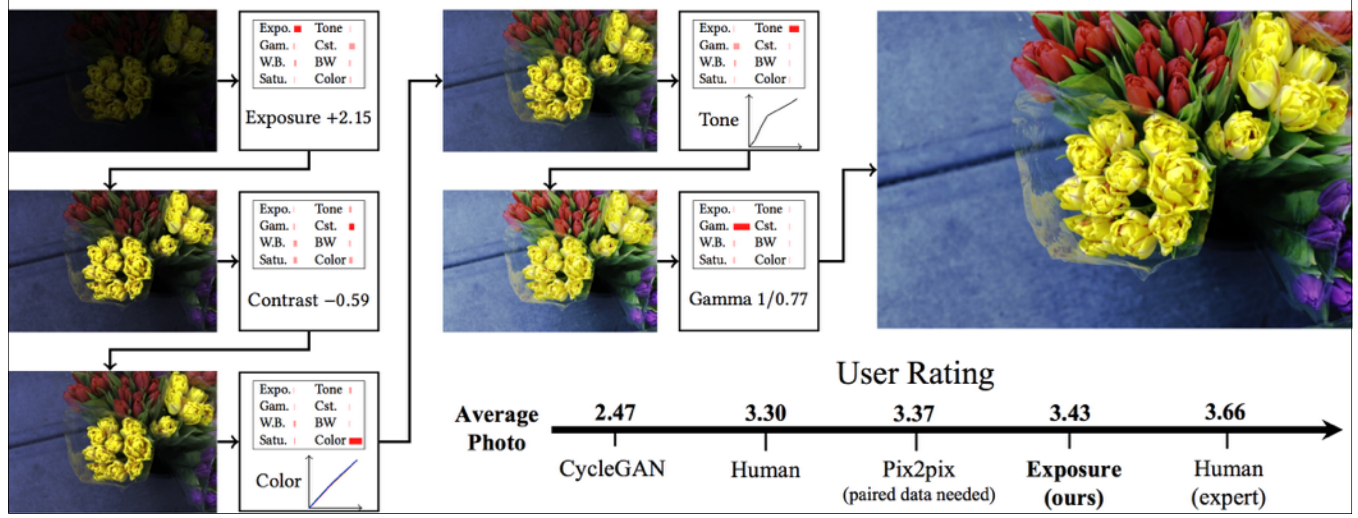
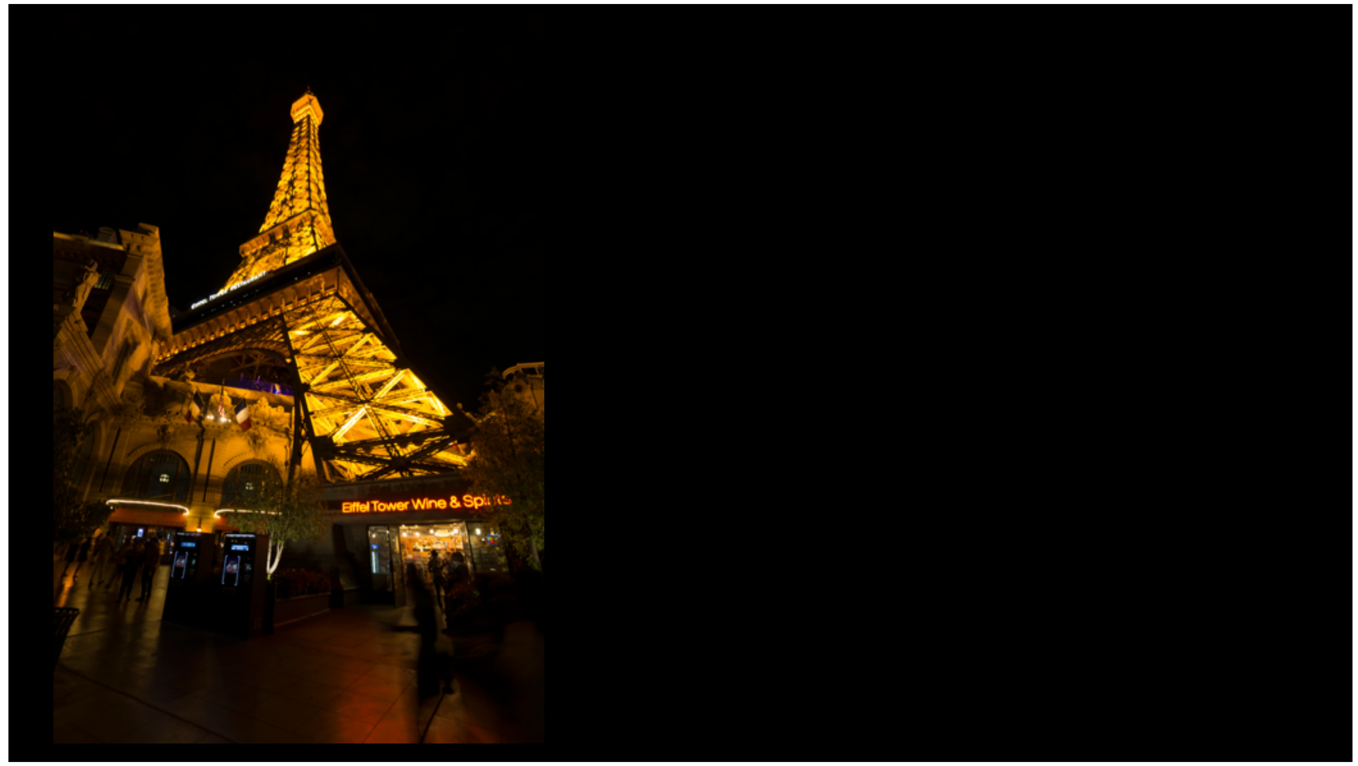
Exposure: A White-Box Photo Post-Processing Framework

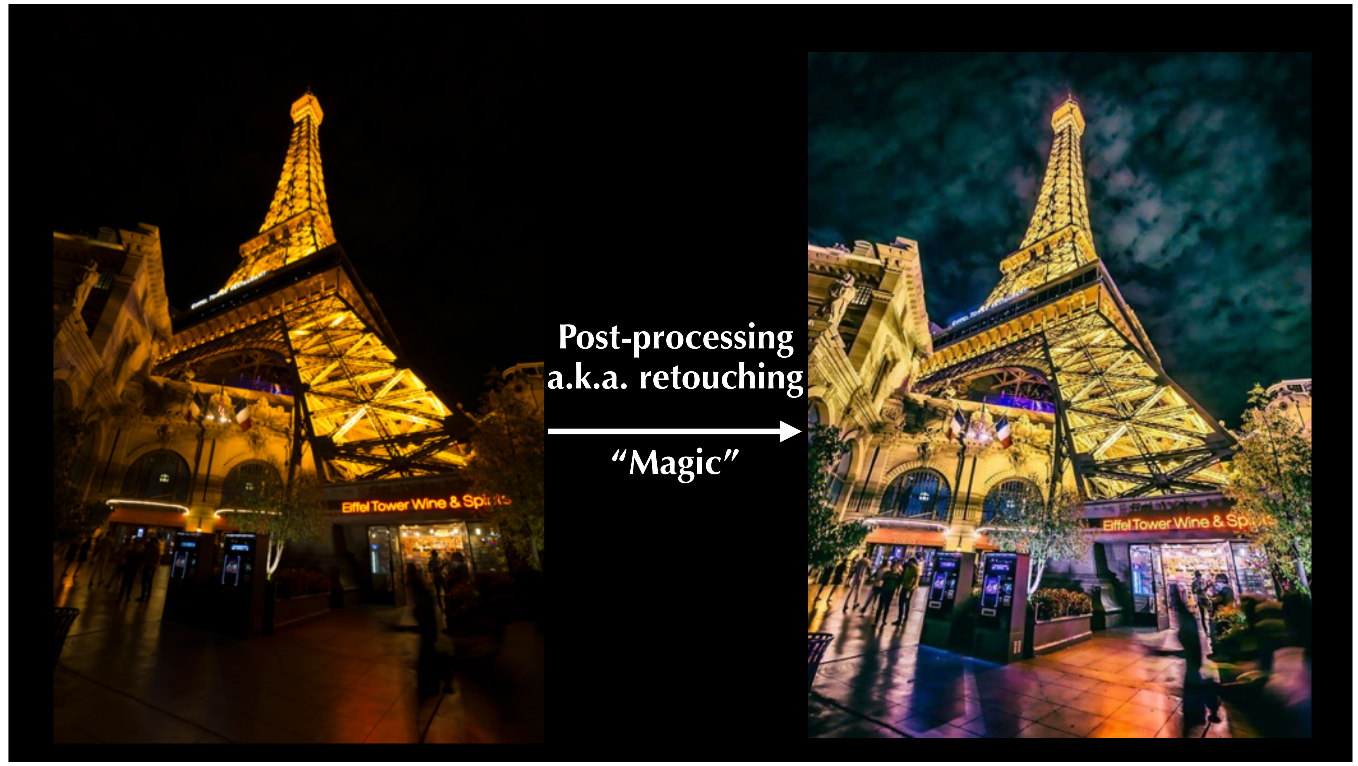
Yuanming Hu[1,2] Hao He[1,2] Chenxi Xu[1,3] Baoyuan Wang[1] Stephen Lin[1]
[1]Microsoft Research [2]MIT CSAIL [3]Peking University



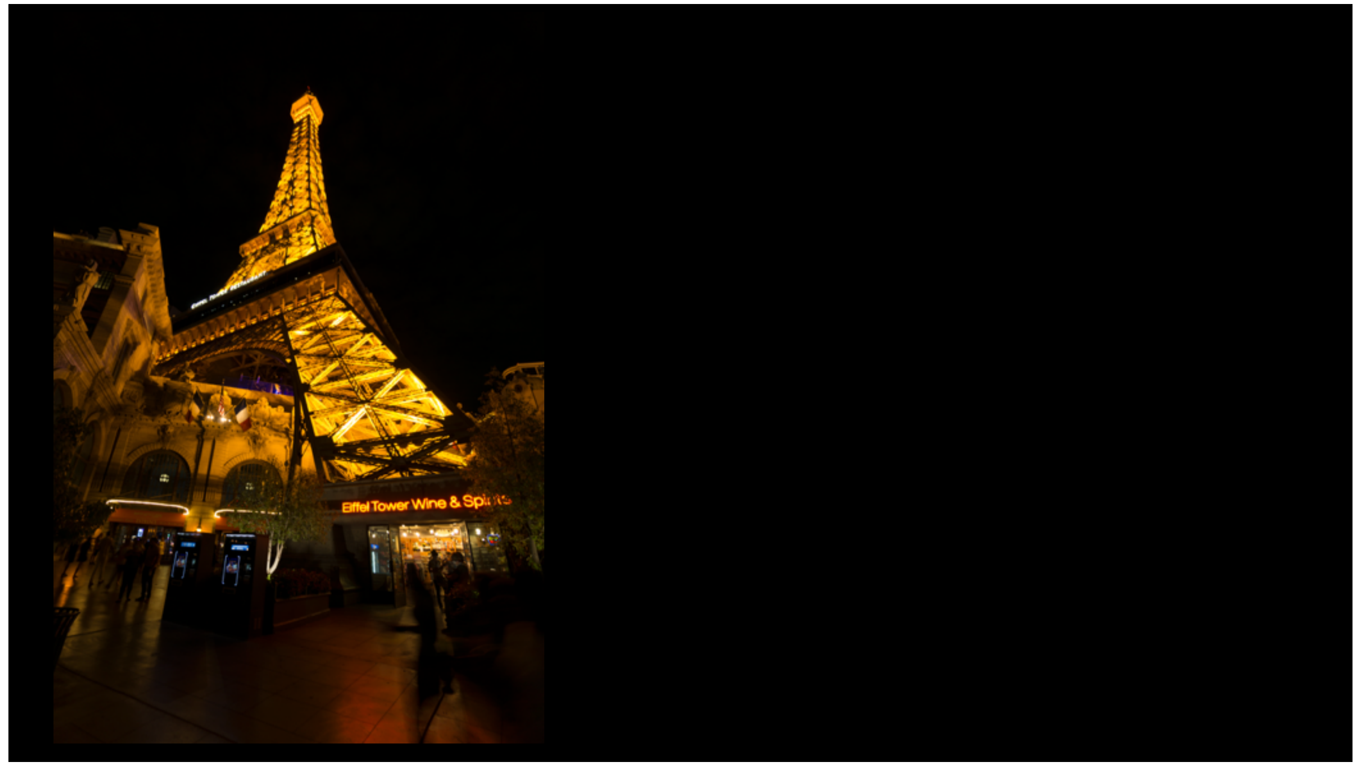
Thank you Jean for the introduction. Good morning everyone. I’m Yuanming from MIT.



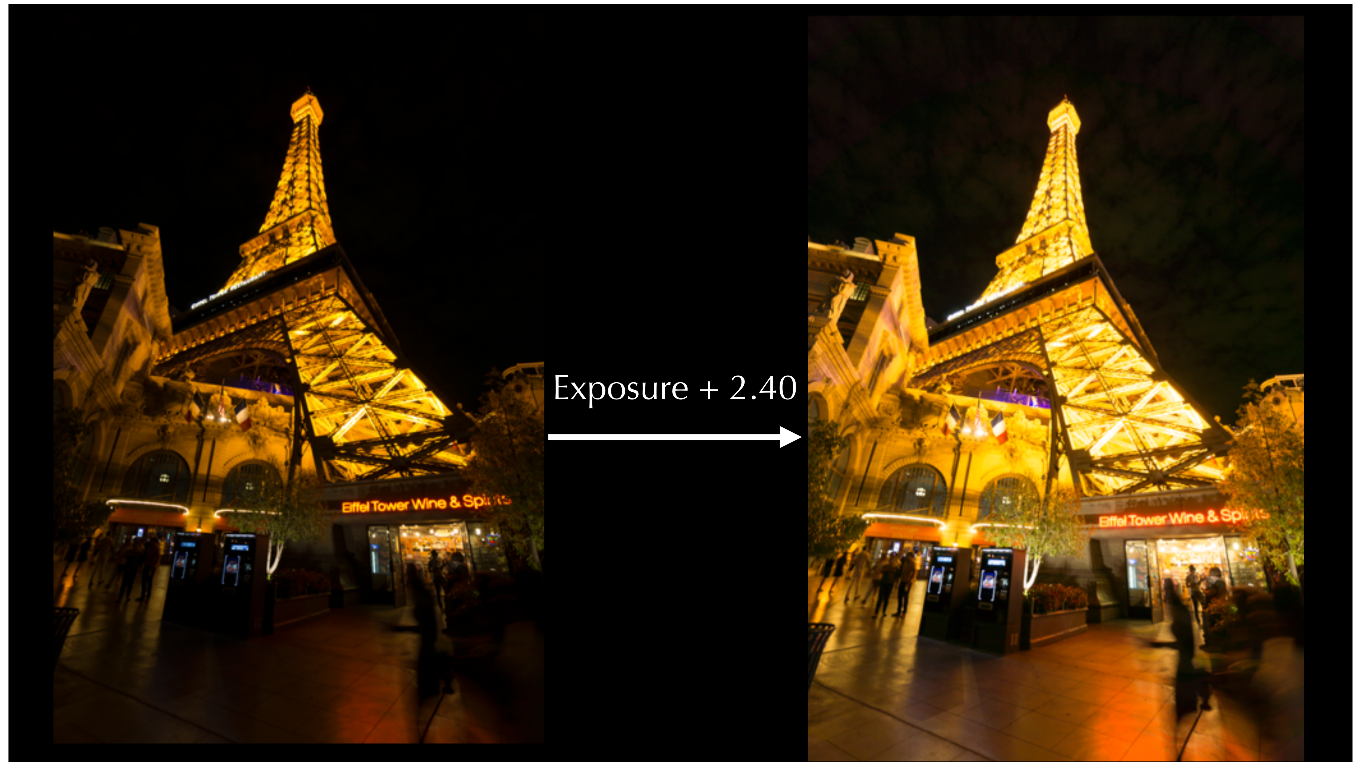
As we all know, post processing or retouching can significantly elevate the visual appeal of photos, [click] but many people lack the expertise to do this well. As a result, photo retouching is typically considered as a magic.



As we all know, post processing or retouching can significantly elevate the visual appeal of photos, [click] but many people lack the expertise to do this well. As a result, photo retouching is typically considered as a magic.



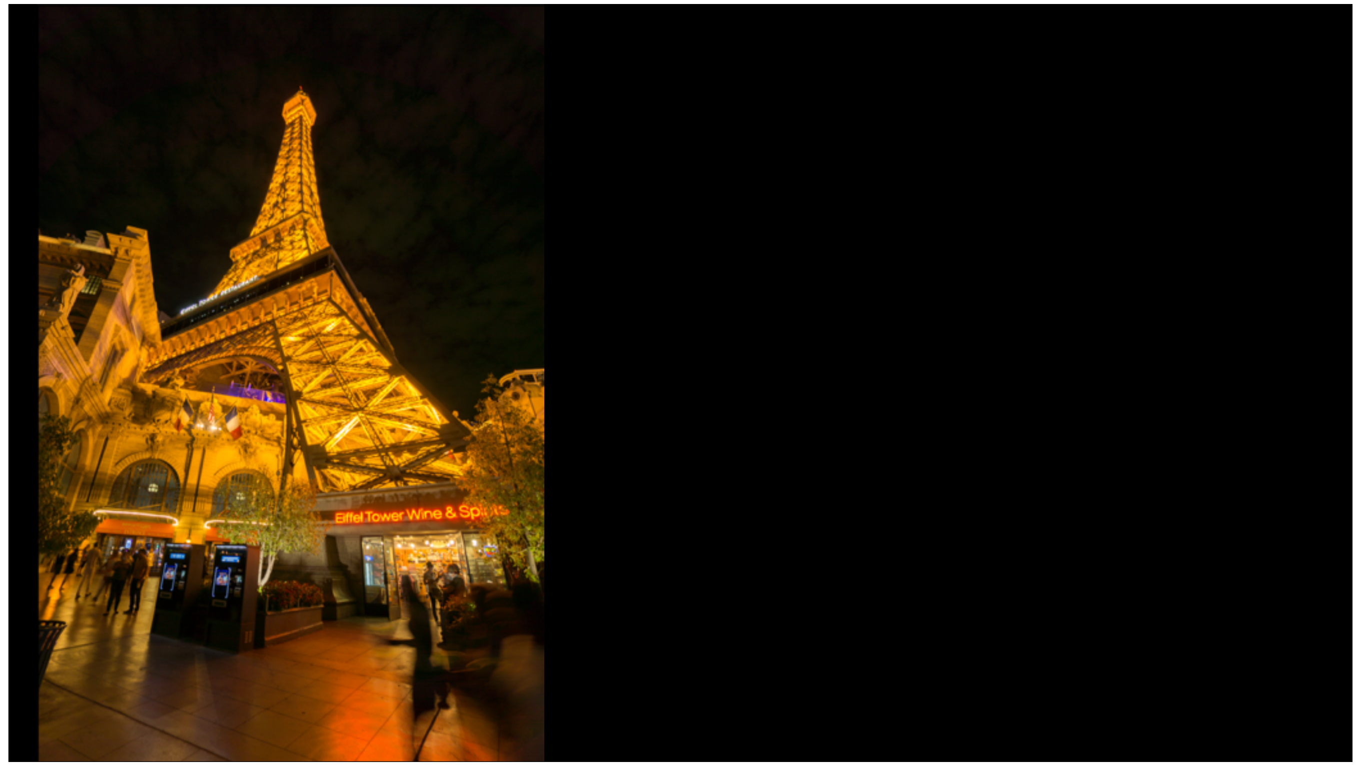
Let me show you an example. Apparently this photo looks too dark and the first thing to do is to increase the exposure. [click] Now it looks much better.



Let me show you an example. Apparently this photo looks too dark and the first thing to do is to increase the exposure. [click] Now it looks much better.

However, the tower is too bright now and you can barely see the nice structure details. [click] That’s why you may want to lower the highlight.

However, the tower is too bright now and you can barely see the nice structure details. [click] That’s why you may want to lower the highlight.

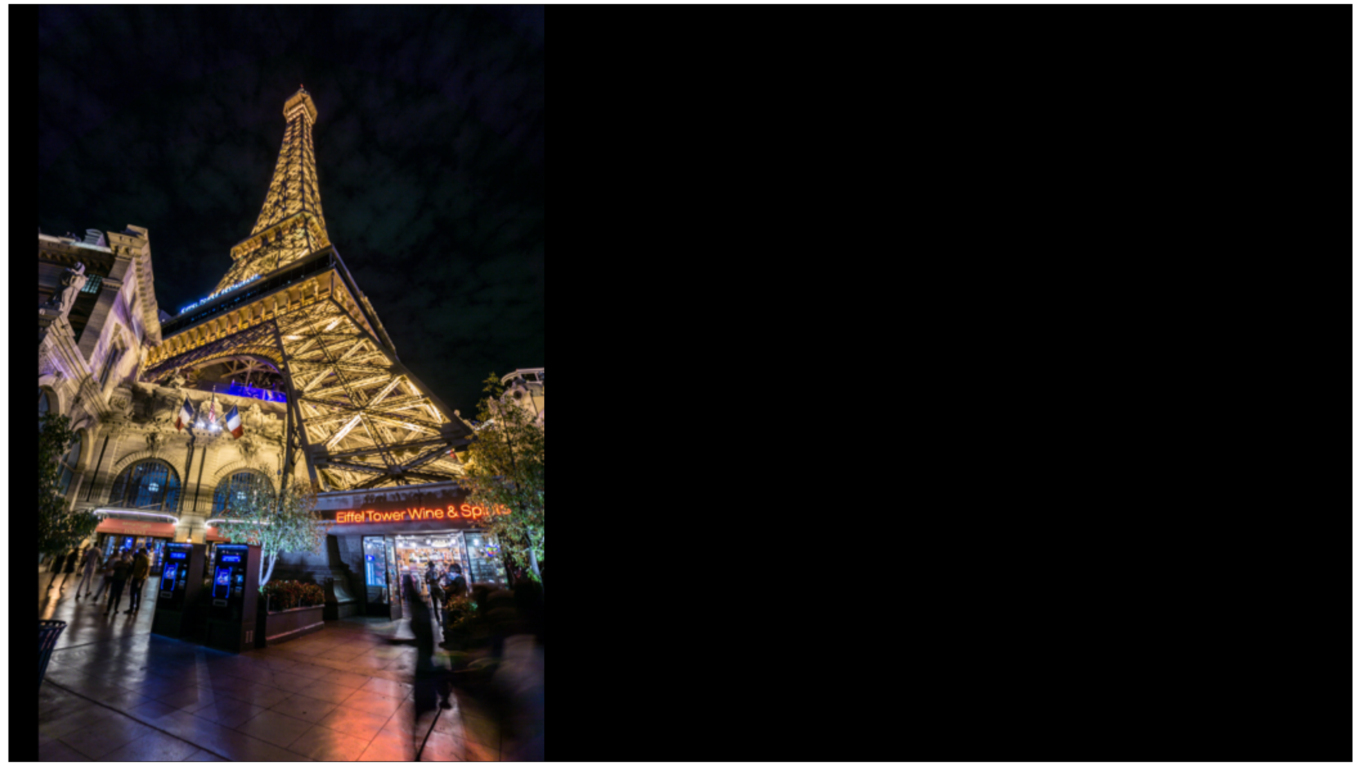


Now the photo appears a bit yellowish. Let’s do a white-balance step to correct the color. [click]
By removing the color cast, everything becomes more colorful.

Now the photo appears a bit yellowish. Let’s do a white-balance step to correct the color. [click]
By removing the color cast, everything becomes more colorful.

We further enhance the structural details [click] by tuning up clarity.

We further enhance the structural details [click] by tuning up clarity.



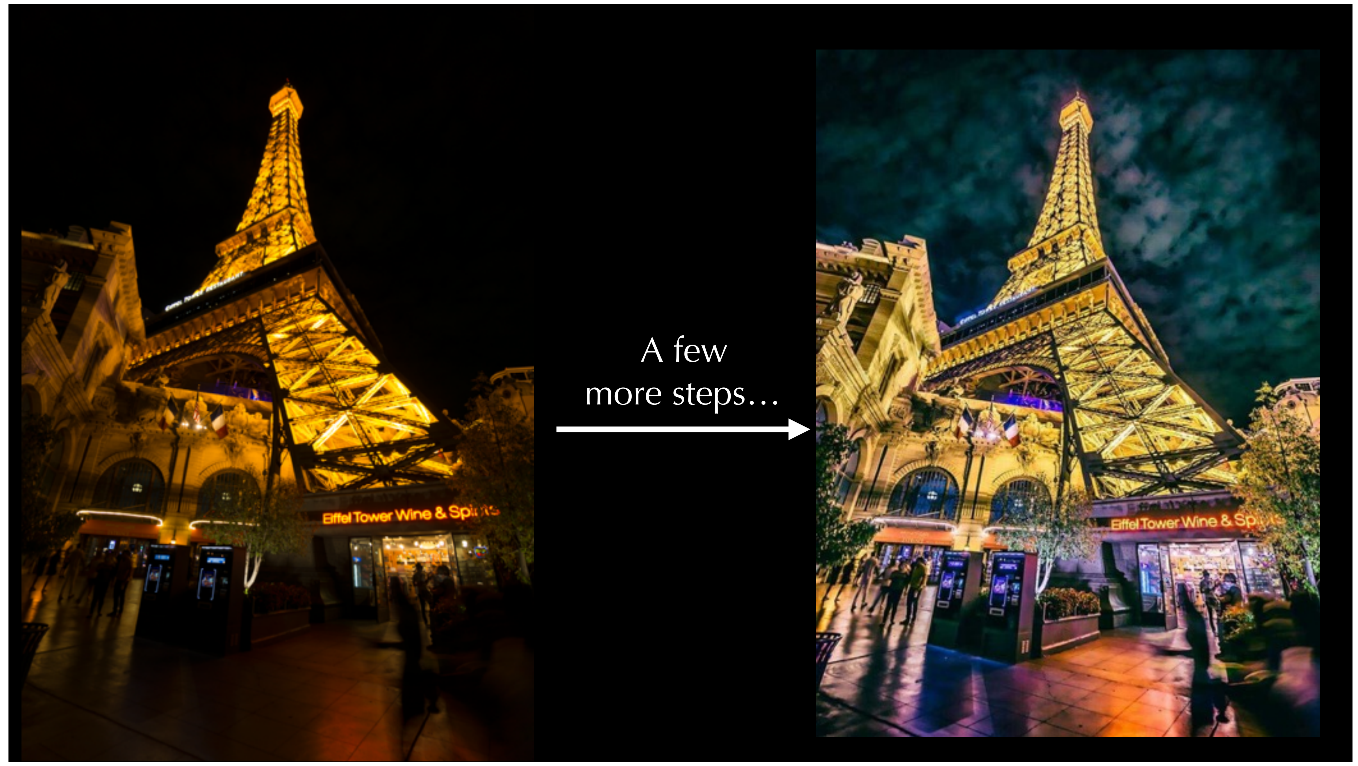
and make it a bit more saturated.[click]

and make it a bit more saturated.[click]

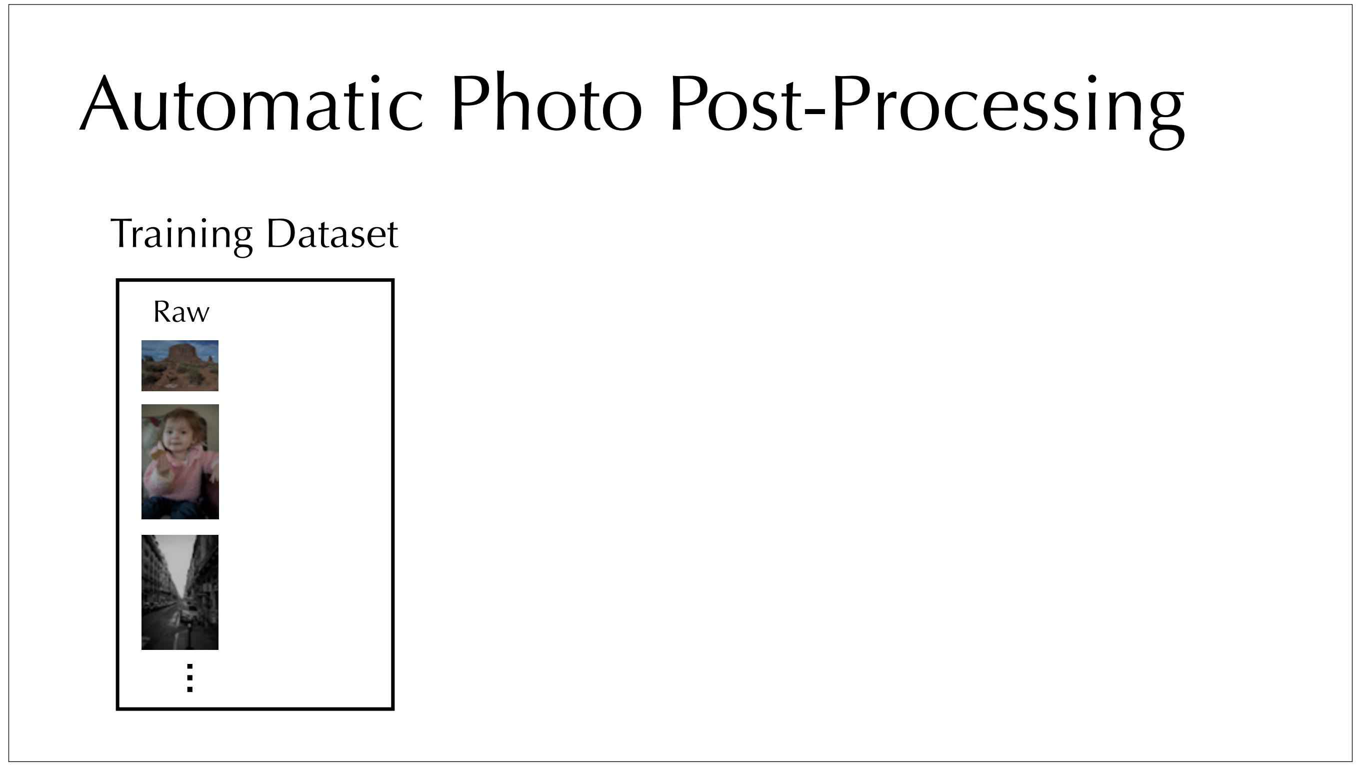
When I shot the photo, it was actually a cloudy day. To recover the lost clouds[click], I boost the shadow parts.

When I shot the photo, it was actually a cloudy day. To recover the lost clouds[click], I boost the shadow parts.

Now we are almost there. More operations can be done to make it even better.
In comparison, [click] this is what we started with. [stop for a while]



Now we are almost there. More operations can be done to make it even better.
In comparison, [click] this is what we started with. [stop for a while]

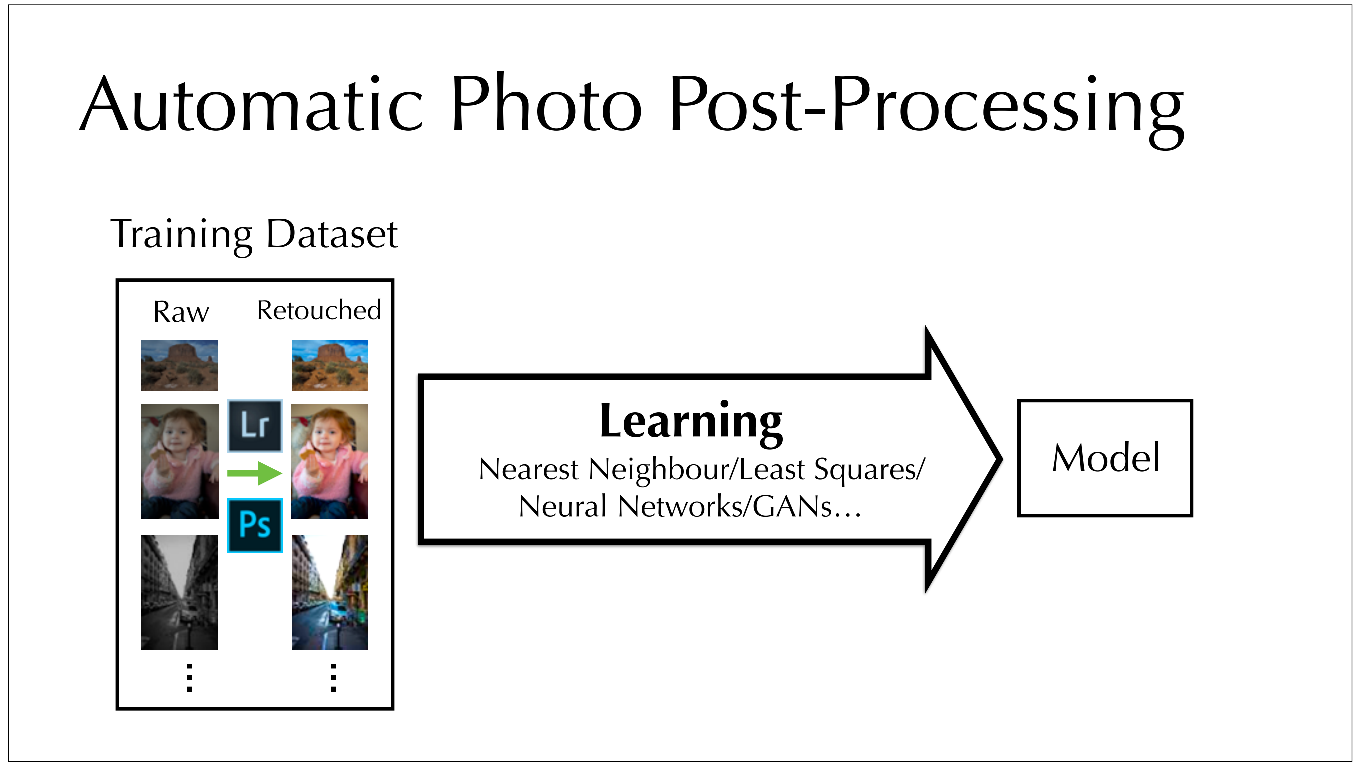


It’s clear that post-processing does require a bit expertise. We want machines to automate this, so that more people can enjoy digital photography. Mathematical rules do exists but in this talk I will focus on data-driven approaches. In general these approaches work in this way:
[click] Firstly, we need to collect a training dataset consisting of raw and retouched images,
[click] Then, using some machine learning techinques, we can train a model based on the dataset.
[click] Finally we can use the model to automatically retouch a photo.

Automatic Photo Post-Processing

Training Dataset

It’s clear that post-processing does require a bit expertise. We want machines to automate this, so that more people can enjoy digital photography. Mathematical rules do exists but in this talk I will focus on data-driven approaches. In general these approaches work in this way:
[click] Firstly, we need to collect a training dataset consisting of raw and retouched images,
[click] Then, using some machine learning techinques, we can train a model based on the dataset.
[click] Finally we can use the model to automatically retouch a photo.

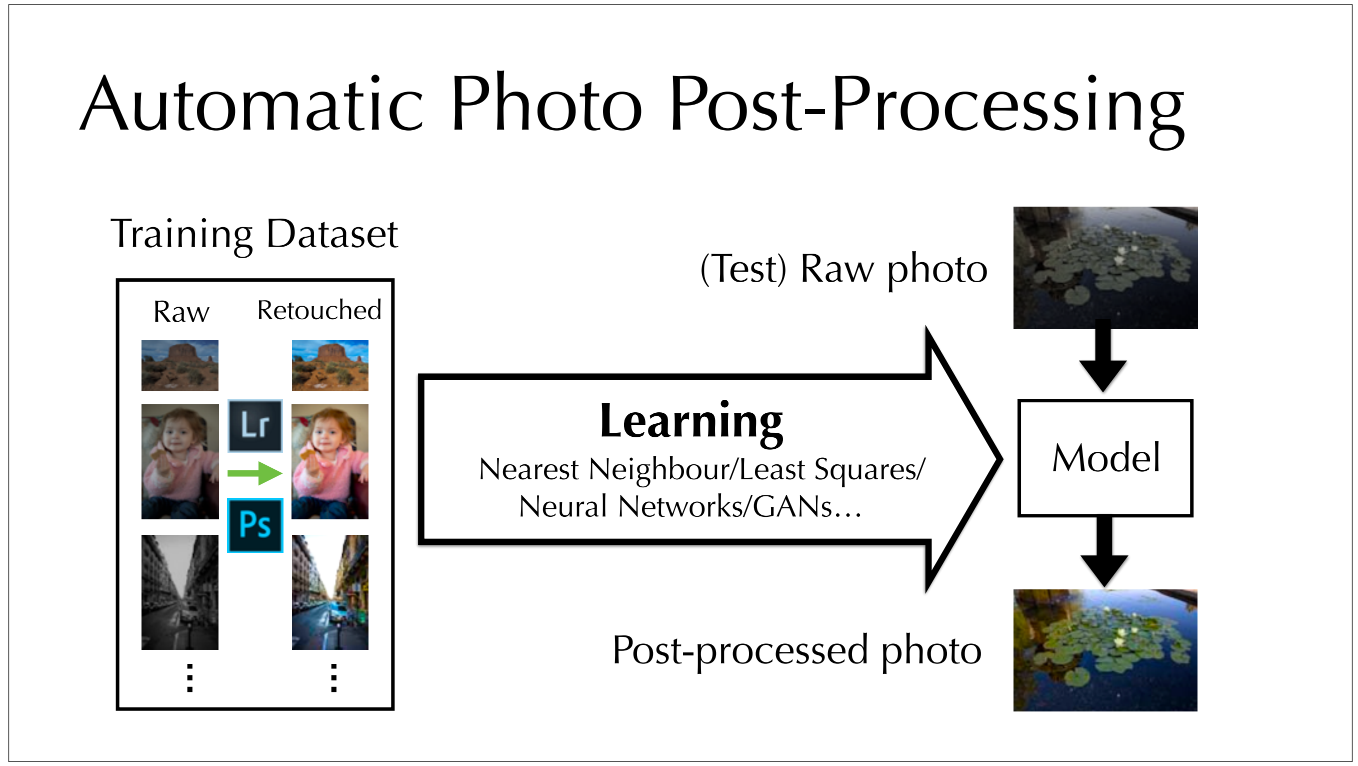


It’s clear that post-processing does require a bit expertise. We want machines to automate this, so that more people can enjoy digital photography. Mathematical rules do exists but in this talk I will focus on data-driven approaches. In general these approaches work in this way:

[click] Firstly, we need to collect a training dataset consisting of raw and retouched images,

[click] Then, using some machine learning techinques, we can train a model based on the dataset.

[click] Finally we can use the model to automatically retouch a photo.



It’s clear that post-processing does require a bit expertise. We want machines to automate this, so that more people can enjoy digital photography. Mathematical rules do exists but in this talk I will focus on data-driven approaches. In general these approaches work in this way:

[click] Firstly, we need to collect a training dataset consisting of raw and retouched images,

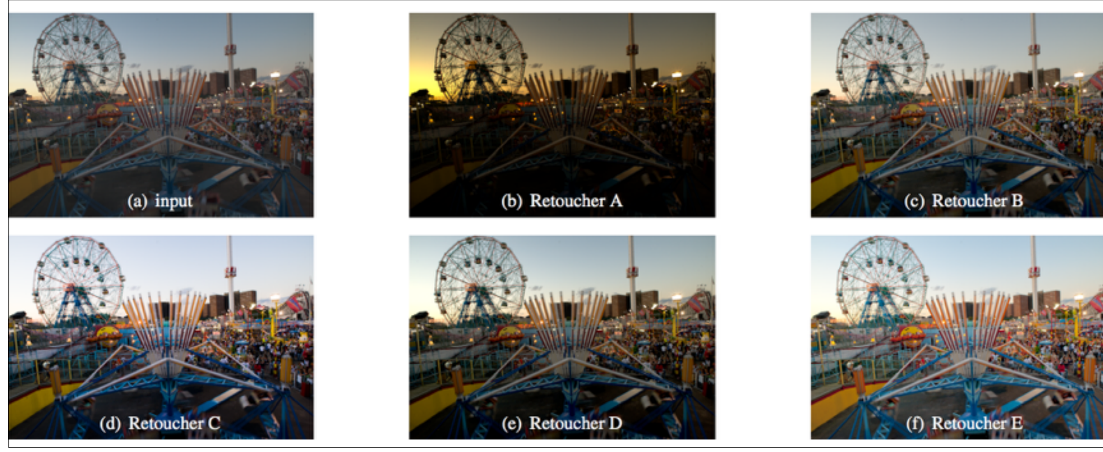
[click] Then, using some machine learning techinques, we can train a model based on the dataset.

[click] Finally we can use the model to automatically retouch a 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

There are many projects along this direction. Learning would be hard without a good dataset. The MIT-Adobe FiveK Dataset was published in 2011 (twenty eleven) by Bychkovsky et al, consisting of 5000 RAW images and the retouched versions by five artists using Adobe Lightroom. Coming along with the data is a learning-based global tonal adjustment approach.

Learning-based Photo Post-Processing

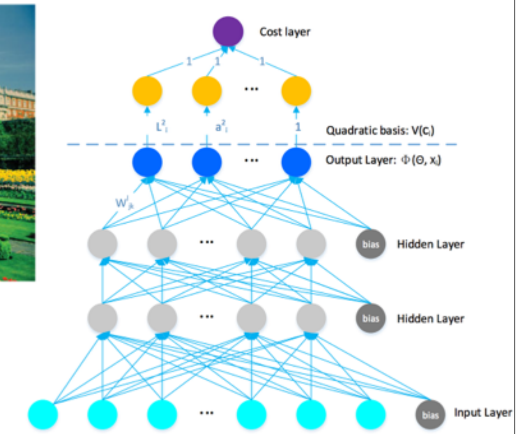
Yan et al. 2014, **Automatic Photo Adjustment Using Deep Neural Networks**

(a)

Input

(b)

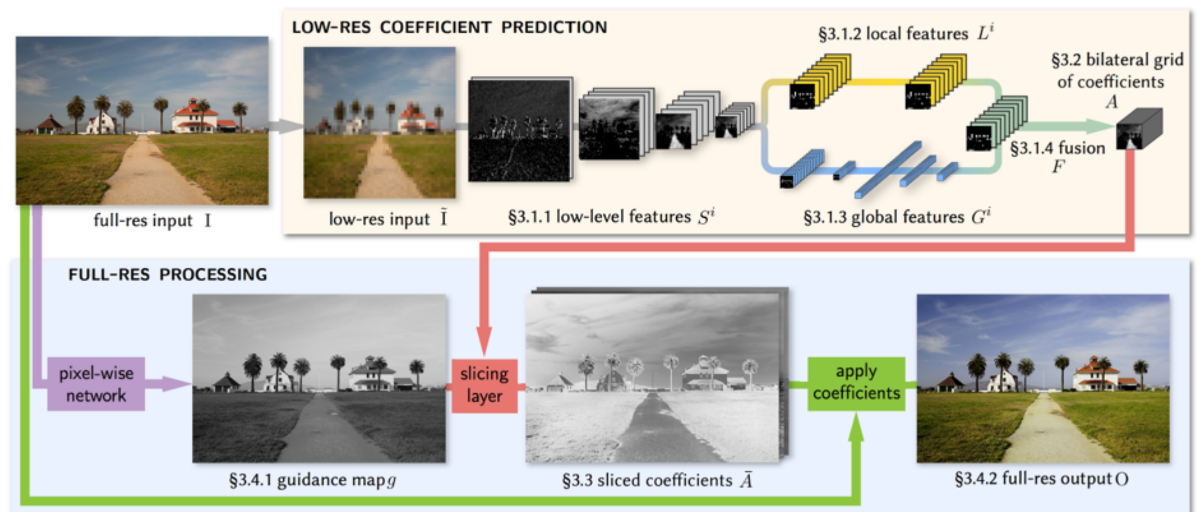
Output



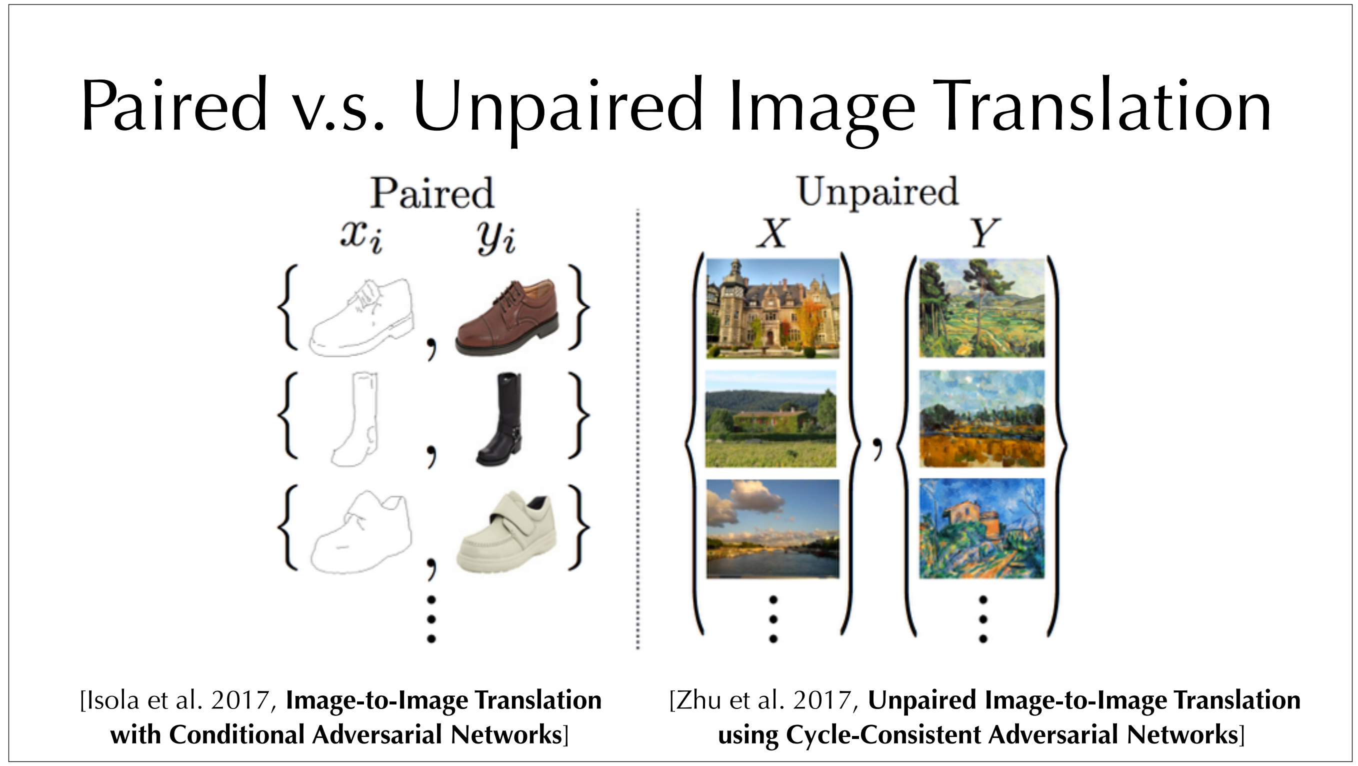
The use of deep learning in this task starts with Yan et al. in 2014 (twenty fourteen), where a deep neural network is used to predict local quadratic color transformation.

Learning-based Photo Post-Processing

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



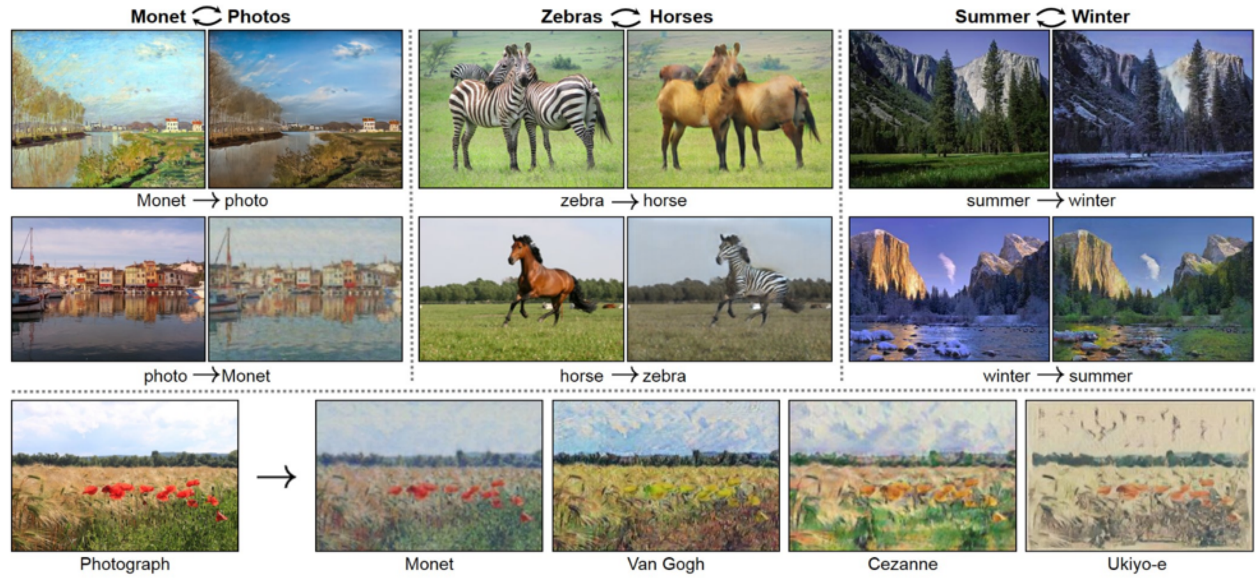
More recently, and actually concurrently to our work, Gharbi et al. used deep-learning to predict the coefficients of locally-affine transforms in bilateral space. Such prediction happens on low-res images and can be applied to high-res ones.



Another thread of work is image translation, which tackles more general problems. One attractive feature of these approaches, is that deep-learning-based image translation can work even without paired training data. This makes it possible to utilise online photo collections where the raw images are not available.

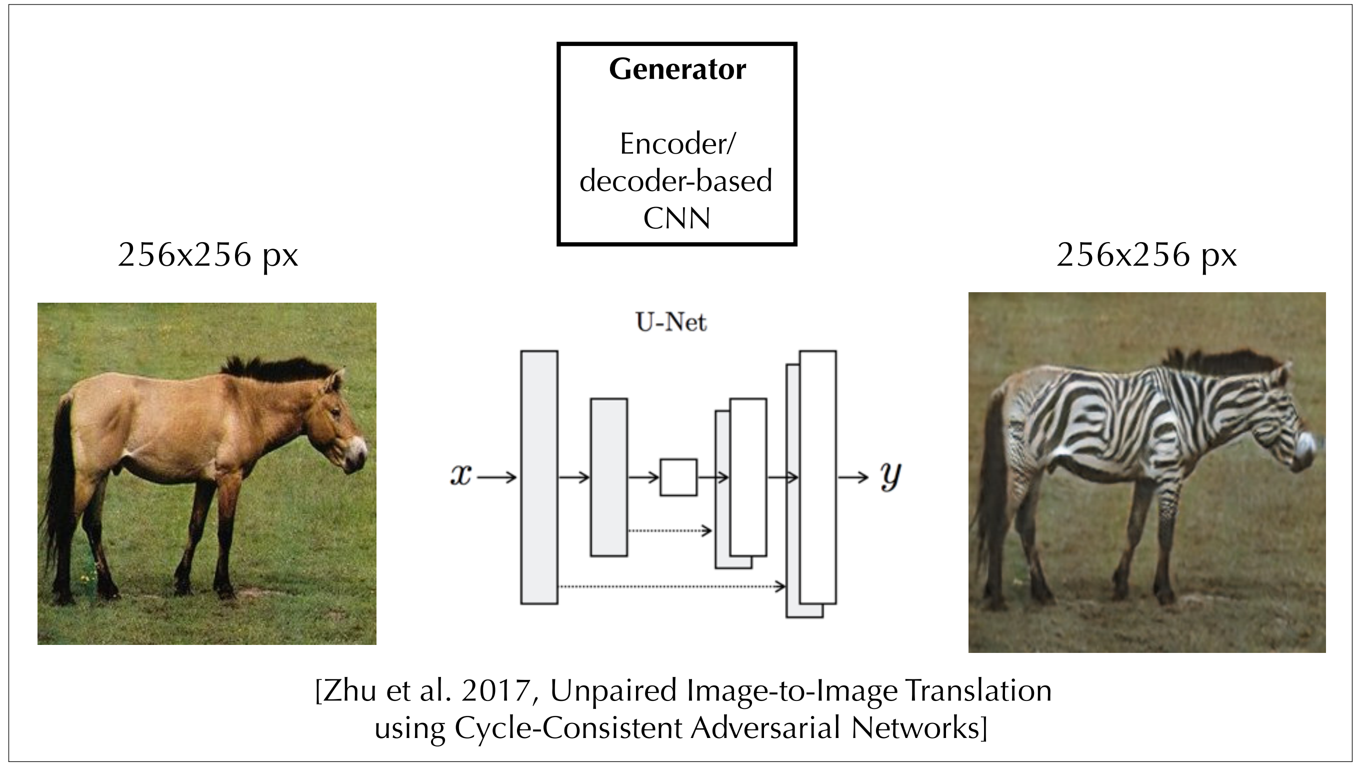
CycleGAN

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

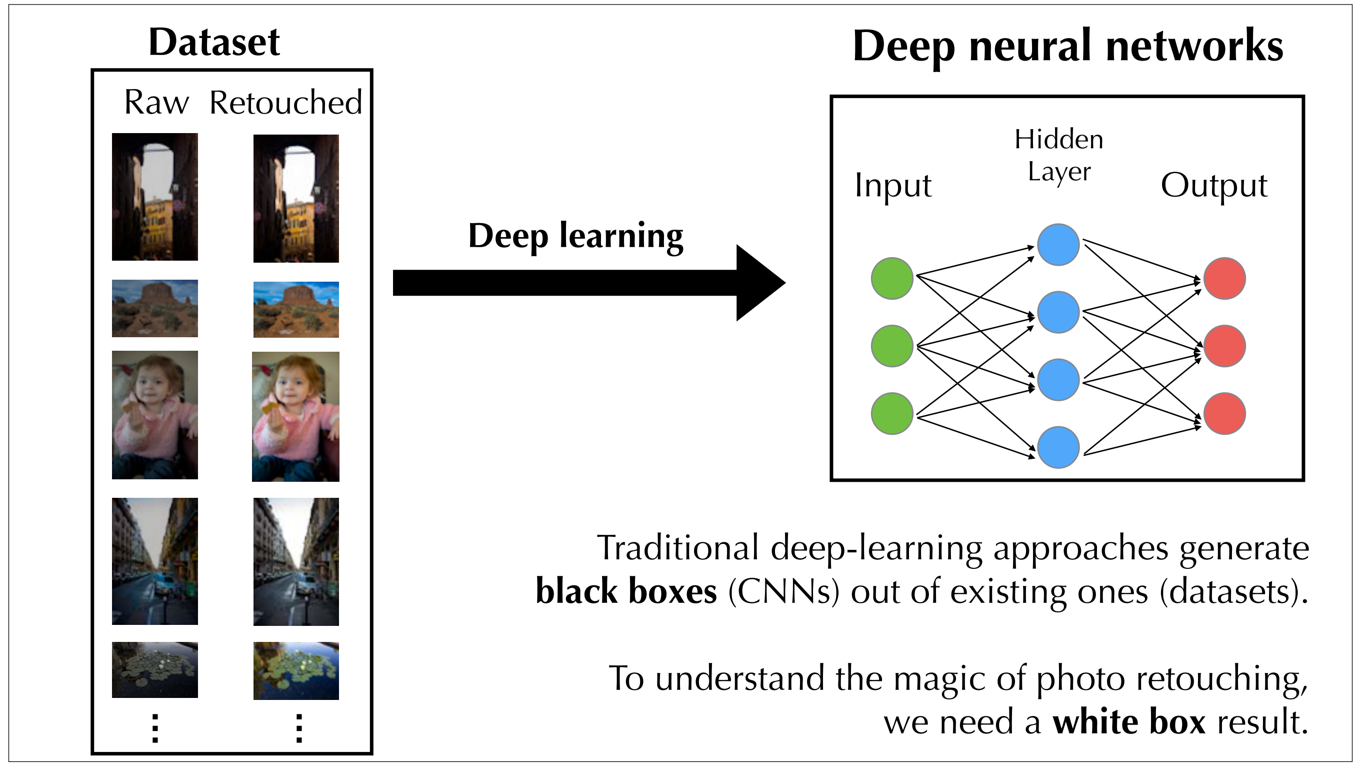


Let's take a close look at CycleGAN, which is a typical unpaired image translation approach. After learning from input/output images, the results are exciting. A painting of Monet can be translated into a photo, zebras can be translated to horses, and vice versa. You can even travel between seasons.

The word GAN here stands for generative adversarial network, which I will talk about later.



For now let's focus on the generator component of traditional GANs. Output images are formed with convolutional and transposed convolutional layers, meaning that every pixel needs to be explicitly generated. Such architecture limits resolution. In this example, the resolution is 256x256. In comparison, a professional photo can have resolution like 6k by 4k. Recently the resolution is boosted, but we want an approach to completely solve the resolution issue.



Another common drawback in previous approaches is human understandability. We start with a black-box style described by a dataset, and after deep-learning we get another black-box. Nothing is actually revealed and we cannot dig into what actually happened.

	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.			○	○

I summaries the previous approaches in this table. We need a method to generate high-resolution outputs. We are more interested in how retouching is done than in just getting the output images, so we want a human-understandable result. Since paired training data is not available in many cases, the approach should work with unpaired data. Finally, it should process the image in an end-to-end manner instead of just a single operation, like tonal adjustment. [click]
We propose a novel system, named “Exposure”, that has these merits simultaneously.

	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)	○	○	○	○

I summaries the previous approaches in this table. We need a method to generate high-resolution outputs. We are more interested in how retouching is done than in just getting the output images, so we want a human-understandable result. Since paired training data is not available in many cases, the approach should work with unpaired data. Finally, it should process the image in an end-to-end manner instead of just a single operation, like tonal adjustment. [click]
We propose a novel system, named “Exposure”, that has these merits simultaneously.

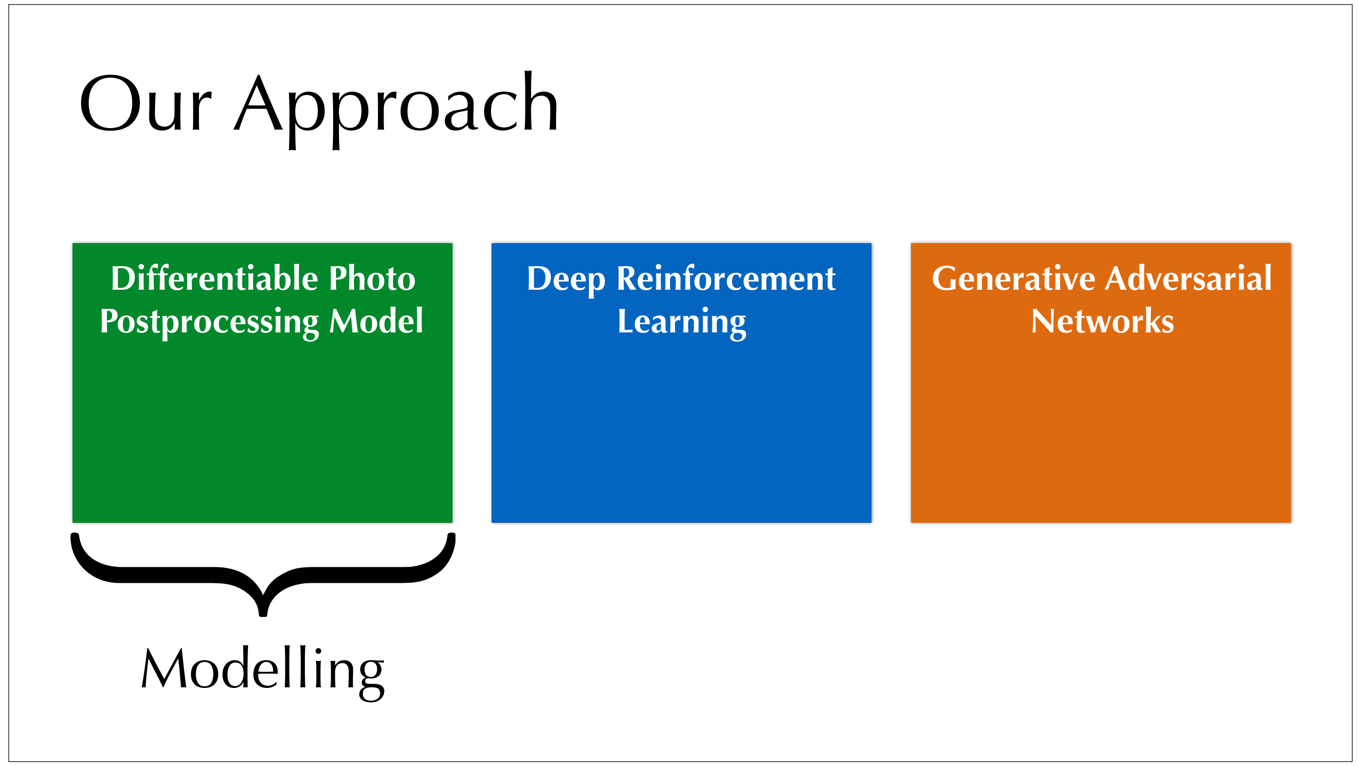
Our Approach

Differentiable Photo Postprocessing Model

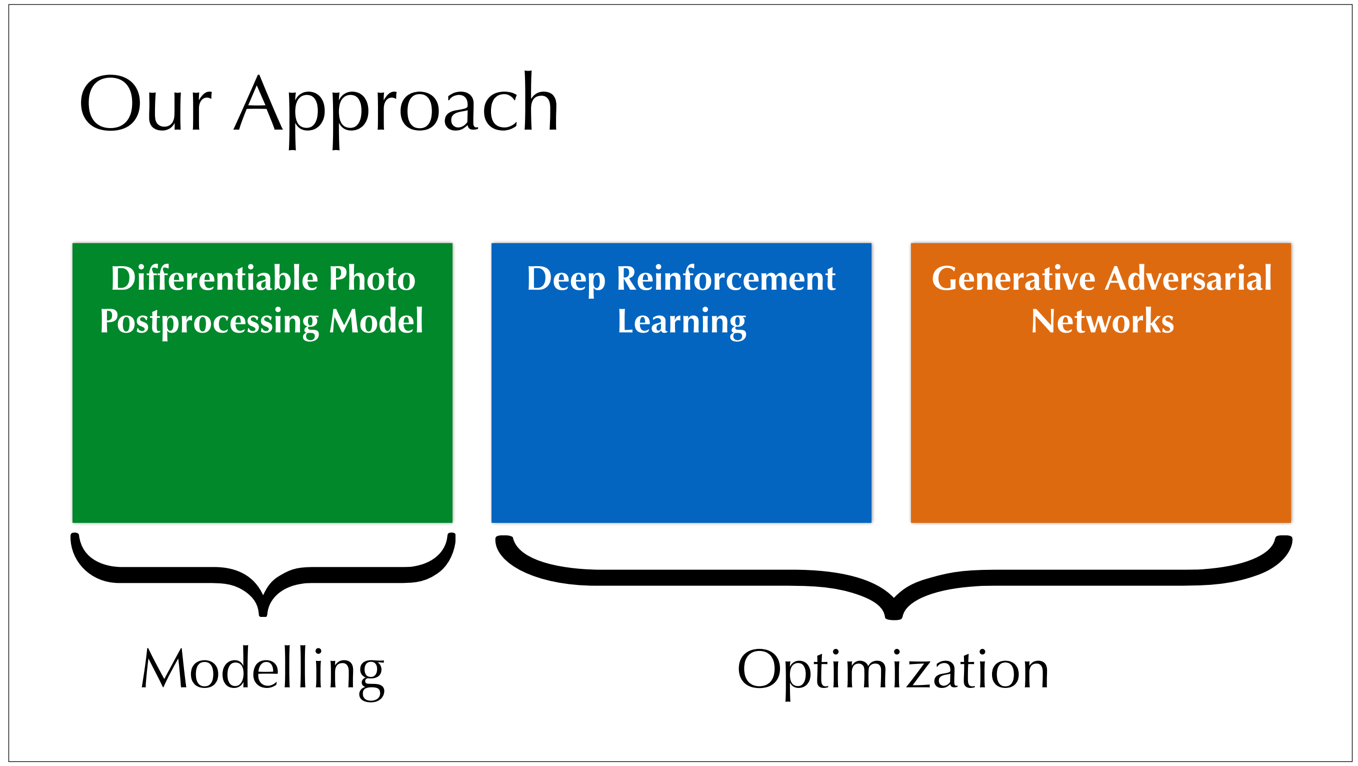
Deep Reinforcement Learning

Generative Adversarial Networks

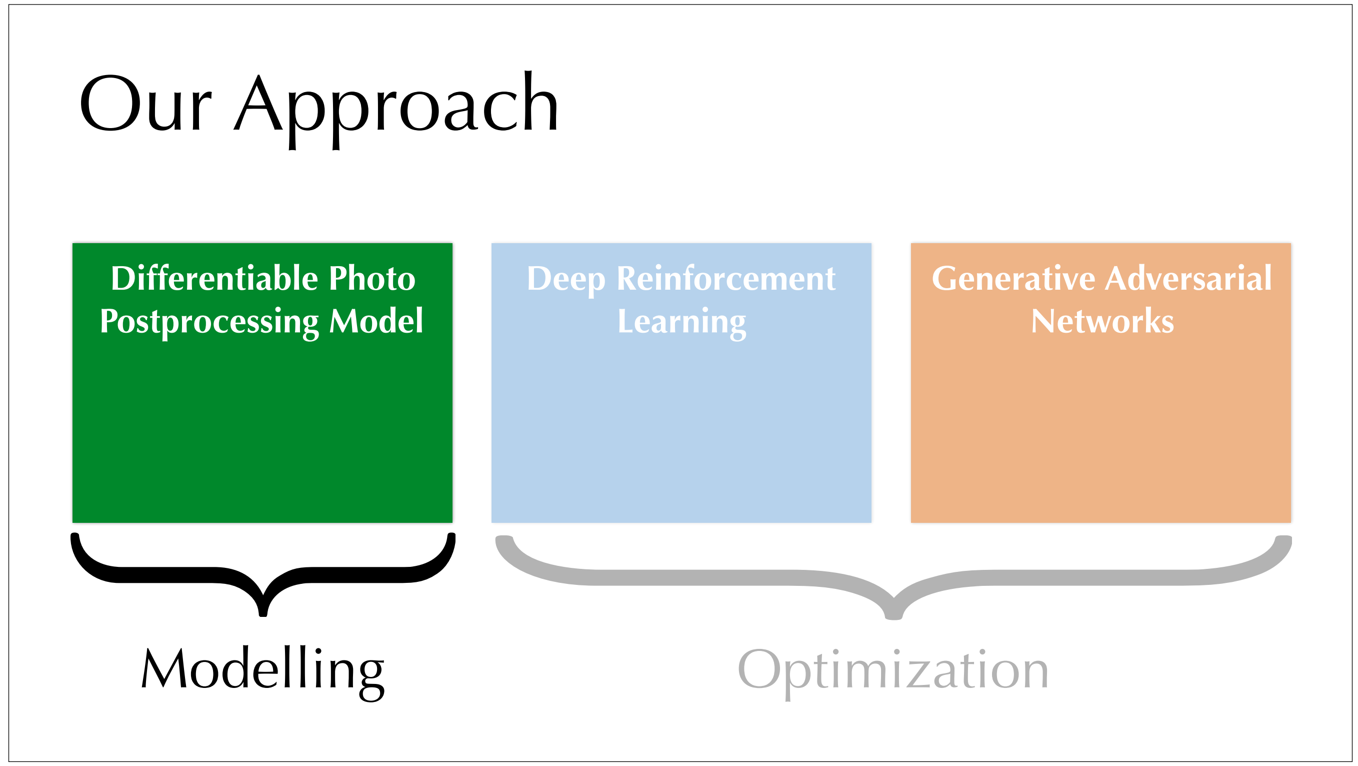
Compared with other deep learning approaches, ours is more tailored for high-resolution photo editing. We first build a model, and then optimise it.
Specifically, [click] we model the way human artists retouch photos. Such model regulates the scope of learning and makes learning much easier. Then [click], we optimize a decision maker using deep reinforcement learning and generative adversarial networks. I'll talk about our model first. [click]
[wait a second] [click]



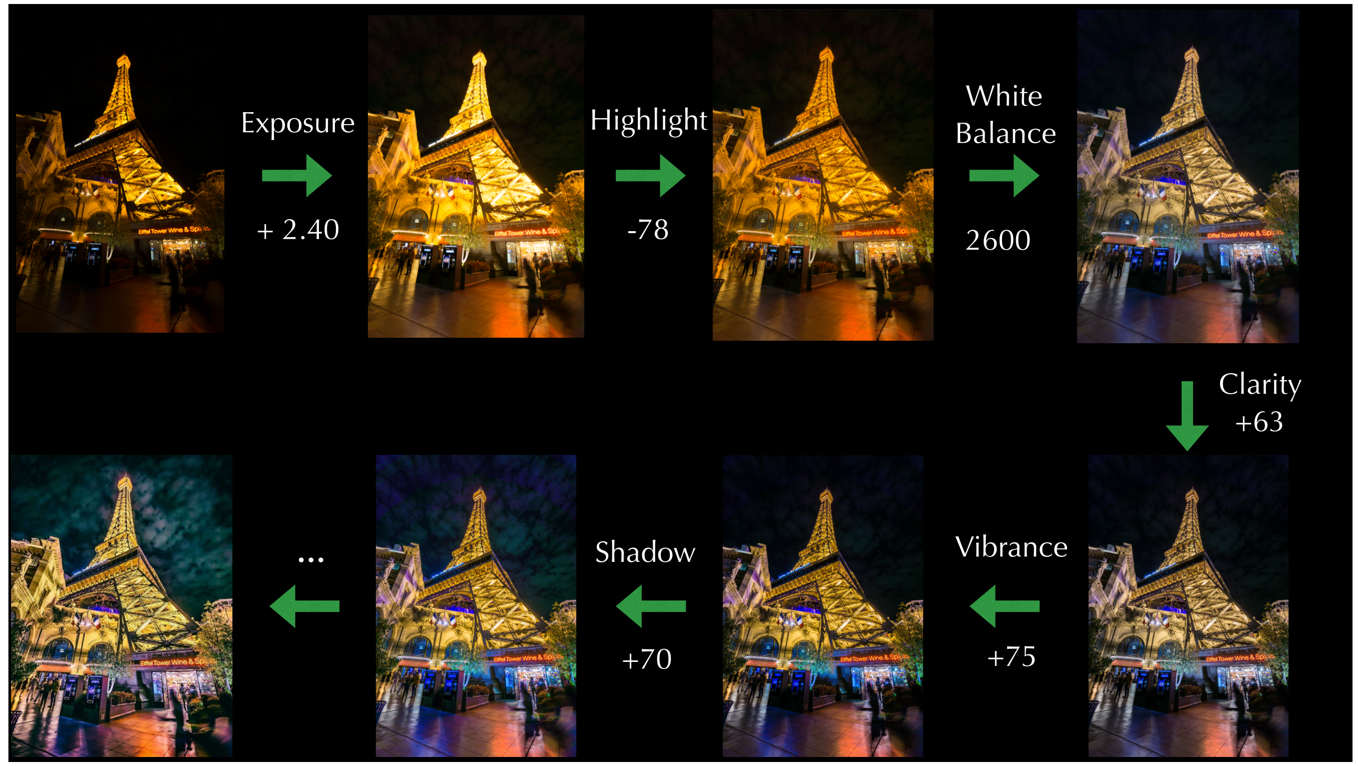
Compared with other deep learning approaches, ours is more tailored for high-resolution photo editing. We first build a model, and then optimise it. Specifically, [click] we model the way human artists retouch photos. Such model regulates the scope of learning and makes learning much easier. Then [click], we optimize a decision maker using deep reinforcement learning and generative adversarial networks. I'll talk about our model first. [click]
[wait a second] [click]



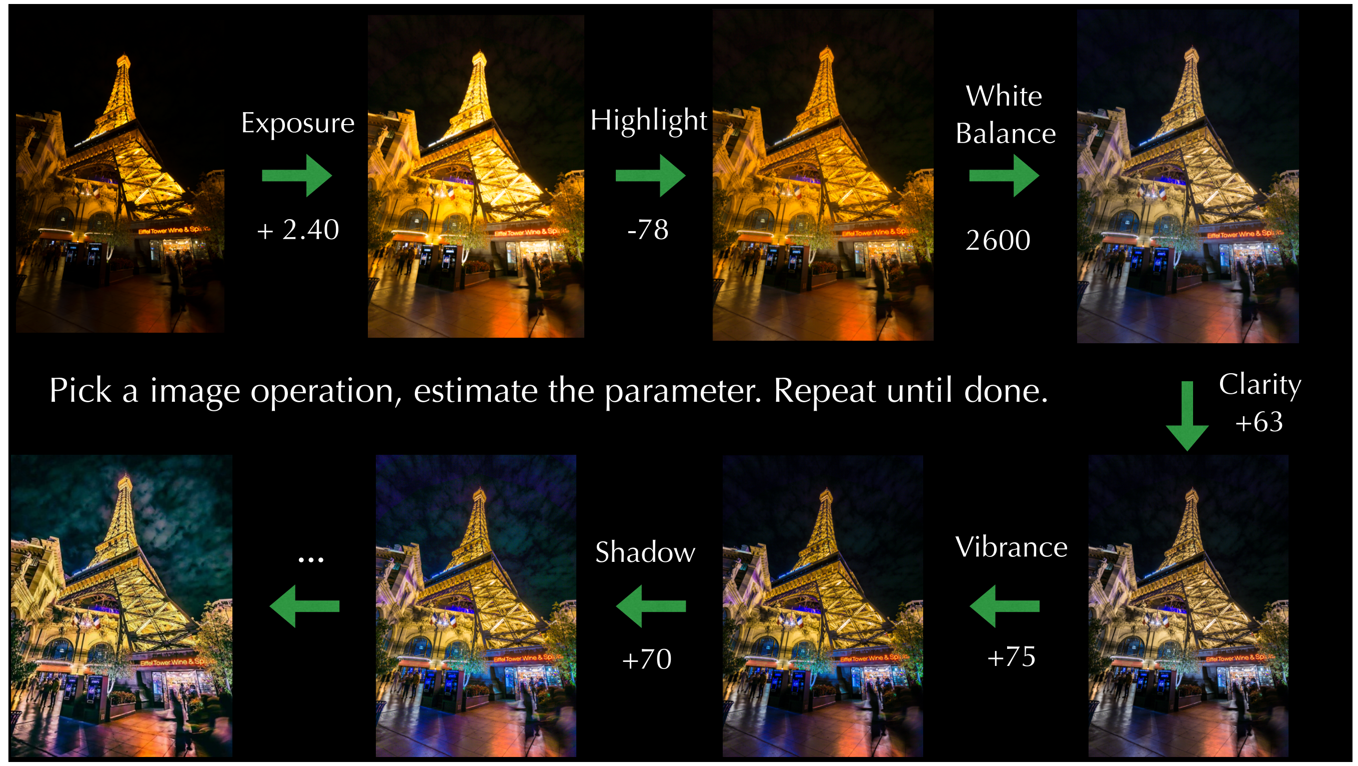
Compared with other deep learning approaches, ours is more tailored for high-resolution photo editing. We first build a model, and then optimise it. Specifically, [click] we model the way human artists retouch photos. Such model regulates the scope of learning and makes learning much easier. Then [click], we optimize a decision maker using deep reinforcement learning and generative adversarial networks. I'll talk about our model first. [click]
[wait a second] [click]



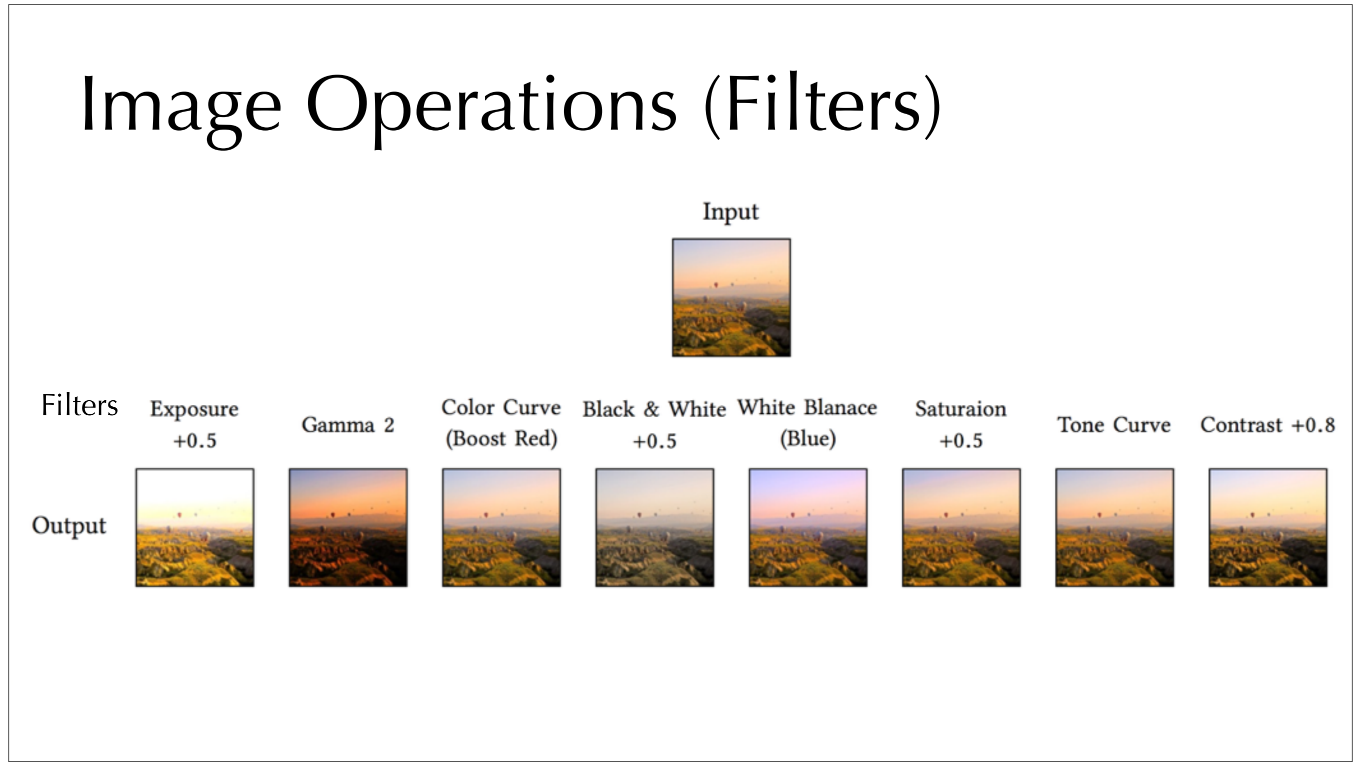
Compared with other deep learning approaches, ours is more tailored for high-resolution photo editing. We first build a model, and then optimise it.
Specifically, [click] we model the way human artists retouch photos. Such model regulates the scope of learning and makes learning much easier. Then [click], we optimize a decision maker using deep reinforcement learning and generative adversarial networks. I'll talk about our model first. [click]
[wait a second] [click]



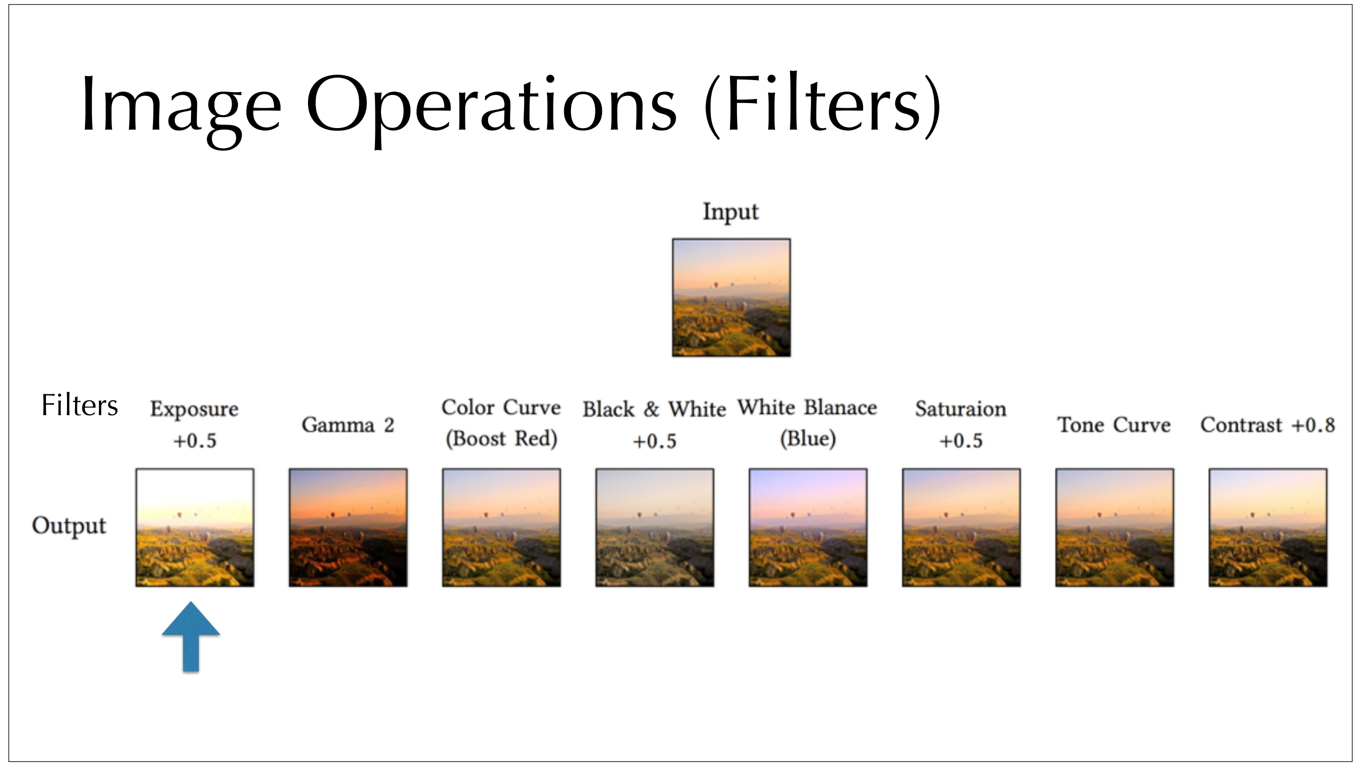
Let's take a step back and recall how humans do the job. Apparently, human brains do not generate a retouched image using convolutions. Instead, with the help of modern photo editing software, we retouch by making a series of decisions. [click] For each step, you first pick an image filter, and then estimate the parameter for that filter.



Let's take a step back and recall how humans do the job. Apparently, human brains do not generate a retouched image using convolutions. Instead, with the help of modern photo editing software, we retouch by making a series of decisions. [click] For each step, you first pick an image filter, and then estimate the parameter for that filter.



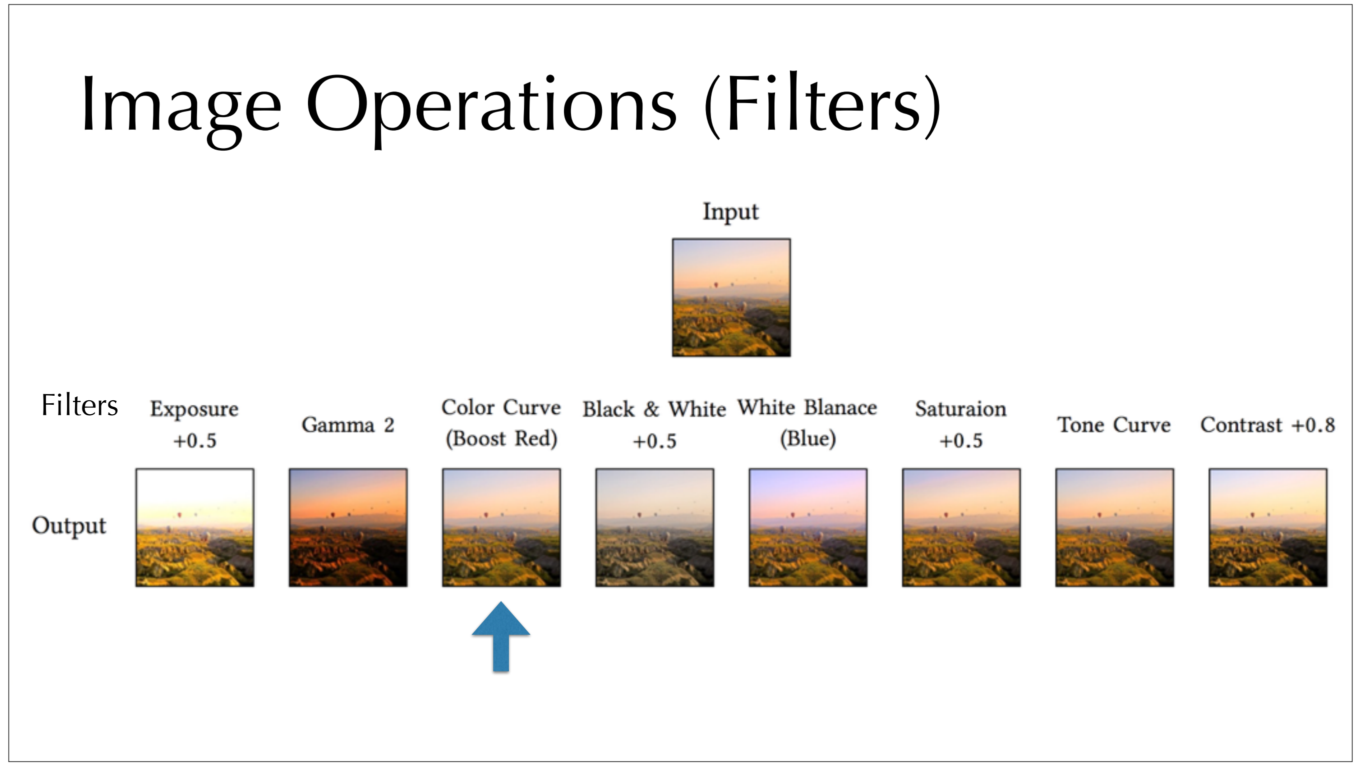
The first part of the model should be image filters.
We designed a set of eight frequently-used filters that can be applied to the images, each of which representing an edit to the image, including [click] exposure adjustment, [click] gamma correction, [click] color curve adjustment and so on. These operations are parameterised by one or more input parameters.
[Click] All of the operations are differentiable. This enables efficient gradient-based optimization.
[Click] Curves are represented using piece-wise linear functions, which is also differentiable.



The first part of the model should be image filters.
We designed a set of eight frequently-used filters that can be applied to the images, each of which representing an edit to the image, including [click] exposure adjustment, [click] gamma correction, [click] color curve adjustment and so on. These operations are parameterised by one or more input parameters.
[Click] All of the operations are differentiable. This enables efficient gradient-based optimization.
[Click] Curves are represented using piece-wise linear functions, which is also differentiable.

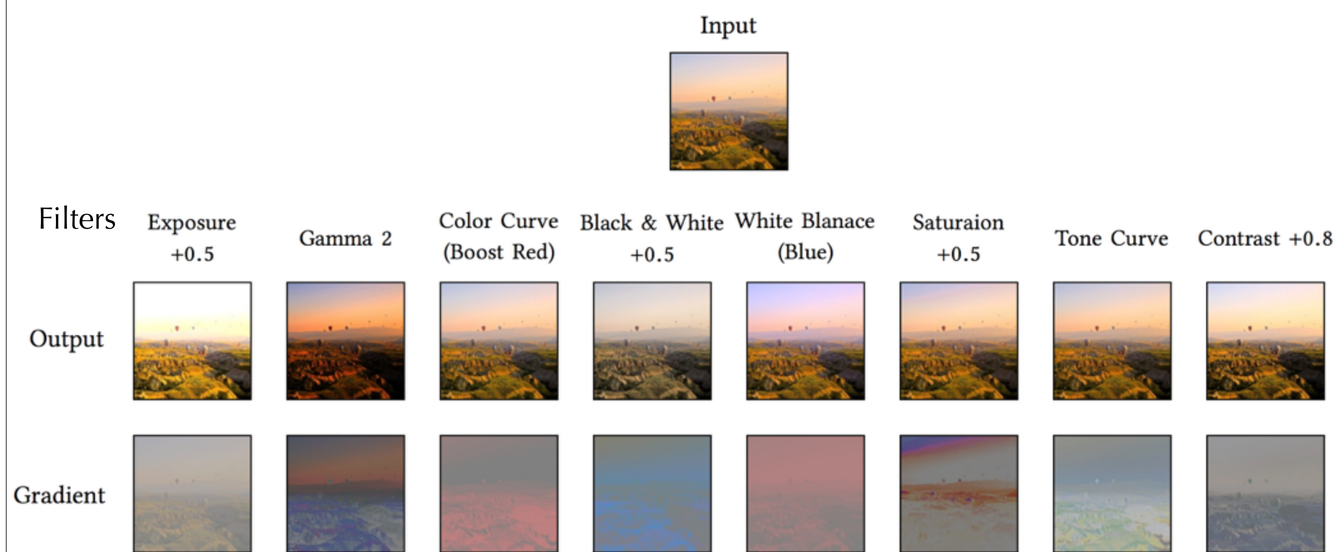
Image Operations (Filters)

The first part of the model should be image filters.
We designed a set of eight frequently-used filters that can be applied to the images, each of which representing an edit to the image, including [click] exposure adjustment, [click] gamma correction, [click] color curve adjustment and so on. These operations are parameterised by one or more input parameters.
[Click] All of the operations are differentiable. This enables efficient gradient-based optimization.
[Click] Curves are represented using piece-wise linear functions, which is also differentiable.



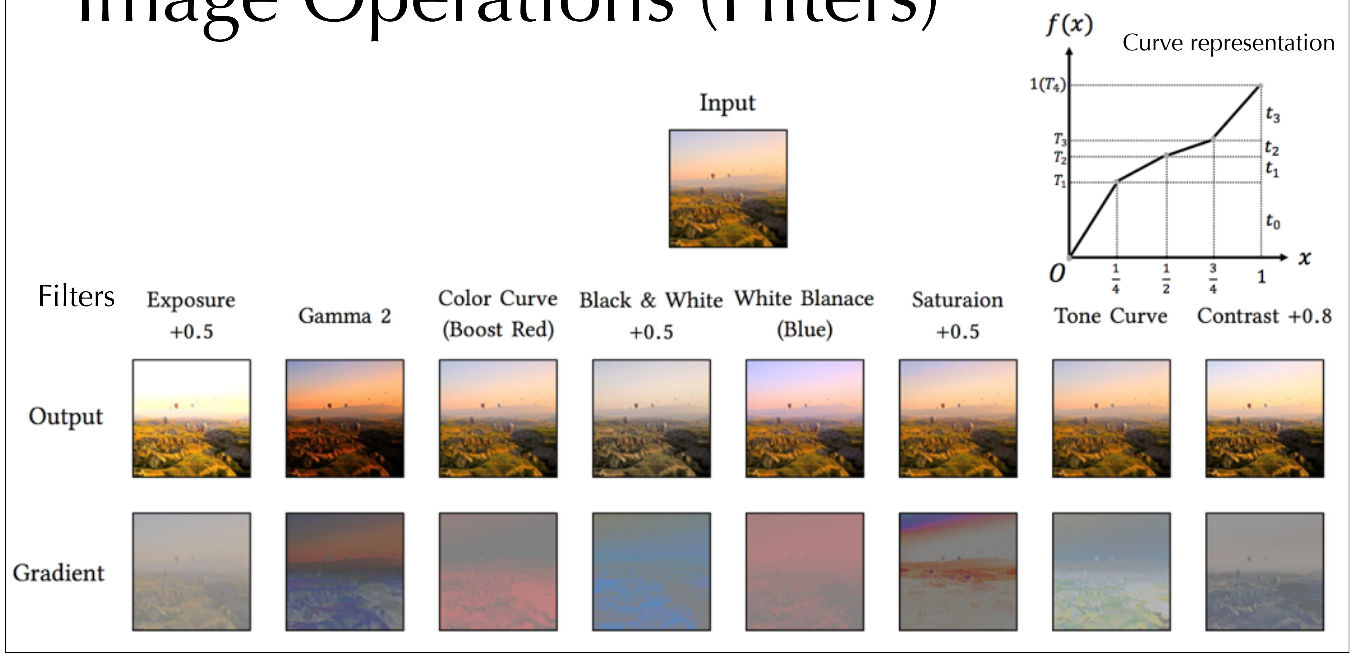
The first part of the model should be image filters.
We designed a set of eight frequently-used filters that can be applied to the images, each of which representing an edit to the image, including [click] exposure adjustment, [click] gamma correction, [click] color curve adjustment and so on. These operations are parameterised by one or more input parameters.
[Click] All of the operations are differentiable. This enables efficient gradient-based optimization.
[Click] Curves are represented using piece-wise linear functions, which is also differentiable.

Image Operations (Filters)

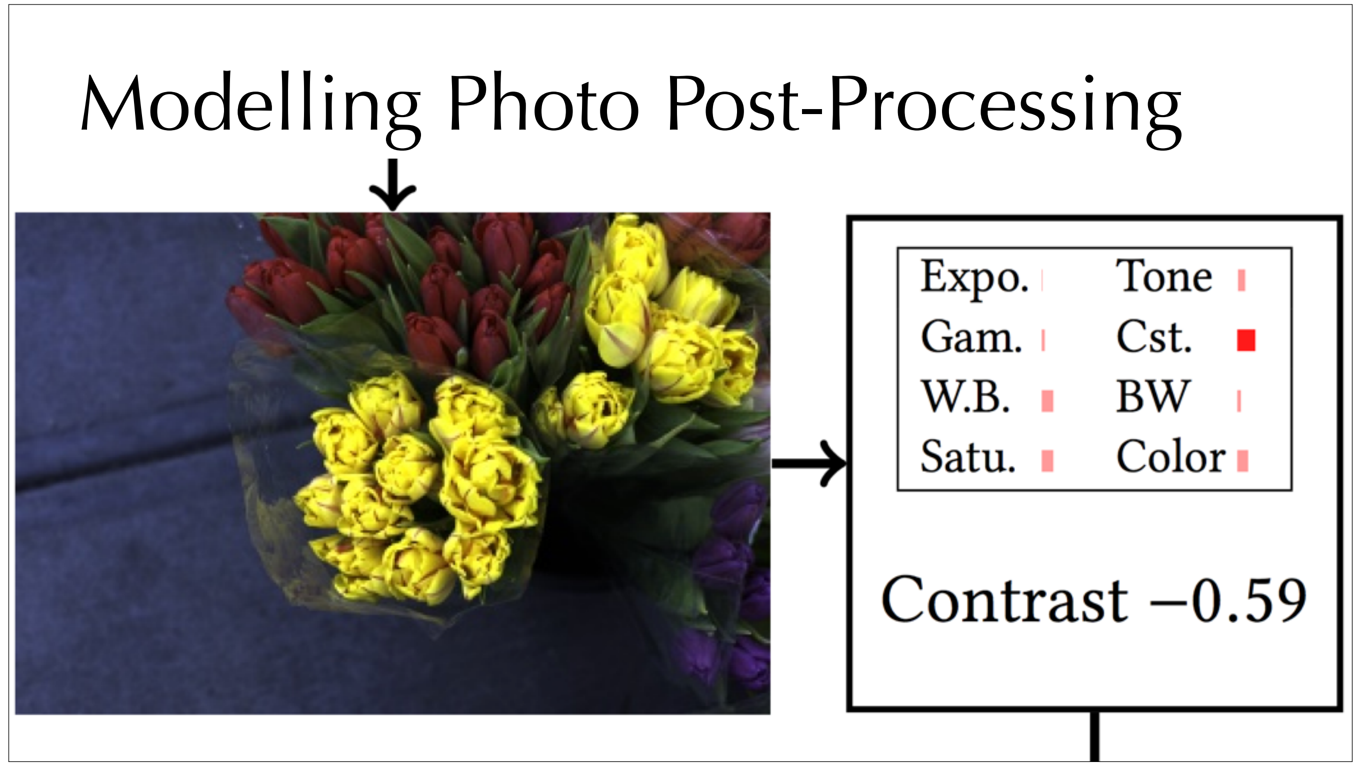


The first part of the model should be image filters.
We designed a set of eight frequently-used filters that can be applied to the images, each of which representing an edit to the image, including [click] exposure adjustment, [click] gamma correction, [click] color curve adjustment and so on. These operations are parameterised by one or more input parameters.
[Click] All of the operations are differentiable. This enables efficient gradient-based optimization.
[Click] Curves are represented using piece-wise linear functions, which is also differentiable.

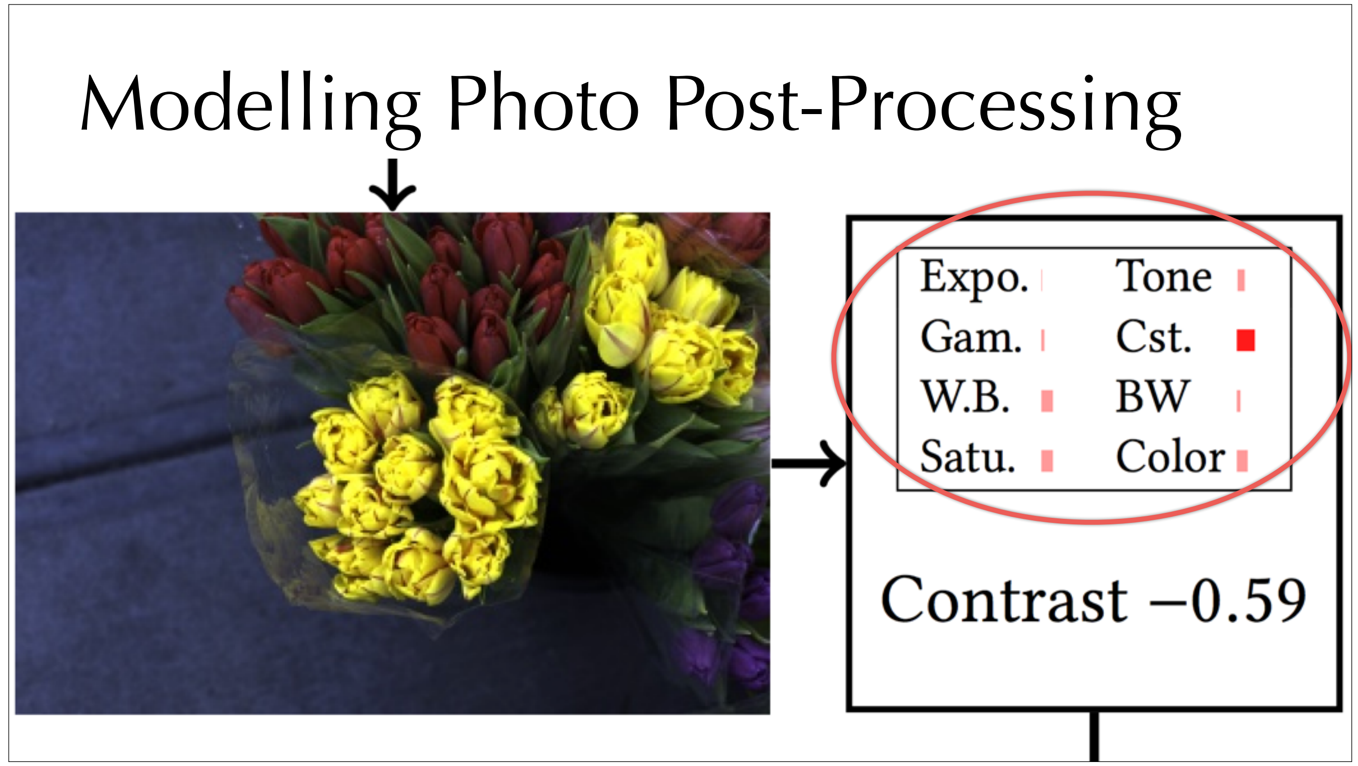
Image Operations (Filters)



The first part of the model should be image filters.
We designed a set of eight frequently-used filters that can be applied to the images, each of which representing an edit to the image, including [click] exposure adjustment, [click] gamma correction, [click] color curve adjustment and so on. These operations are parameterised by one or more input parameters.
[Click] All of the operations are differentiable. This enables efficient gradient-based optimization.
[Click] Curves are represented using piece-wise linear functions, which is also differentiable.



Given an image, either high-definition or not, [click] the photographer choses a filter to apply, and then [Click] estimate a parameter for that filter. After several steps, [Click] he gets a nice looking image.

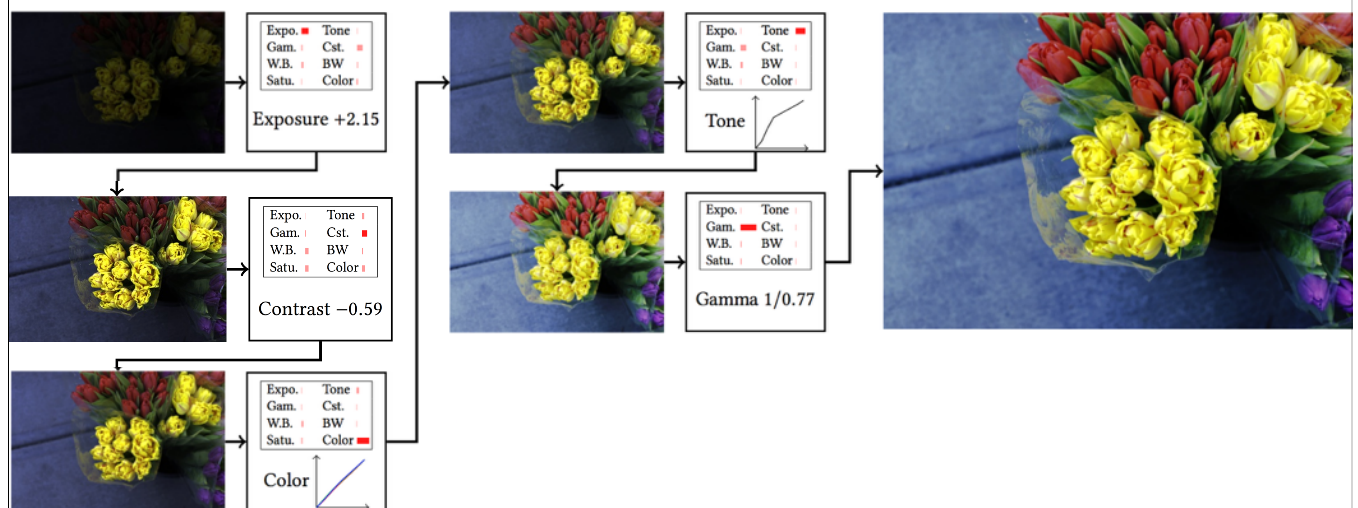


Given an image, either high-definition or not, [click] the photographer choses a filter to apply, and then [Click] estimate a parameter for that filter. After several steps, [Click] he gets a nice looking image.

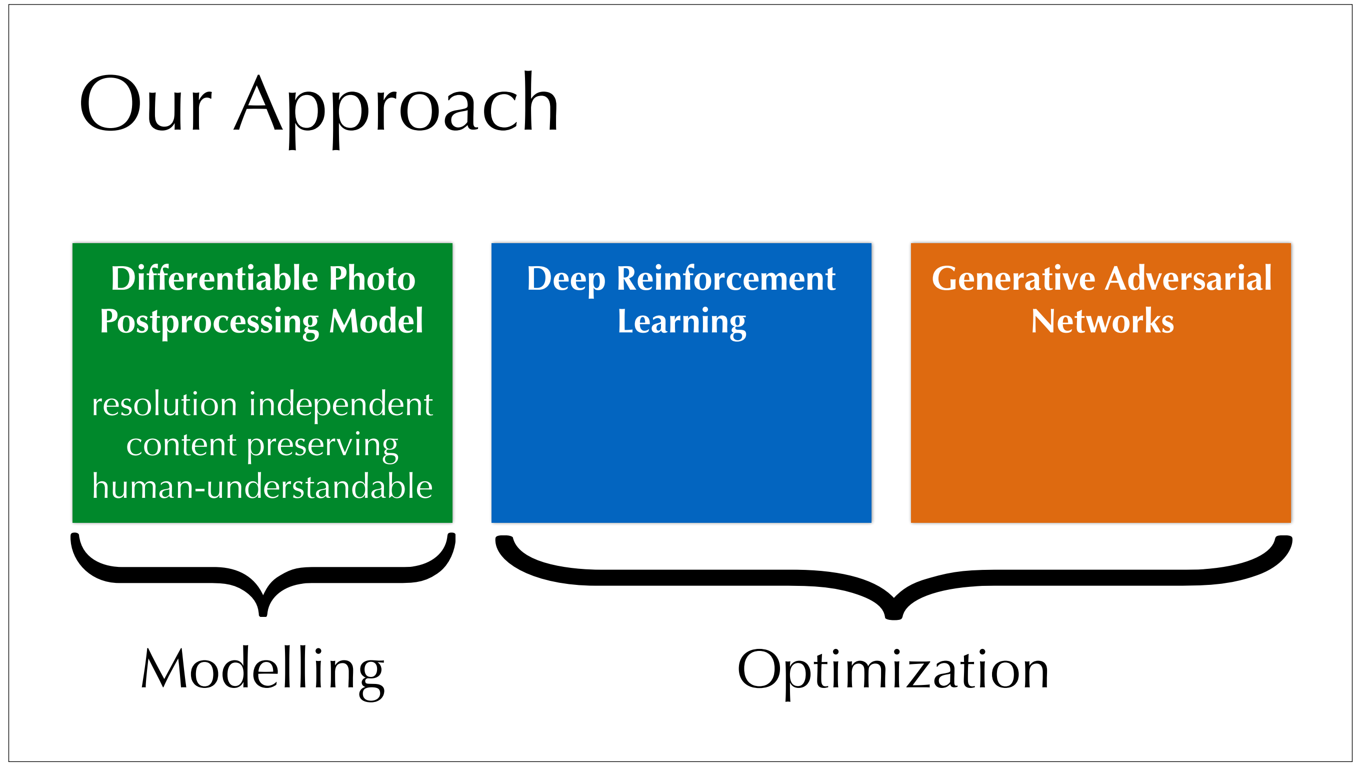
Modelling Photo Post-Processing

Given an image, either high-definition or not, [click] the photographer choses a filter to apply, and then [Click] estimate a parameter for that filter. After several steps, [Click] he gets a nice looking image.

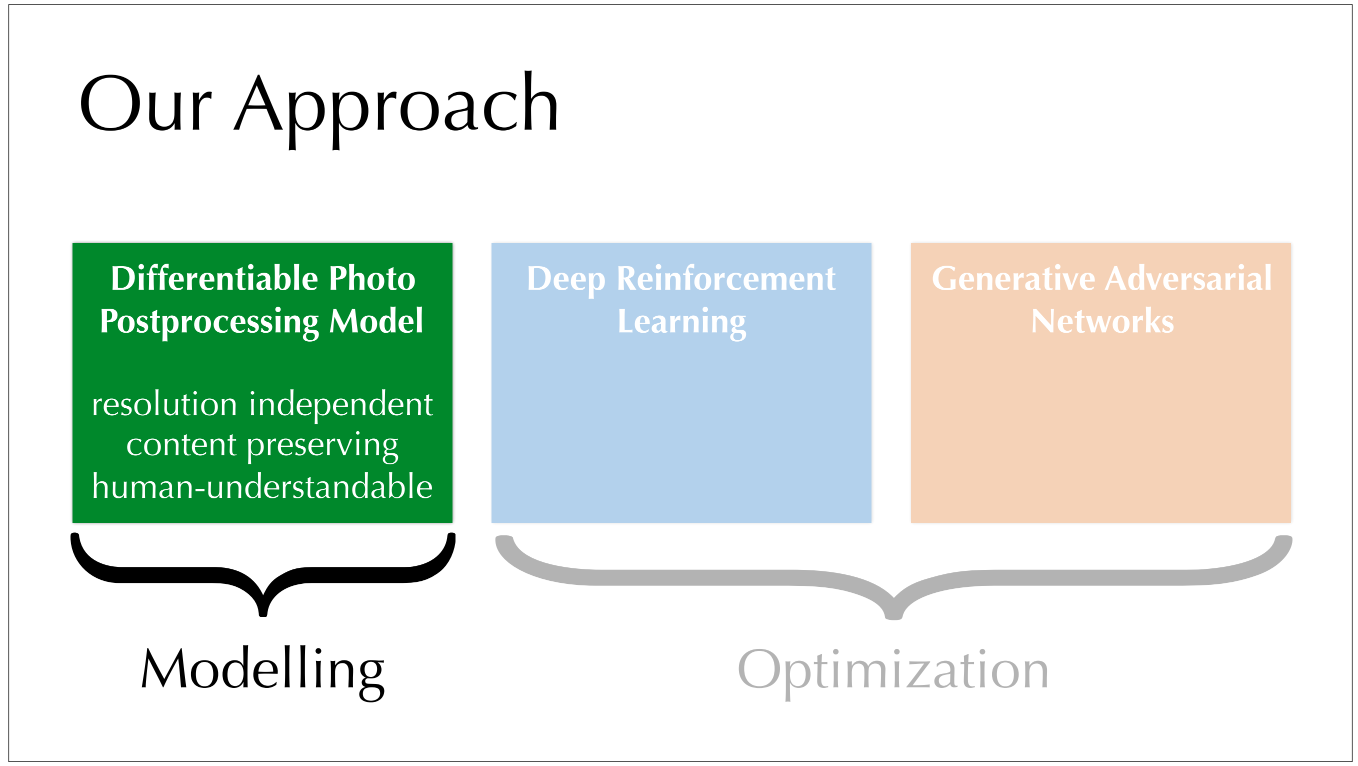
Modelling Photo Post-Processing



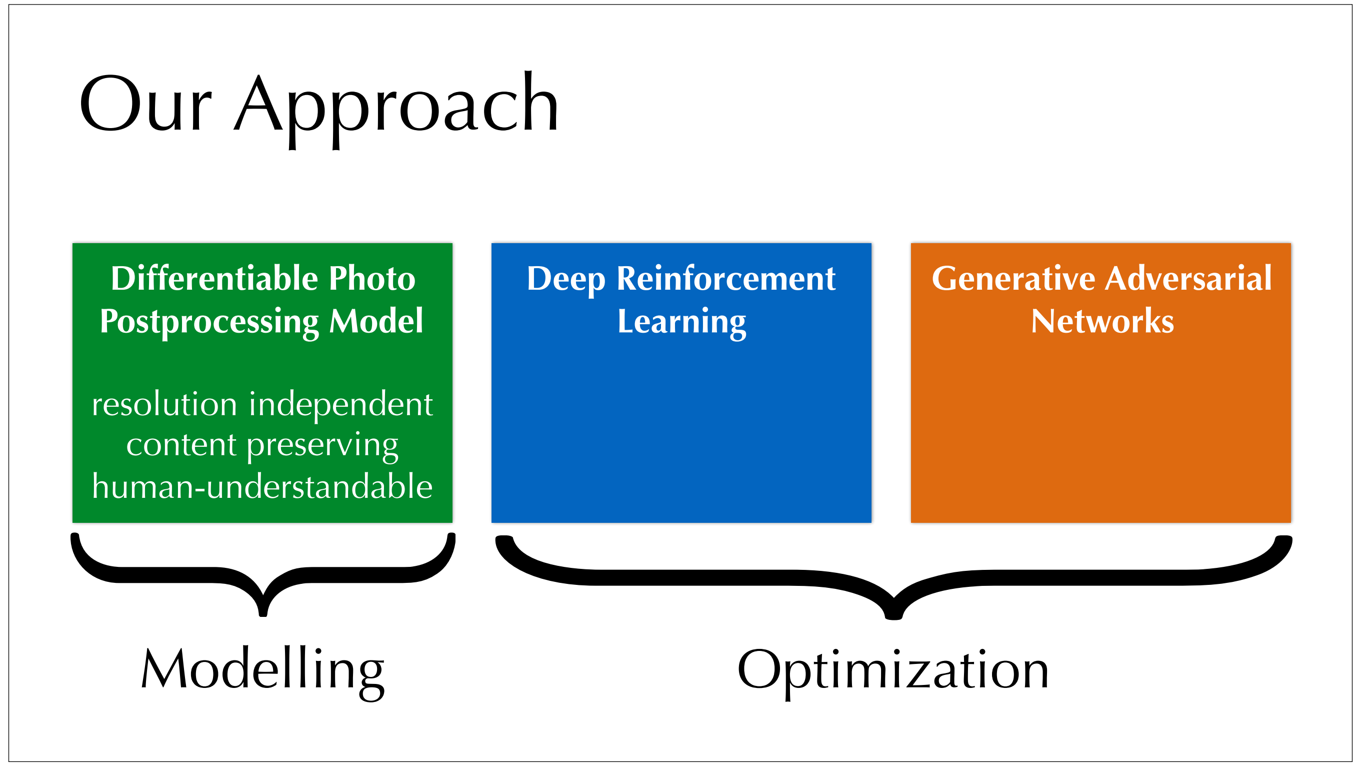
Given an image, either high-definition or not, [click] the photographer choses a filter to apply, and then [Click] estimate a parameter for that filter. After several steps, [Click] he gets a nice looking image.



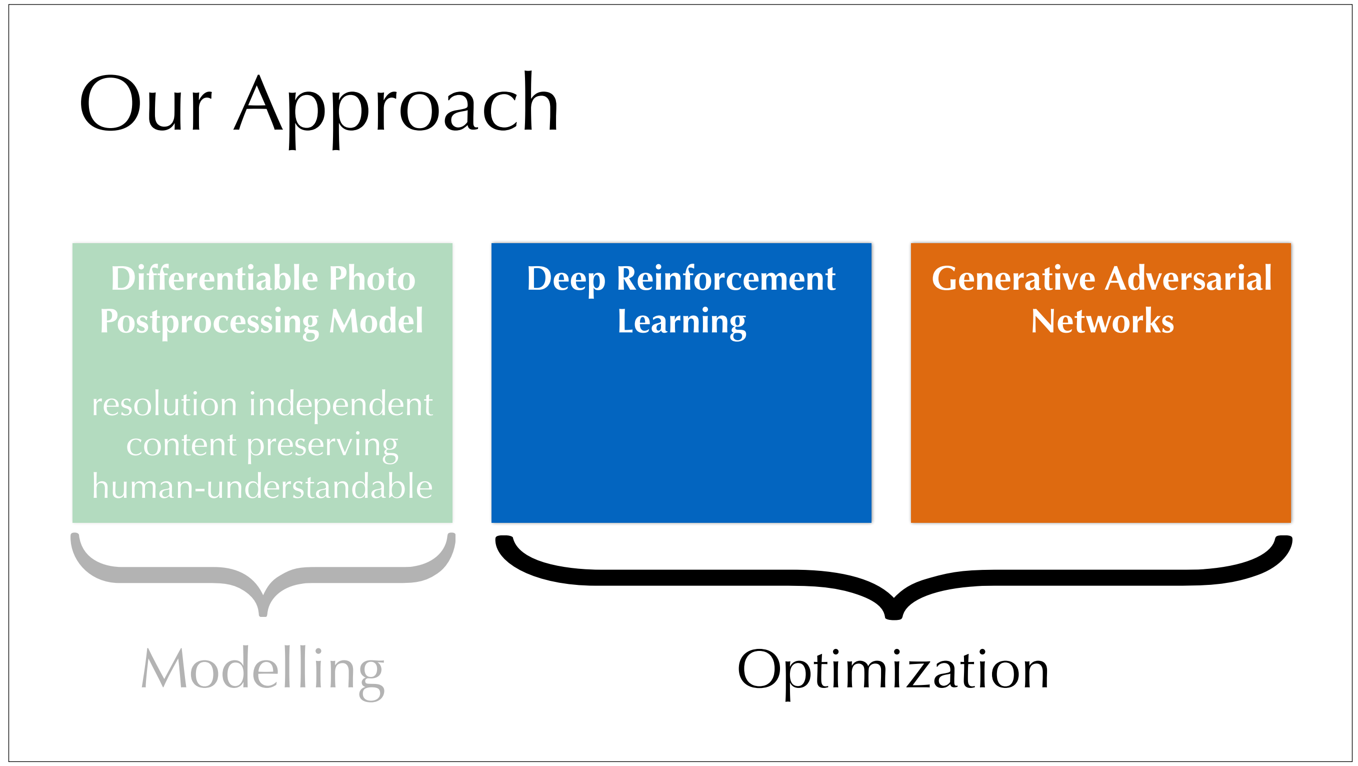
One of the highlights of our work is the differentiable model. Though straightforward, it does solve a lot of problems.
Firstly, the filters are resolution independent, that means a decision made on a low-res image can be applied to a high-res one directly. Secondly, they preserve content perfectly. Lastly, the operation sequence are easily understandable.



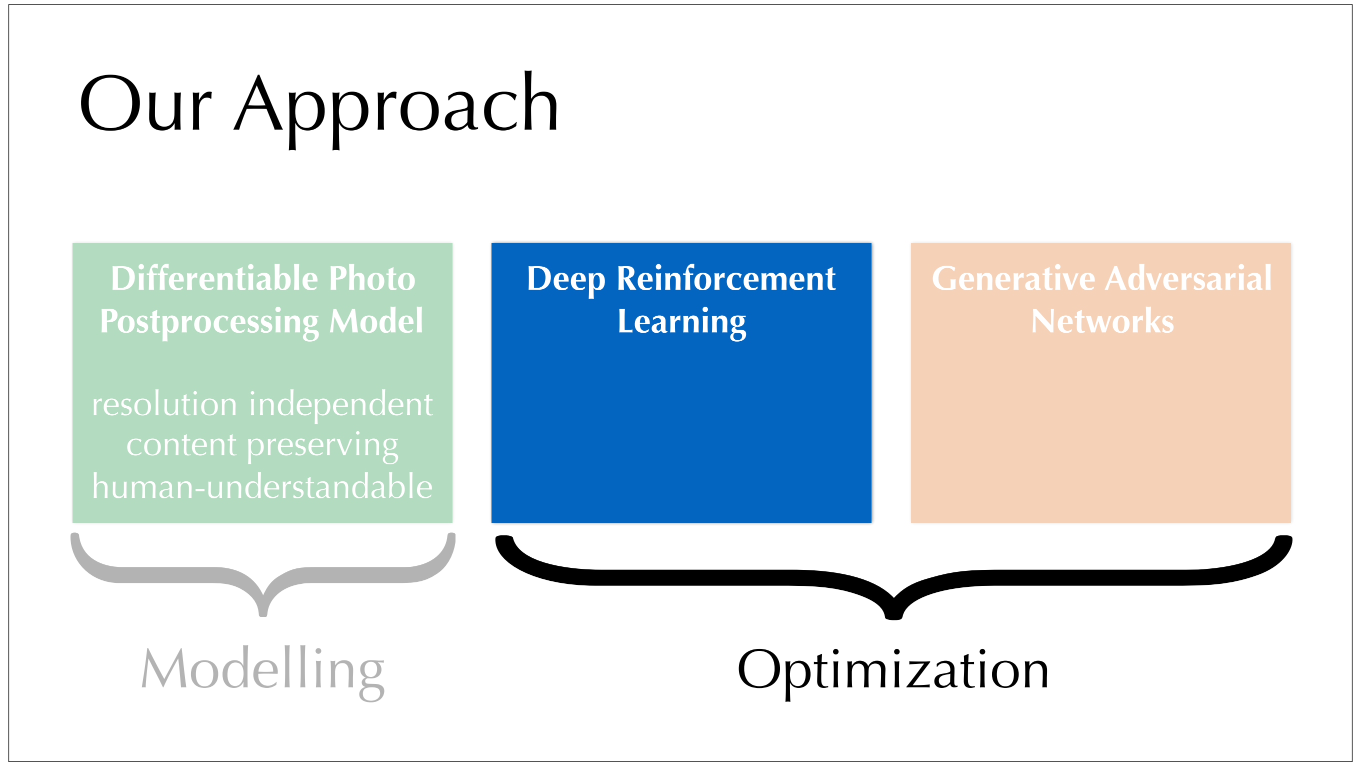
One of the highlights of our work is the differentiable model. Though straightforward, it does solve a lot of problems.
Firstly, the filters are resolution independent, that means a decision made on a low-res image can be applied to a high-res one directly. Secondly, they preserve content perfectly. Lastly, the operation sequence are easily understandable.



Now the problems is how to optimise image operations. The first component of the optimisation framework [click] is deep reinforcement learning.

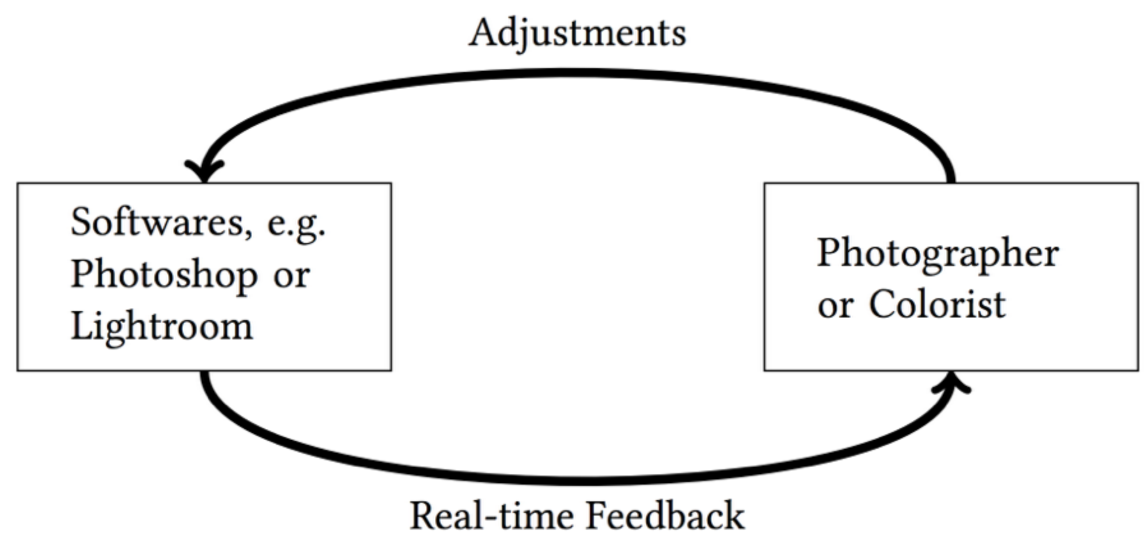


Now the problems is how to optimise image operations. The first component of the optimisation framework [click] is deep reinforcement learning.

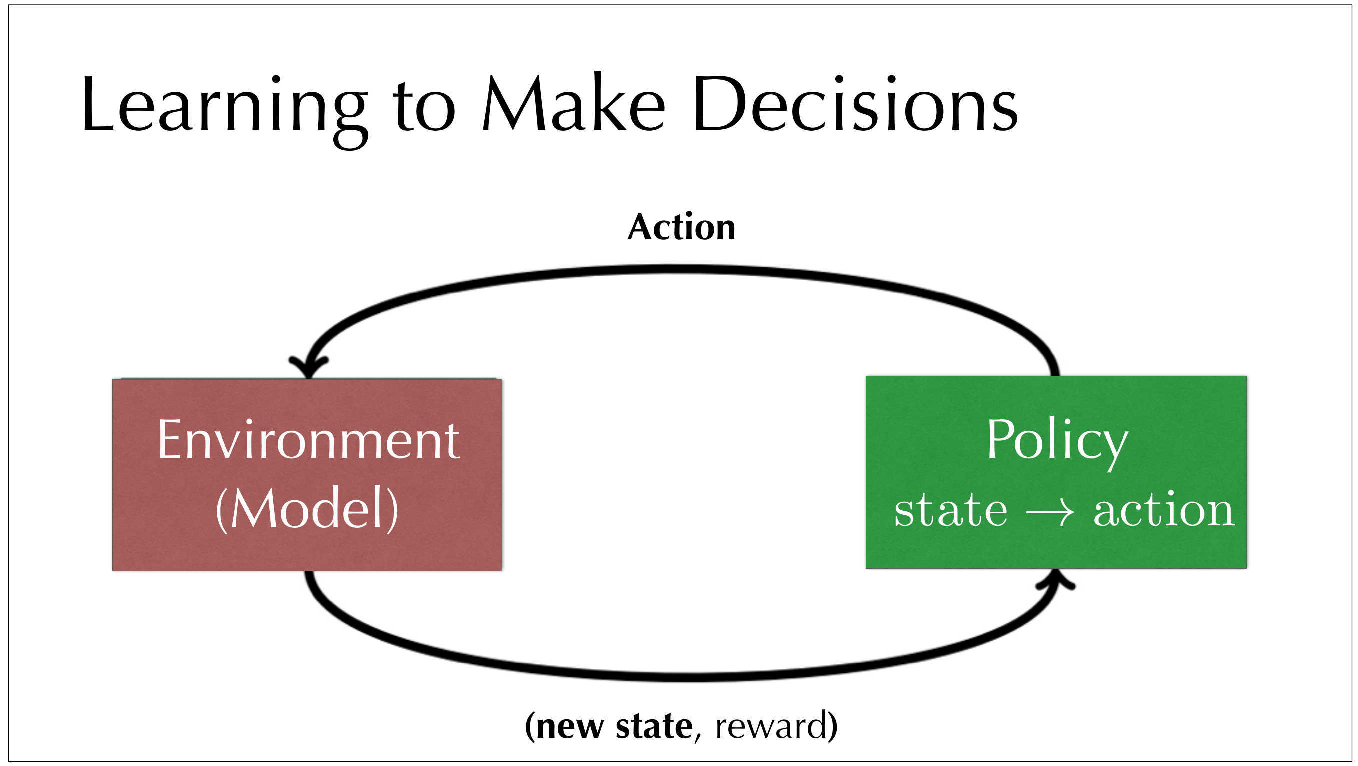


Now the problems is how to optimise image operations. The first component of the optimisation framework [click] is deep reinforcement learning.

Learning to Make Decisions

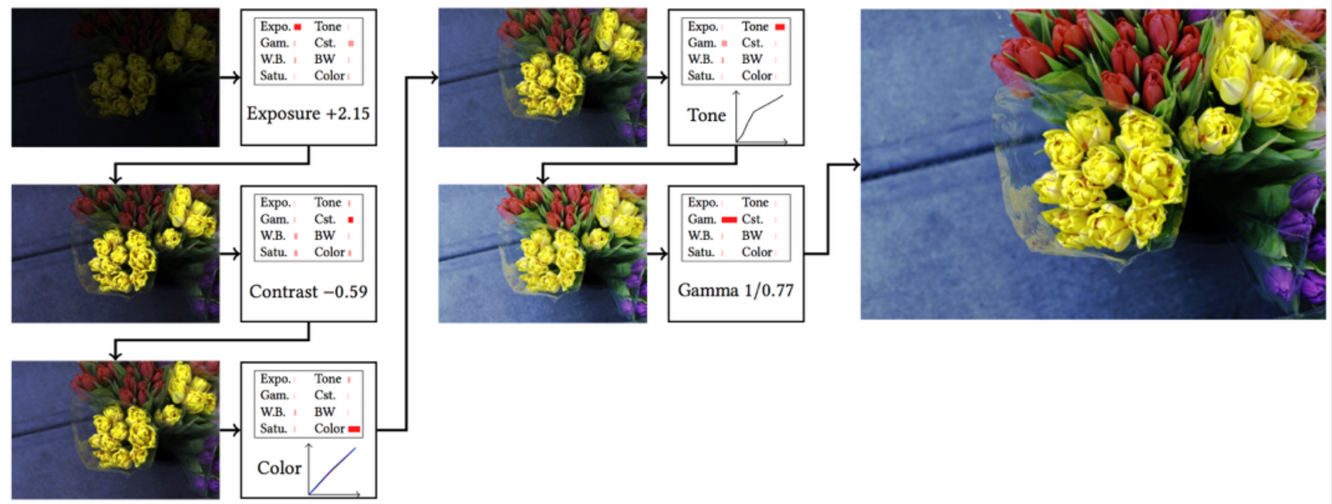


Looking at the workflow of human artists, you will find it a very good fit to reinforcement learning.



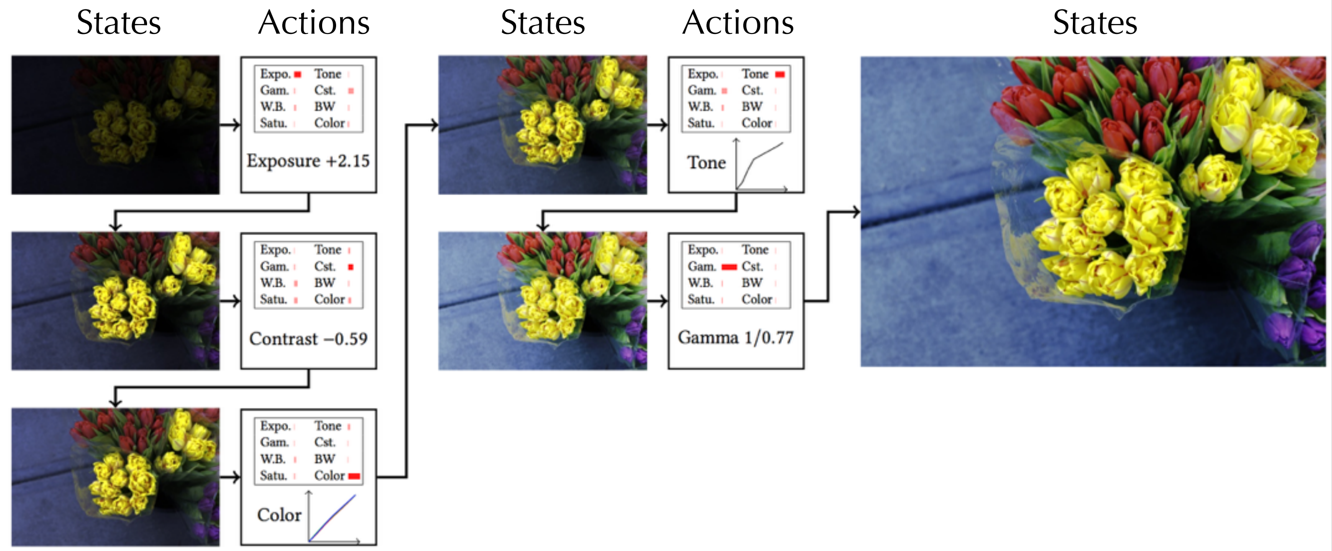
In reinforcement learning, an self-optimising agent takes action based on the current state. The environment changes its state given the action, and feedback the agent with a reward. The decision making strategy is called a policy.

Learning to Make Decisions



In our case, [click] the states are images and actions are image filters. The policy proposes image operations based on the current visual appearance of the image.

Learning to Make Decisions



In our case, [click] the states are images and actions are image filters. The policy proposes image operations based on the current visual appearance of the image.

Gradient-based Policy Optimization

Monte-Carlo Estimation of (Stochastic) Policy Gradient

Loss Policy Value

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

$a_1\sim\pi_1(s)$ ← **Discrete policy (filter selection)**

$a_2=\pi_2(s,a_1)$ ← **Continuous (deterministic) policy parameter estimation**

To optimise the policy, we utilise gradient-based methods.

In fact, we have two policies. The first one picks a filter out of the eight possibilities, and the second one estimates the filter parameters. Those policies are in practice approximated by neural networks. The selection of filter is optimise using a monte-carol estimation of policy gradient.

Gradient-based Policy Optimization

Deterministic Policy Gradient Theorem

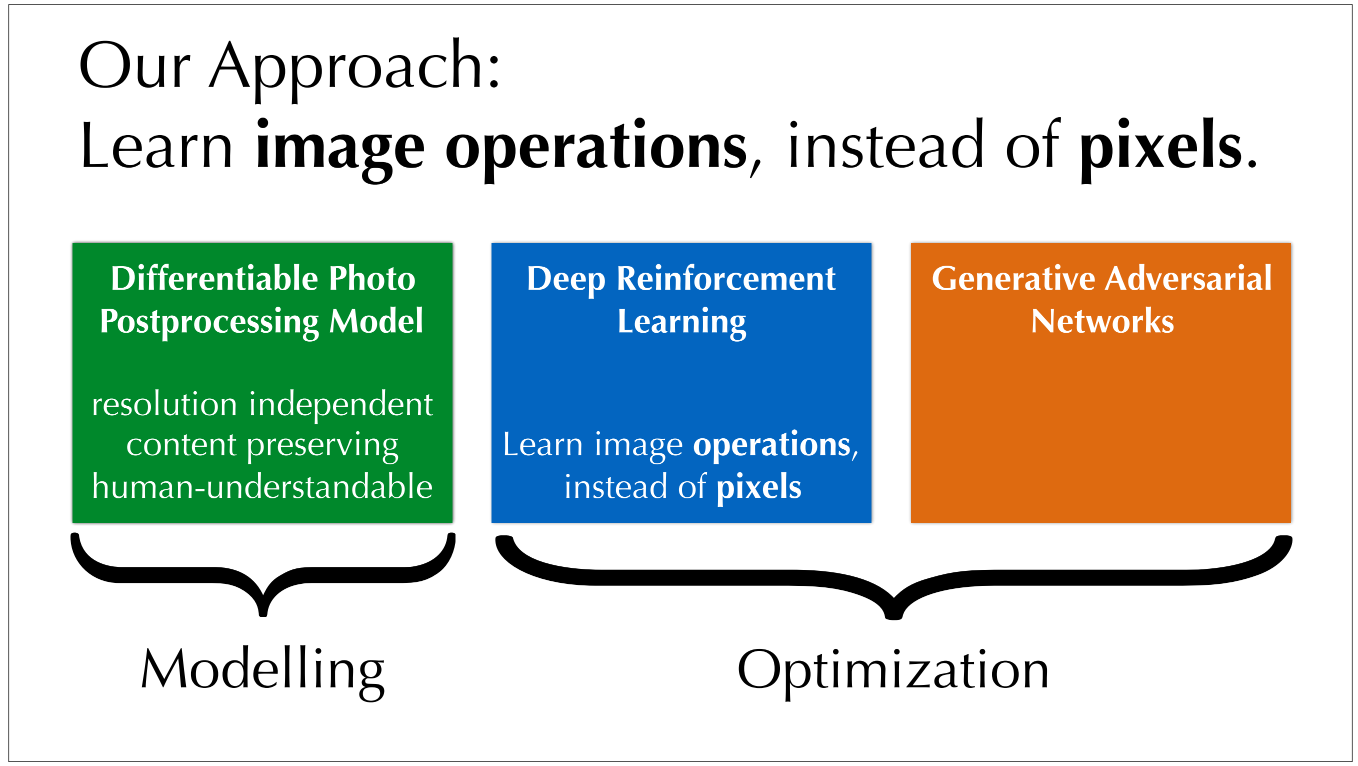
Loss Policy Value

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

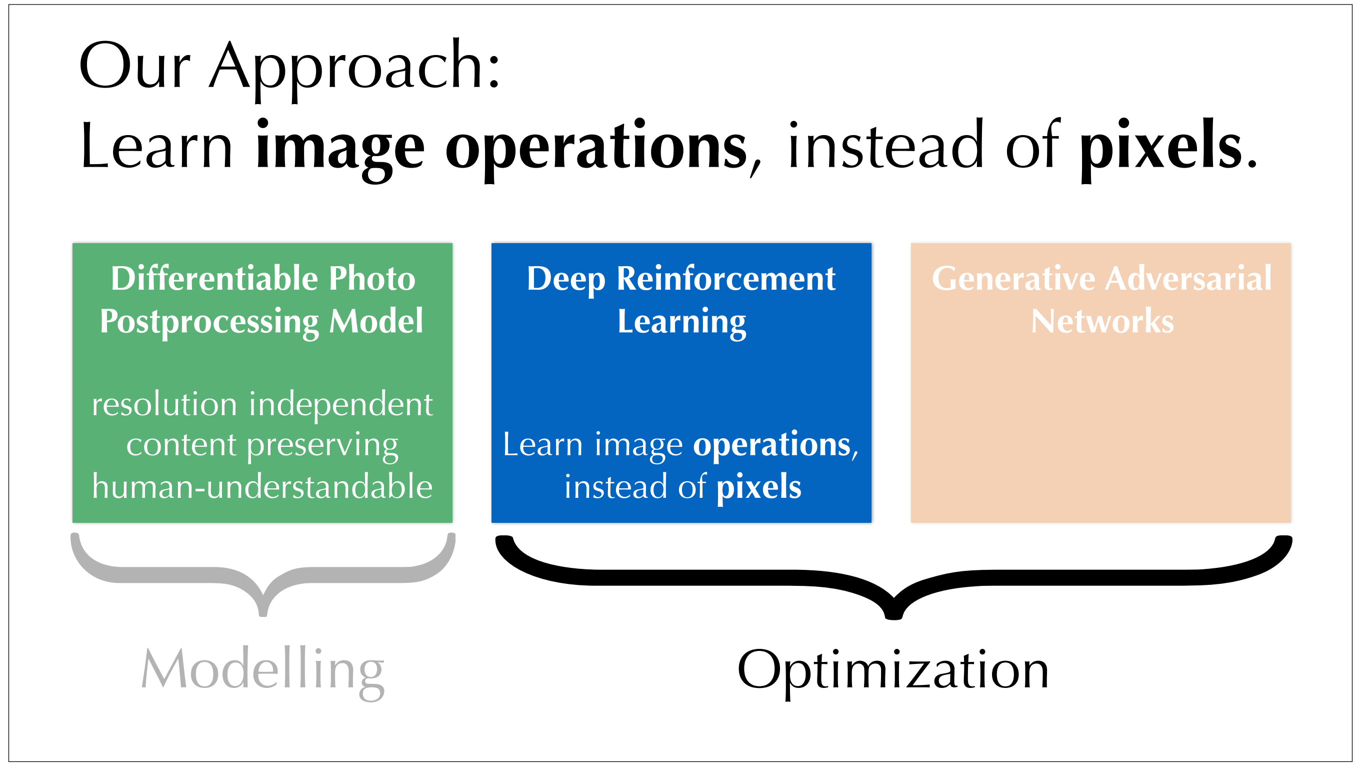
NN weights

Continuous (deterministic) policy parameter estimation

The continuous parameter estimation is optimised via the deterministic policy gradient theorem.
Please see our paper for more details.



Deep reinforcement learning allows the system to learn actually post-processing a photo step-by-step, instead of to generating one pixel-by-pixel. This design makes infinite resolution and human understandability practical.



Deep reinforcement learning allows the system to learn actually post-processing a photo step-by-step, instead of to generating one pixel-by-pixel. This design makes infinite resolution and human understandability practical.

Our Approach:

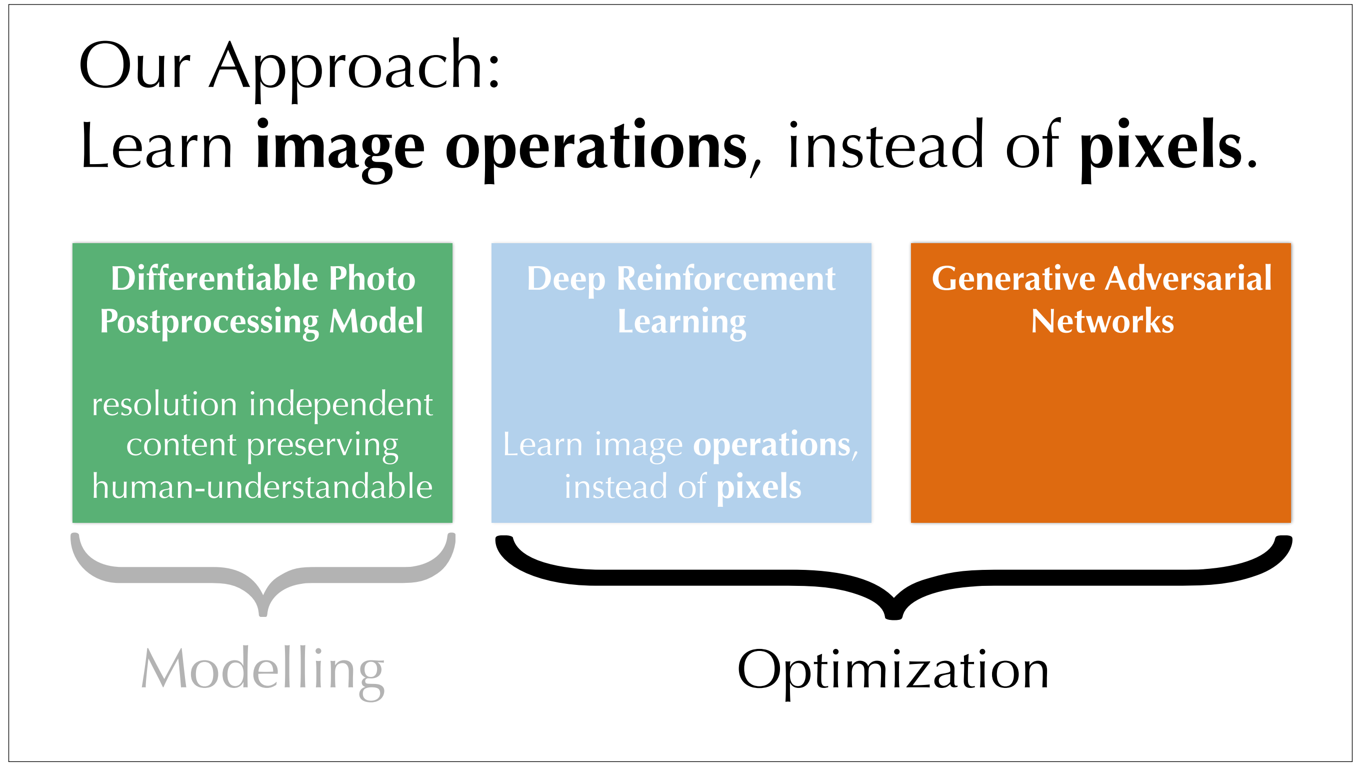
Learn **image operations**, instead of **pixels**.

Differentiable Photo Postprocessing Model	Deep Reinforcement Learning	Generative Adversarial Networks
resolution independent content preserving human-understandable	Learn image **operations**, instead of **pixels**	

Modelling

Optimization

There is still one missing component, which is the reward function for reinforcement learning. [click] We used a GAN framework for this purpose.



There is still one missing component, which is the reward function for reinforcement learning. [click] We used a GAN framework for this purpose.

Designing the Reward Function

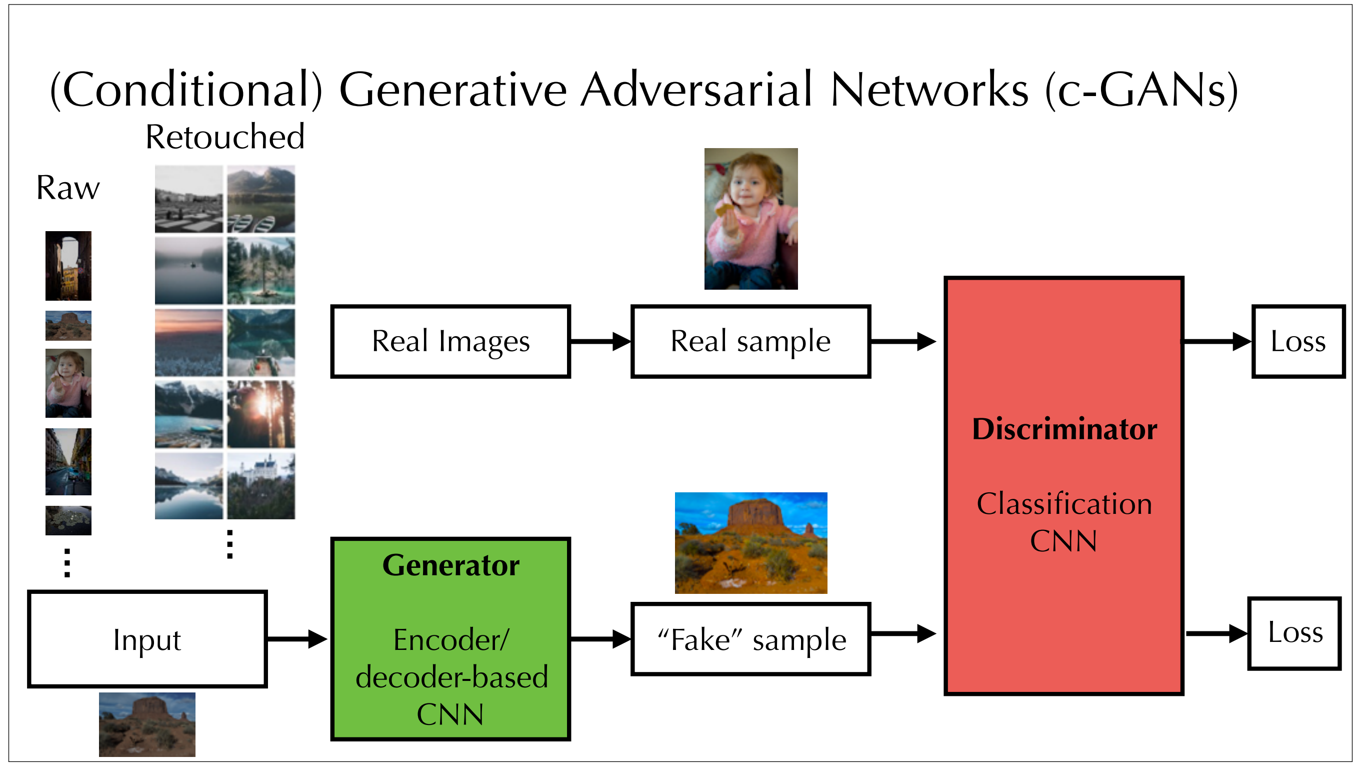
$r = -D($  $,$ 

$)$

Generated

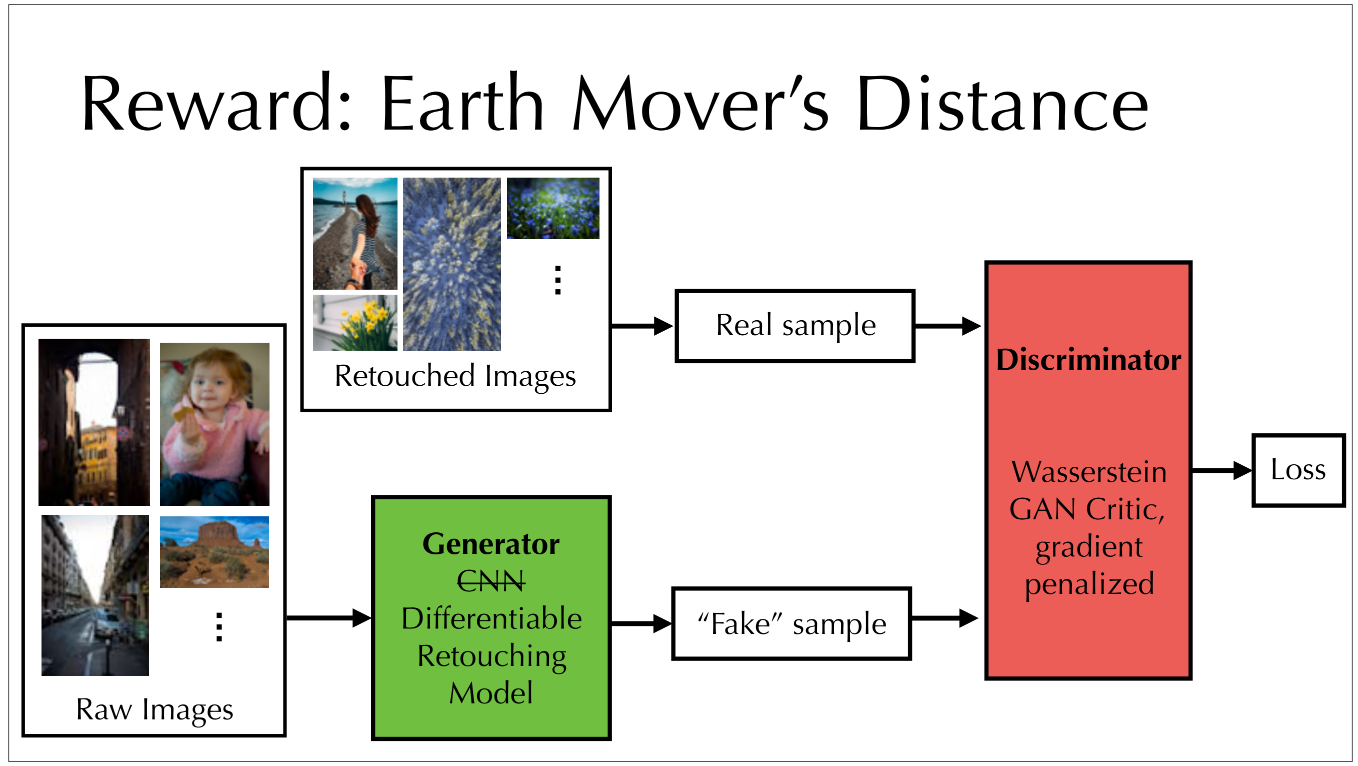
Target (i.e. "ground truth")

Since our goal is to learn the image style of the target dataset, it is natural to use the negative distance from the generated image to target images as reward. The closer image we are getting, the smaller distance will be evaluated, and the policy should be rewarded more.

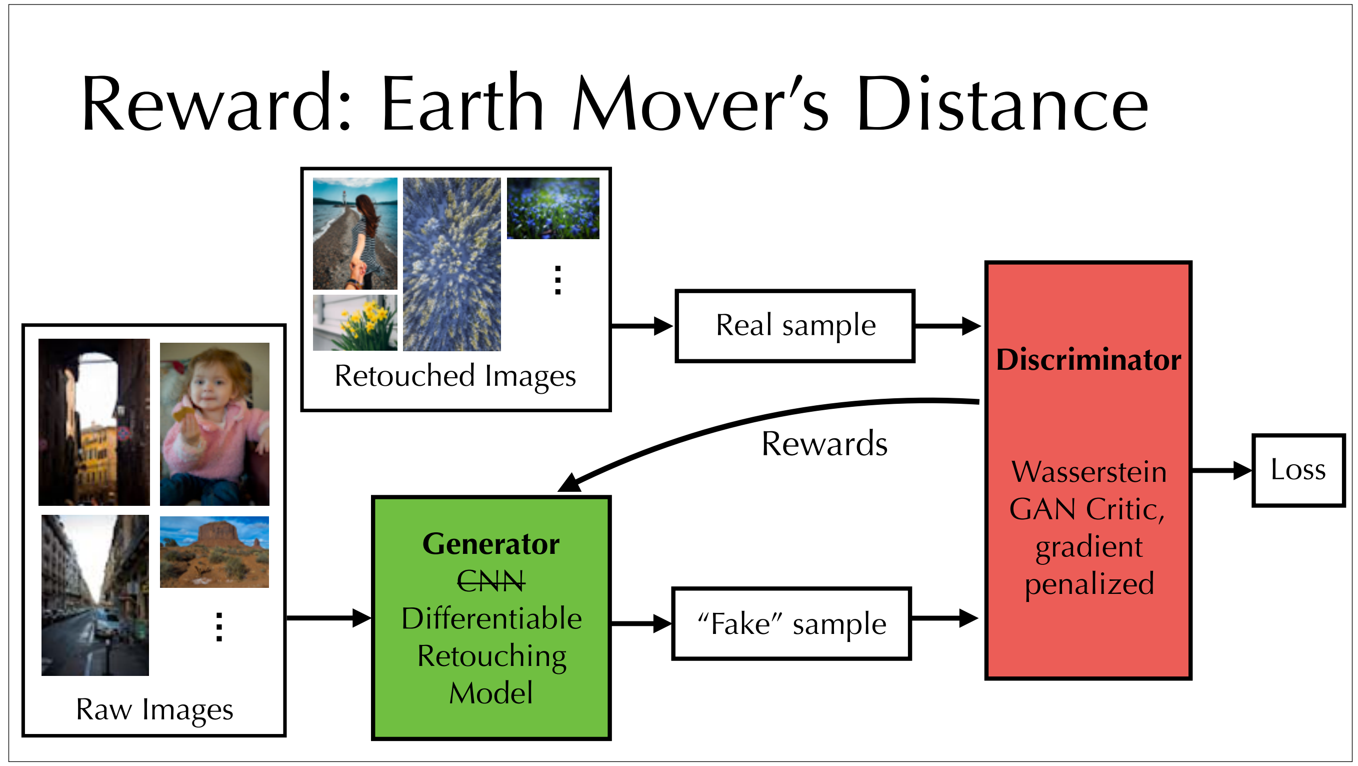


Such distance is computed with a conditional GAN. Here's a quick recap of GANs. A GAN consists of a generator and a discriminator.
The generator takes as input an raw image, and tries to convert it into a retouched one similar to the target retouched set, so that it can fool the discriminator. The discriminator, however, tries instead to distinguish the "fake" retouched images generated by the generator, from those "real" ones in the target image set. Equilibrium in this zero-sum game implies high-quality image generation.

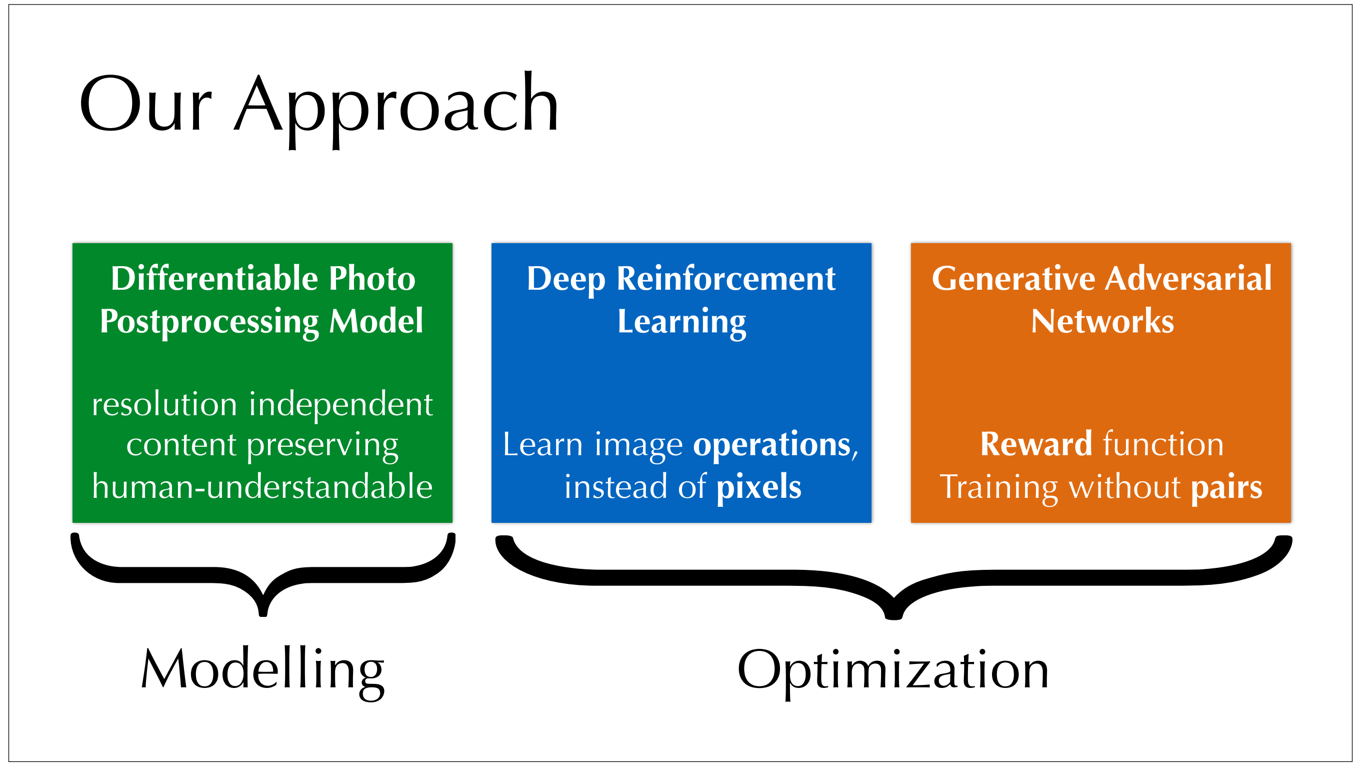
However, there is no guarantee that the generator preserves the content of the image, and in CycleGAN a mechanism called cycle consistency is used to enforce the preservation of content.



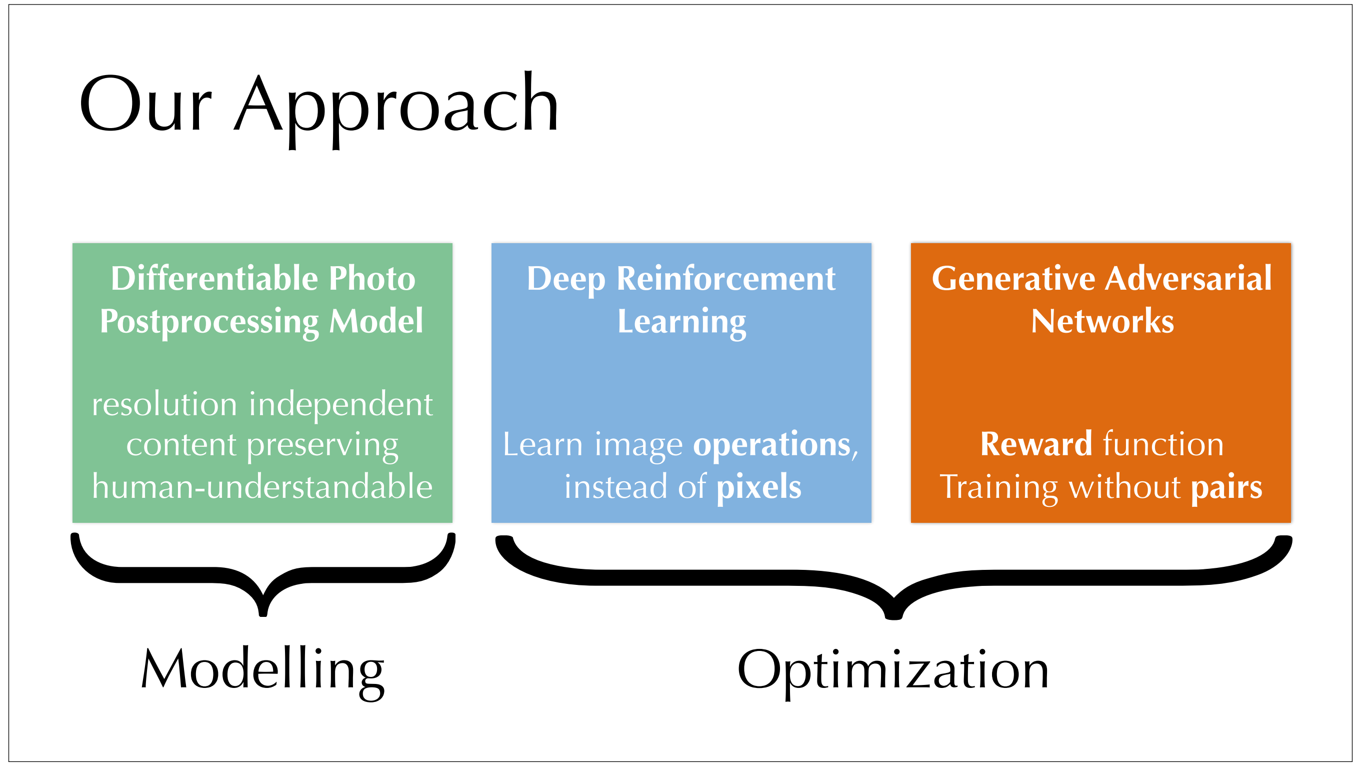
In our case, instead of using a CNN, we plug in our differentiable photo generation model into the generator. The discriminator provides an earth mover's distance estimate as a distance metric and reward [click] for reinforcement learning.



In our case, instead of using a CNN, we plug in our differentiable photo generation model into the generator. The discriminator provides an earth mover's distance estimate as a distance metric and reward [click] for reinforcement learning.



In summary, the GAN component acts as the reward function, allowing us to train without pairs. [Pause]. So that’s our whole system. Let look at some results.



In summary, the GAN component acts as the reward function, allowing us to train without pairs.
[Pause]. So that's our whole system. Let look at some results.

Results: Retouching and Stylisation

The main goal of the development is retouching and stylization. Here we show our step-wise retouching results learned from three artists [raise], one style per row. RAW images look dark on LDR displays, so this is what we started with.

Results: Retouching and Stylisation

Step one, the policy usually decides to brighten the images, via exposure or gamma adjustments.

Results: Retouching and Stylisation

Step two, the images are getting better

Results: Retouching and Stylisation

Step three

Results: Retouching and Stylisation

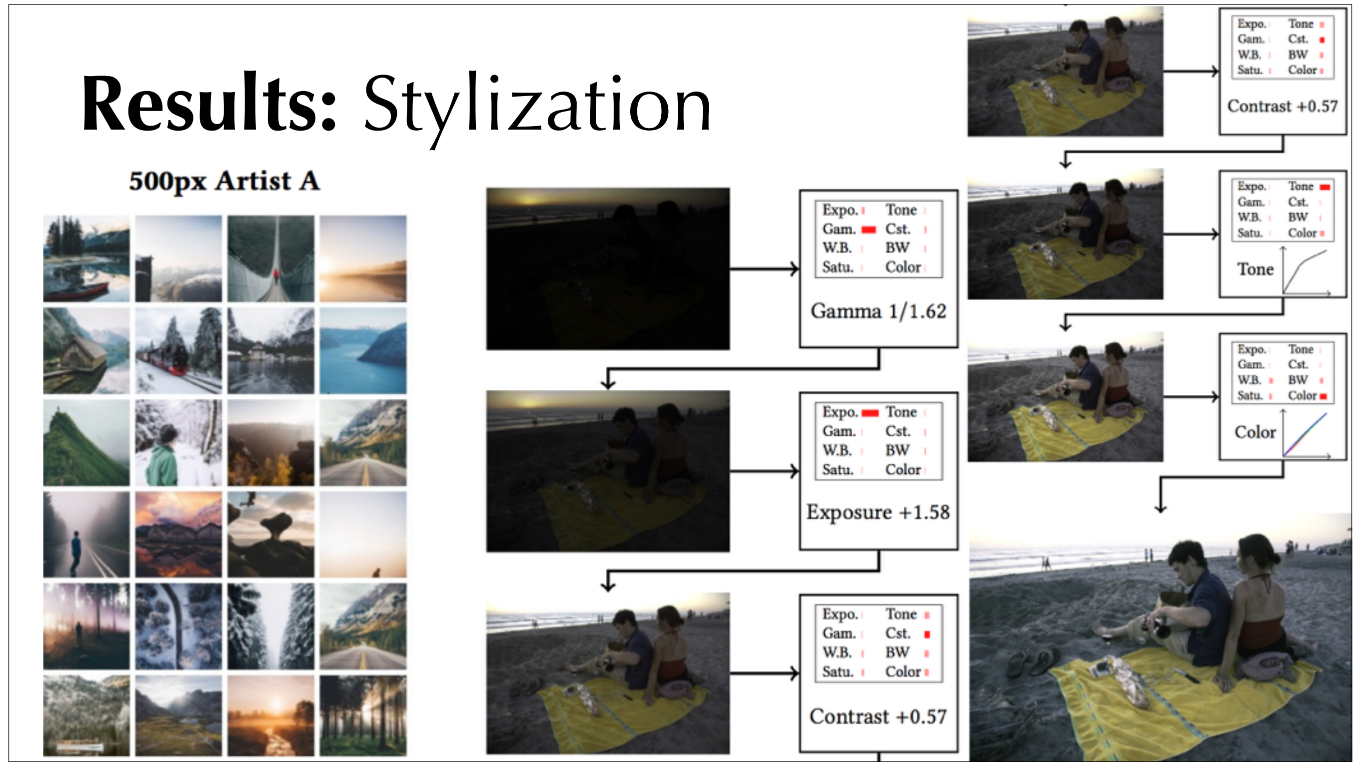
Step four

Results: Retouching and Stylisation

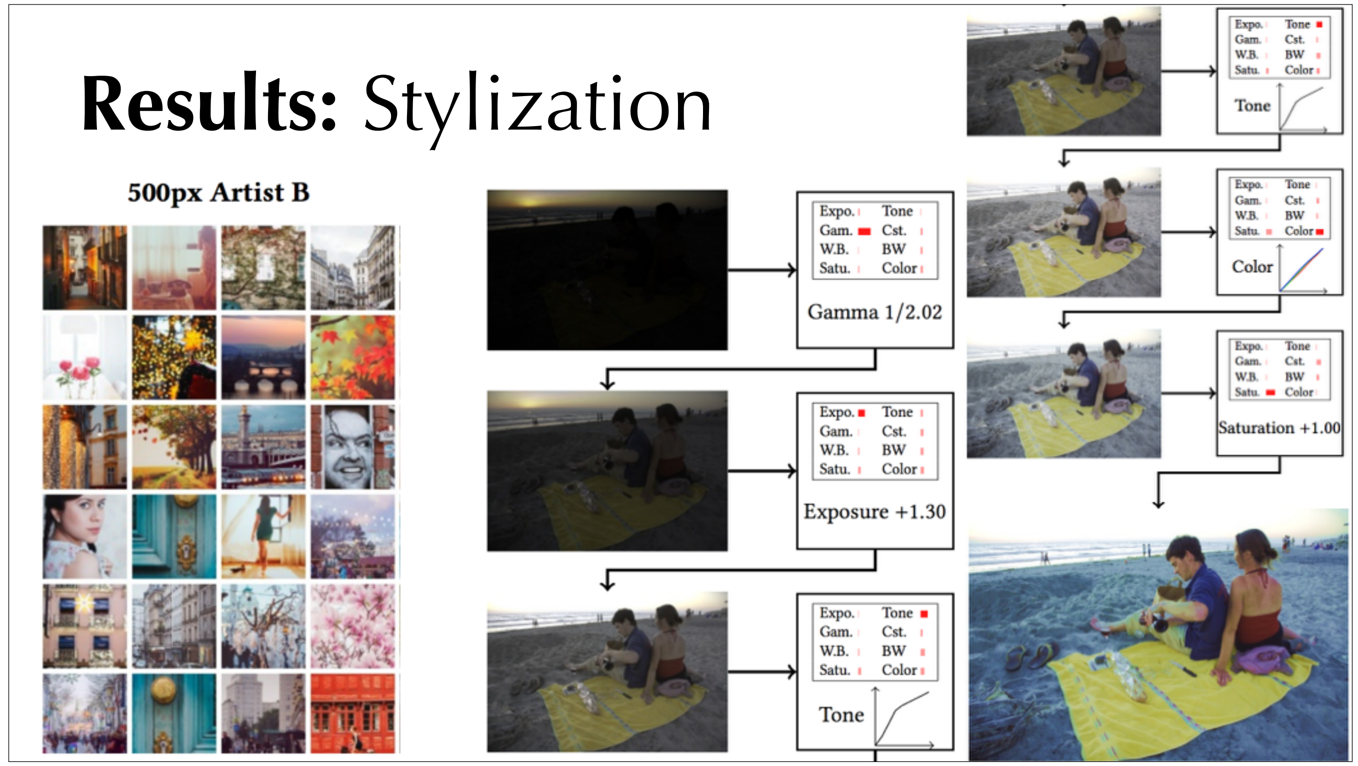
And finally, step five. This is what Exposure gives us.

Results: Retouching and Stylisation

In comparison, this is what naive tone-mapping gives.



Exposure can learn the unique style of artists. We pick two artists from 500px with different styles. Here we show the results and learned image operation sequences. Artist A tends to use less saturated colours and sadder mood.



In contrast, artist B likes saturated colours. This can be reflected in the last step, where saturation is maximized.

An “Infinite-Resolution” GAN

Moreover, our model is resolution-independent. This is the first GAN to the best of our knowledge that is practical for high-resolution image processing.

An “Infinite-Resolution” GAN

Moreover, our model is resolution-independent. This is the first GAN to the best of our knowledge that is practical for high-resolution image processing.

An “Infinite-Resolution” GAN

Since learning is easier in our framwork, we can even get better results than approaches needing much stronger supervision.

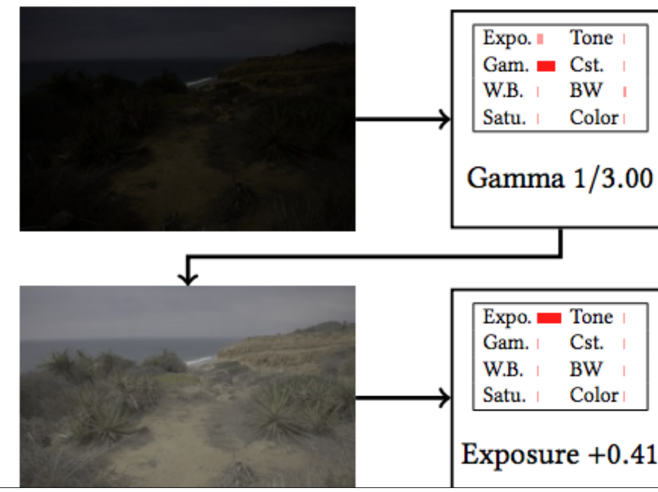
An
“Infinite-
Resolution”
GAN

Since learning is easier in our framwork, we can even get better results than approaches needing much stronger supervision.

Results: Reverse Engineering

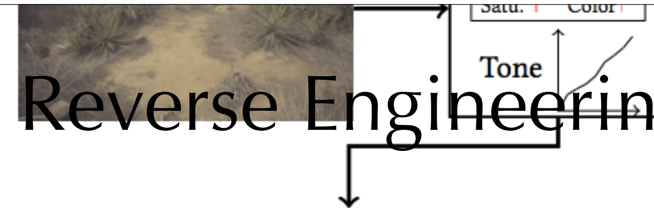
We can even reverse-engineer a certain artistic style. For example, this is a filter from Instagram with no explicit rule. [click] Our system can learn the underlying secrets given a bunch of output images from this black-box filter.

Results: Reverse Engineering

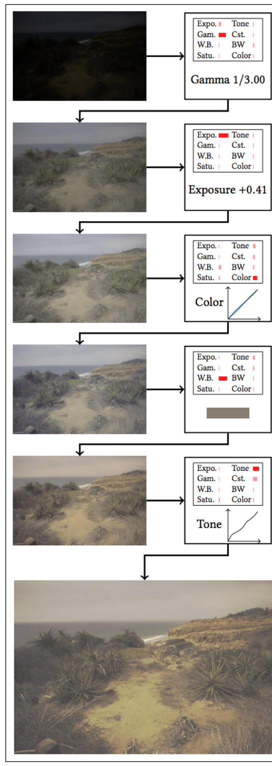


We can even reverse-engineer a certain artistic style. For example, this is a filter from Instagram with no explicit rule. [click] Our system can learn the underlying secrets given a bunch of output images from this black-box filter.

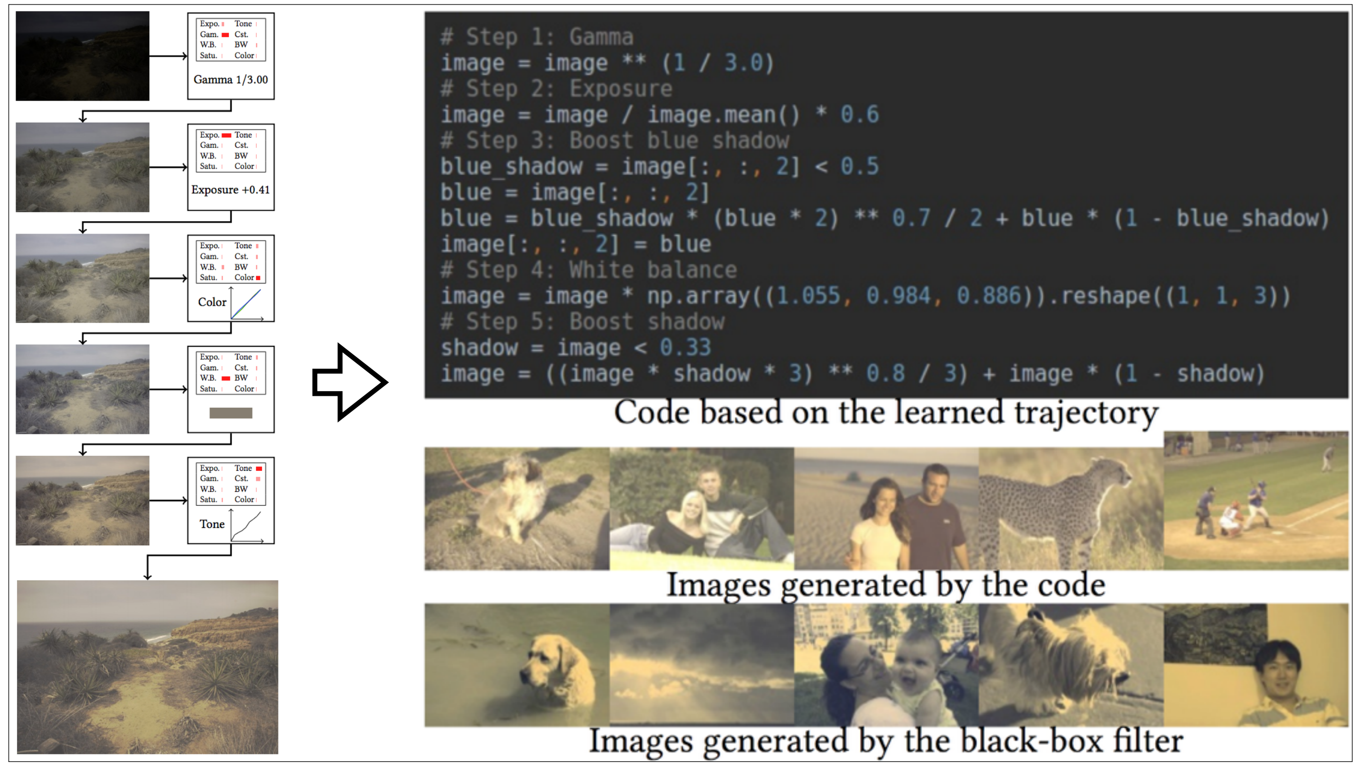
Results: Reverse Engineering



We can even reverse-engineer a certain artistic style. For example, this is a filter from Instagram with no explicit rule. [click] Our system can learn the underlying secrets given a bunch of output images from this black-box filter.



To prove that we have learned the style, we write [click] a piece of python code based on the operation sequence to mimic the style. As you can see, what we have generated using the python code is quite close to those from the black-box filter.



To prove that we have learned the style, we write [click] a piece of python code based on the operation sequence to mimic the style. As you can see, what we have generated using the python code is quite close to those from the black-box filter.

Summary

Differentiable Photo Postprocessing Model

resolution independent
content preserving
human-understandable

Deep Reinforcement Learning

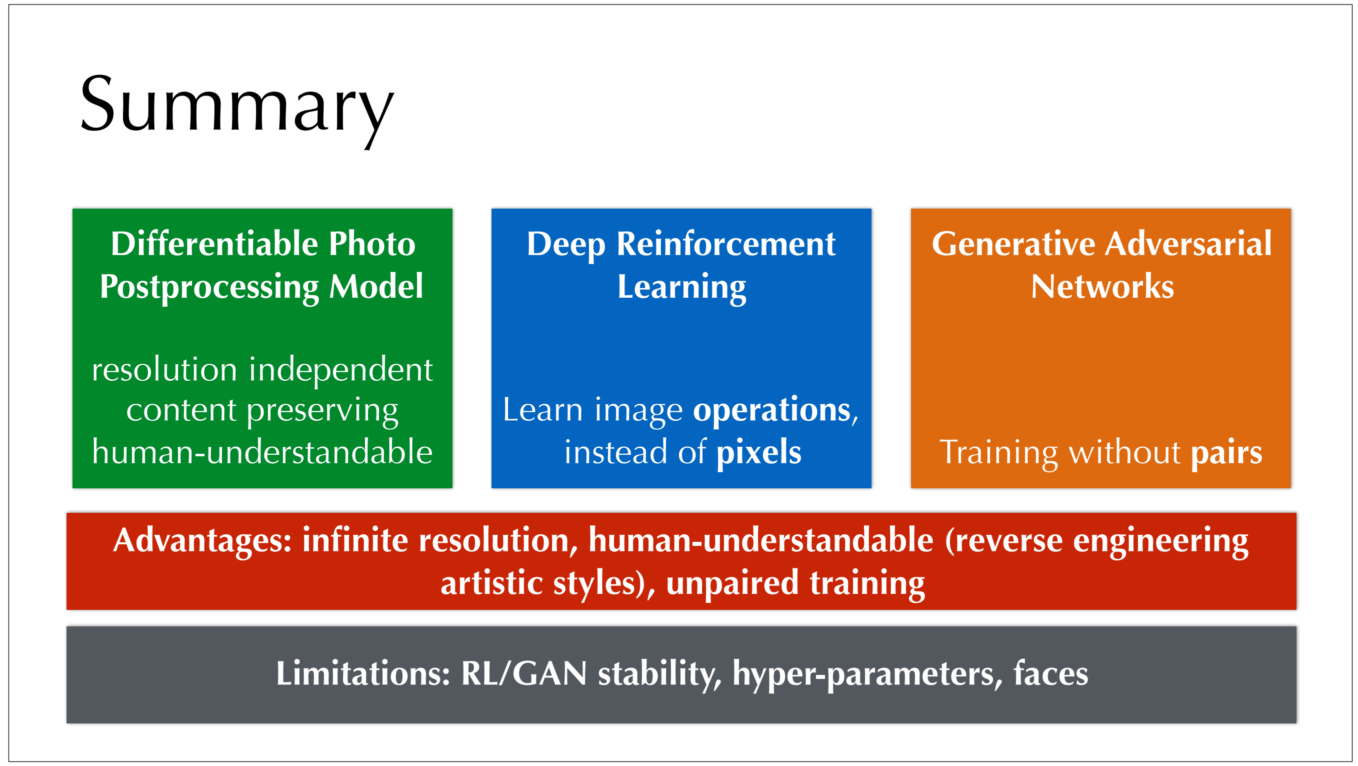
Learn image **operations**, instead of **pixels**

Generative Adversarial Networks

Training without **pairs**

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

The promising results have validated the effectiveness of our system. This is the first GAN, to the best of our knowledge, that has no limitation in resolution, and generates human-understandable results.



On the other hand, the system still has some limitations. For example, like many deep reinforcement learning or GAN approaches, stabilities and parameter-tuning is an inevitable issue. We also need some additional special treatments of human faces to grant this feature higher importance.

Retouch your photos with **Exposure**!

Input (tone-mapped) | Retouched | Input (tone-mapped) | Retouched | Input (tone-mapped) | Retouched

Reproducible research:
https://github.com/yuanming-hu/exposure

Finally, I also applied Exposure to my own photo collection, and it proves to work well.
The code and data are published on github and you are welcome to try it.
We hope that not only machines but also all interested people can understand the secrets of digital photography better, with the help of our “Exposure” system.
[click] That concludes my talk and I’m happy to take some questions. I would like to thank everyone who made this project possible, and thank you all for listening!

Finally, I also applied Exposure to my own photo collection, and it proves to work well.
The code and data are published on github and you are welcome to try it.
We hope that not only machines but also all interested people can understand the secrets of digital photography better, with the help of our “Exposure” system.
[click] That concludes my talk and I’m happy to take some questions. I would like to thank everyone who made this project possible, and thank you all for listening!