

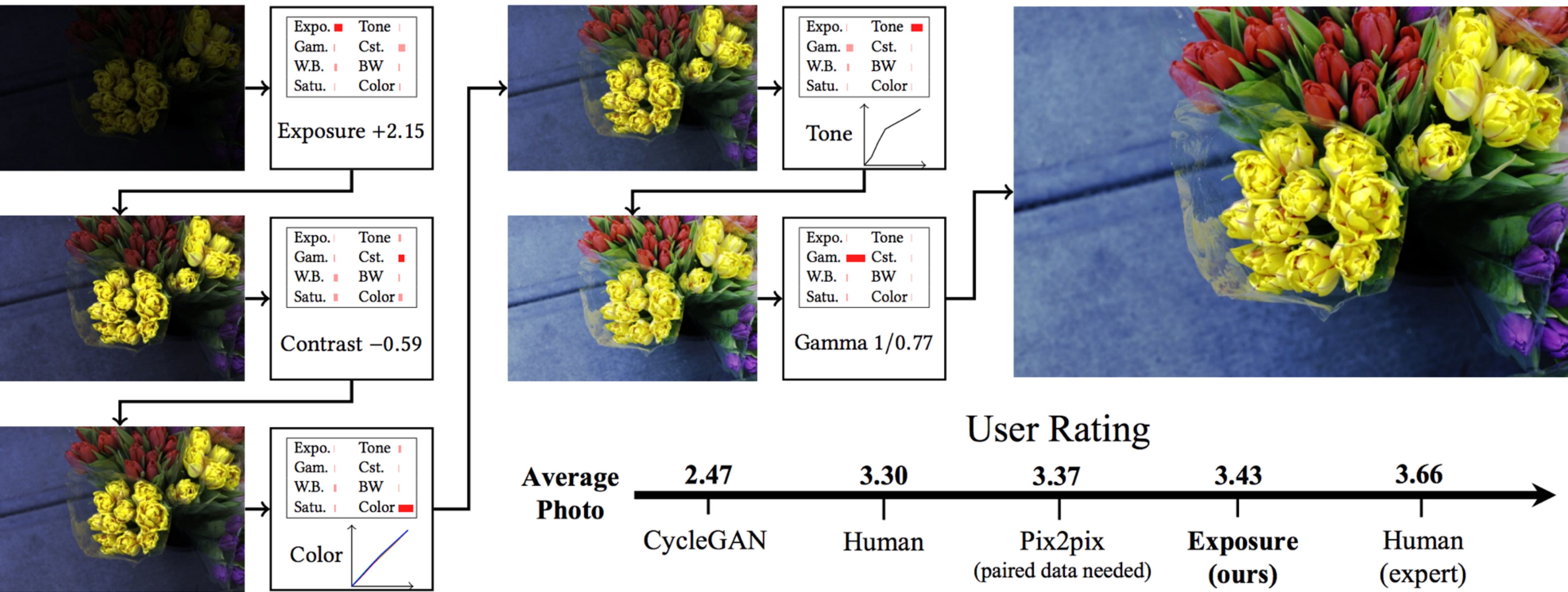
# Exposure: A White-Box Photo Post-Processing Framework

Yuanming Hu<sup>1,2</sup> Hao He<sup>1,2</sup> Chenxi Xu<sup>1,3</sup> Baoyuan Wang<sup>1</sup> Stephen Lin<sup>1</sup>

<sup>1</sup>Microsoft Research

<sup>2</sup>MIT CSAIL

<sup>3</sup>Peking University



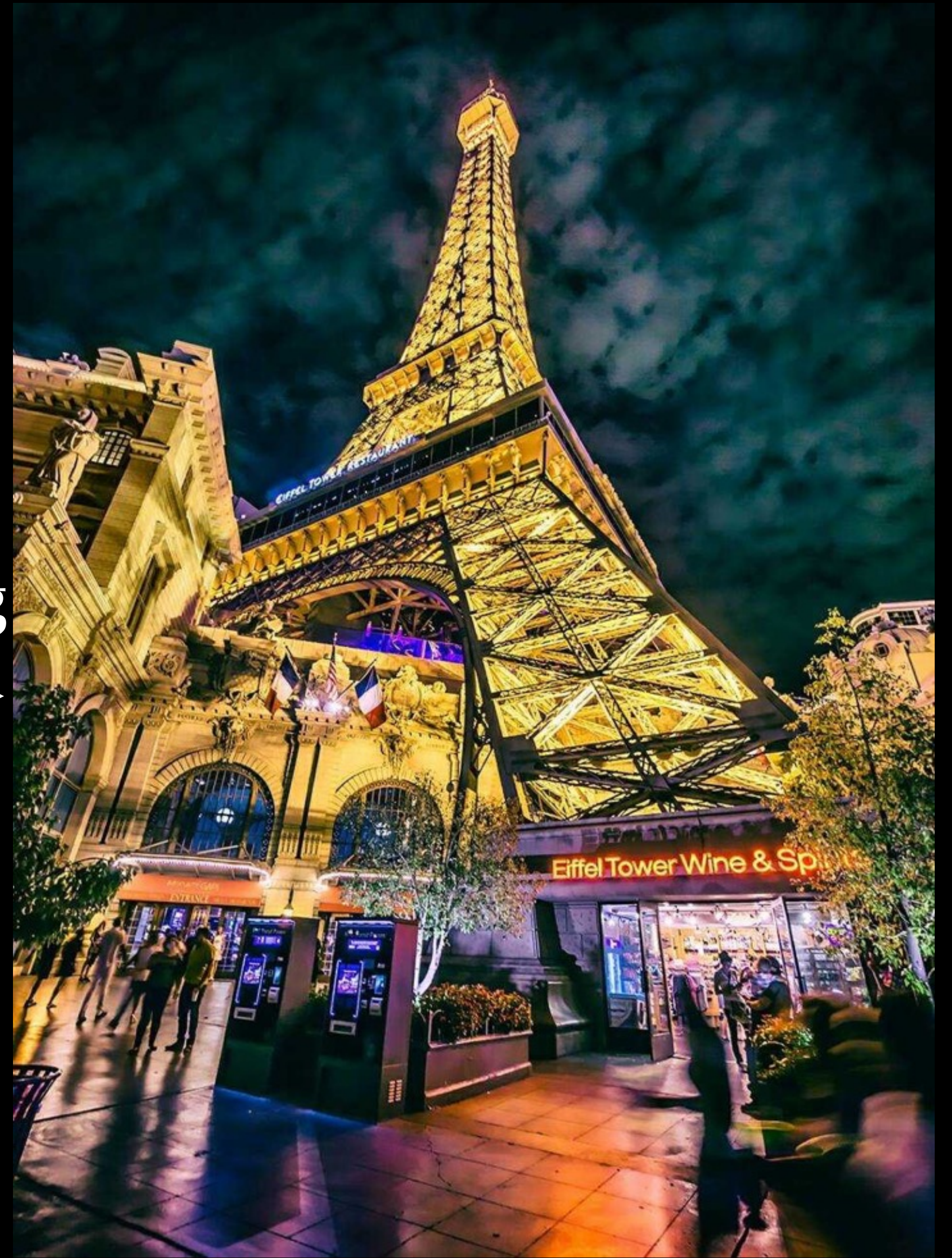




Post-processing  
a.k.a. retouching



“Magic”







Exposure + 2.40

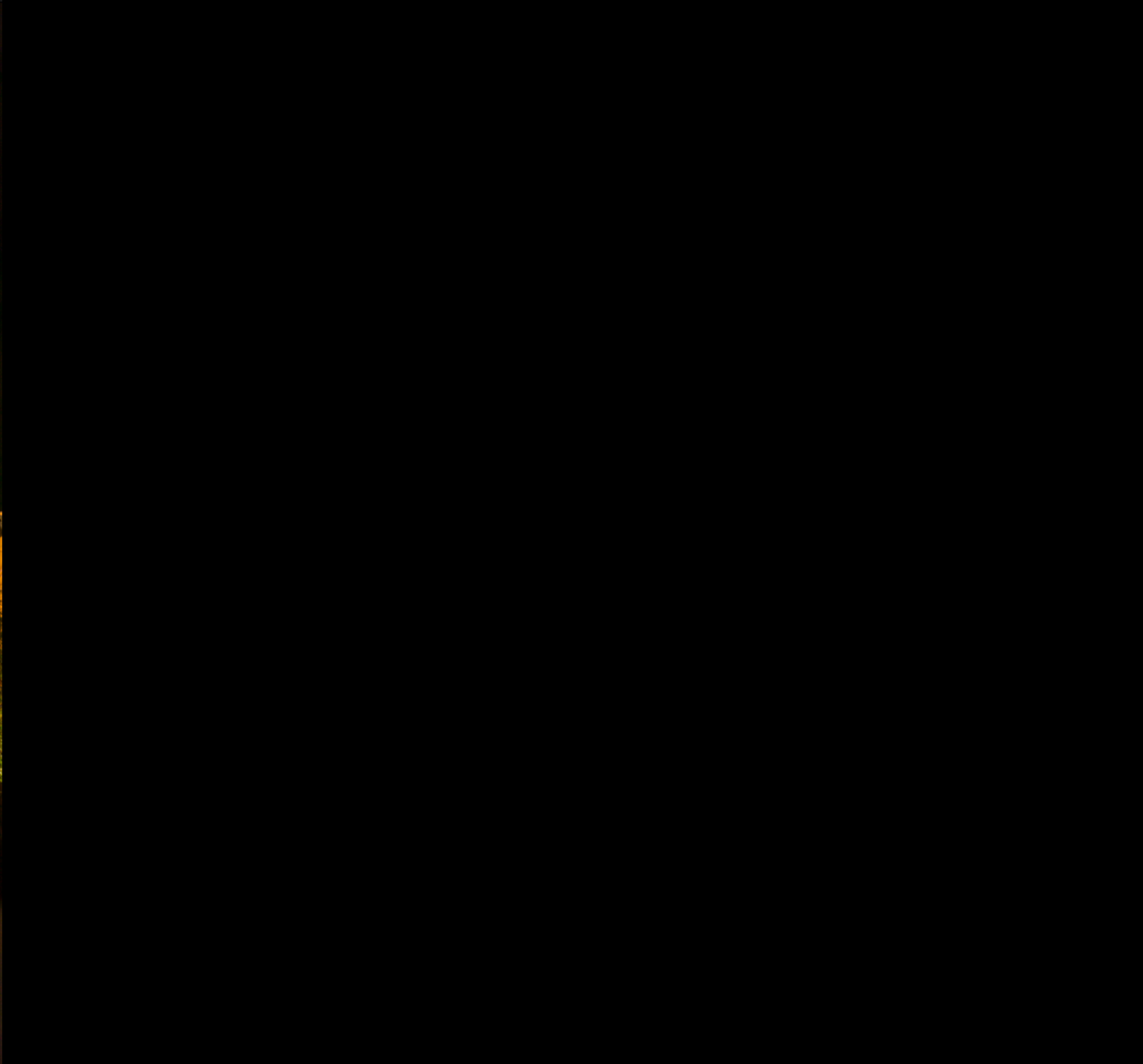






Highlight -78









White balance  
→  
Temperature 2600  
Tint +23







Clarity + 63







Vibrance +75







Shadow + 70  
←





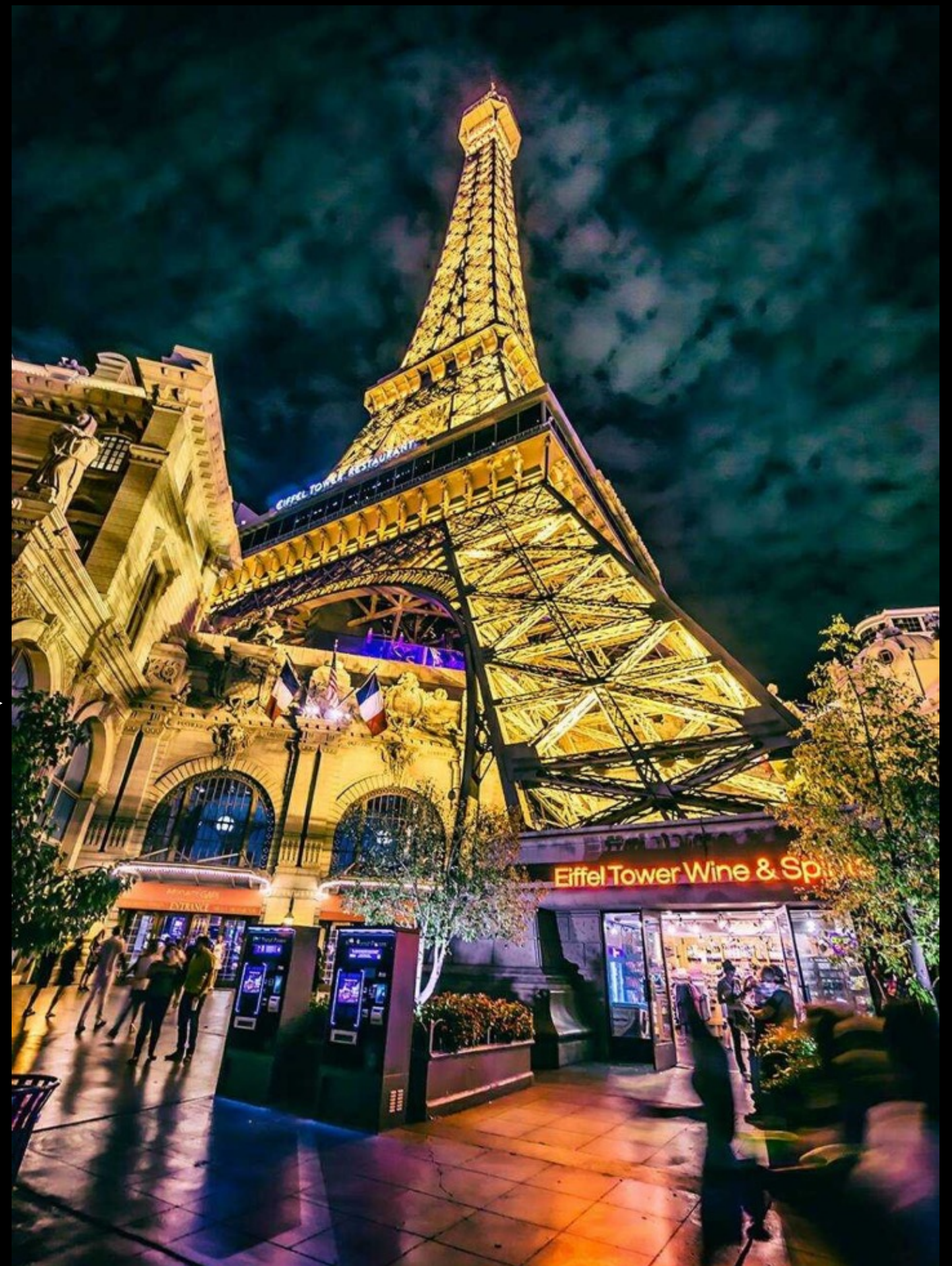
A few  
more steps...





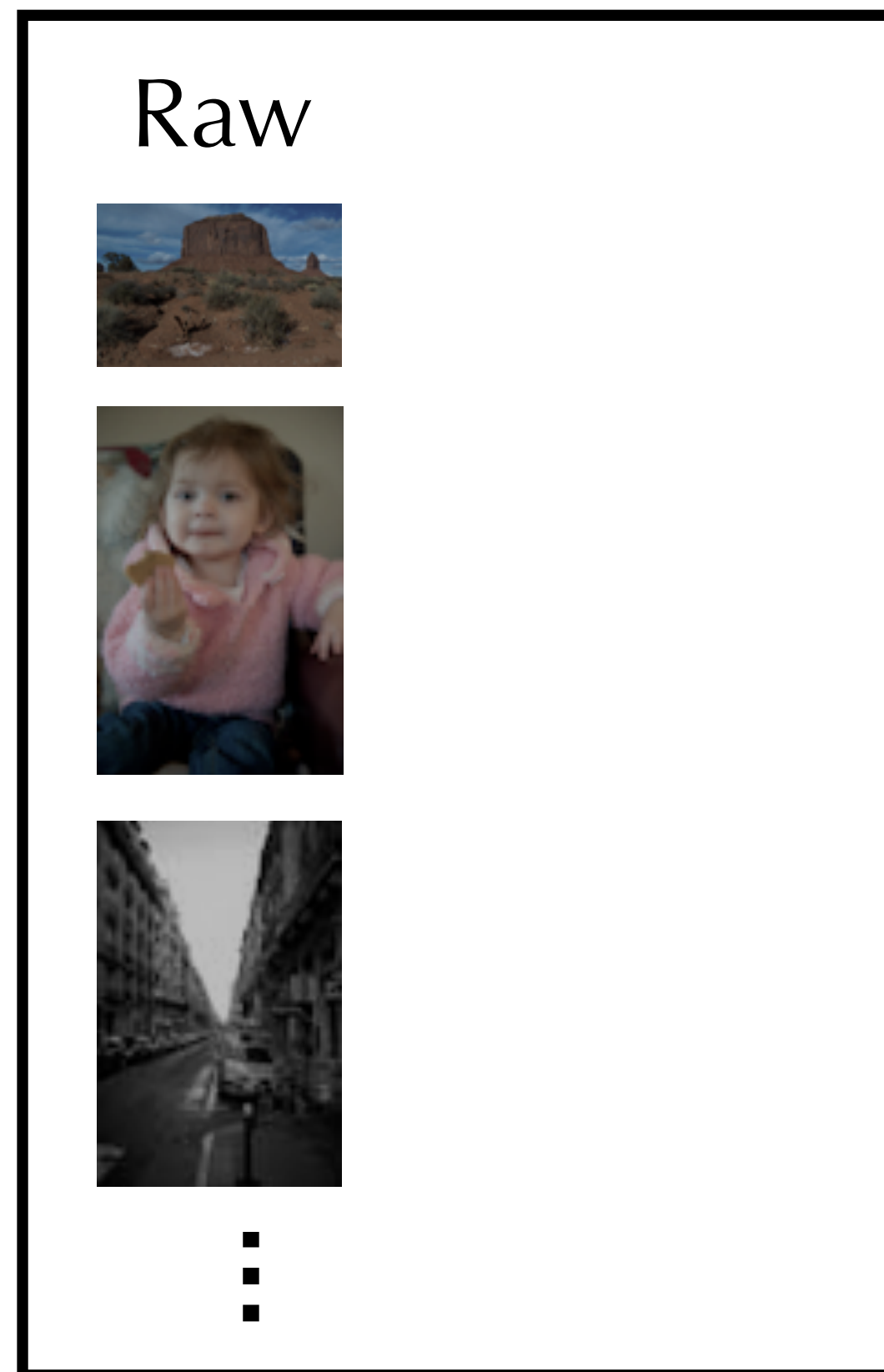


A few  
more steps...



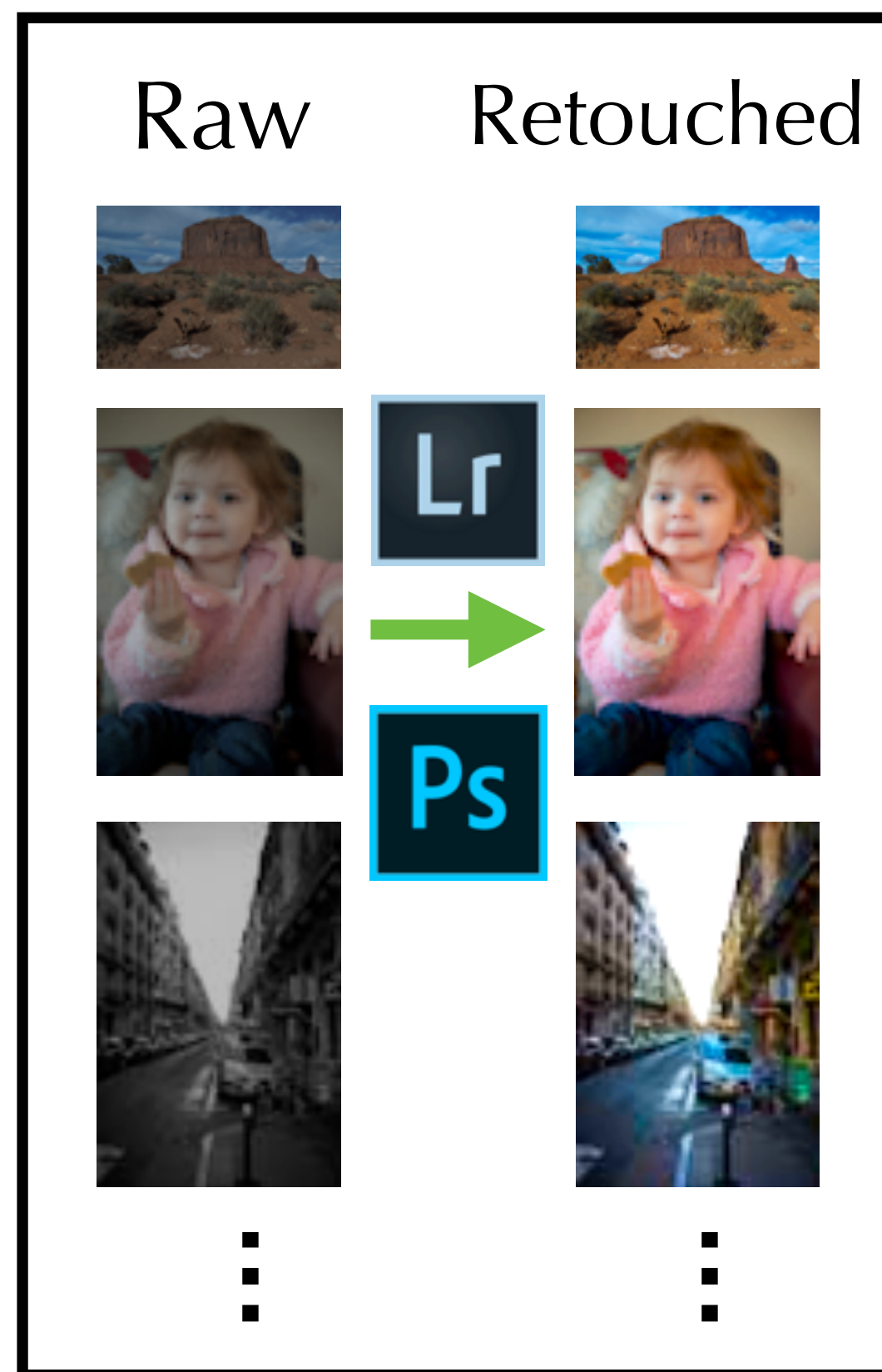
# Automatic Photo Post-Processing

Training Dataset



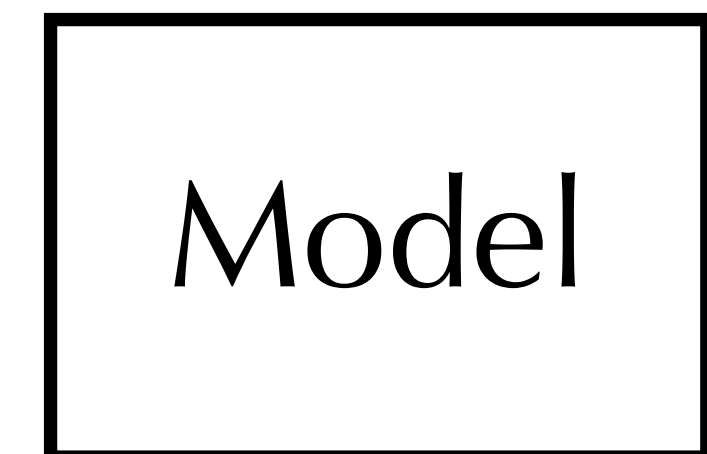
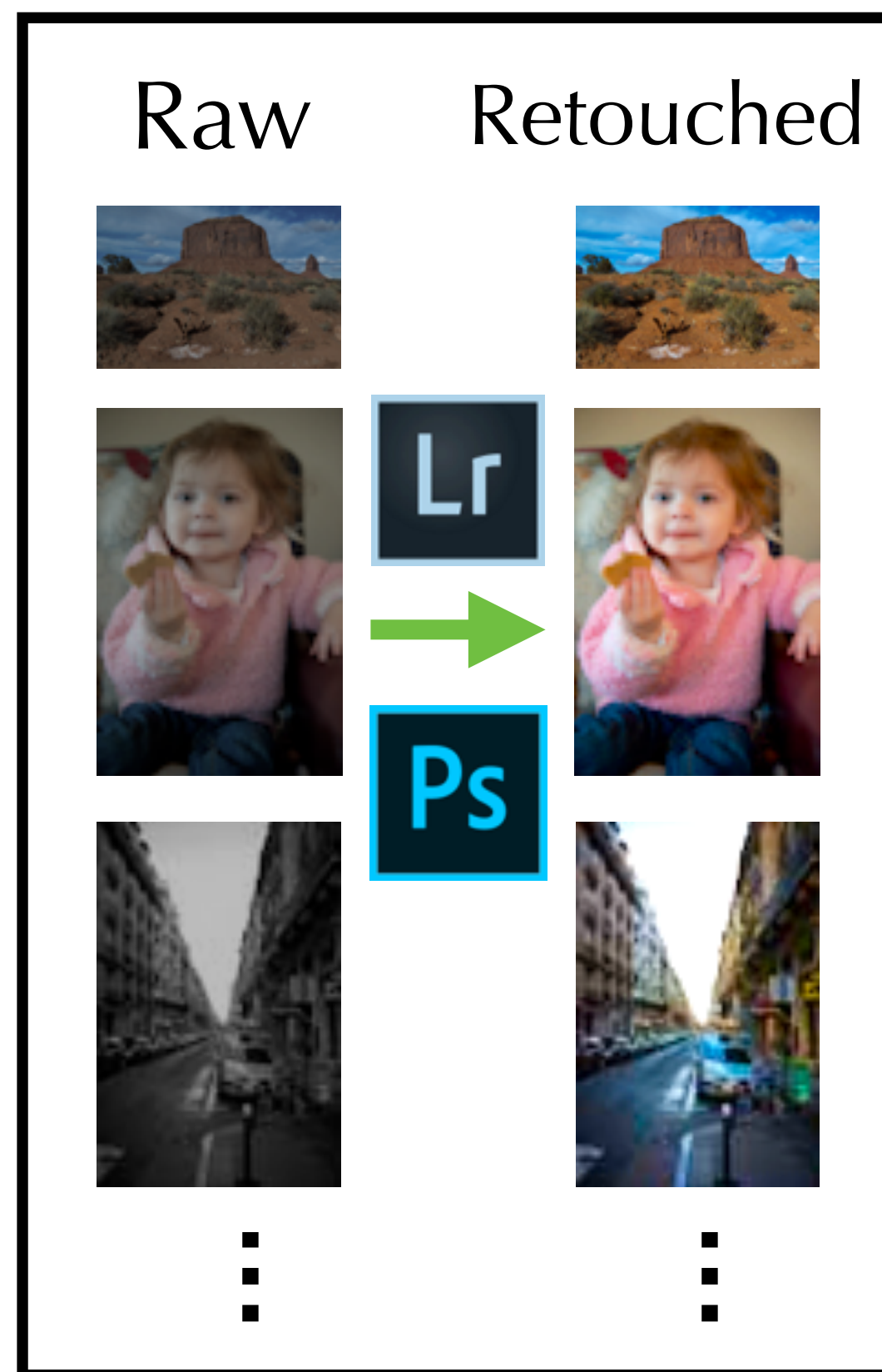
# Automatic Photo Post-Processing

Training Dataset



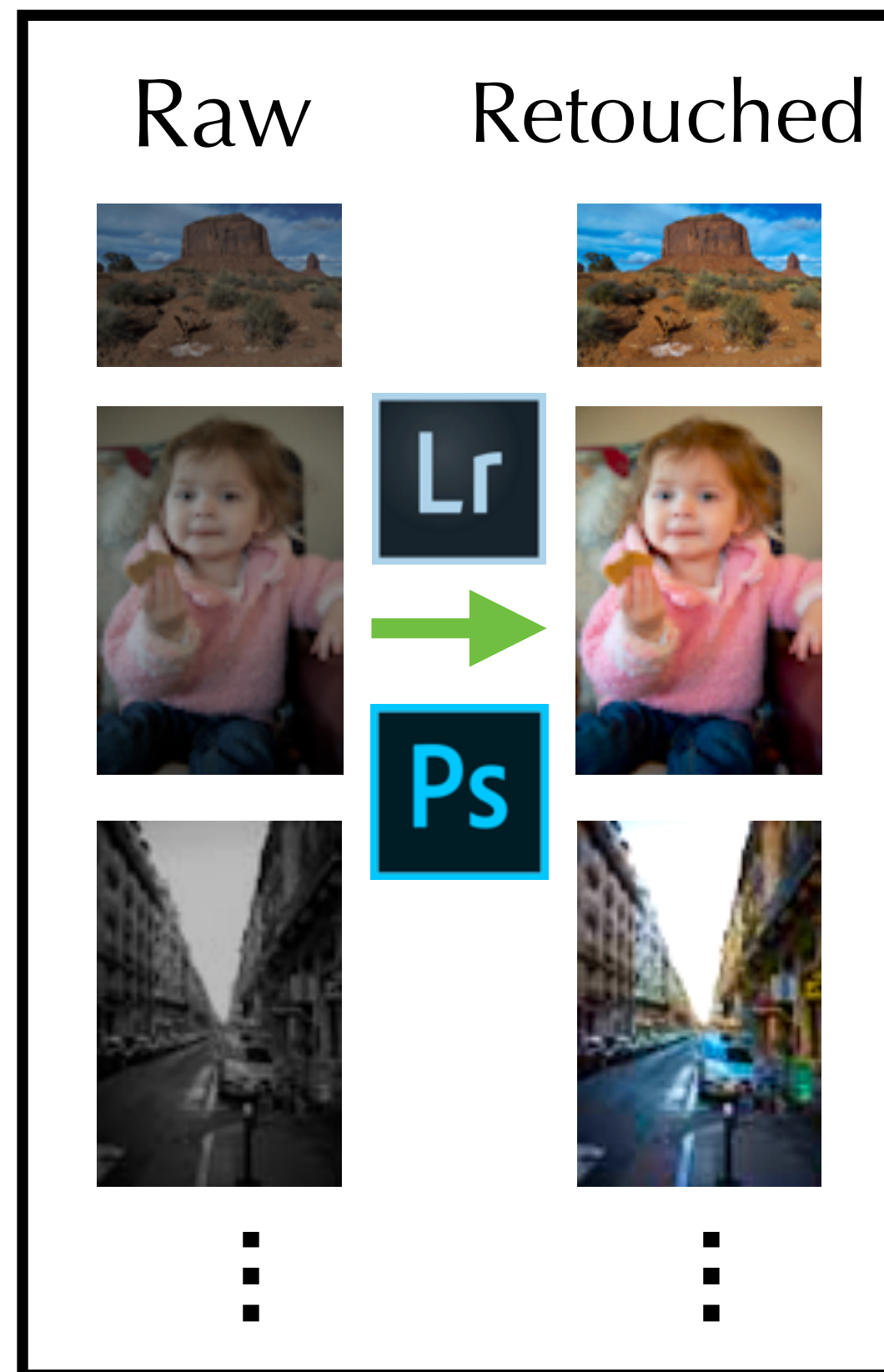
# Automatic Photo Post-Processing

Training Dataset

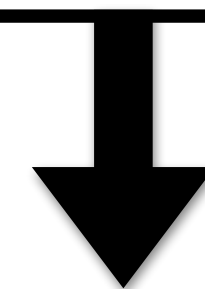
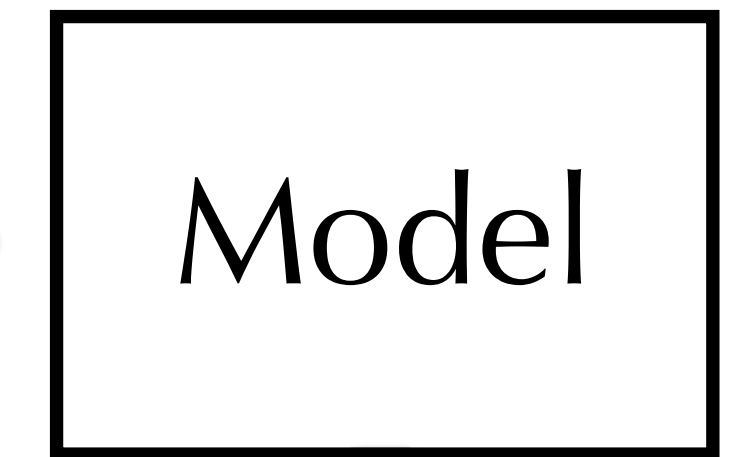
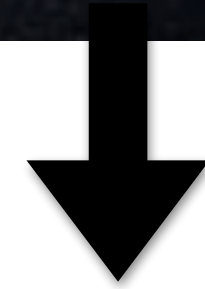


# Automatic Photo Post-Processing

Training Dataset



(Test) Raw photo



Post-processed photo



# Learning-based Photo Post-Processing

Bychkovsky et al. 2011, **Learning Photographic Global Tonal Adjustment with a Database of Input / Output Image Pairs**

MIT-Adobe FiveK Dataset



x5000

+

Learning-based  
Global Tonal  
Adjustment

# Learning-based Photo Post-Processing

Yan et al. 2014, **Automatic Photo Adjustment Using Deep Neural Networks**



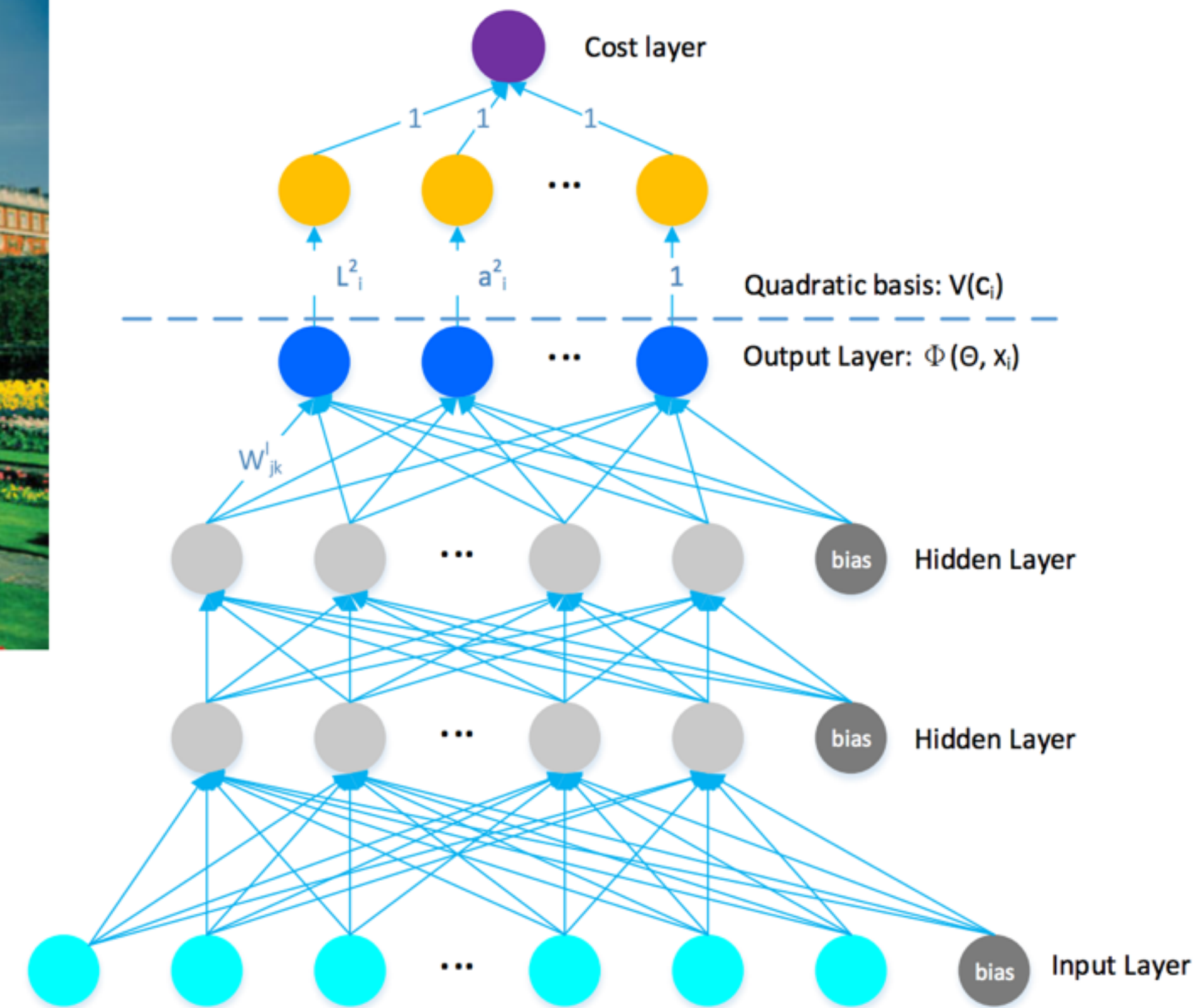
(a)

Input



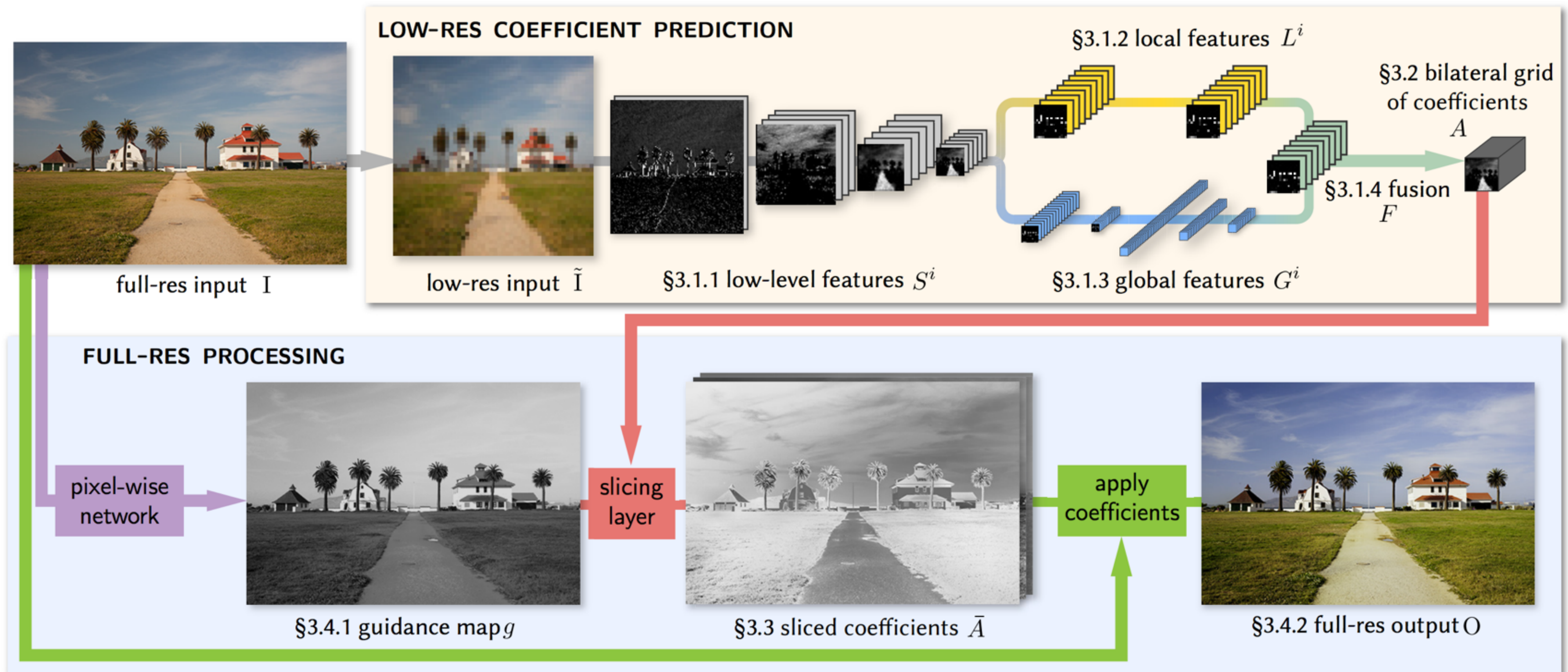
(b)

Output



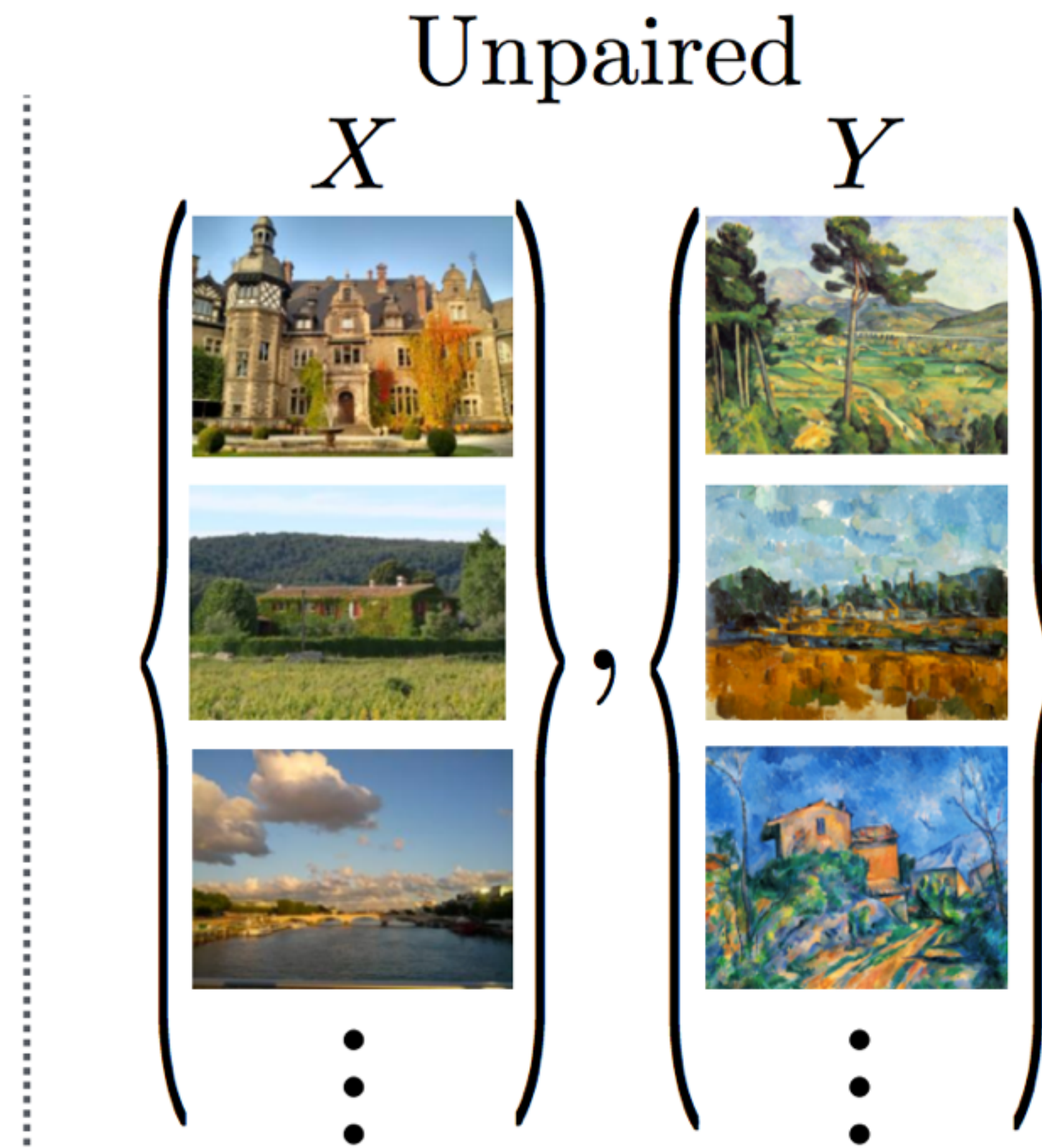
# Learning-based Photo Post-Processing

Gharbi et al., **Deep Bilateral Learning for Real-Time Image Enhancement**





# Paired v.s. Unpaired Image Translation

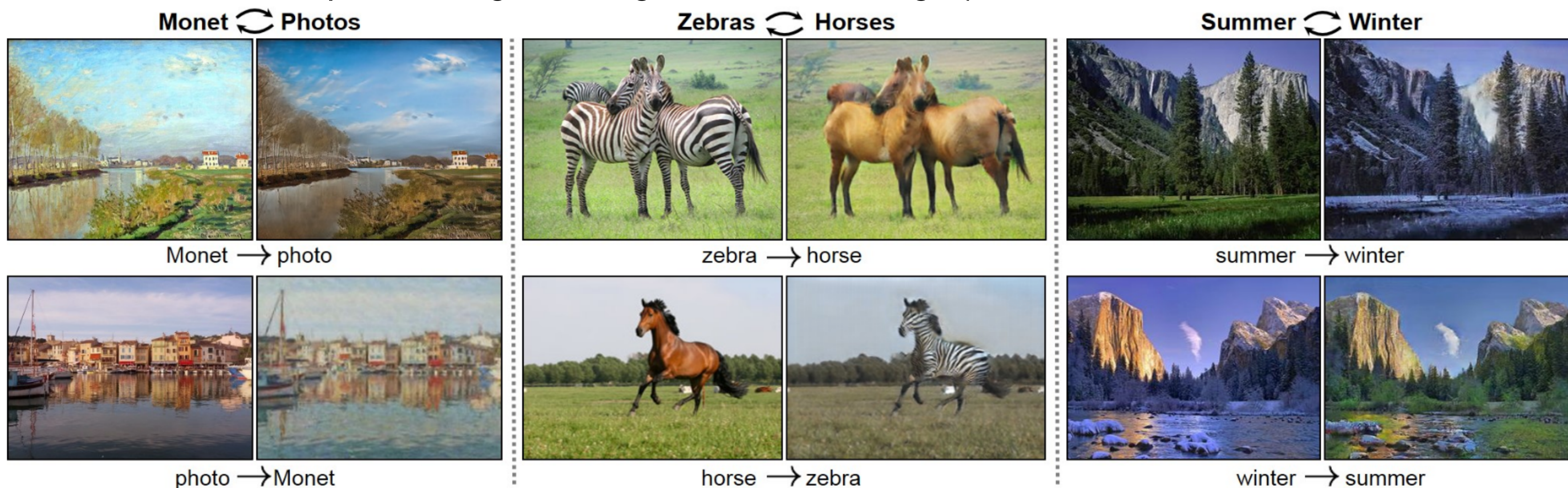


[Isola et al. 2017, **Image-to-Image Translation with Conditional Adversarial Networks**]

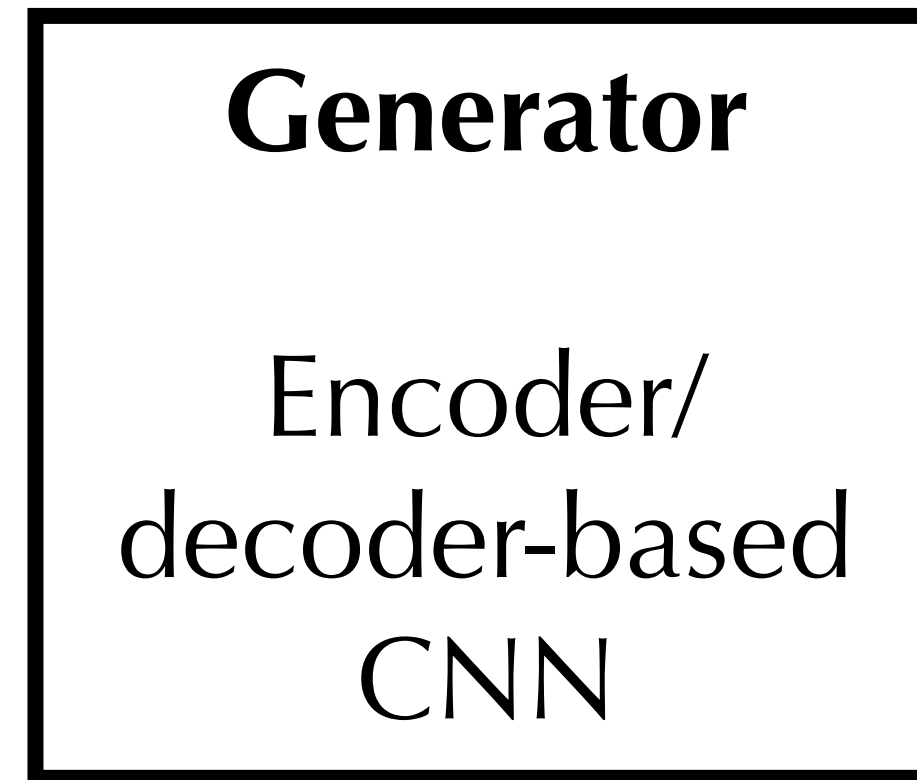
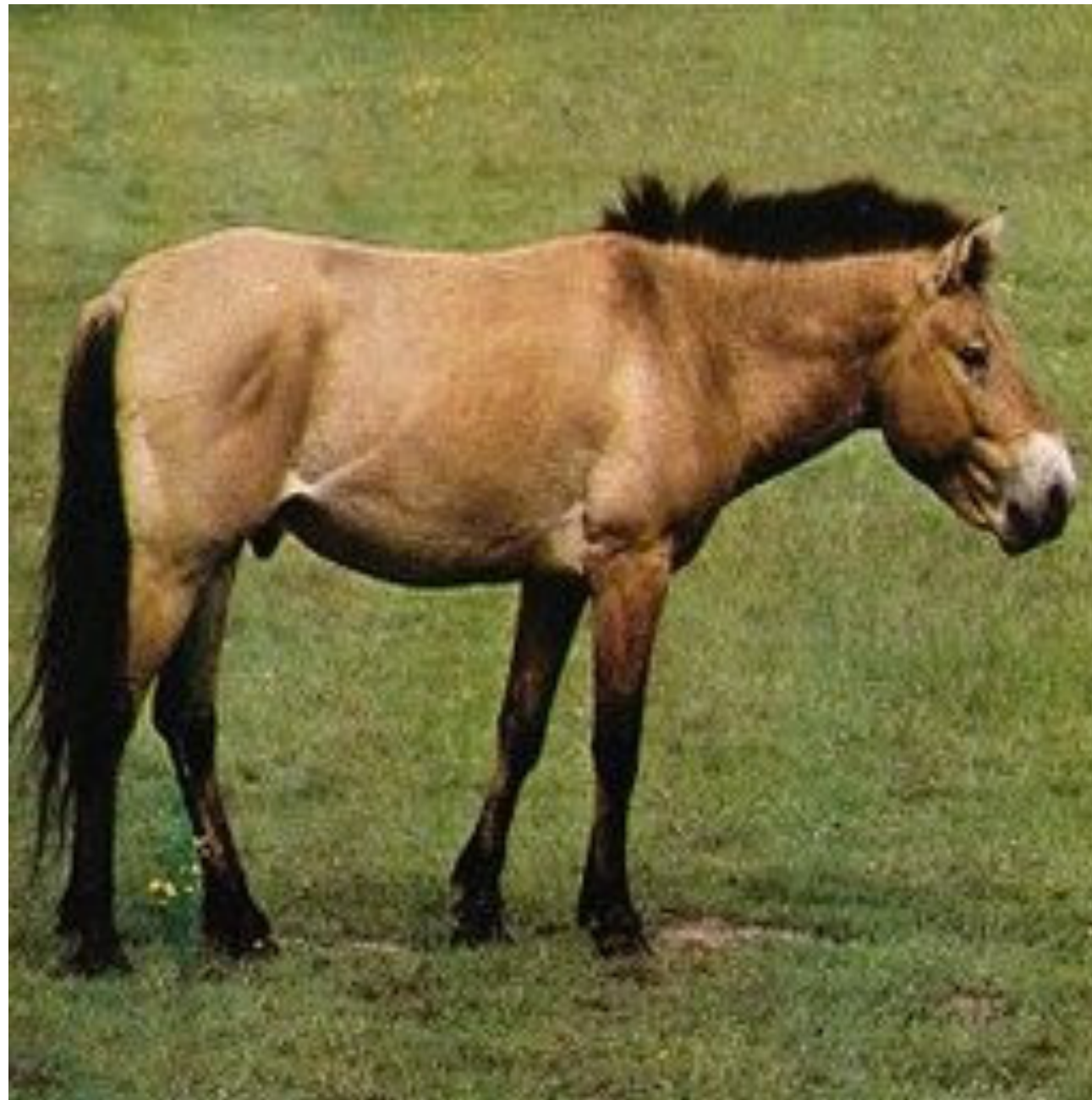
[Zhu et al. 2017, **Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks**]

# CycleGAN

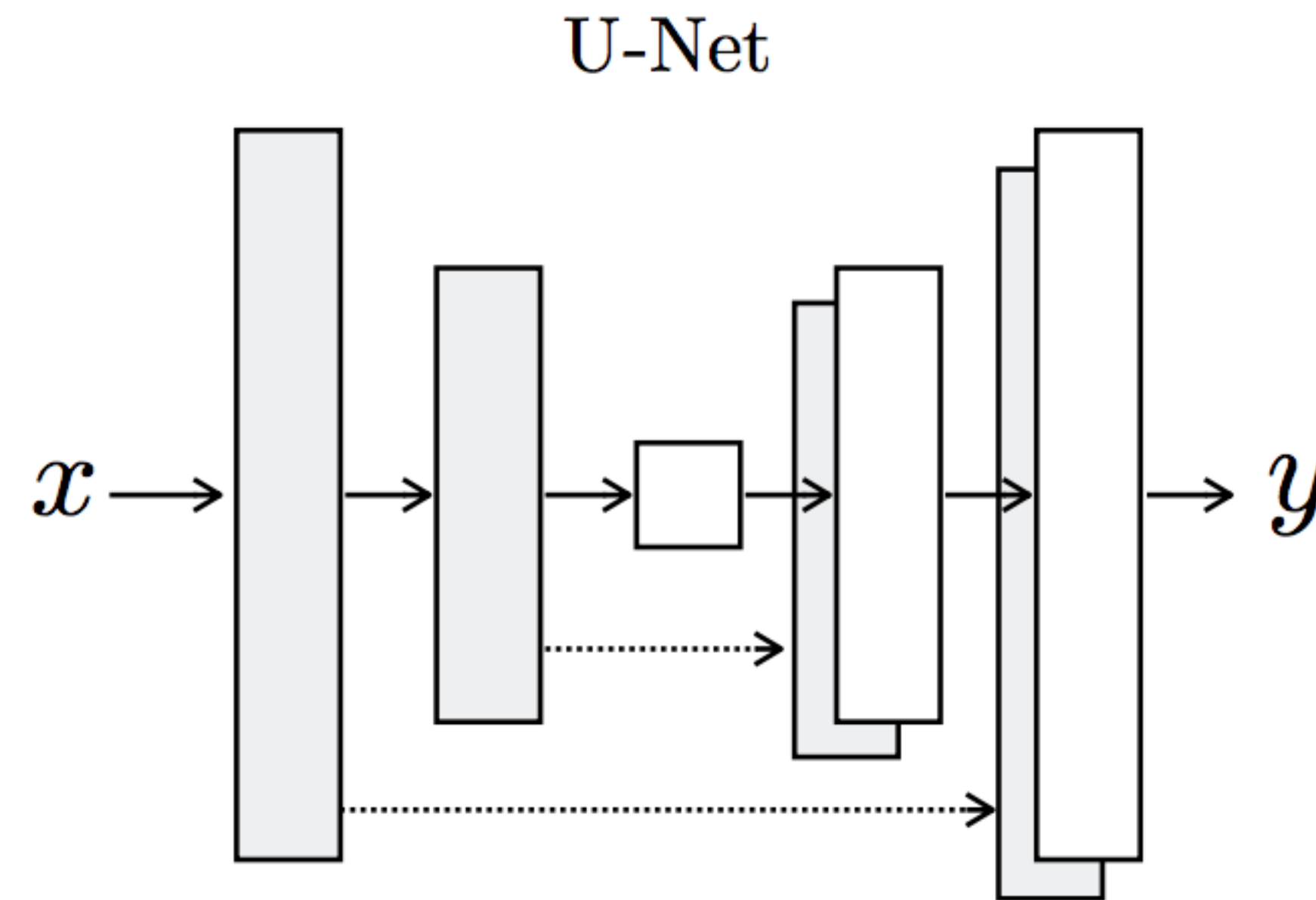
[Zhu et al. 2017, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks]



256x256 px

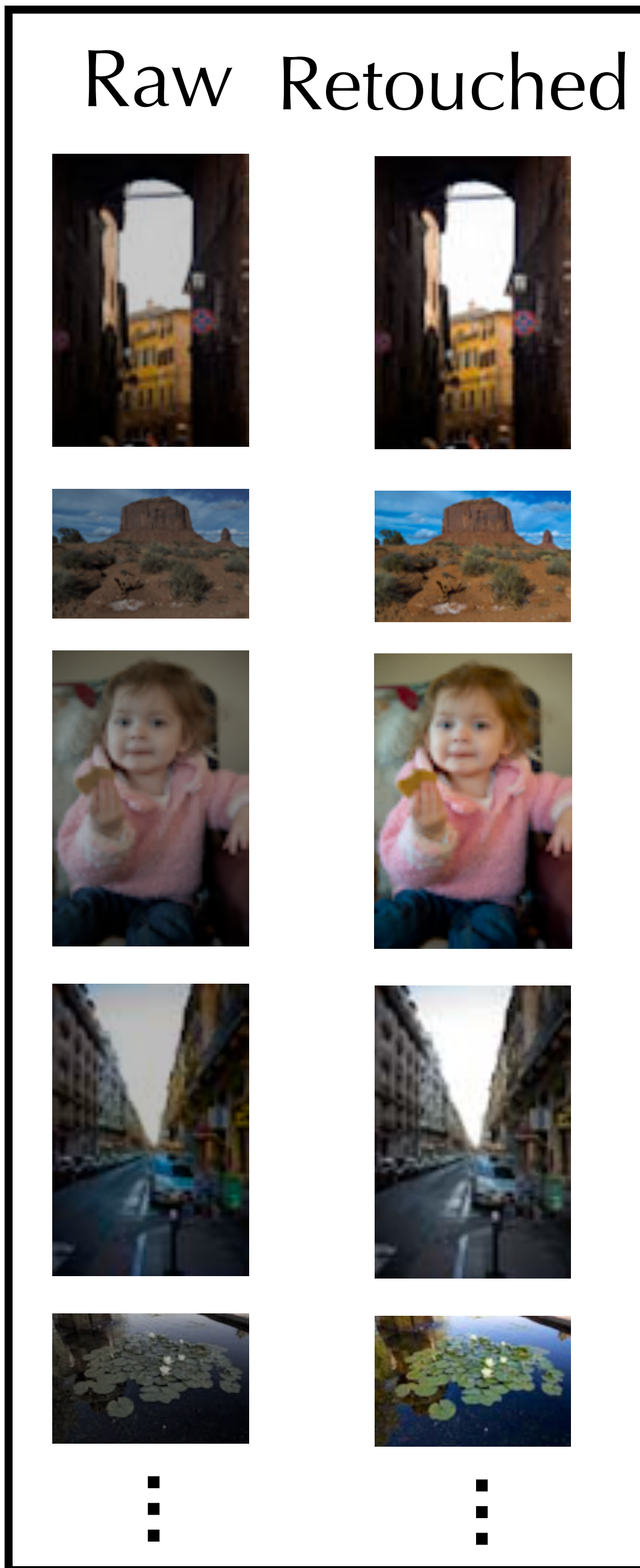


256x256 px



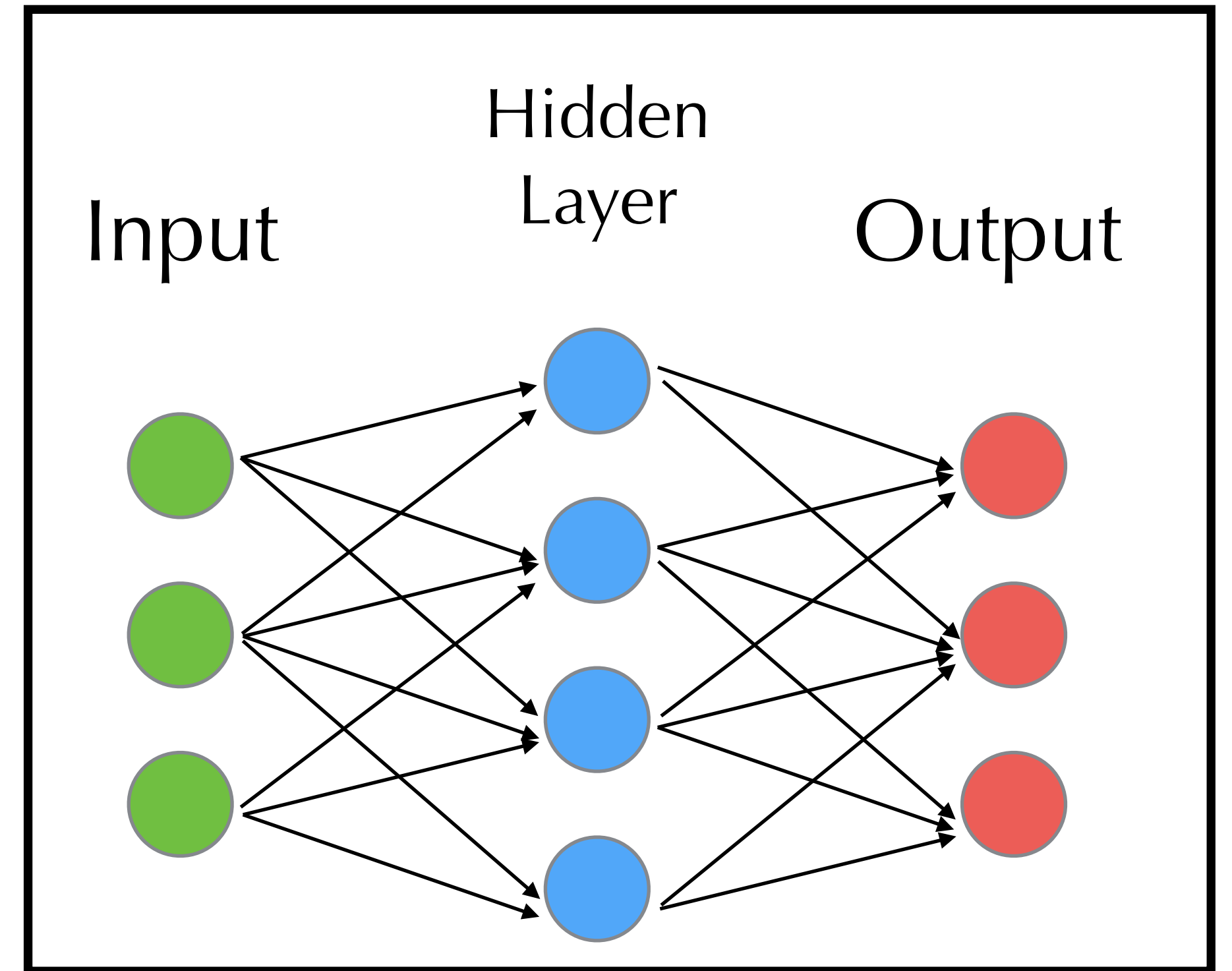
[Zhu et al. 2017, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks]

# Dataset



# Deep neural networks

Deep learning



Traditional deep-learning approaches generate **black boxes** (CNNs) out of existing ones (datasets).

To understand the magic of photo retouching, we need a **white box** result.

High Resolution

Human Understandable

Unpaired Training

End-to-end Processing

Tonal Adjustment Learning  
Bychkovsky et al. 2011



Color transform learning  
Yan et al.



Deep Bilateral Learning  
Gharbi et al.



CycleGAN,  
Zhu et al.



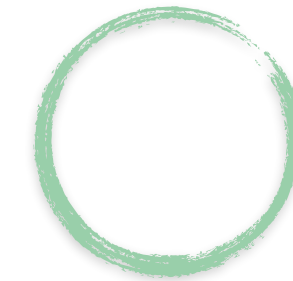
High Resolution

Human Understandable

Unpaired Training

End-to-end Processing

Tonal Adjustment Learning  
Bychkovsky et al. 2011



Color transform learning  
Yan et al.



Deep Bilateral Learning  
Gharbi et al.

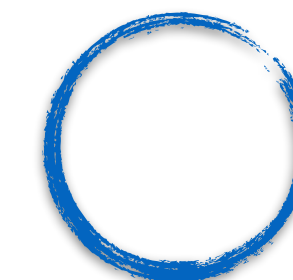
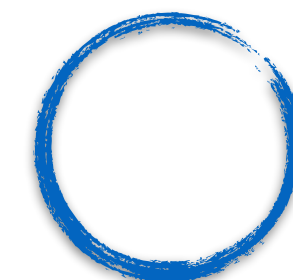
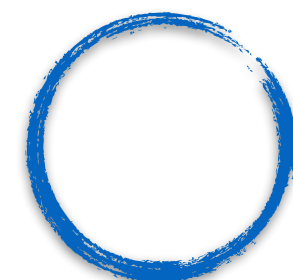


CycleGAN,  
Zhu et al.



---

Exposure (ours)



# Our Approach

**Differentiable Photo  
Postprocessing Model**

**Deep Reinforcement  
Learning**

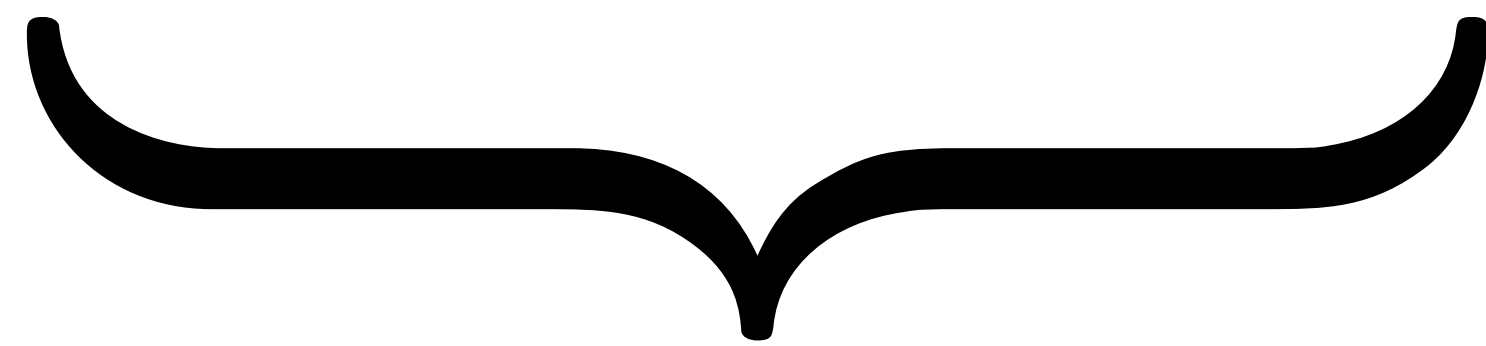
**Generative Adversarial  
Networks**

# Our Approach

**Differentiable Photo  
Postprocessing Model**

**Deep Reinforcement  
Learning**

**Generative Adversarial  
Networks**



**Modelling**



# Our Approach

**Differentiable Photo  
Postprocessing Model**

**Deep Reinforcement  
Learning**

**Generative Adversarial  
Networks**

**Modelling**

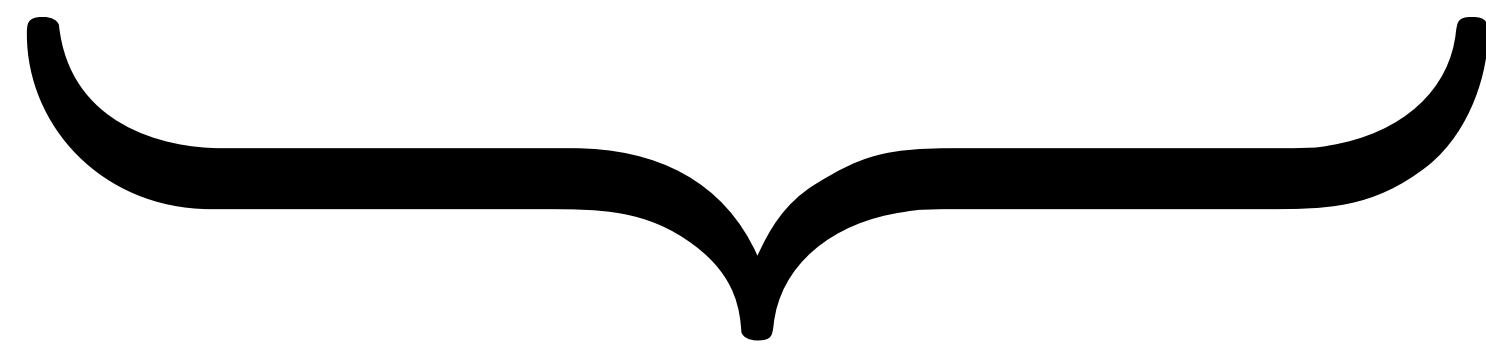
**Optimization**

# Our Approach

**Differentiable Photo  
Postprocessing Model**

**Deep Reinforcement  
Learning**

**Generative Adversarial  
Networks**



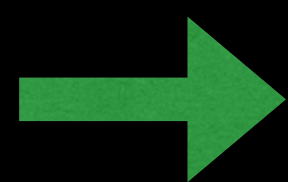
Modelling



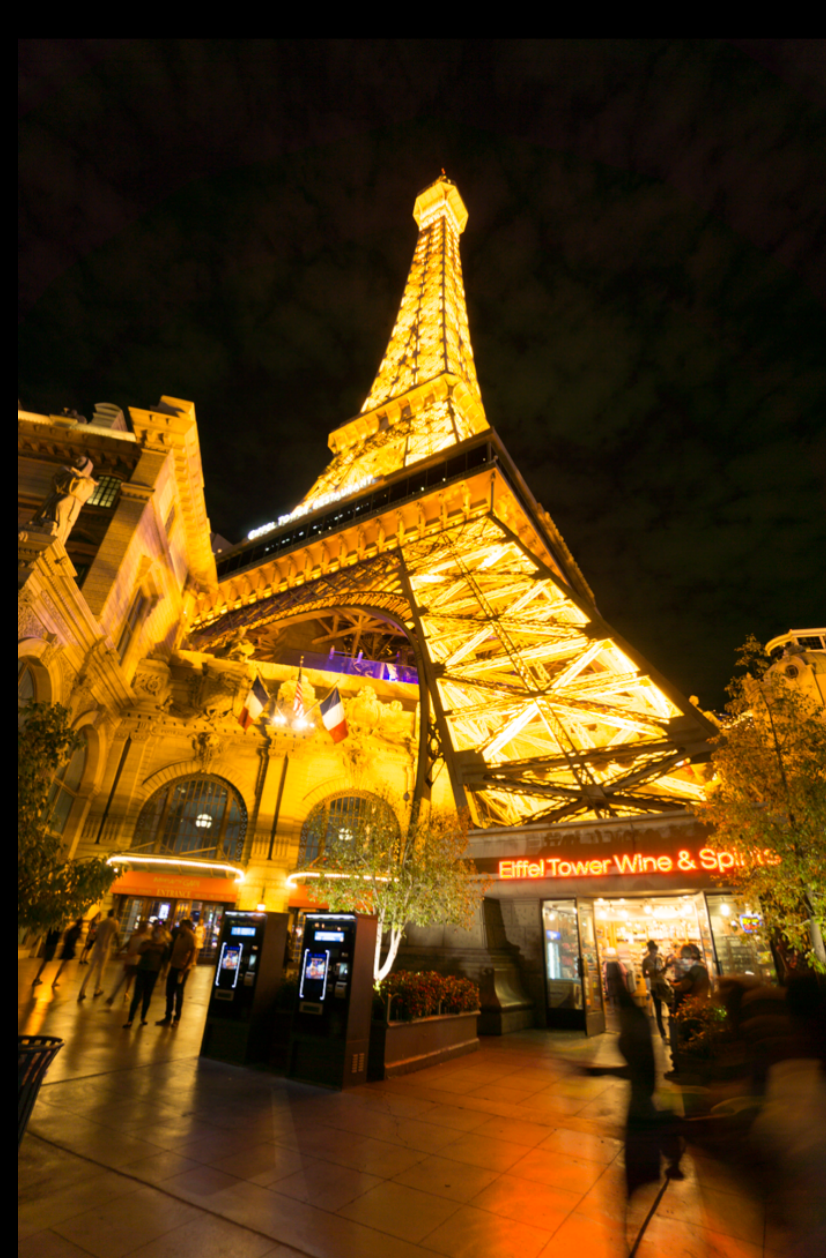
Optimization



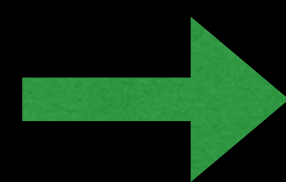
Exposure



+ 2.40



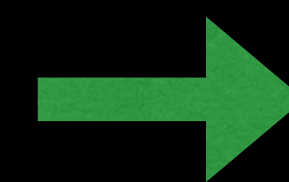
Highlight



-78



White Balance



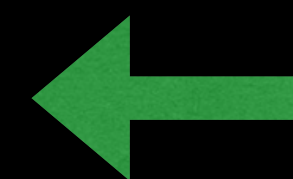
2600



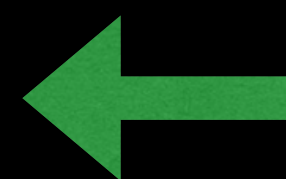
Clarity  
+63



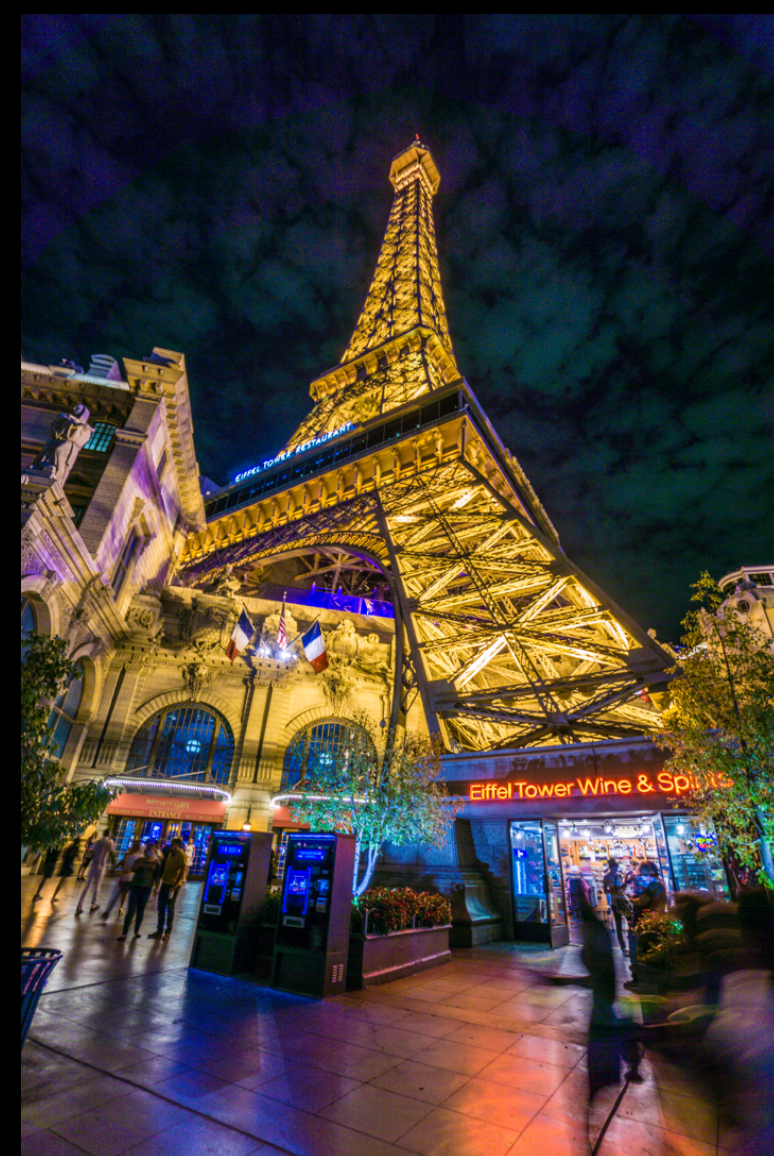
...



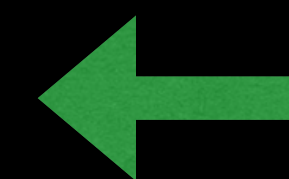
Shadow



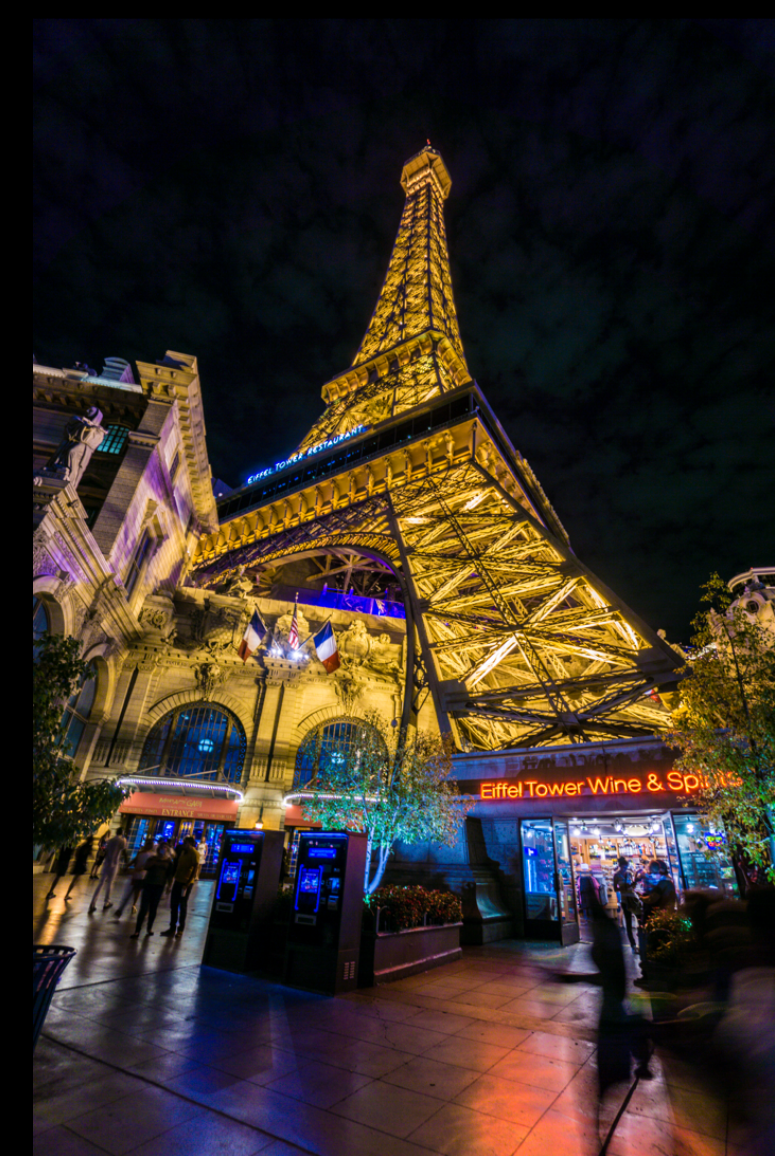
+70



Vibrance

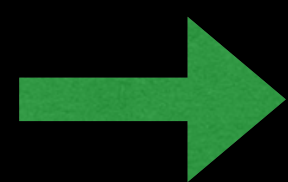


+75

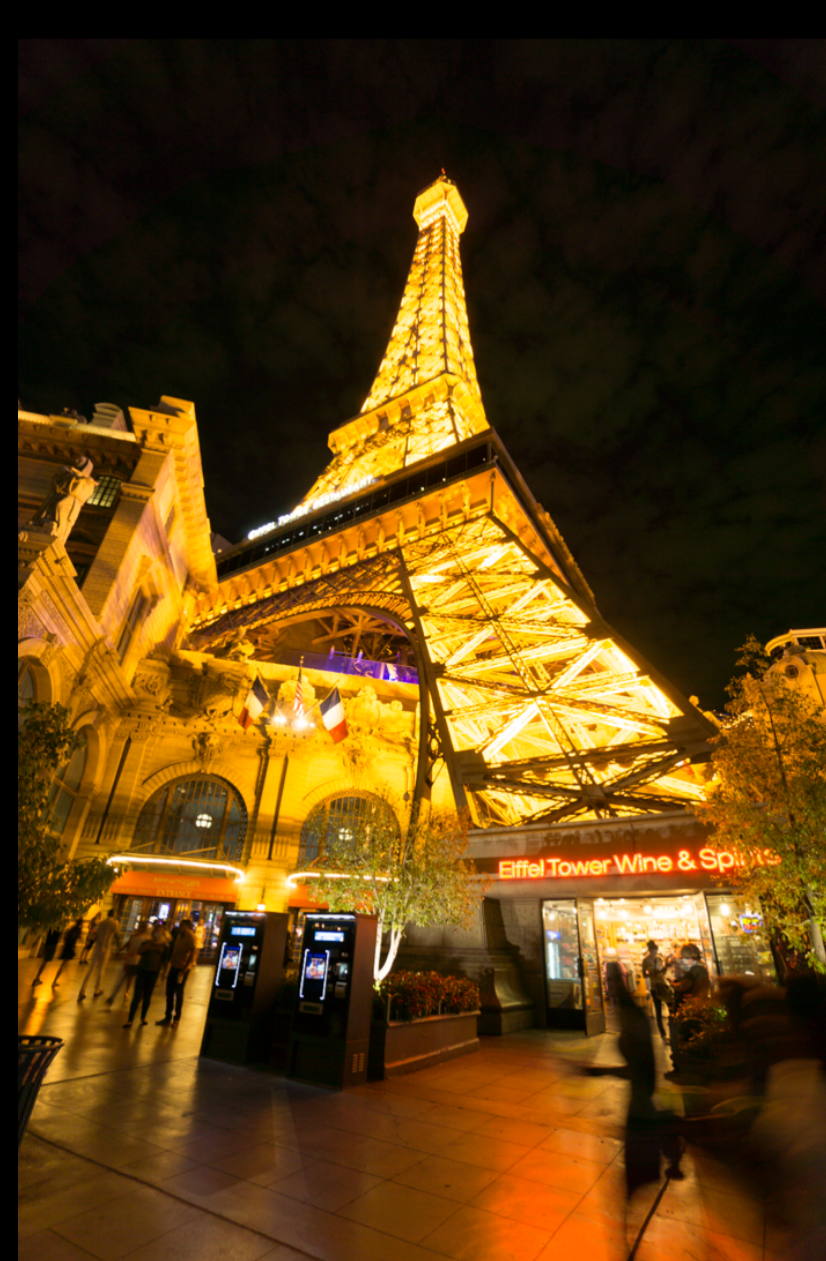




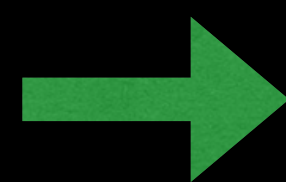
Exposure



+ 2.40



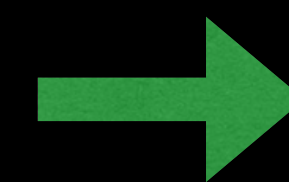
Highlight



-78



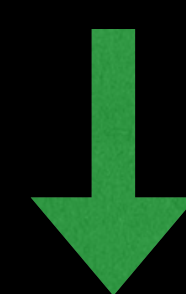
White Balance



2600



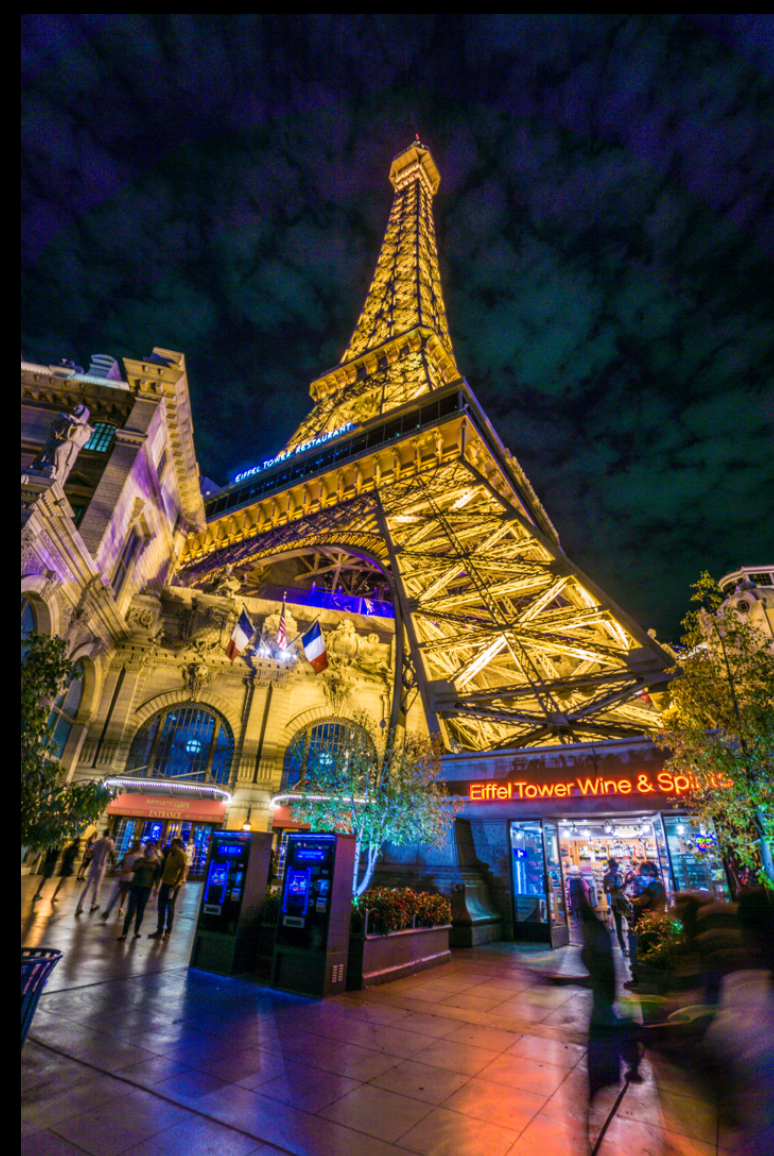
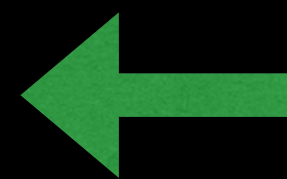
Pick a image operation, estimate the parameter. Repeat until done.



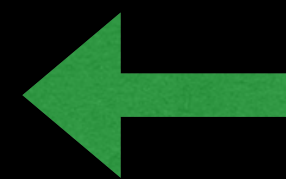
Clarity  
+63



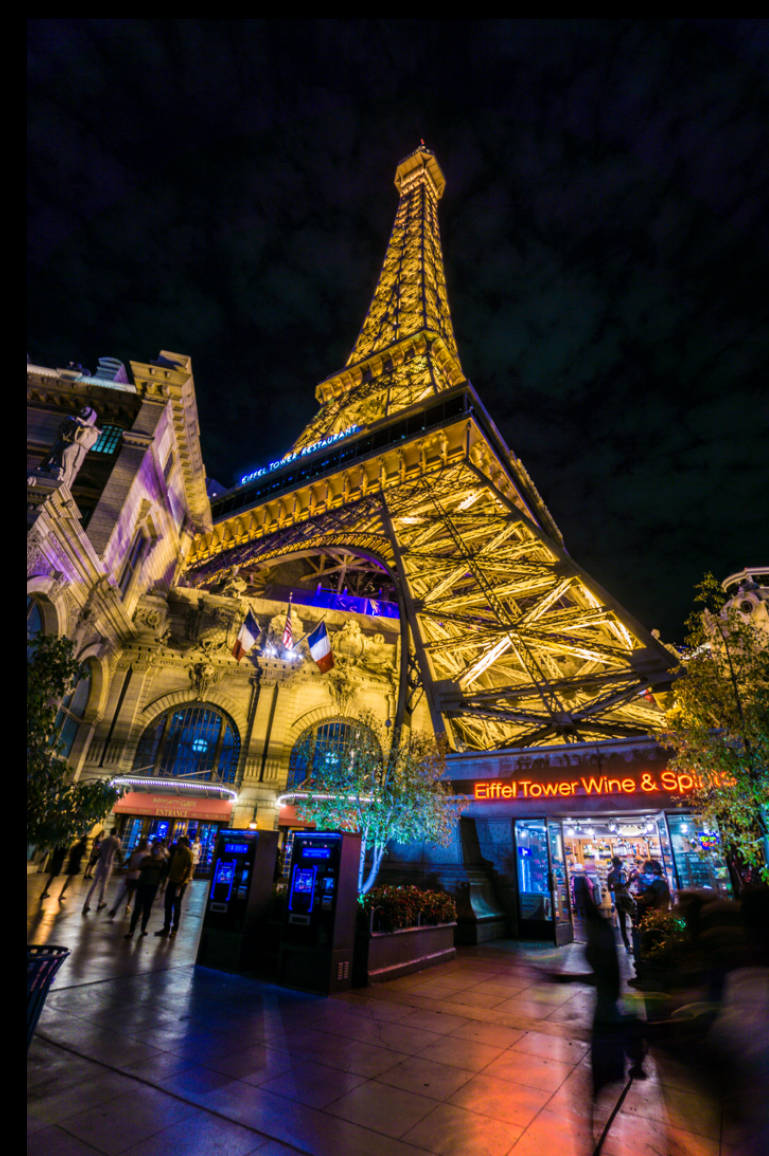
...



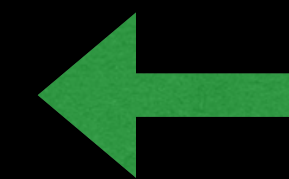
Shadow



+70



Vibrance



+75



# Image Operations (Filters)

Input



Filters

Exposure  
+0.5

Gamma 2

Color Curve  
(Boost Red)

Black & White  
+0.5

White Balance  
(Blue)

Saturation  
+0.5

Tone Curve

Contrast +0.8

Output



# Image Operations (Filters)

Input



Filters

Exposure  
+0.5

Gamma 2

Color Curve  
(Boost Red)

Black & White  
+0.5

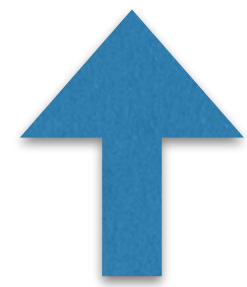
White Balance  
(Blue)

Saturation  
+0.5

Tone Curve

Contrast +0.8

Output



# Image Operations (Filters)

Input



Filters

Exposure  
+0.5

Gamma 2

Color Curve  
(Boost Red)

Black & White  
+0.5

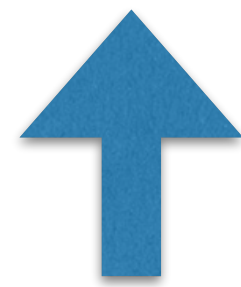
White Balance  
(Blue)

Saturation  
+0.5

Tone Curve

Contrast +0.8

Output



# Image Operations (Filters)

Input



Filters

Exposure  
+0.5

Gamma 2

Color Curve  
(Boost Red)

Black & White  
+0.5

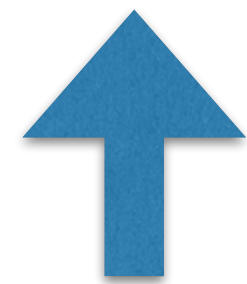
White Balance  
(Blue)

Saturation  
+0.5

Tone Curve

Contrast +0.8

Output





# Image Operations (Filters)

Input



Filters

Exposure  
+0.5

Gamma 2

Color Curve  
(Boost Red)

Black & White  
+0.5

White Balance  
(Blue)

Saturaion  
+0.5

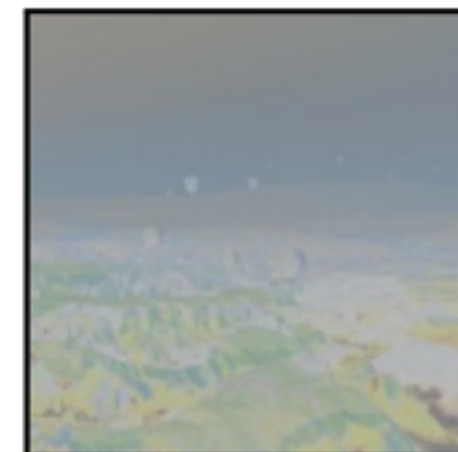
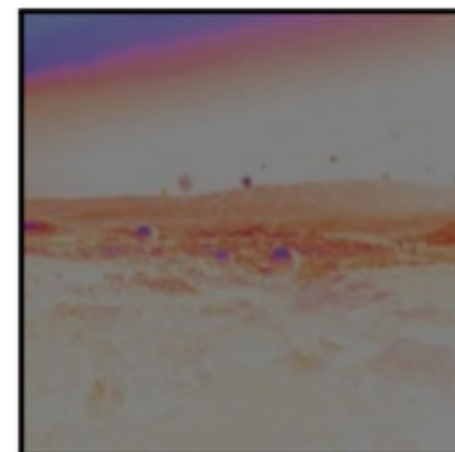
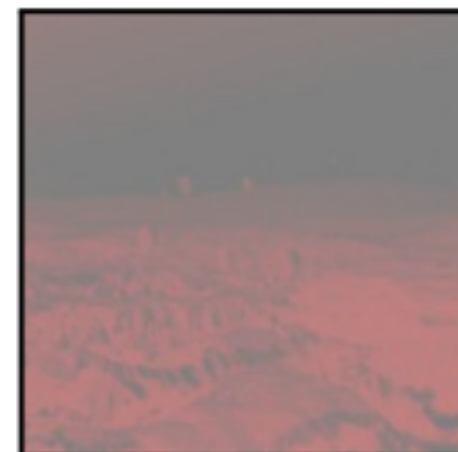
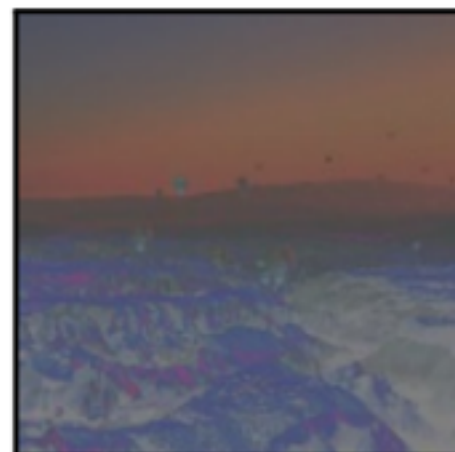
Tone Curve

Contrast +0.8

Output

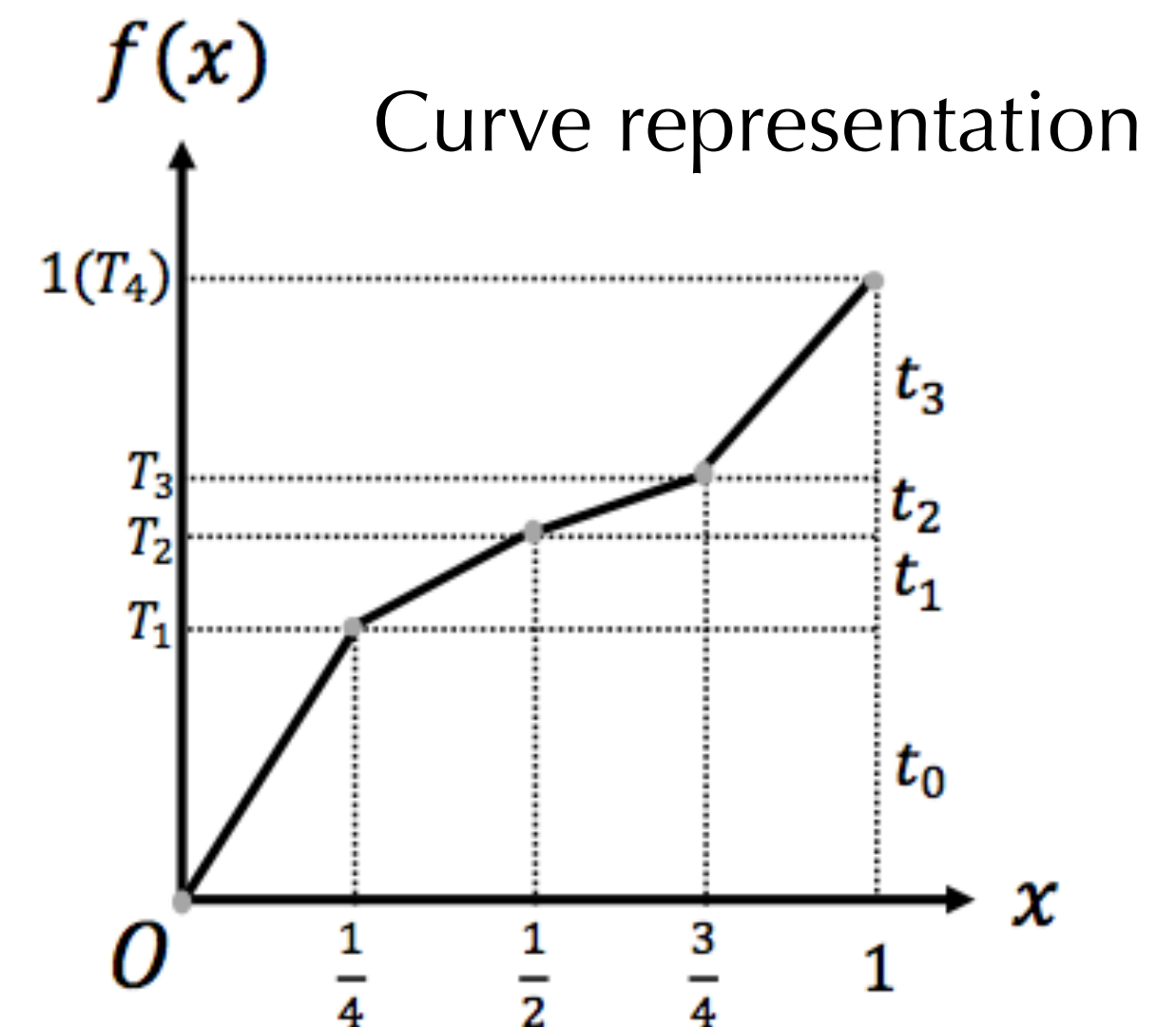


Gradient



# Image Operations (Filters)

Input



Filters

Exposure  
+0.5

Gamma 2

Color Curve  
(Boost Red)

Black & White  
+0.5

White Balance  
(Blue)

Saturation  
+0.5

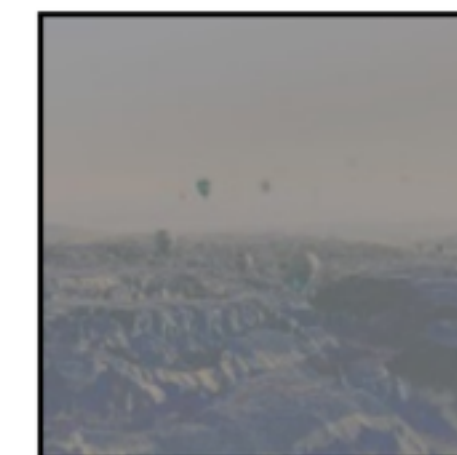
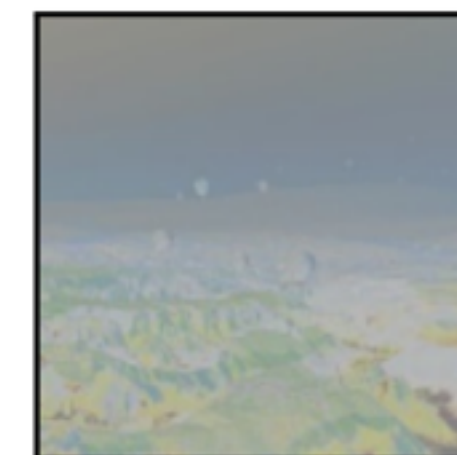
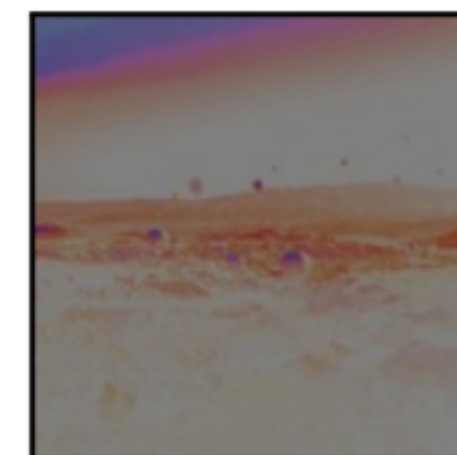
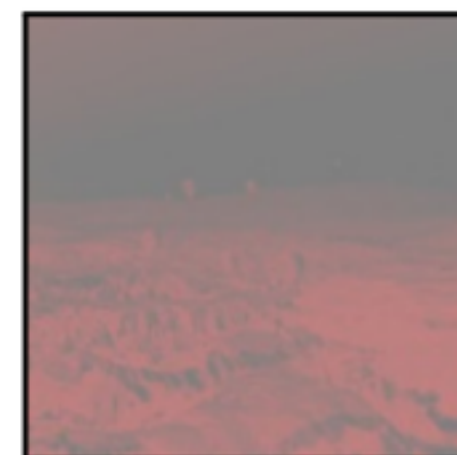
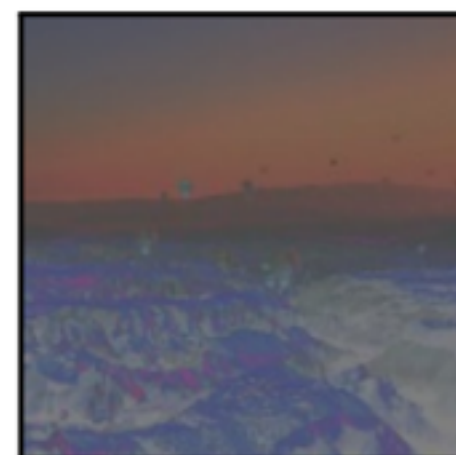
Tone Curve

Contrast +0.8

Output



Gradient



# Modelling Photo Post-Processing



Expo.		Tone	
Gam.		Cst.	■
W.B.	■	BW	
Satu.	■	Color	■

Contrast  $-0.59$

# Modelling Photo Post-Processing



Expo.		Tone	
Gam.		Cst.	■
W.B.	■	BW	
Satu.	■	Color	■

Contrast  $-0.59$

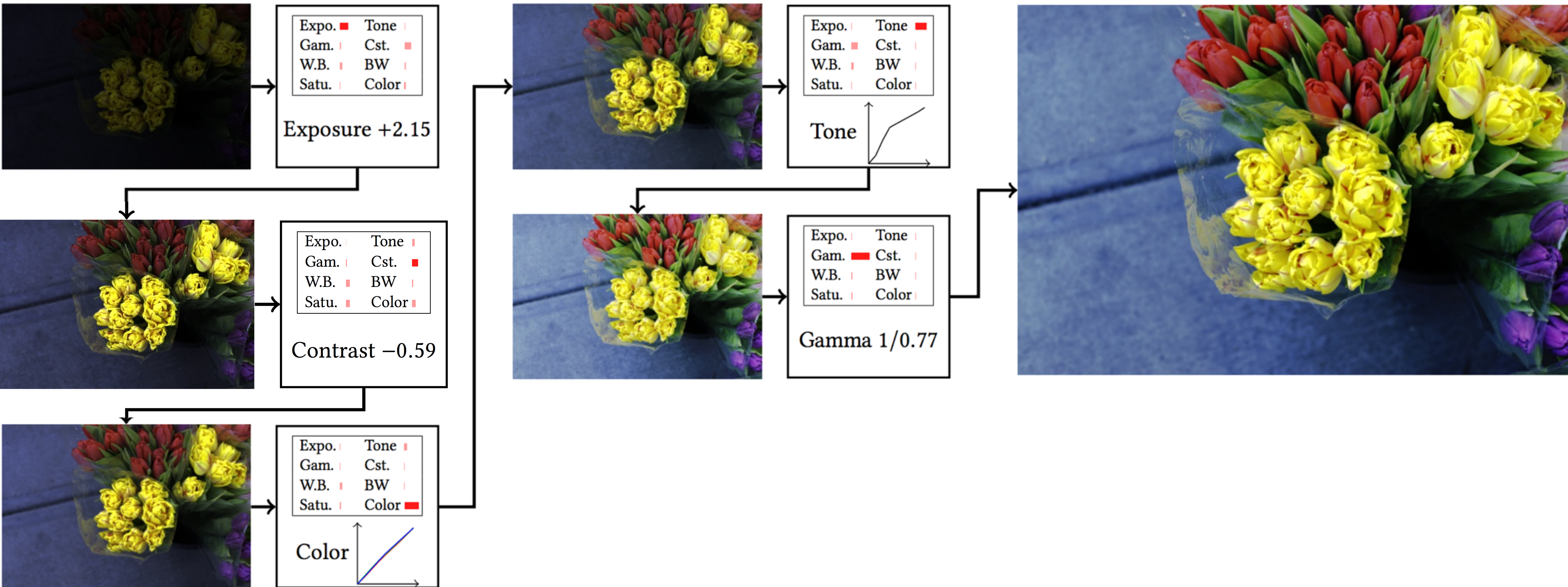
# Modelling Photo Post-Processing



Expo.		Tone	
Gam.		Cst.	■
W.B.	■	BW	
Satu.	■	Color	■

Contrast **-0.59**

# Modelling Photo Post-Processing



# Our Approach

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

**Deep Reinforcement  
Learning**

**Generative Adversarial  
Networks**

Modelling

Optimization

# Our Approach

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

**Deep Reinforcement  
Learning**

**Generative Adversarial  
Networks**

Modelling

Optimization



# Our Approach

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

**Deep Reinforcement  
Learning**

**Generative Adversarial  
Networks**

Modelling

Optimization

# Our Approach

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

**Deep Reinforcement  
Learning**

**Generative Adversarial  
Networks**

Modelling

Optimization

# Our Approach

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

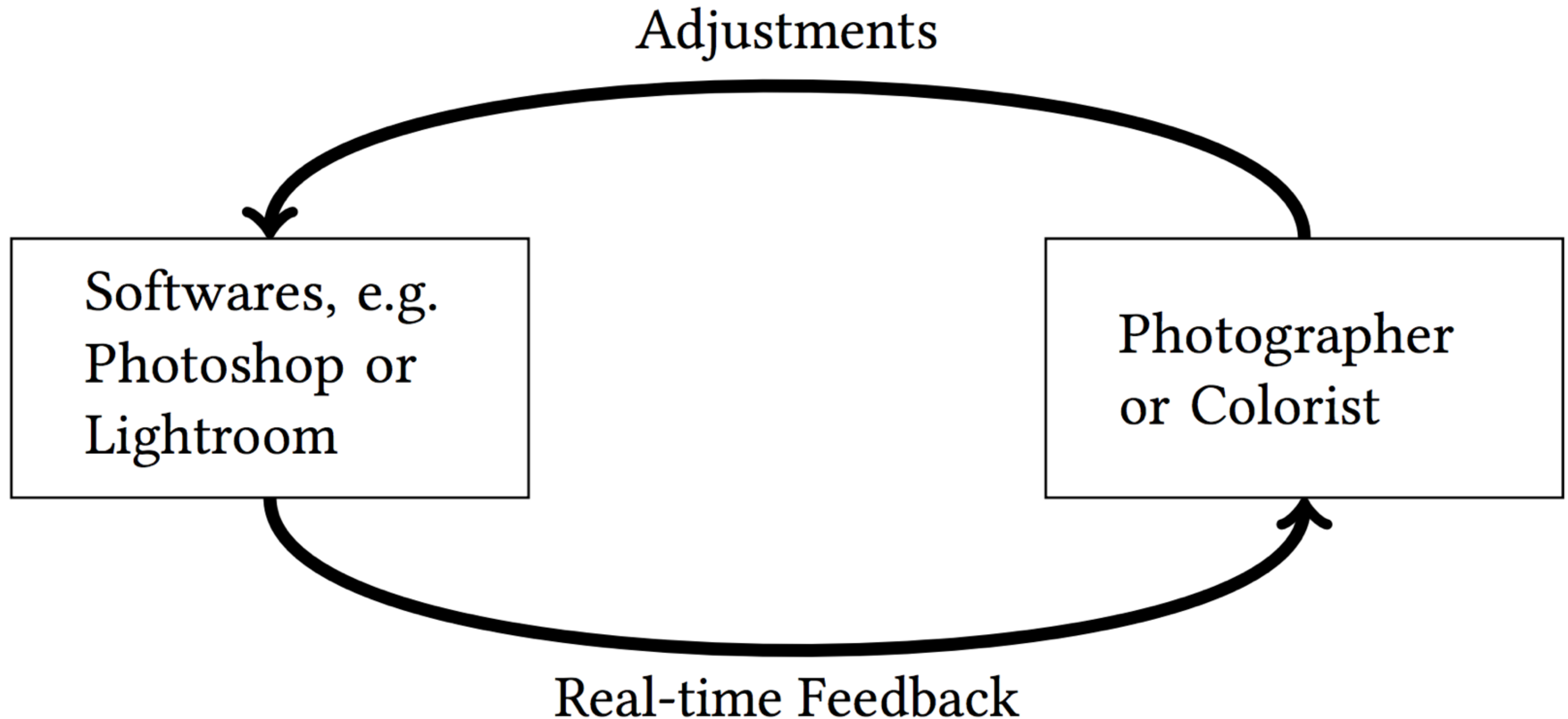
**Deep Reinforcement  
Learning**

**Generative Adversarial  
Networks**

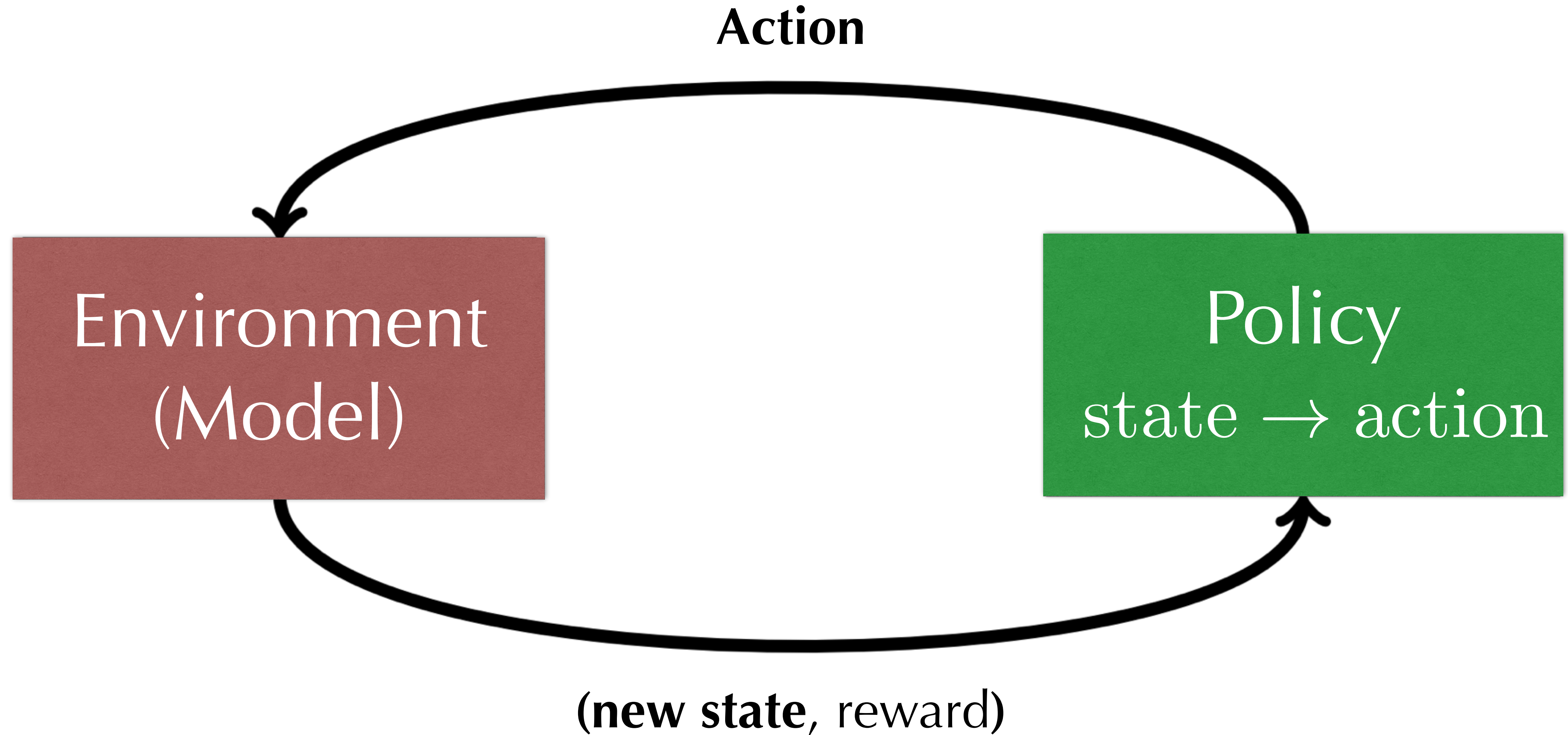
Modelling

Optimization

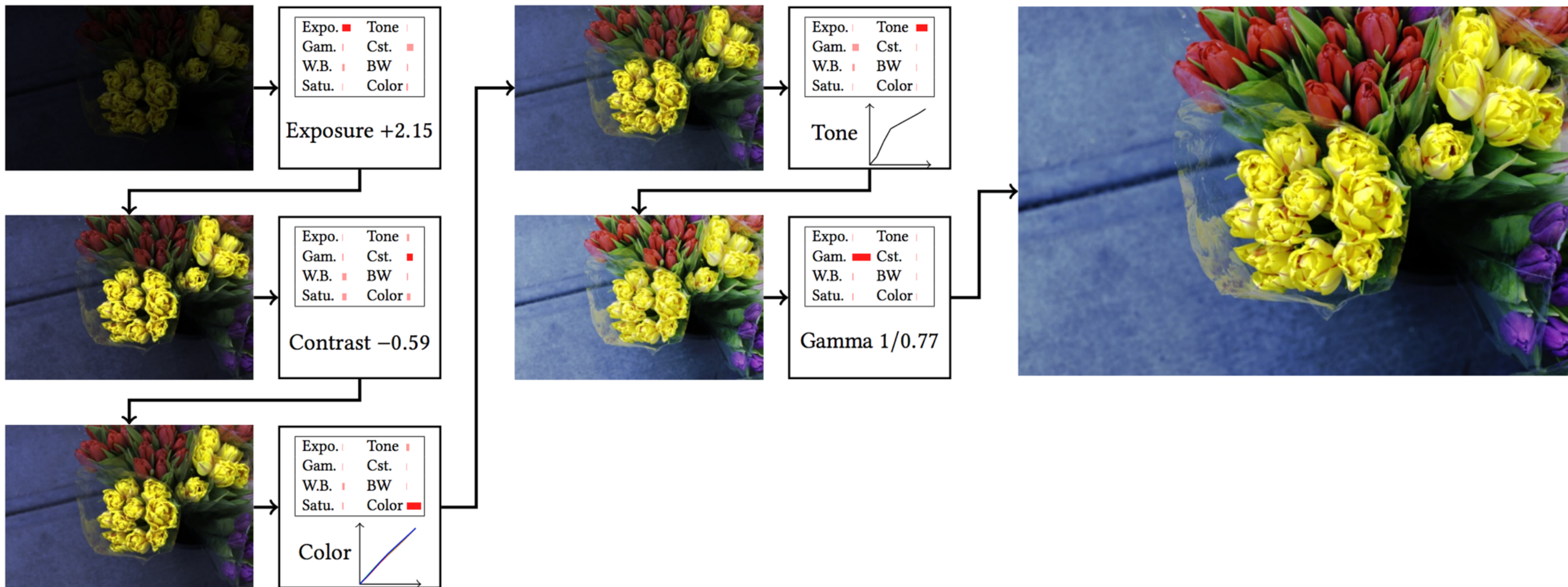
# Learning to Make Decisions



# Learning to Make Decisions



# Learning to Make Decisions



# Learning to Make Decisions

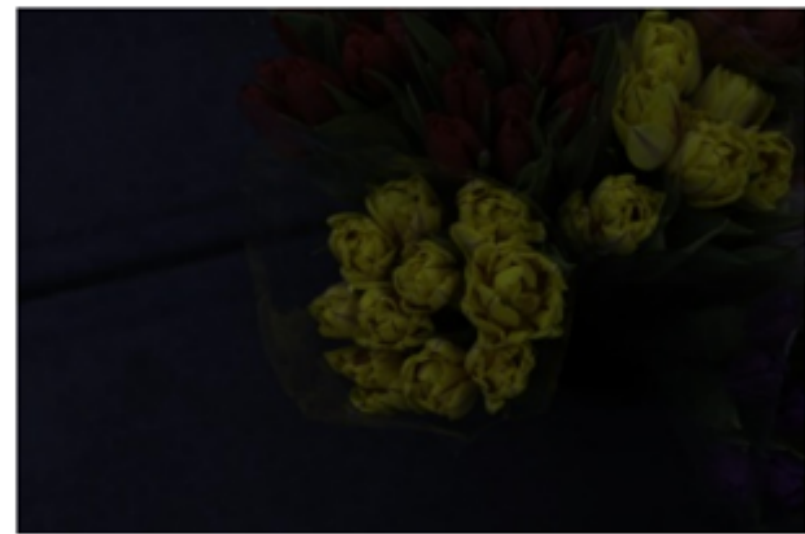
States

Actions

States

Actions

States



Expo. ■	Tone
Gam.	Cst. ■
W.B.	BW
Satu.	Color

Exposure +2.15



Expo.	Tone
Gam.	Cst. ■
W.B. ■	BW
Satu. ■	Color ■

Contrast -0.59



Expo.	Tone
Gam.	Cst.
W.B.	BW
Satu.	Color ■

Color



Expo.	Tone ■
Gam. ■	Cst.
W.B.	BW
Satu.	Color

Tone



Expo.	Tone
Gam. ■	Cst.
W.B.	BW
Satu.	Color

Gamma 1/0.77



# Gradient-based Policy Optimization

## Monte-Carlo Estimation of (Stochastic) Policy Gradient

Loss      Policy

Value

$$\nabla_{\theta_1} J(\pi_{\theta}) = \mathbb{E}_{\substack{s \sim \rho^{\pi} \\ a_1 \sim \pi_1(s) \\ a_2 = \pi_2(s, a_1)}} [\nabla_{\theta_1} \log \pi_1(a_1 | s) Q(s, (a_1, a_2))]$$

NN weights

Discrete policy (filter selection)

Continuous (deterministic) policy parameter estimation

Detailed description: The diagram illustrates the Monte-Carlo estimation of the policy gradient. On the left, the expression  $\nabla_{\theta_1} J(\pi_{\theta})$  is shown. An arrow labeled 'Loss' points to  $J(\pi_{\theta})$ , and another arrow labeled 'Policy' points to  $\pi_{\theta}$ . Below this expression, an arrow labeled 'NN weights' points upwards. The equation is set equal to an expectation over states  $s$  and actions  $a_1, a_2$ . An arrow labeled 'Value' points to the  $Q$  function in the expectation. Below the expectation, two lines describe the action sampling:  $a_1 \sim \pi_1(s)$  is labeled 'Discrete policy (filter selection)', and  $a_2 = \pi_2(s, a_1)$  is labeled 'Continuous (deterministic) policy parameter estimation'.



# Gradient-based Policy Optimization

## Deterministic Policy Gradient Theorem

The diagram illustrates the Deterministic Policy Gradient Theorem with several annotations. On the left, the expression  $\nabla_{\theta_2} J(\pi_{\theta})$  is shown. An arrow labeled "Loss" points to  $J(\pi_{\theta})$ , and an arrow labeled "Policy" points to  $\pi_{\theta}$ . Below this expression, an arrow labeled "NN weights" points up to  $\theta_2$ . The theorem is represented by an equals sign followed by an expectation operator  $\mathbb{E}_{s \sim \rho^{\pi}}$ . Above the expectation operator, an arrow labeled "Value" points to  $Q(s, (a_1, a_2))$ . The expression inside the expectation is  $[\nabla_{a_2} Q(s, (a_1, a_2)) \nabla_{\theta_2} \pi_2(s, a_1)]$ . Below the expectation operator, the text  $a_2 = \pi_2(s, a_1)$  is shown with an arrow pointing to the  $a_1$  argument of the policy function. To the right of this text is the phrase "Continuous (deterministic) policy parameter estimation" with an arrow pointing left towards the  $a_1$  argument.

$$\nabla_{\theta_2} J(\pi_{\theta}) = \mathbb{E}_{s \sim \rho^{\pi}} [\nabla_{a_2} Q(s, (a_1, a_2)) \nabla_{\theta_2} \pi_2(s, a_1)]$$

Continuous (deterministic) policy parameter estimation

# Our Approach: Learn **image operations**, instead of **pixels**.

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

**Deep Reinforcement  
Learning**

Learn **image operations**,  
instead of **pixels**

**Generative Adversarial  
Networks**

Modelling

Optimization

# Our Approach: Learn **image operations**, instead of **pixels**.

Differentiable Photo  
Postprocessing Model

resolution independent  
content preserving  
human-understandable

Deep Reinforcement  
Learning

Learn image **operations**,  
instead of **pixels**

Generative Adversarial  
Networks

Modelling

Optimization

# Our Approach: Learn **image operations**, instead of **pixels**.

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

**Deep Reinforcement  
Learning**

Learn **image operations**,  
instead of **pixels**

**Generative Adversarial  
Networks**

Modelling

Optimization

# Our Approach: Learn **image operations**, instead of **pixels**.

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

**Deep Reinforcement  
Learning**

Learn **image operations**,  
instead of **pixels**

**Generative Adversarial  
Networks**

Modelling

Optimization

# Designing the Reward Function

$$r = -D($$



Generated

,



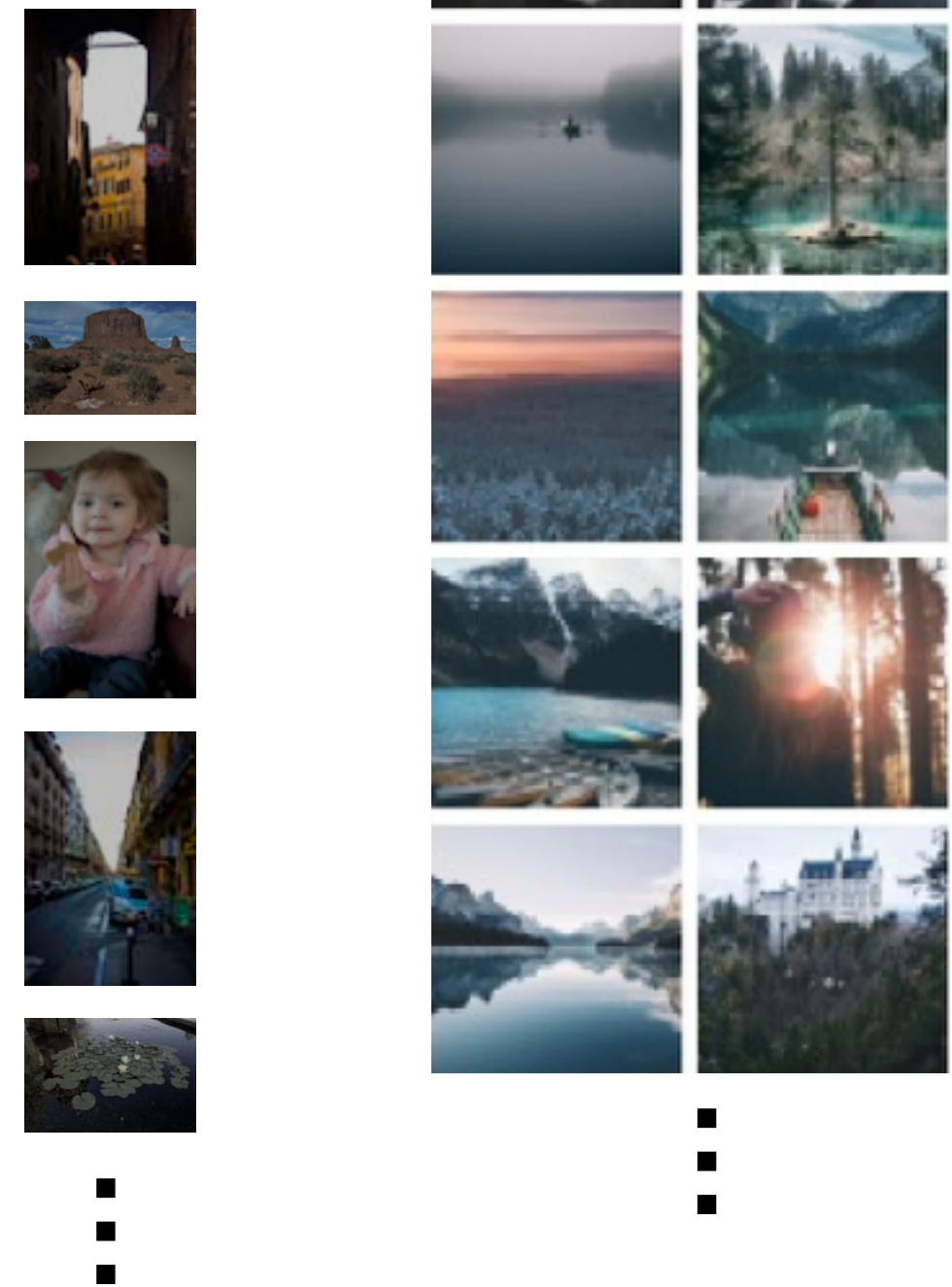
Target (i.e. "ground truth")

)

# (Conditional) Generative Adversarial Networks (c-GANs)

Retouched

Raw



Real Images

Real sample

**Generator**  
Encoder/  
decoder-based  
CNN

"Fake" sample

**Discriminator**  
Classification  
CNN

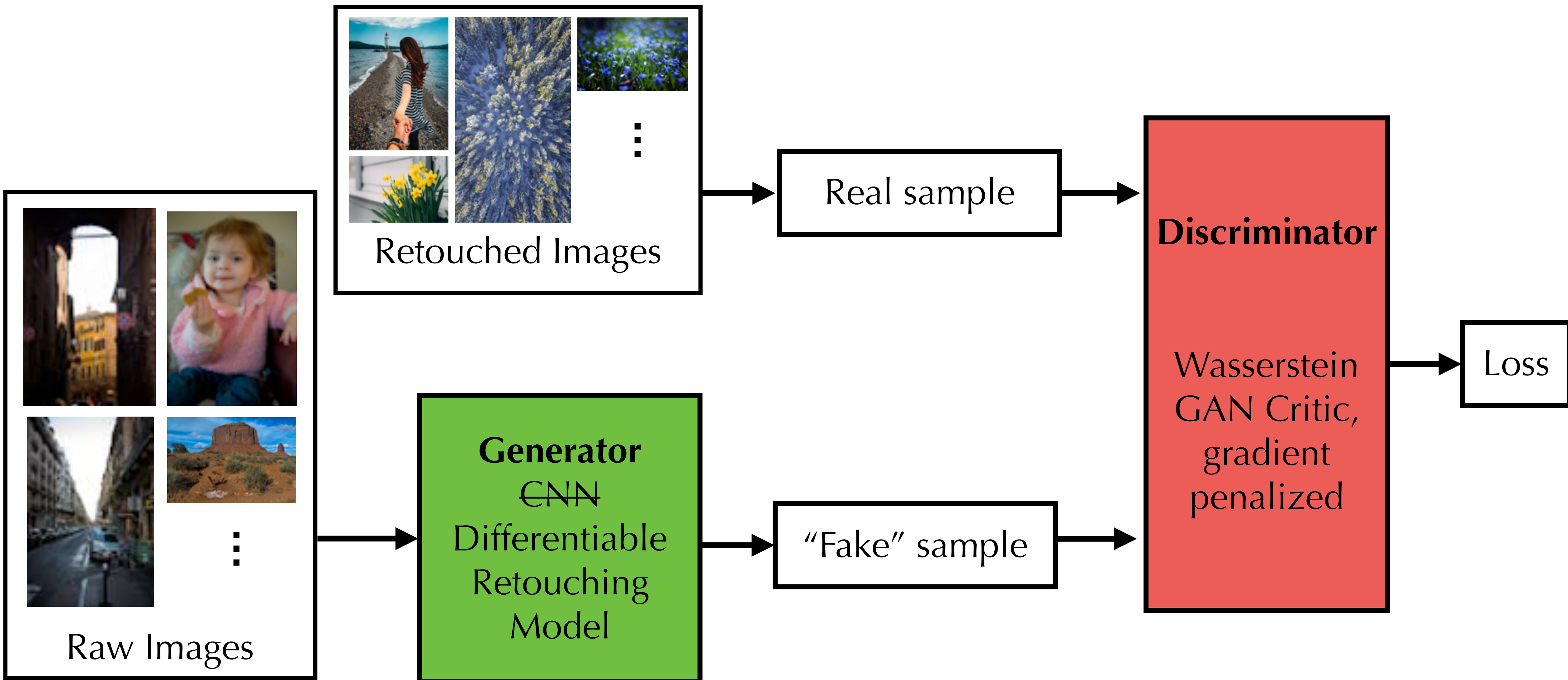
Loss

Loss

Input

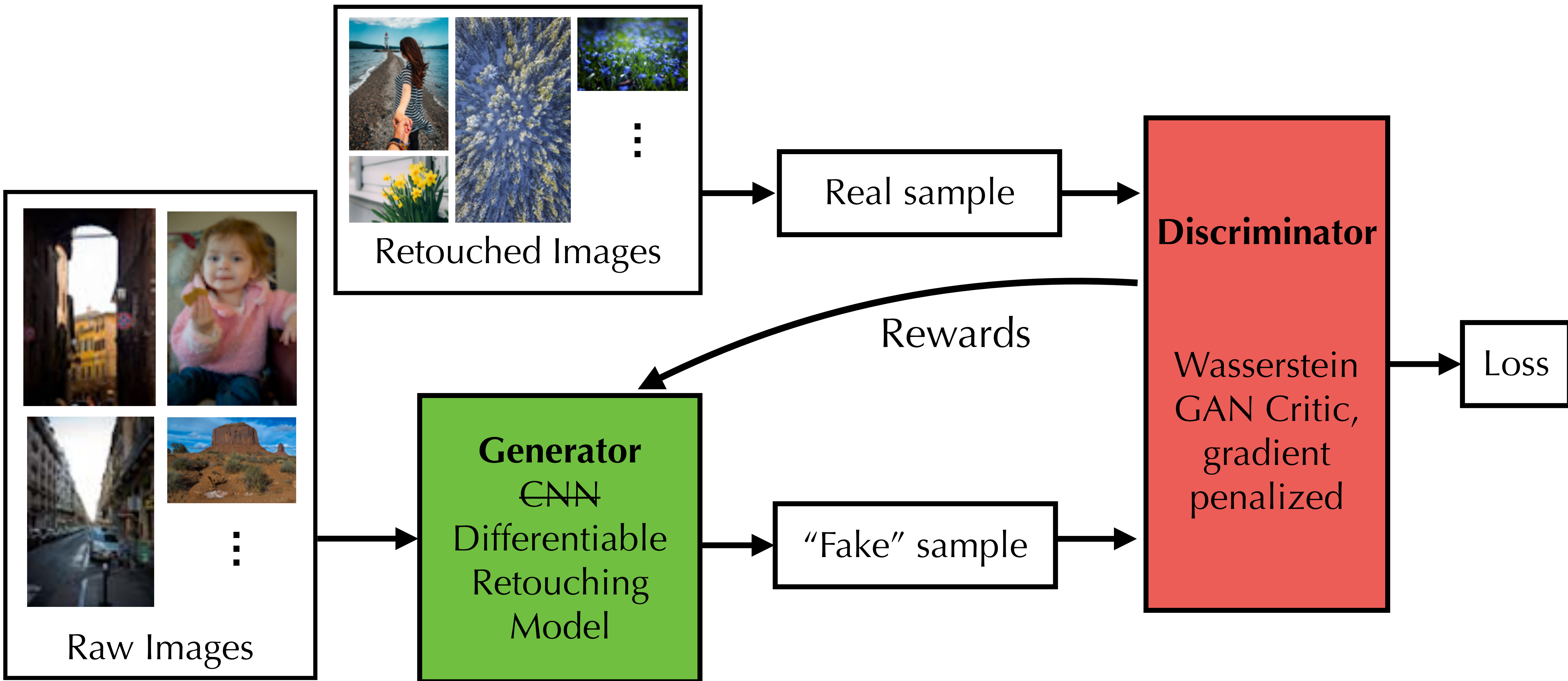


# Reward: Earth Mover's Distance





# Reward: Earth Mover's Distance



# Our Approach

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

**Deep Reinforcement  
Learning**

Learn image **operations**,  
instead of **pixels**

**Generative Adversarial  
Networks**

**Reward function**  
Training without **pairs**

Modelling

Optimization

# Our Approach

**Differentiable Photo  
Postprocessing Model**

resolution independent  
content preserving  
human-understandable

**Deep Reinforcement  
Learning**

Learn image **operations**,  
instead of **pixels**

**Generative Adversarial  
Networks**

**Reward function**  
Training without **pairs**

Modelling

Optimization

# Results: Retouching and Stylisation



# Results: Retouching and Stylisation



# Results: Retouching and Stylisation



# Results: Retouching and Stylisation



# Results: Retouching and Stylisation





# Results: Retouching and Stylisation

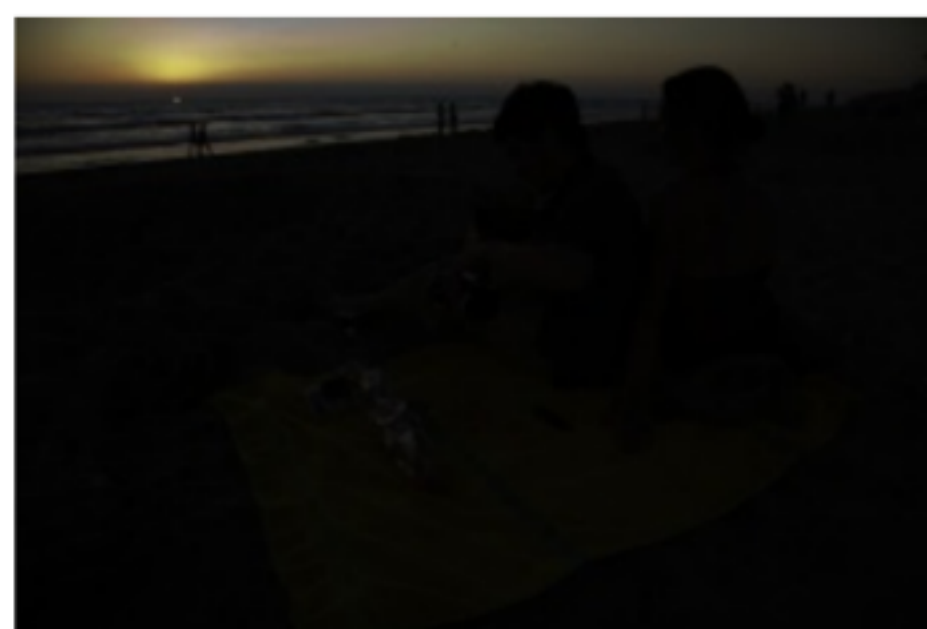
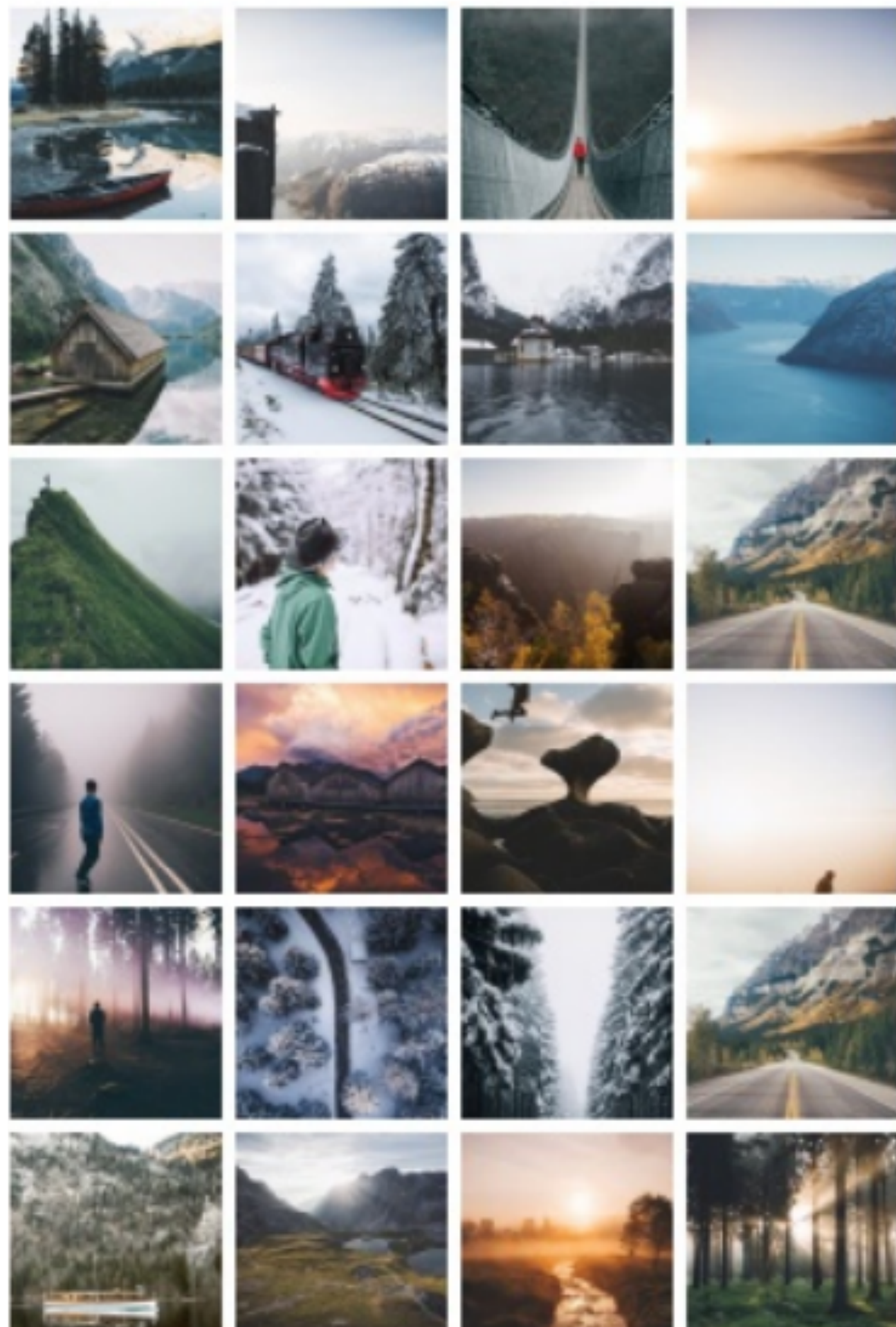


# Results: Retouching and Stylisation



# Results: Stylization

## 500px Artist A



Expo.	Tone
Gam. <input checked="" type="checkbox"/>	Cst.
W.B.	BW
Satu.	Color

Gamma 1/1.62



Expo. <input checked="" type="checkbox"/>	Tone
Gam.	Cst.
W.B.	BW
Satu.	Color

Exposure +1.58



Expo.	Tone
Gam.	Cst. <input checked="" type="checkbox"/>
W.B.	BW
Satu.	Color <input checked="" type="checkbox"/>

Contrast +0.57



Expo.	Tone
Gam.	Cst. <input checked="" type="checkbox"/>
W.B.	BW
Satu.	Color <input checked="" type="checkbox"/>

Contrast +0.57



Expo.	Tone <input checked="" type="checkbox"/>
Gam.	Cst.
W.B.	BW
Satu.	Color

Tone



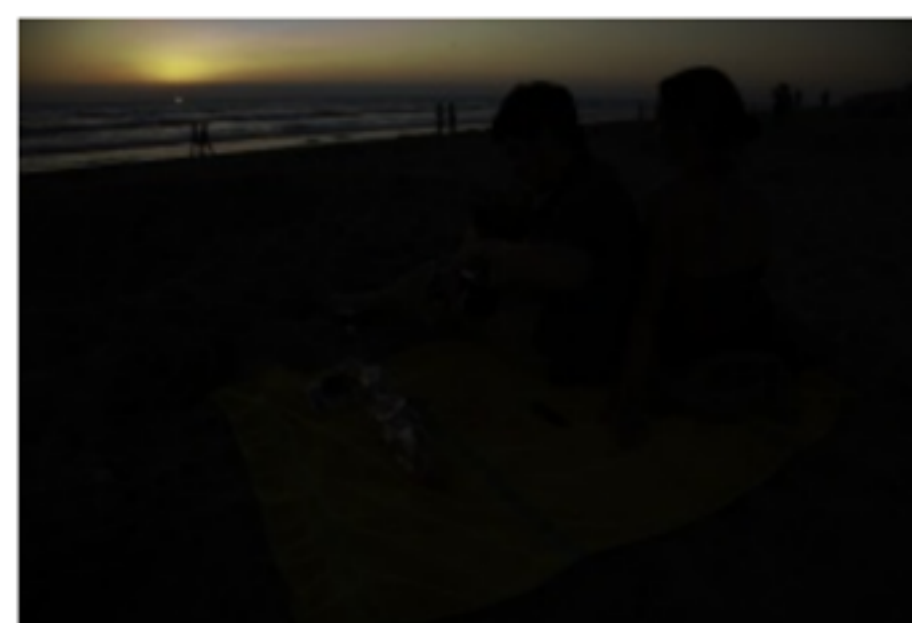
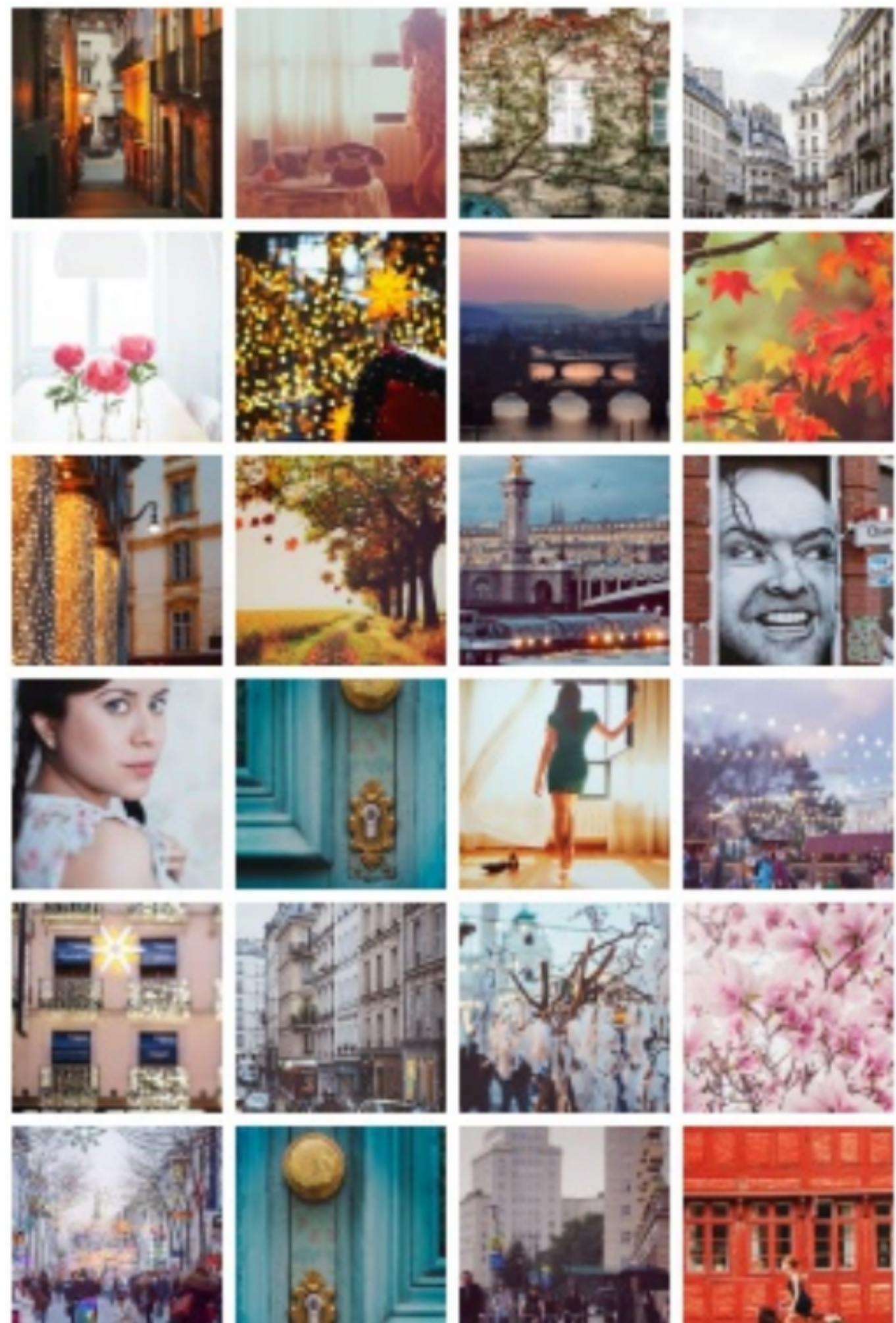
Expo.	Tone
Gam.	Cst.
W.B.	BW
Satu. <input checked="" type="checkbox"/>	Color <input checked="" type="checkbox"/>

Color



# Results: Stylization

500px Artist B



Expo.	Tone
Gam.	Cst.
W.B.	BW
Satu.	Color

Gamma 1/2.02



Expo.	Tone
Gam.	Cst.
W.B.	BW
Satu.	Color

Exposure +1.30



Expo.	Tone
Gam.	Cst.
W.B.	BW
Satu.	Color

Tone



Expo.	Tone
Gam.	Cst.
W.B.	BW
Satu.	Color

Tone



Expo.	Tone
Gam.	Cst.
W.B.	BW
Satu.	Color

Color



Expo.	Tone
Gam.	Cst.
W.B.	BW
Satu.	Color

Saturation +1.00



An  
“Infinite-  
Resolution”  
GAN



CycleGAN



Ours

An  
“Infinite-  
Resolution”  
GAN



CycleGAN



Ours

# An “Infinite- Resolution” GAN



**Pix2pix (paired data needed)**



**Ours (unpaired training)**

# An “Infinite- Resolution” GAN



**Pix2pix (paired data needed)**



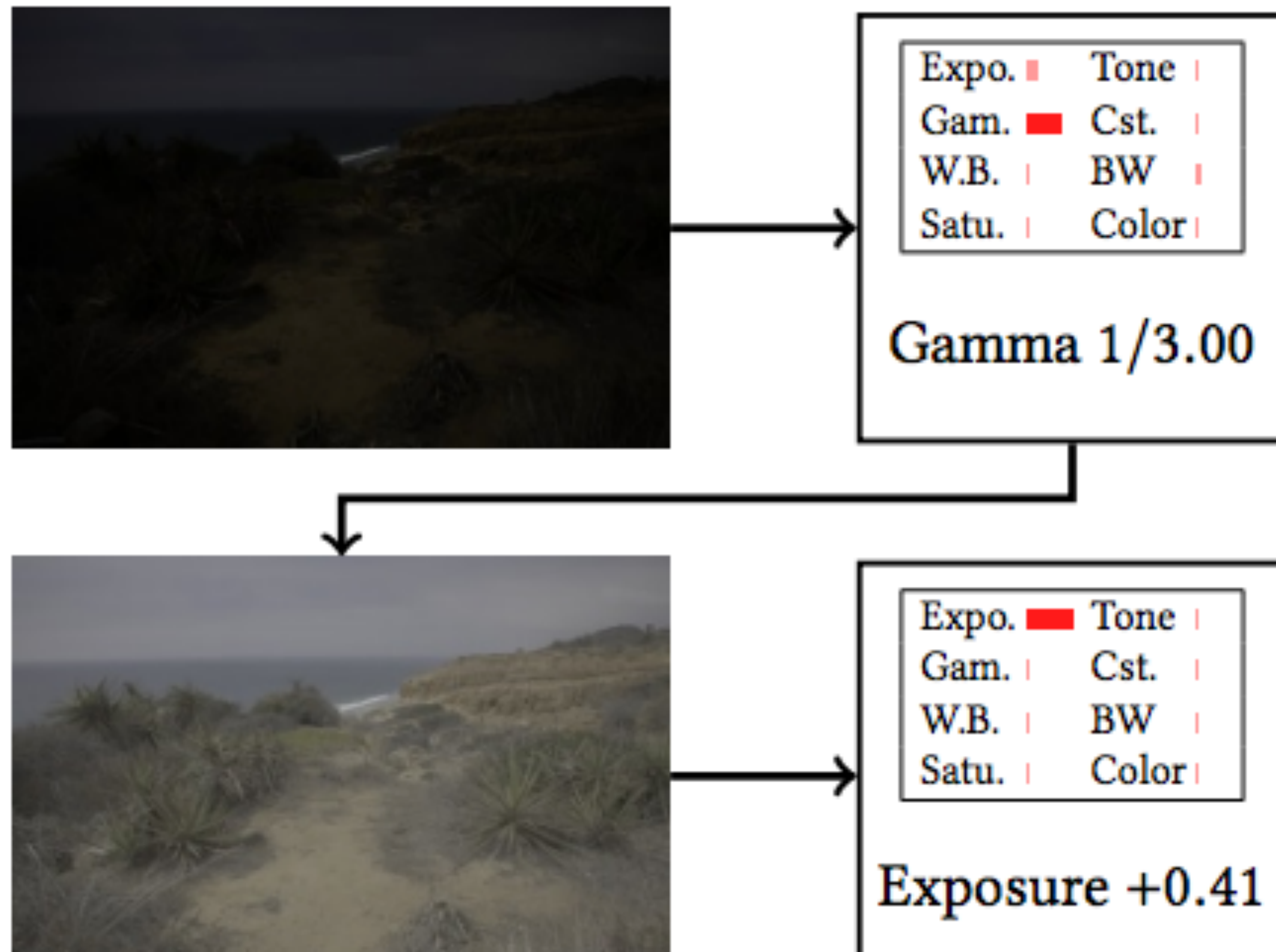
**Ours (unpaired training)**



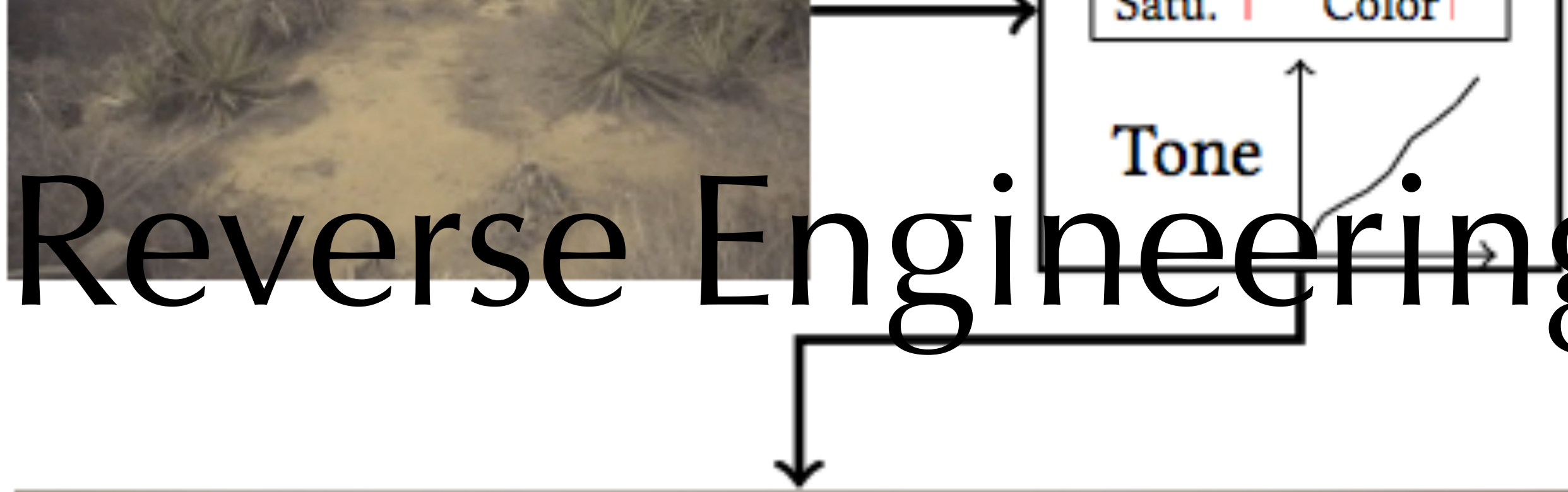
# Results: Reverse Engineering



# Results: Reverse Engineering



# Results: Reverse Engineering





Expo.  Tone   
Gam.  Cst.   
W.B.  BW   
Satu.  Color

Gamma 1/3.00



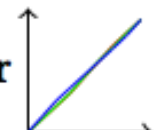
Expo.  Tone   
Gam.  Cst.   
W.B.  BW   
Satu.  Color

Exposure +0.41




Expo.  Tone   
Gam.  Cst.   
W.B.  BW   
Satu.  Color

Color




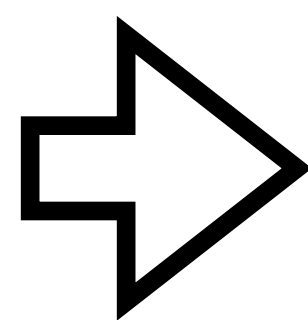
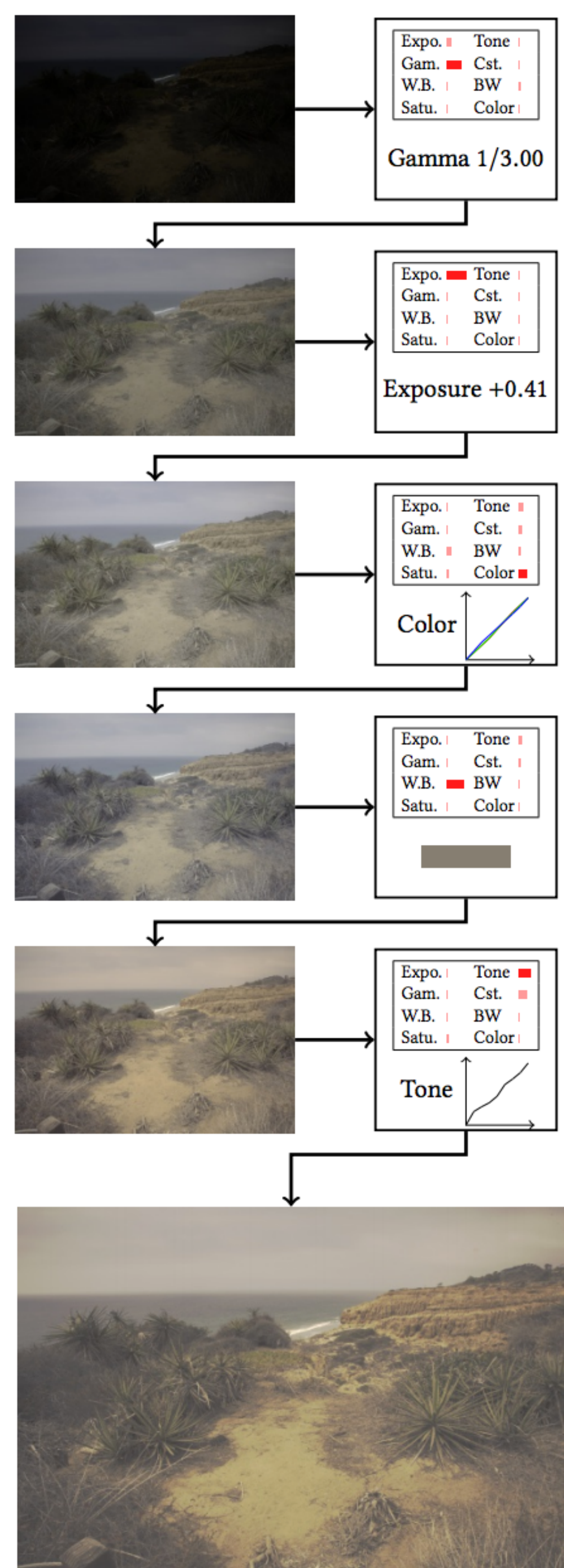
Expo.  Tone   
Gam.  Cst.   
W.B.  BW   
Satu.  Color



Expo.  Tone   
Gam.  Cst.   
W.B.  BW   
Satu.  Color

Tone





```
# Step 1: Gamma
image = image ** (1 / 3.0)
# Step 2: Exposure
image = image / image.mean() * 0.6
# Step 3: Boost blue shadow
blue_shadow = image[:, :, 2] < 0.5
blue = image[:, :, 2]
blue = blue_shadow * (blue * 2) ** 0.7 / 2 + blue * (1 - blue_shadow)
image[:, :, 2] = blue
# Step 4: White balance
image = image * np.array((1.055, 0.984, 0.886)).reshape((1, 1, 3))
# Step 5: Boost shadow
shadow = image < 0.33
image = ((image * shadow * 3) ** 0.8 / 3) + image * (1 - shadow)
```

Code based on the learned trajectory



Images generated by the code



Images generated by the black-box filter

# Summary

## Differentiable Photo Postprocessing Model

resolution independent  
content preserving  
human-understandable

## Deep Reinforcement Learning

Learn image **operations**,  
instead of **pixels**

## Generative Adversarial Networks

Training without **pairs**

**Advantages: infinite resolution, human-understandable (reverse engineering artistic styles), unpaired training**

# Summary

## Differentiable Photo Postprocessing Model

resolution independent  
content preserving  
human-understandable

## Deep Reinforcement Learning

Learn image **operations**,  
instead of **pixels**

## Generative Adversarial Networks

Training without **pairs**

**Advantages: infinite resolution, human-understandable (reverse engineering artistic styles), unpaired training**

**Limitations: RL/GAN stability, hyper-parameters, faces**

# Retouch your photos with **Exposure!**

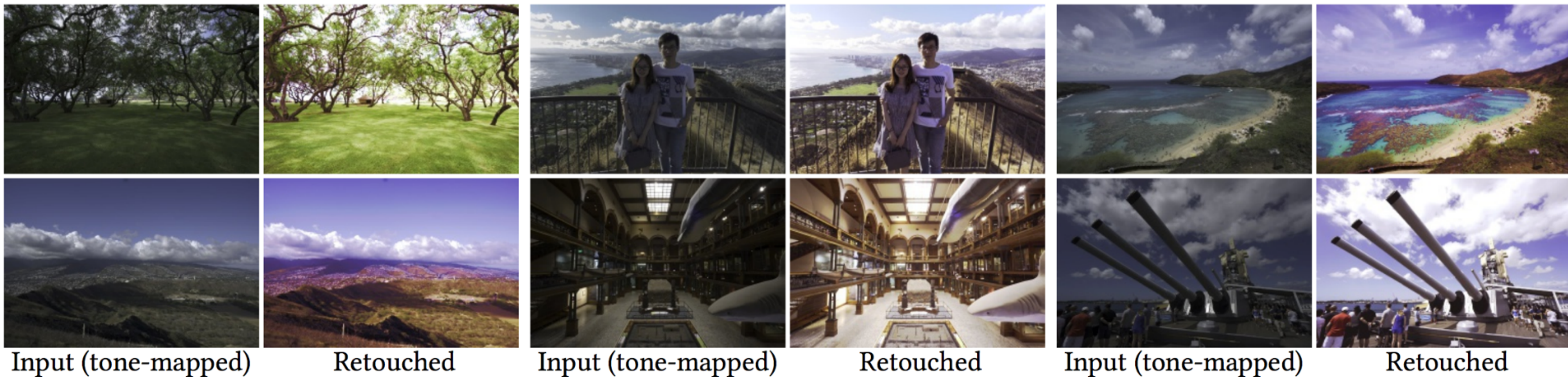


Reproducible research:

<https://github.com/yuanming-hu/exposure>



# Retouch your photos with **Exposure!**



Reproducible research:

<https://github.com/yuanming-hu/exposure>

Thank you!

Questions are welcome!