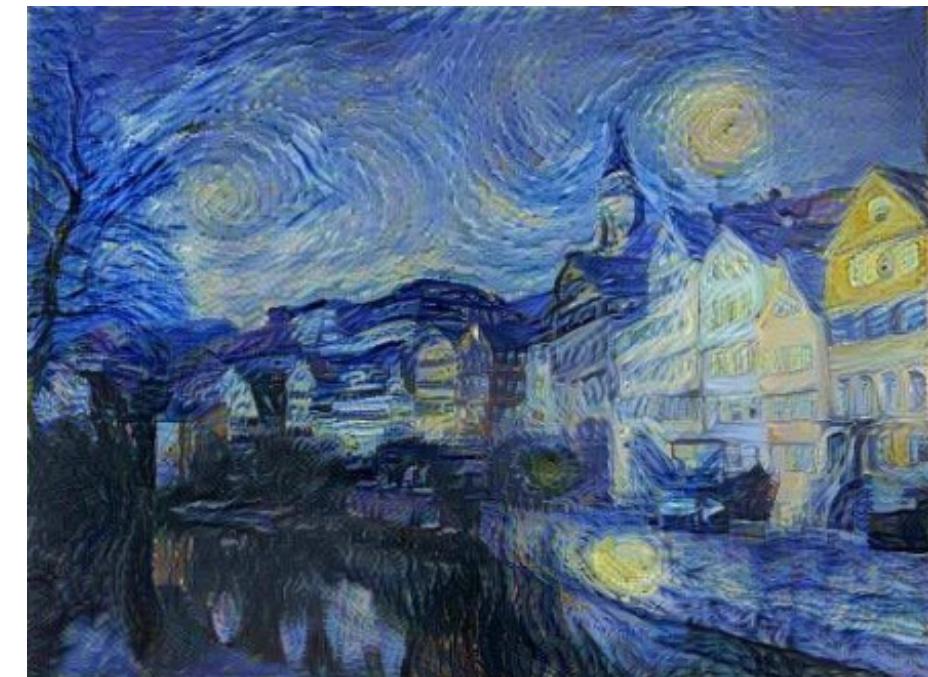
content image

+



style image

=



output result

Content Reconstruction (Perceptual Loss)

$$|\mathbf{F}(\hat{y}) - \mathbf{F}(x)|$$

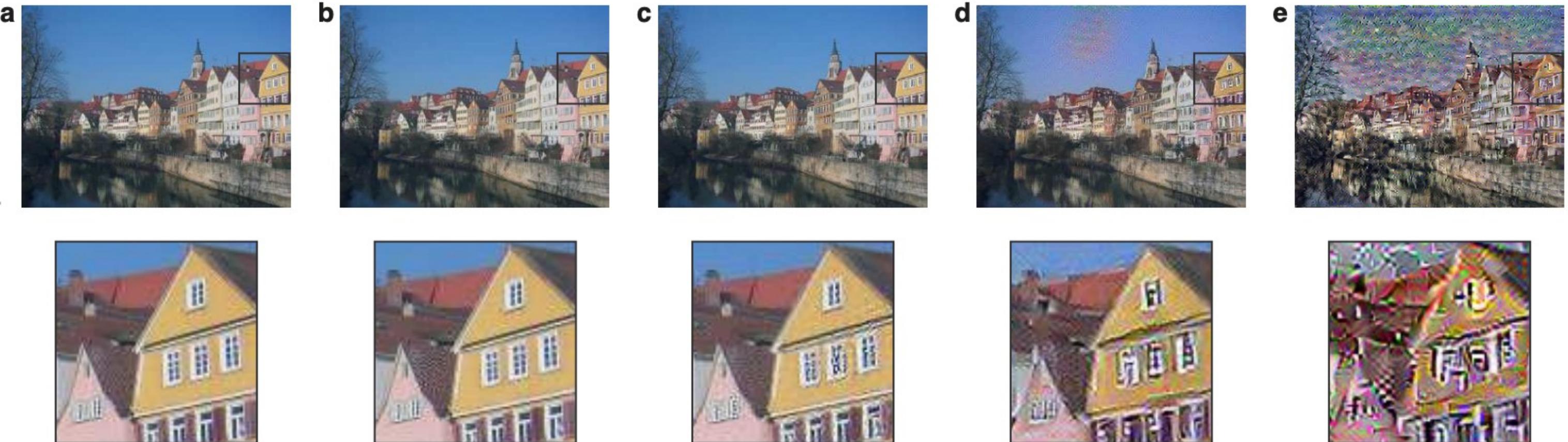
 optimized output  content image

\mathbf{F} is a deep network (e.g., ImageNet classifier)

Content Loss

$$\arg \min_{\hat{y}} \sum_i^N \lambda_i \underset{\text{weight}}{\downarrow} ||F^{(i)}(\hat{y}) - F^{(i)}(x)||_1 \underset{(i)\text{-th layer}}{\swarrow}$$

Content Reconstruction (Perceptual Loss)



Conv1_2

Conv2_2

Conv3_2

Conv4_2

Conv5_2

Neural Style Transfer

$$|\text{Gram}(\hat{y}) - \text{Gram}(y)|$$


optimized output

style image

$$+ |\mathbf{F}(\hat{y}) - \mathbf{F}(x)|$$


optimized output

content image

$$\arg \min_{\hat{y}} \mathcal{L}_{\text{style}}(\hat{y}, y) + \lambda \mathcal{L}_{\text{content}}(\hat{y}, x)$$



Different Initializations

A



B



C



Fast Neural Style Transfer

- Optimization-based method

$$\arg \min_{\hat{y}} \mathcal{L}_{\text{style}}(\hat{y}, y) + \lambda \mathcal{L}_{\text{content}}(\hat{y}, x)$$

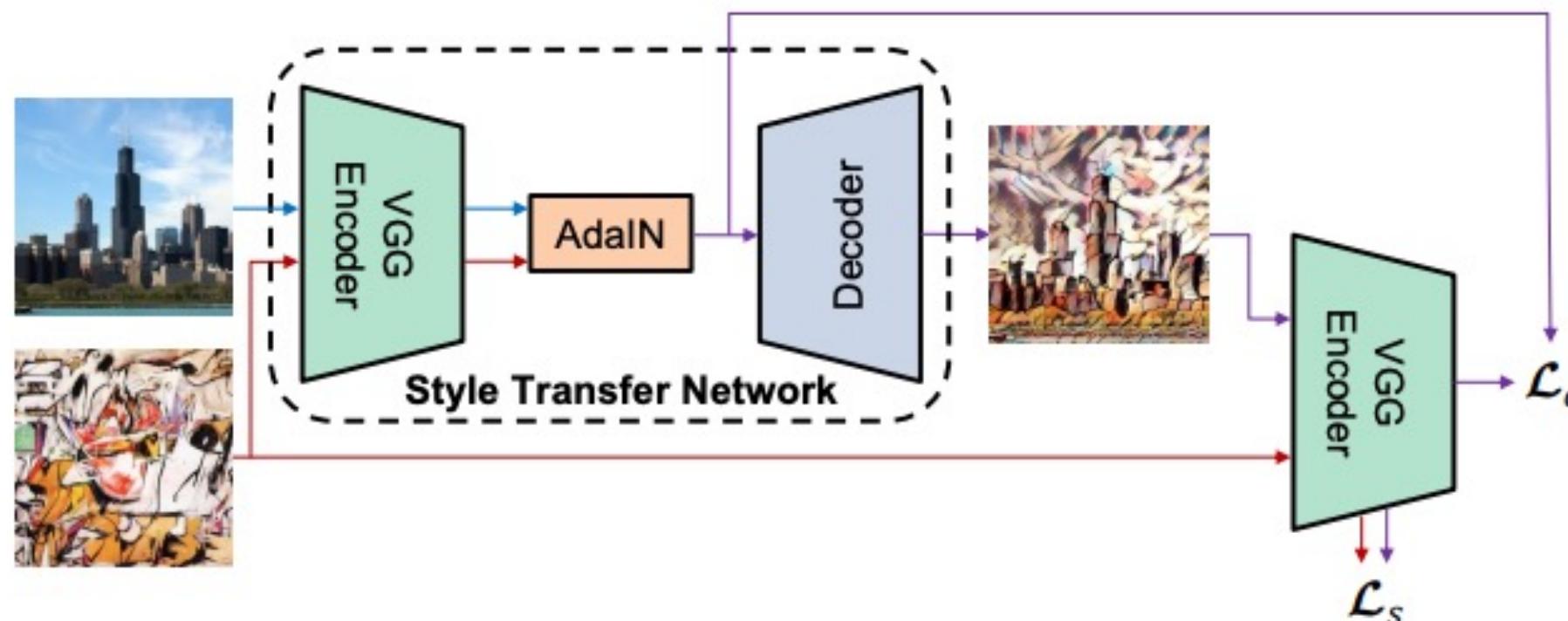
- Feedforward network

$$\arg \min_G \mathbb{E}_x \mathcal{L}_{\text{style}}(G(x), y) + \lambda \mathcal{L}_{\text{content}}(G(x), x)$$

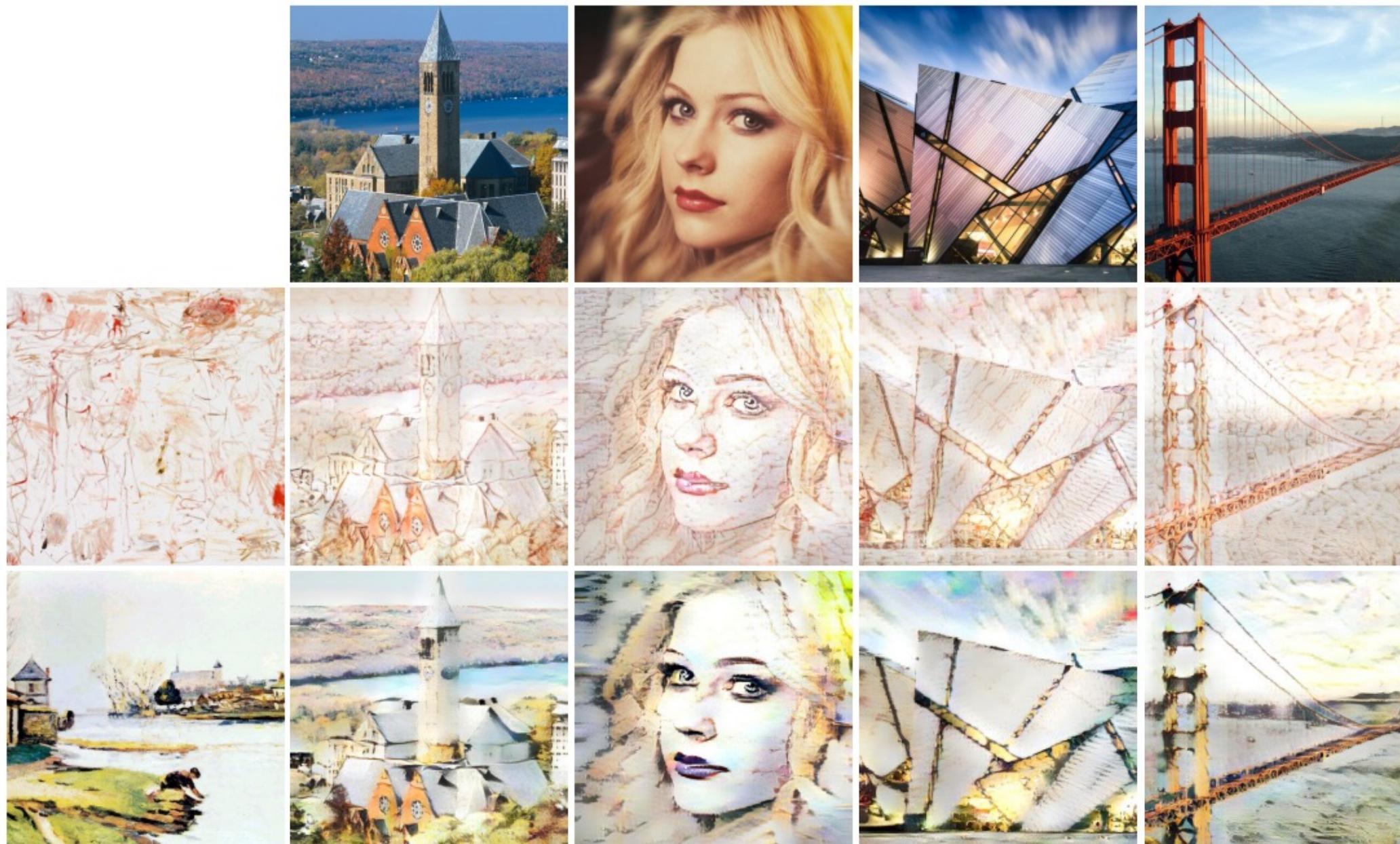
Arbitrary Style Transfer with AdaIN

- Feedforward network with any style

$$\arg \min_G \mathbb{E}_{x,y} \mathcal{L}_{\text{style}}(G(x,y), y) + \lambda \mathcal{L}_{\text{content}}(G(x,y), x)$$

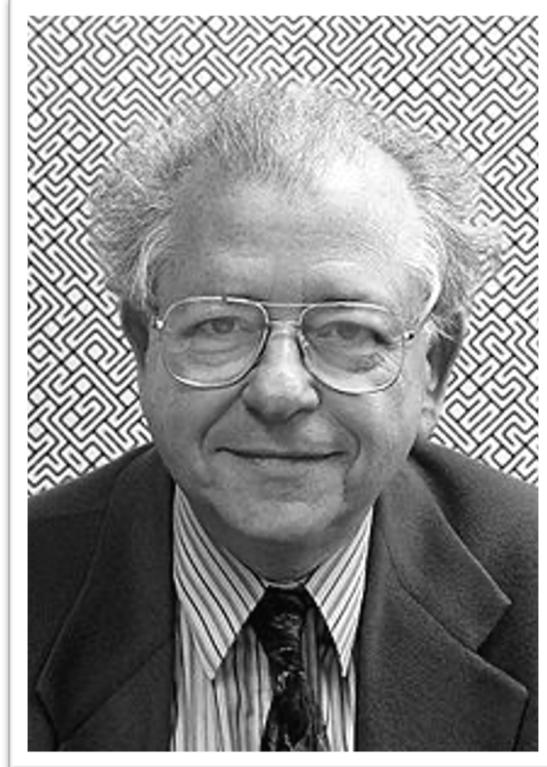


Arbitrary Style Transfer with AdaIN

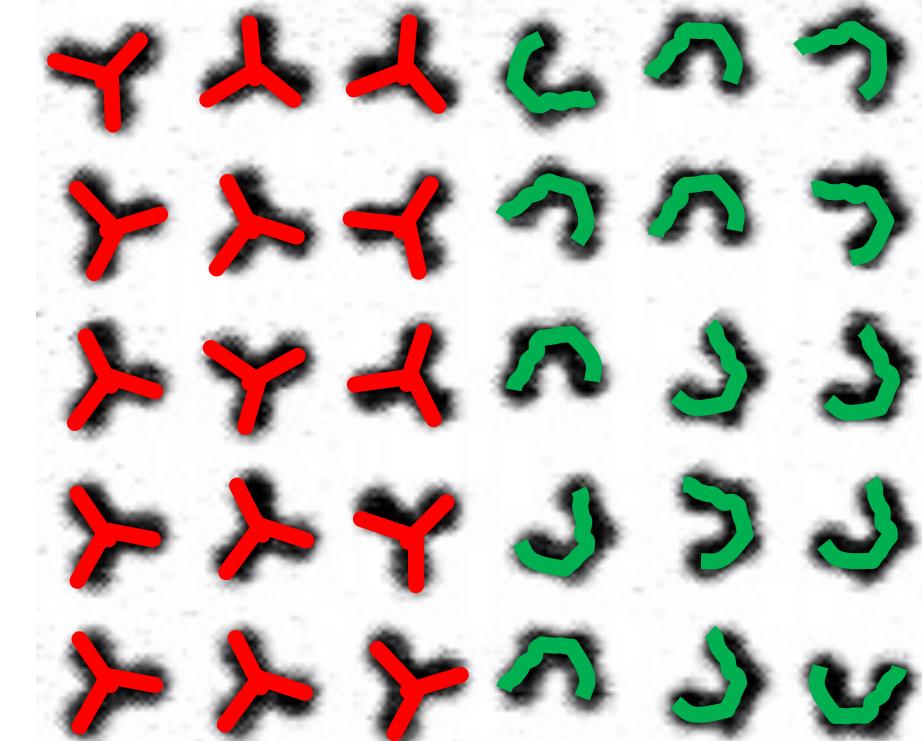


Texture Discrimination

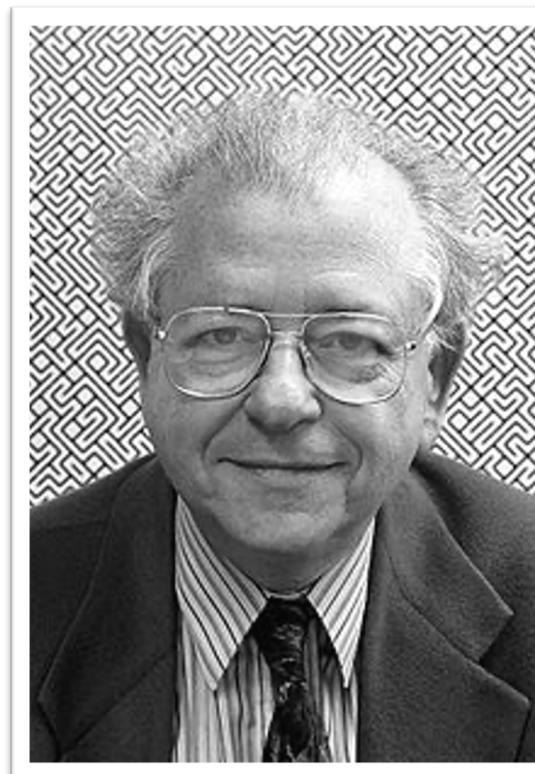
Texture Discrimination in Human Perception



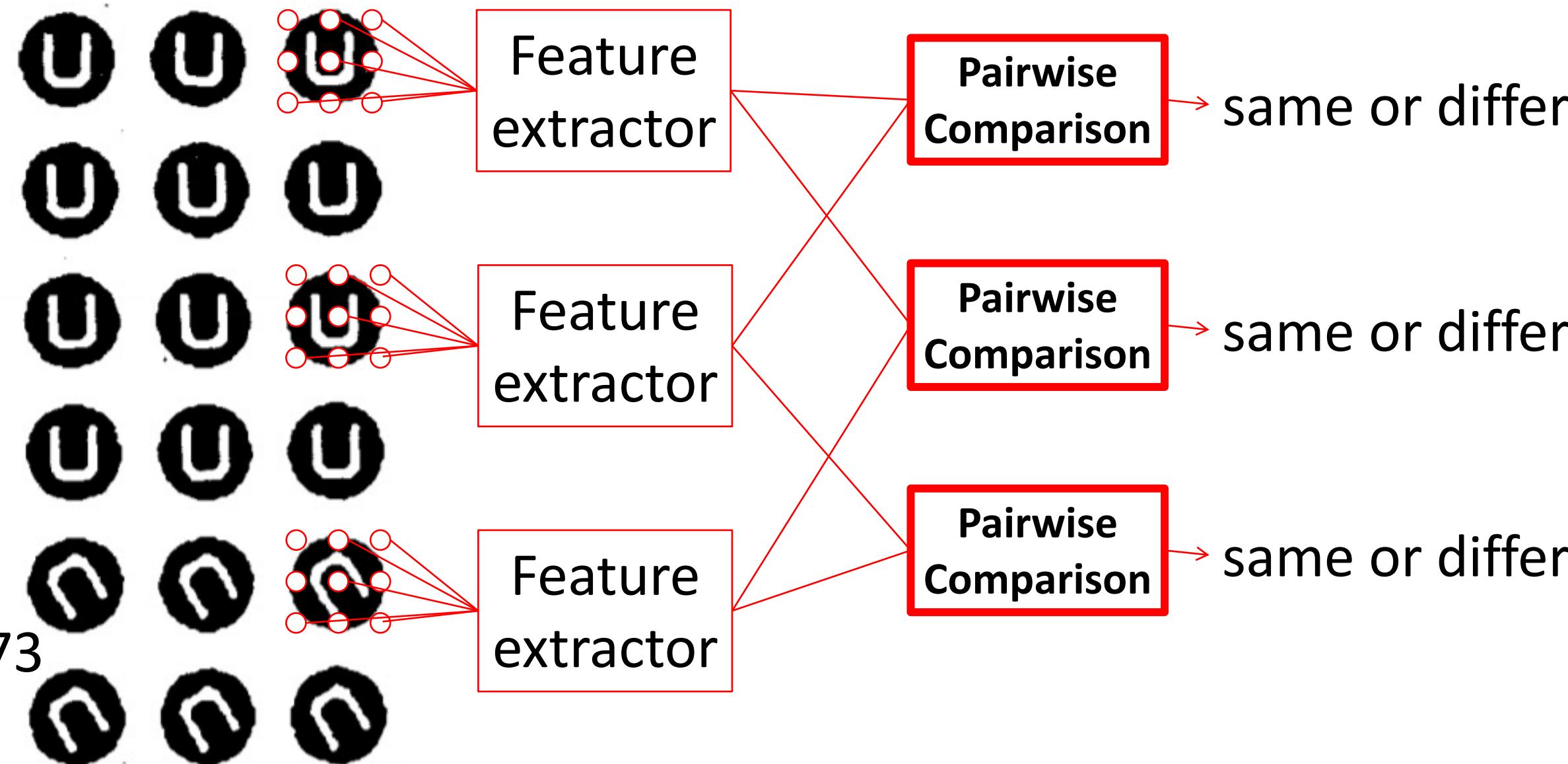
Bela Julesz, 1973



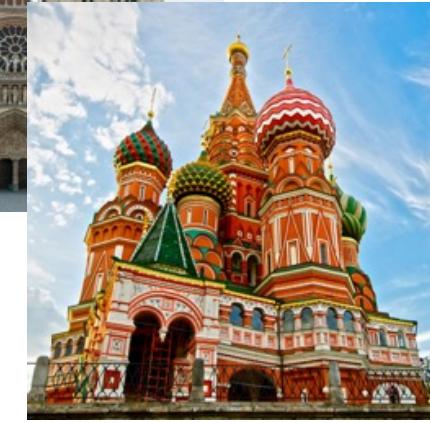
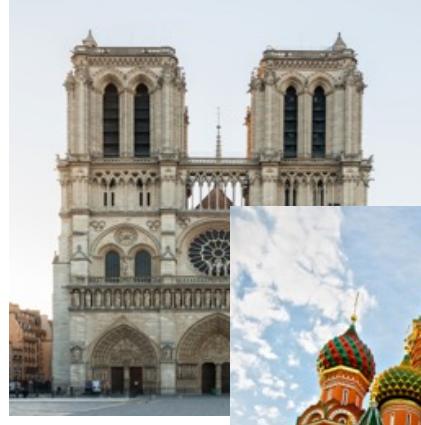
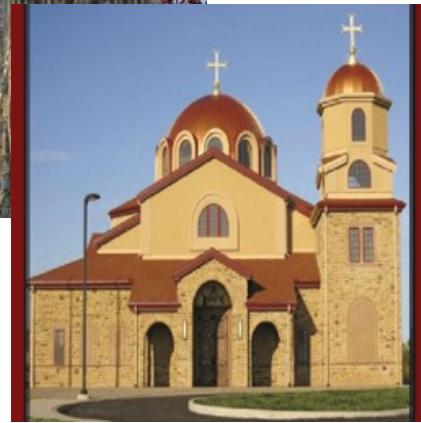
Texture Discrimination in Human Perception



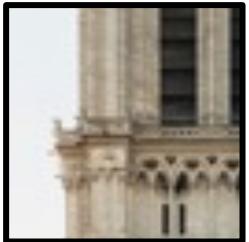
Bela Julesz, 1973



Church images



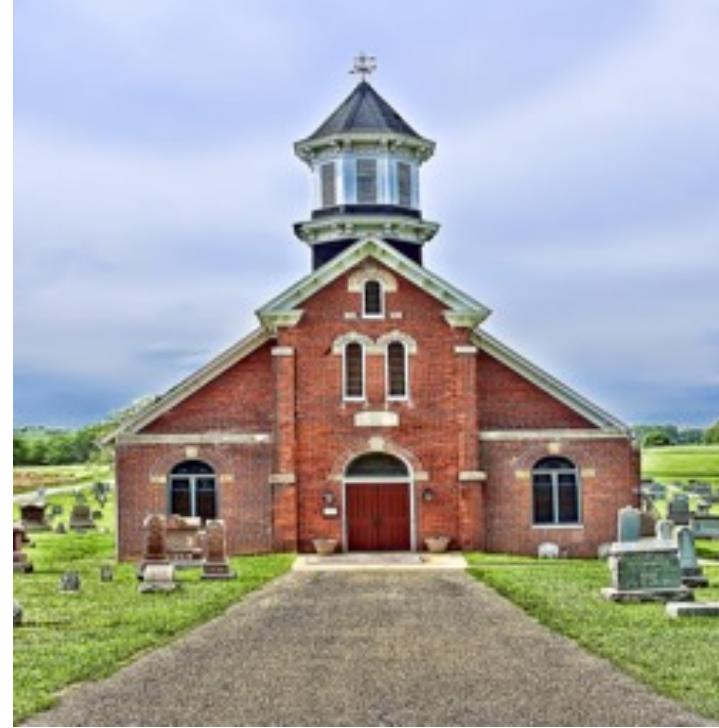
Are

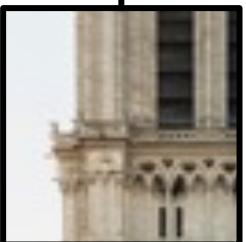


and

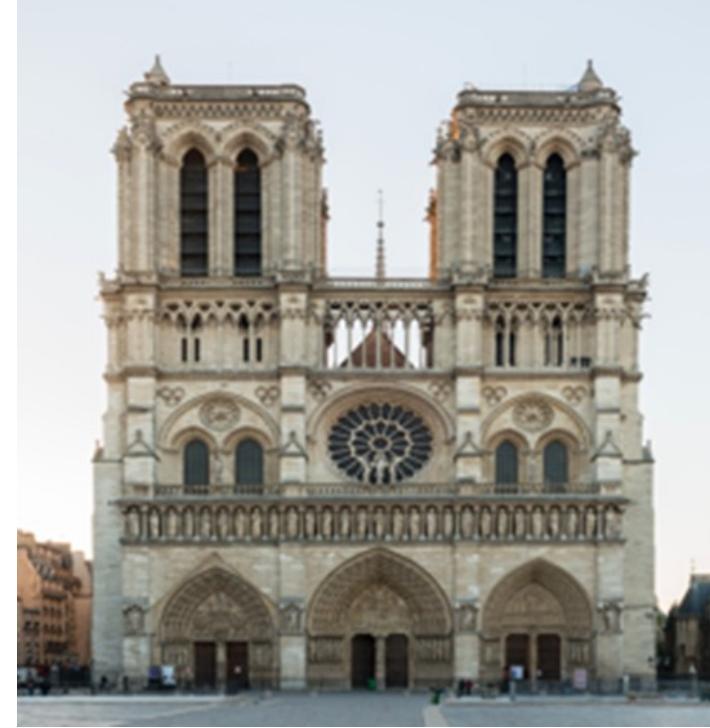


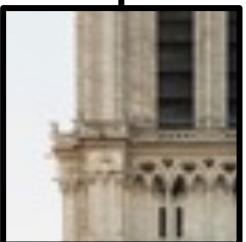
from the same image?



Are  and  from the same image?

Answer:
No



Are  and  from the same image?

Answer:
Yes

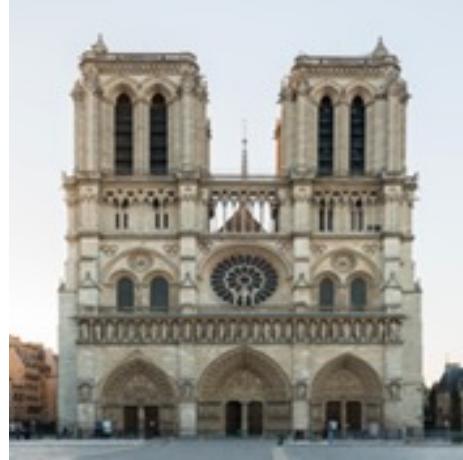


Answer:
...?

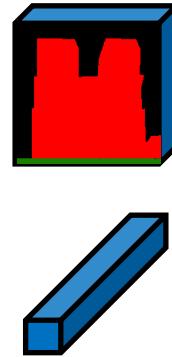
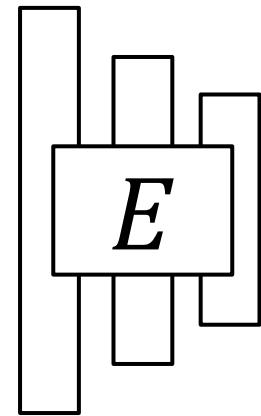
Are  and  from the same image?

Patch co-occurrence discriminator

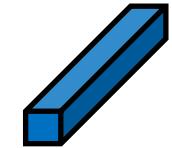
Auto-
encode



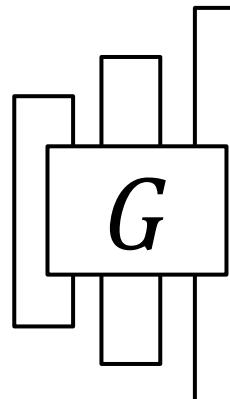
Reconstruction



structure code

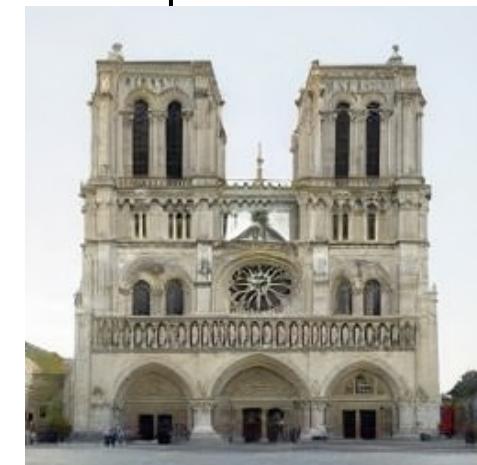
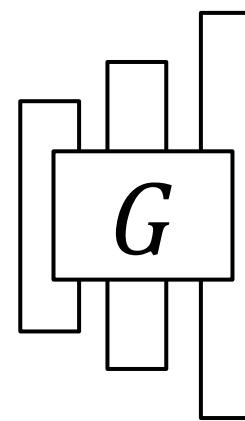
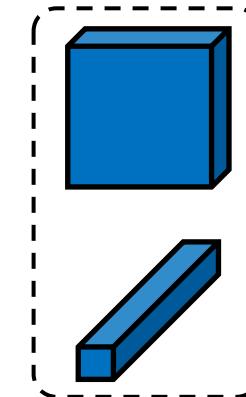
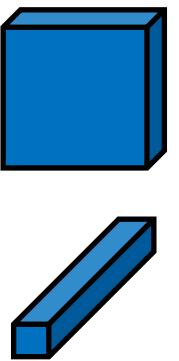
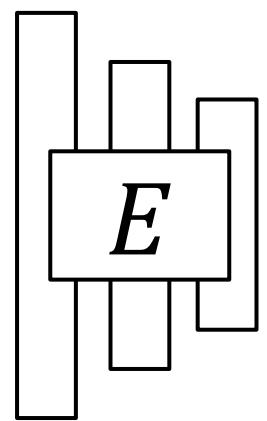
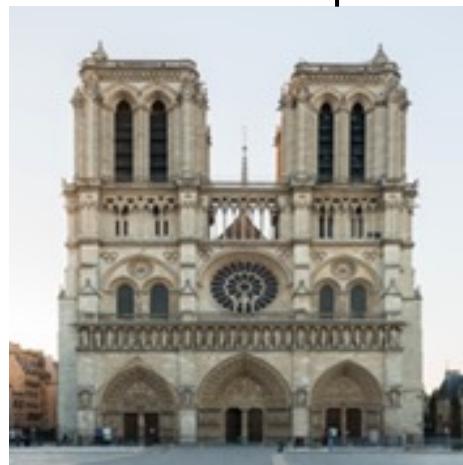


texture code

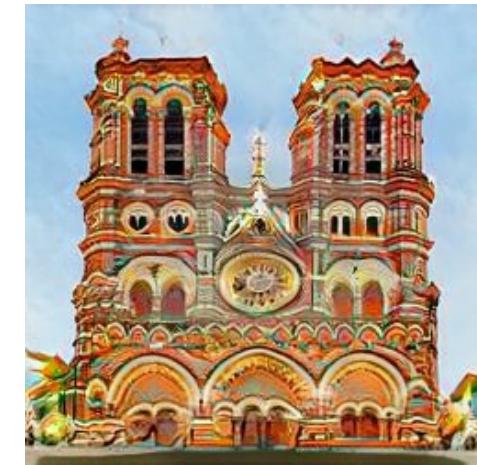
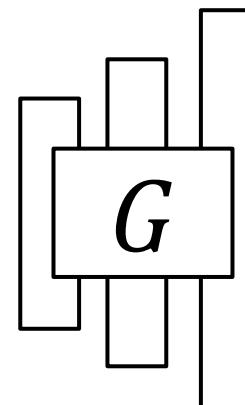
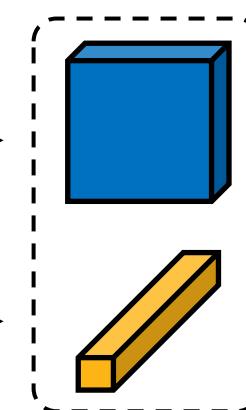
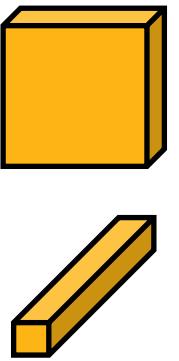
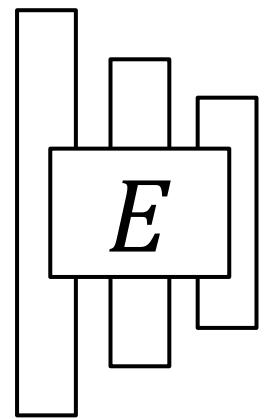
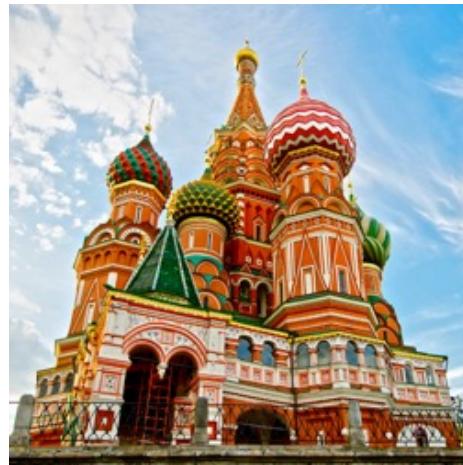


Reconstruction

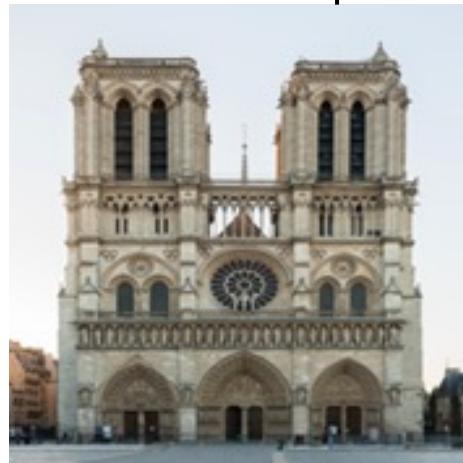
Auto-
encode



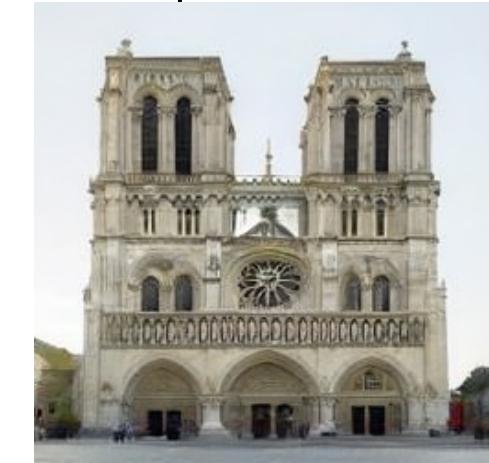
Swap



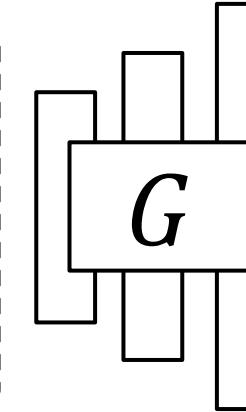
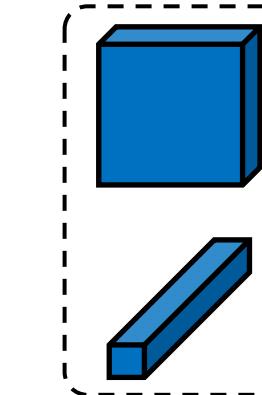
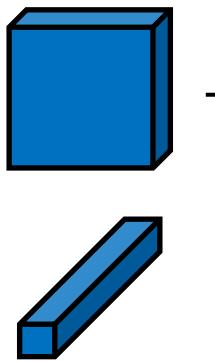
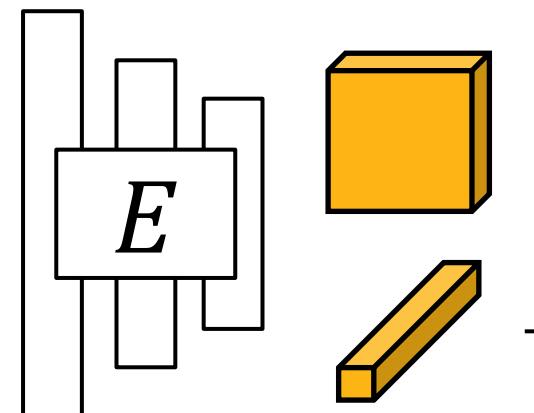
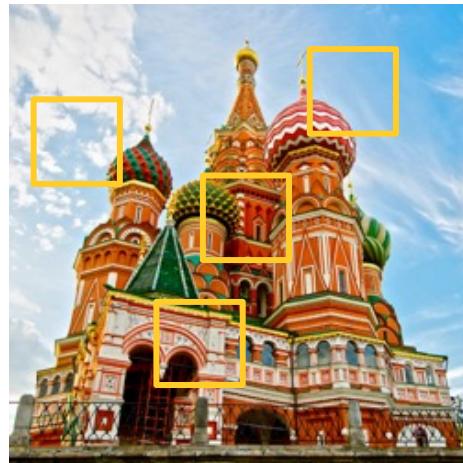
Auto-
encode



Reconstruction



Swap



texture



structure

