Seminar

Research Center for Technology and Art

" Real-Time User-Guided Image Colorization with Learned Deep Priors ", ACM SIGGRAPH 2017

https://arxiv.org/abs/1705.02999

IPHD Yang, YuanFu

Agenda

01 Art Statement

02 Method

03 Experiment Results

04 Connection&Demo

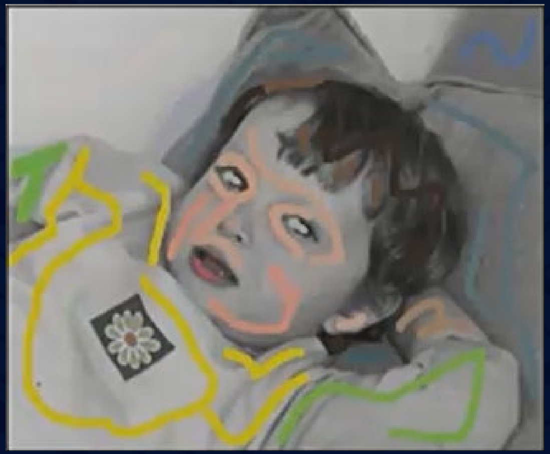
1 Art Statement

There is something uniquely and powerfully satisfying about the simple act of adding color to black and white imagery. Whether as a way of rekindling old, dormant memories or expressing artistic creativity, people continue to be fascinated by colorization.

Art Statement

- This paper propose a deep learning approach for user-guided image colorization. The system directly maps a grayscale image, along with sparse, local user "hints" to an output colorization with a Convolutional Neural Network (CNN). Rather than using hand-defined rules, the network propagates user edits by fusing low-level cues along with high-level semantic information, learned from large-scale data.
- To guide the user towards efficient input selection, the system recommends likely colors based on the input image and current user inputs. The colorization is performed in a single feed-forward pass, enabling real-time use. Even with randomly simulated user inputs, it show that the proposed system helps novice users quickly create realistic colorizations, and offers large improvements in colorization quality with just a minute of use. In addition, authors demonstrate that the framework can incorporate other user "hints" to the desired colorization, showing an application to color histogram transfer.

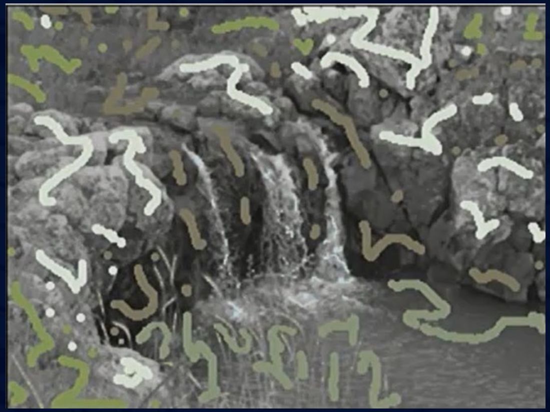
Art Statement



Grayscale image + user strokes

Colorization

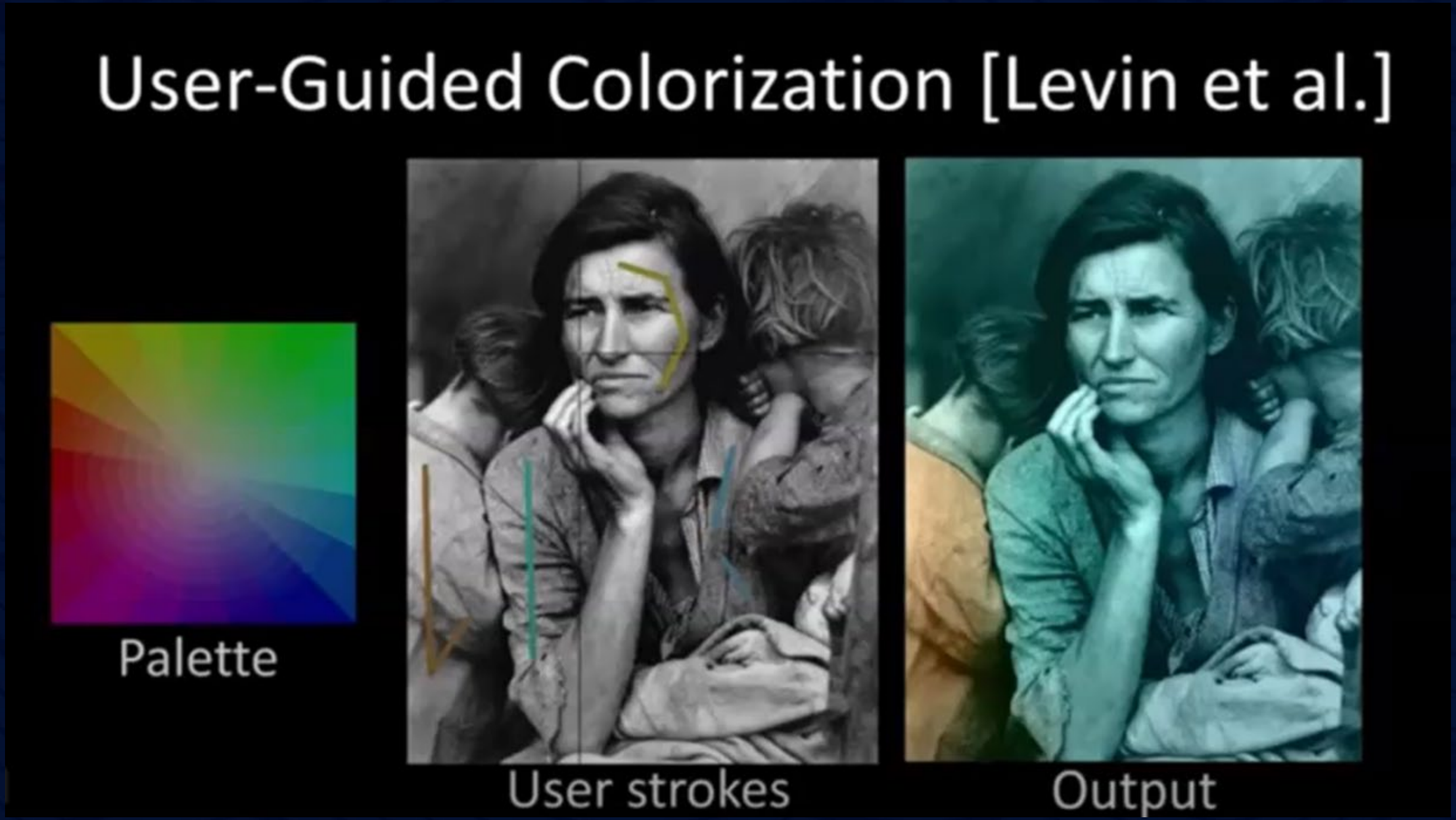
Hertzmann et al., SIGGRAPH, 2001.
Welsh et al., ACM Transactions on Graphics, 2002.
Irony et al., Eurographics, 2005.
Liu et al., ACM Transactions on Graphics, 2008.
Wang et al., ACM Transactions on Graphics, 2010.
Chia et al., ACM Transactions on Graphics, 2011.
Gupta et al., ACM Multimedia, 2012.
Chang et al., ACM Transactions on Graphics, 2015.



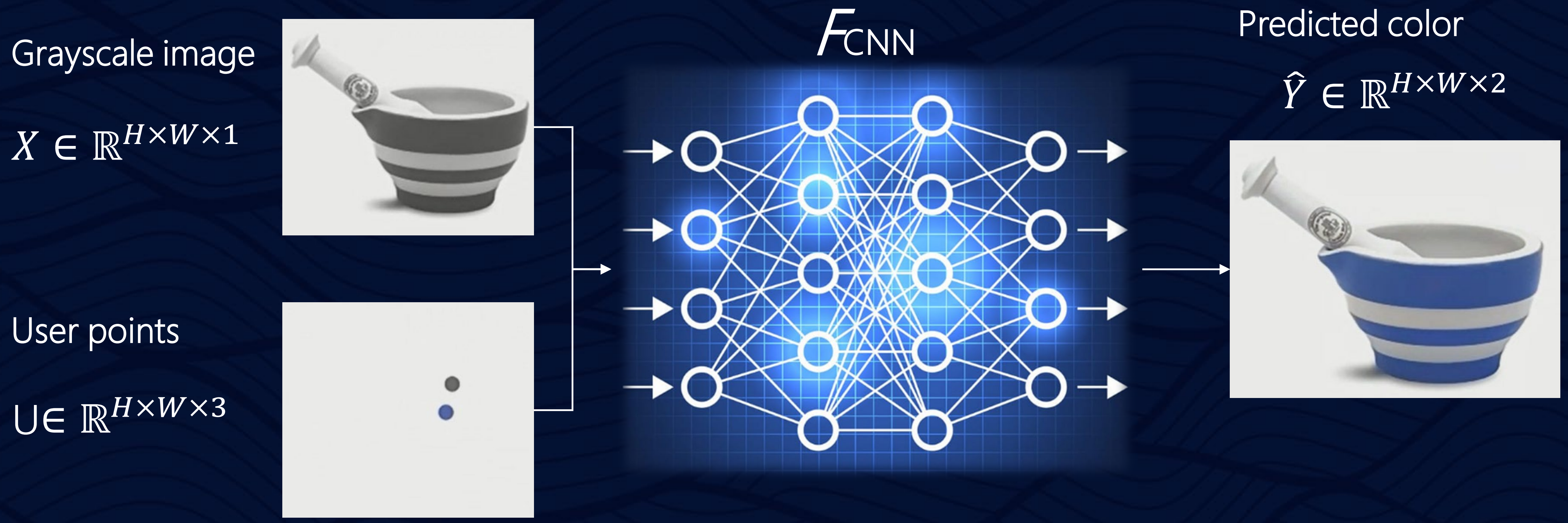
Many user strokes often needed
➔ Desired: Learn natural image priors and edit propagation from large-scale data

~ 80 input strokes/points

Art Statement



Art Statement

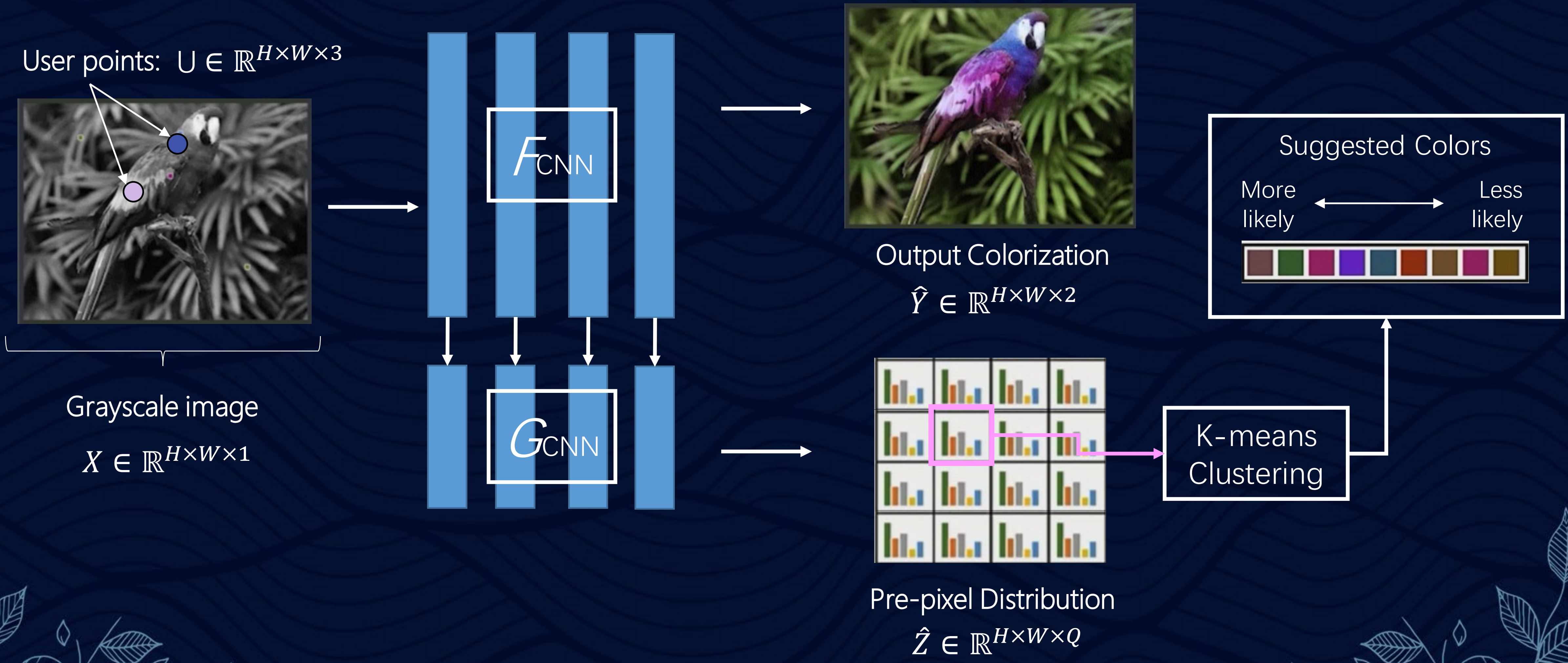


2 Method

We train a deep network to predict the color of an image, given the grayscale version and user inputs. First, we describe the objective of the network. Second, we describe the two variants of our system (i) the Local Hints Network, which uses sparse user points, and (ii) the Global Hints Network, which uses global statistics. Finally, we define our network architecture.

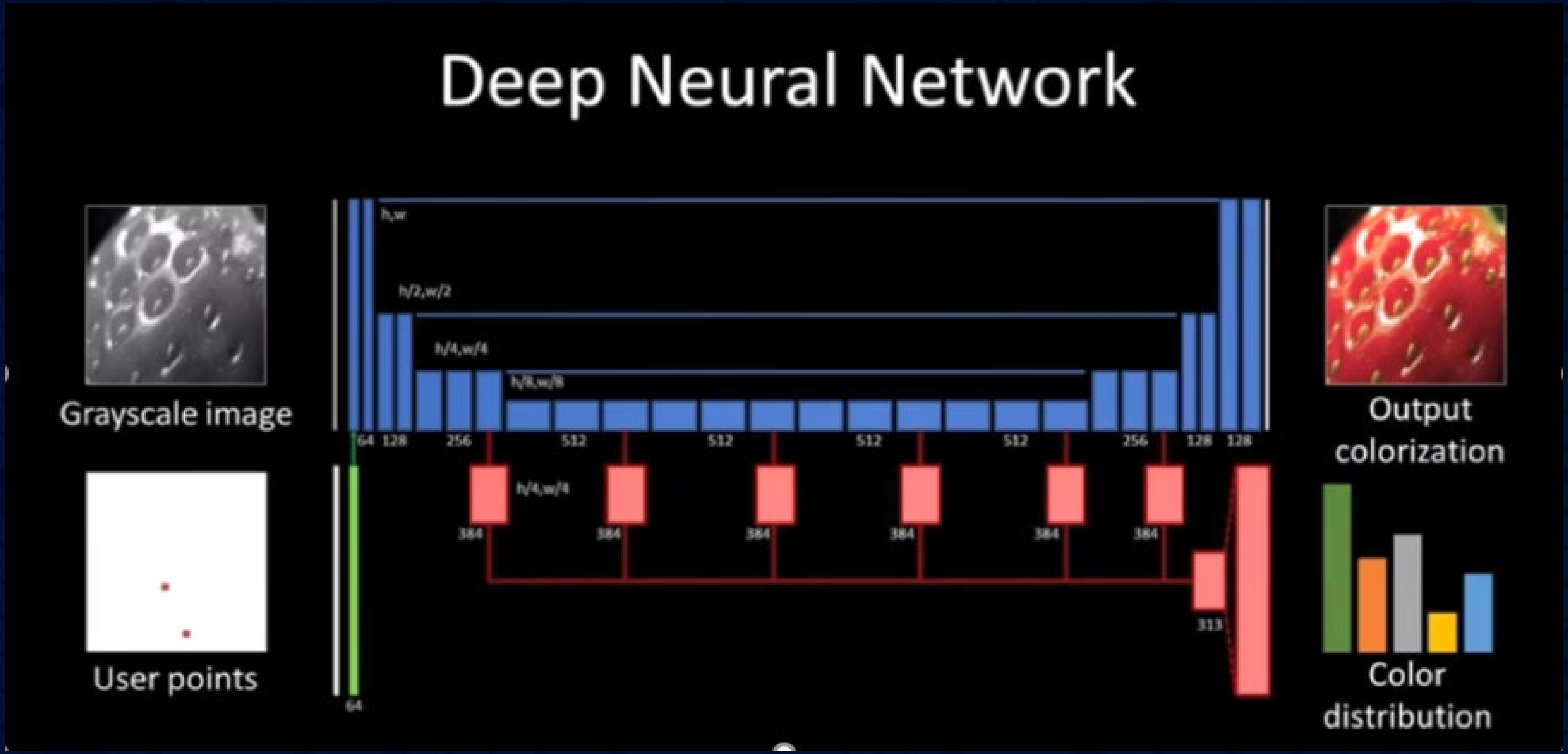
Method – Architecture

- Deep Neural Networks of User-Guided Image Colorization:



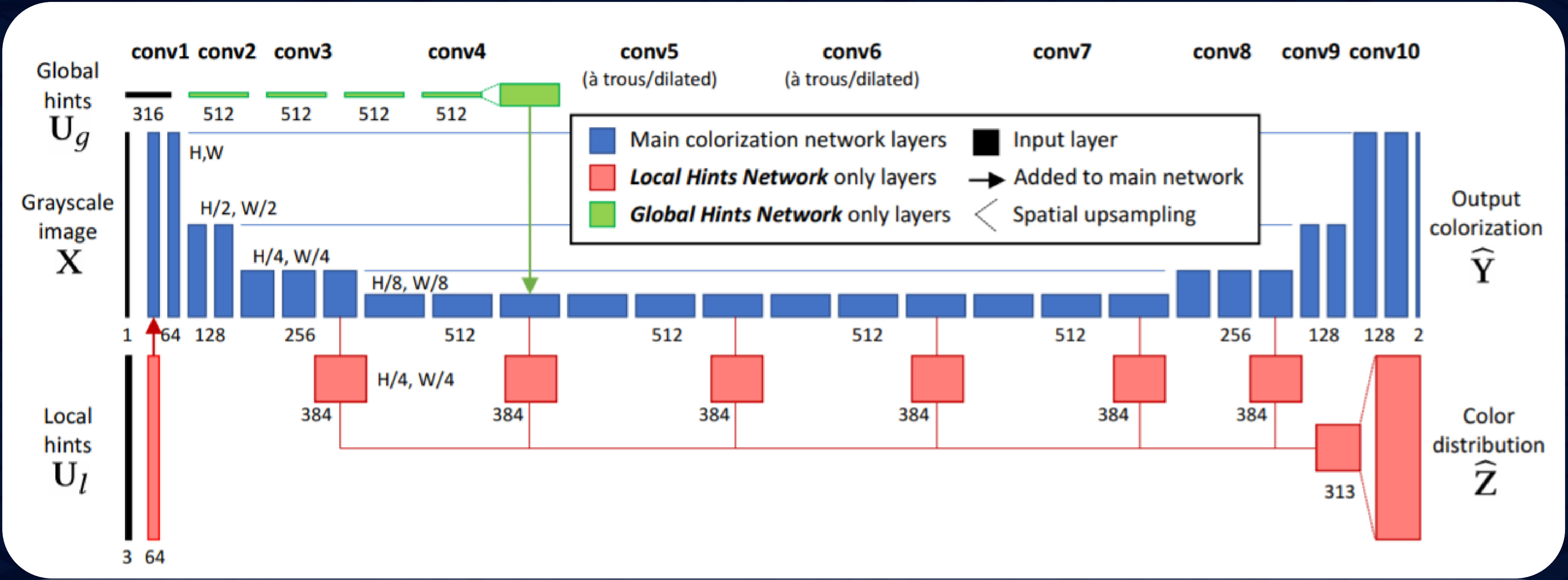
Method – Architecture

- Deep Neural Networks of User-Guided Image Colorization:



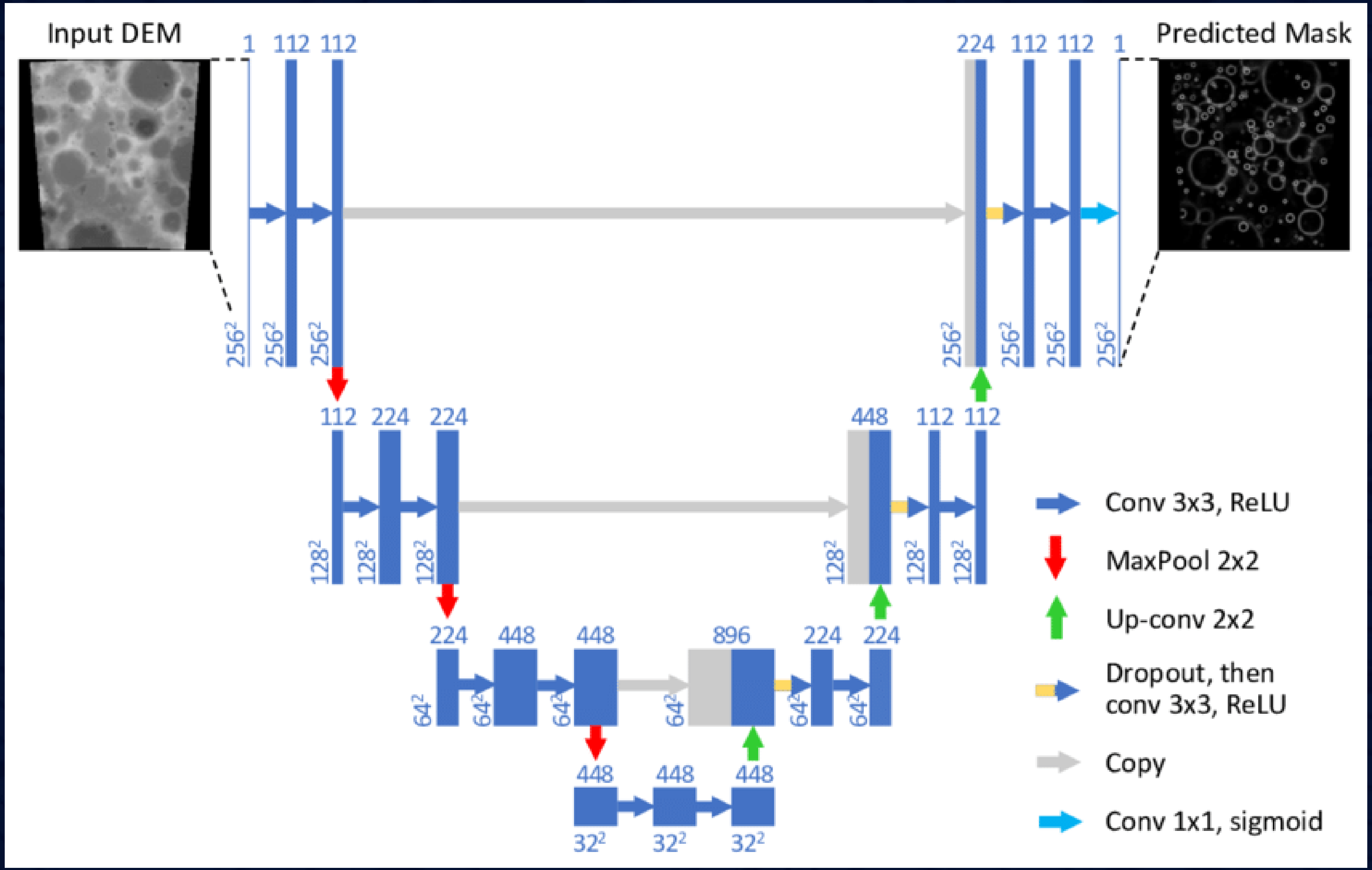
Method – Architecture

- Deep Neural Networks of User-Guided Image Colorization:
  This study use a hyper column approach (Hariharan et al., 2015; Larsson et al., 2016) by concatenating features from multiple layers of the main branch, and learning a two-layer classifier on top.



Method – U-Net

- U-Net Introduction:



O. Ronneberger, P. Fischer and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", Lecture Notes in Computer Science Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, pp. 234-241, 2015.

Method – Algorithm

- Learning to Colorize:

Training Data

X $\quad$ Y

Automatic $\quad$ $\theta = arg\min_{\theta} \mathcal{L}(\mathcal{F}(X), Y)$

User-guided $\quad$ $\theta = arg\min_{\theta} \mathcal{L}(\mathcal{F}(X, U), Y)$

Randomly Simulated User Interactions $\quad$ $\theta = arg\min_{\theta} \mathcal{L}(\mathcal{F}(X, Hints(Y)), Y)$

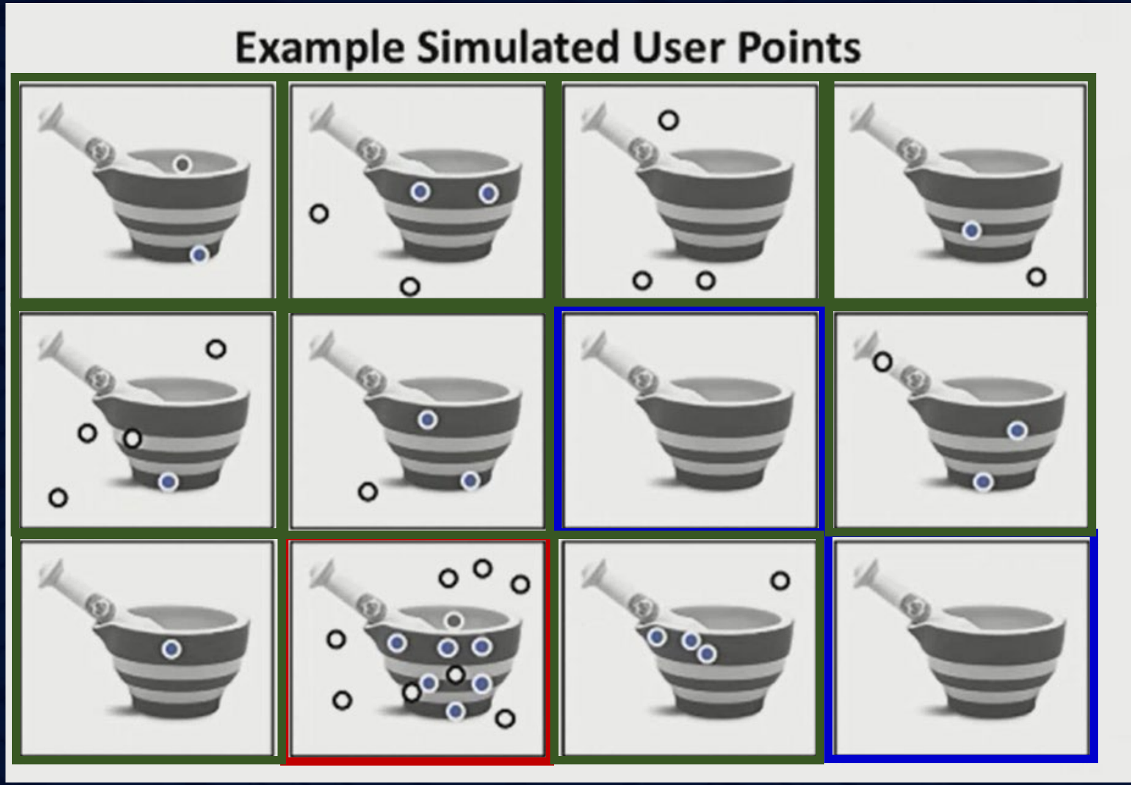
⋮

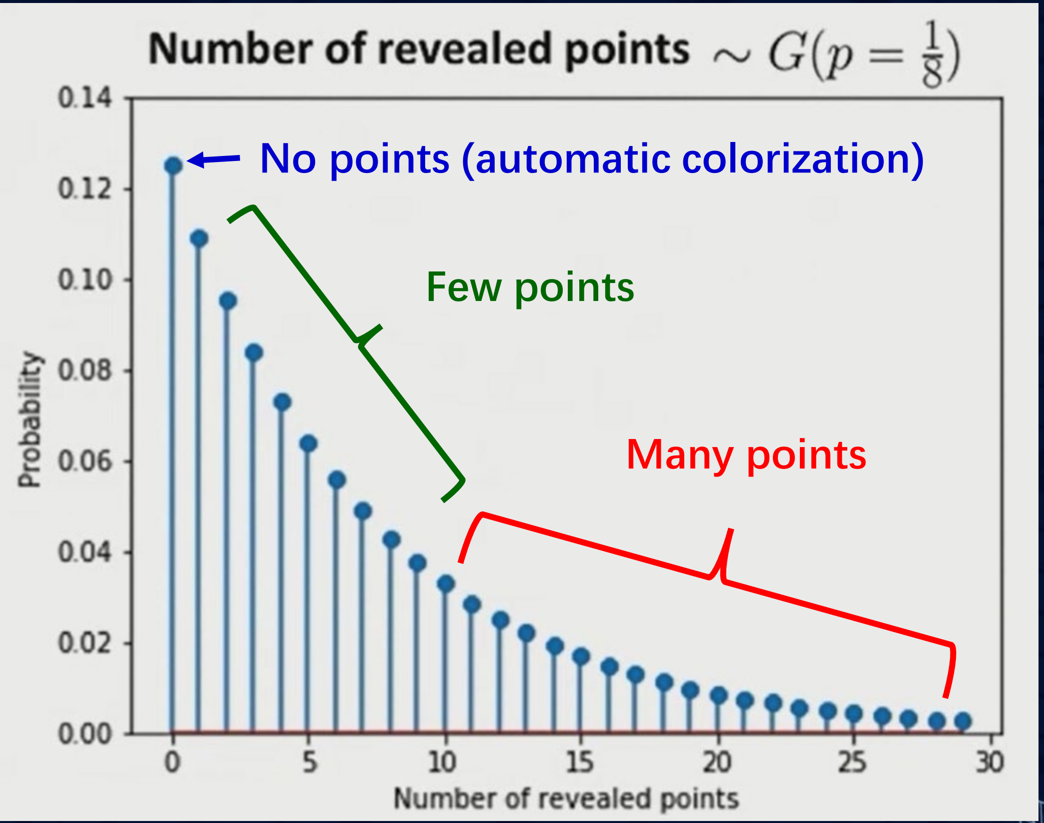
ImageNet Database (1.3M train/10k validation/10k test)

Method – Algorithm

- Randomly Simulated User Interactions:

$$\theta = arg\min_{\theta} \mathcal{L}(\mathcal{F}(X, Hints(Y)), Y)$$

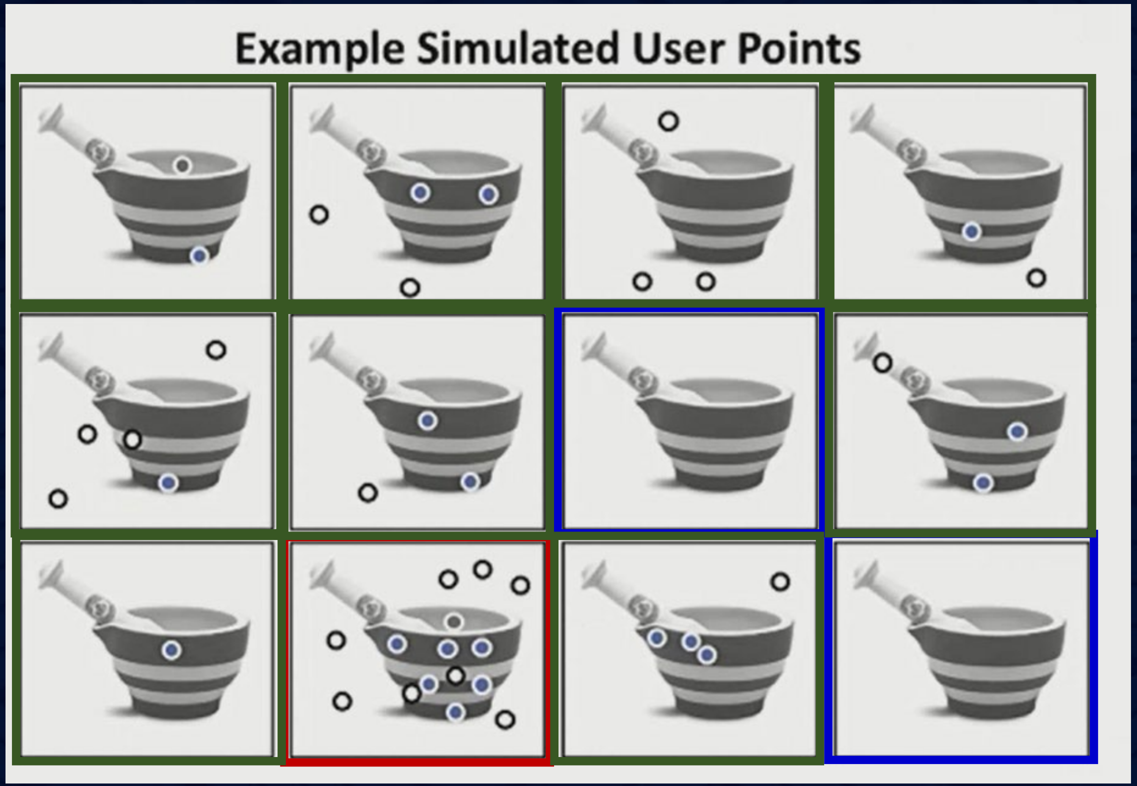


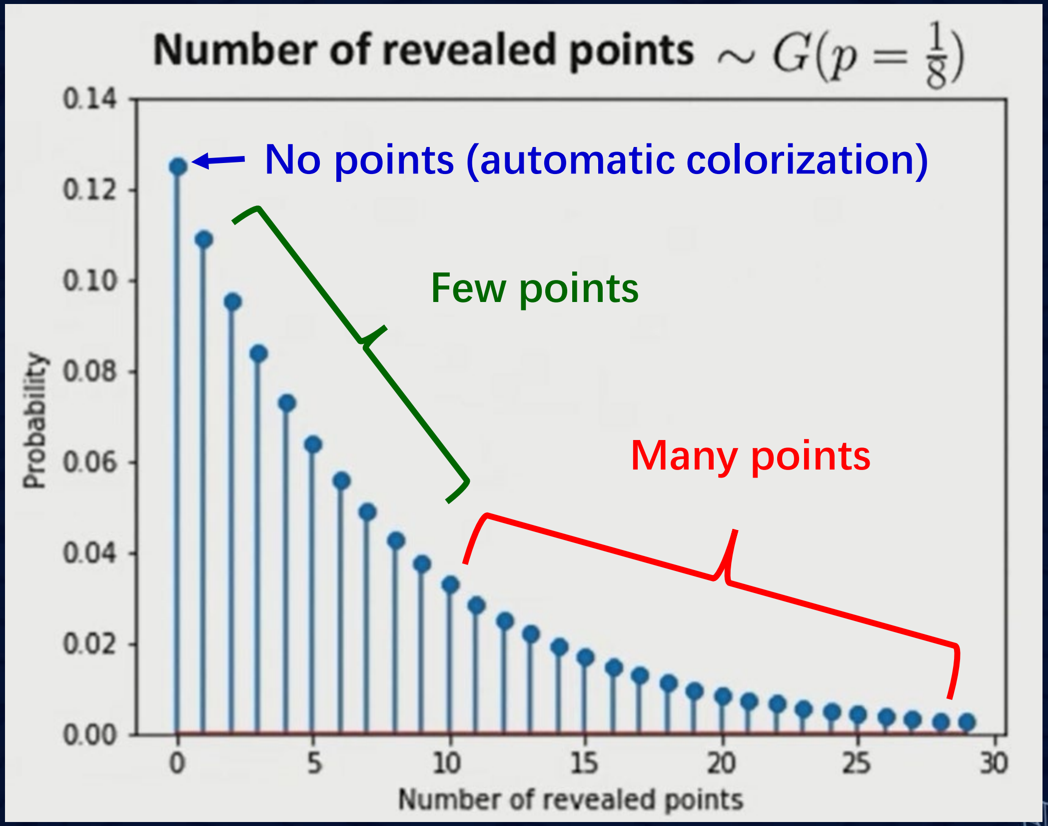


Method – Algorithm

- Randomly Simulated User Interactions:

$$\theta = arg\min_{\theta} \mathcal{L}(\mathcal{F}(X, Hints(Y)), Y)$$





Method – Architecture

- Learning to Colorize:

The inputs to our system are a grayscale image X ∈ R^(H×W×1), along with an input user tensor U. The grayscale image is the L, or lightness in the CIE Lab color space, channel. The output of the system is Y ∈ R^(H×W×2), the estimate of the ab color channels of the image. The mapping is learned with a CNN F, parameterized by θ, with the network architecture specified shown in below Figure.

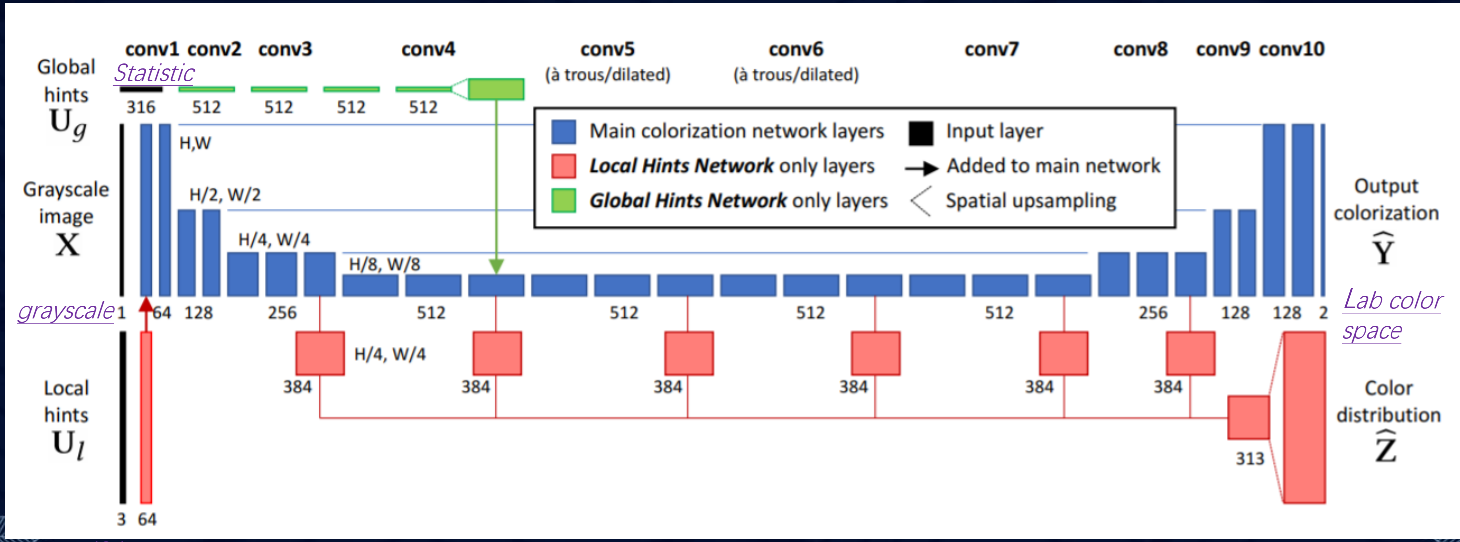
$$\theta^* = arg\min_{\theta} \mathbb{E}_{X,U,Y\sim\mathcal{D}}[\mathcal{L}(\mathcal{F}(X,U;\theta),Y)]$$

Across D, argue min Loss Function L, estimate θ of CNN function F with input tensor U

$$U_l = P_l(Y),\ U_g = P_g(Y),$$

Ul Local hints network, Ug: Global hints network.
Ul and Ug are generated by giving the network a "peek", or projection, of the ground truth color Y using functions Pl and Pg, respectively.

$X \in \mathbb{R}^{H\times W\times 1}$

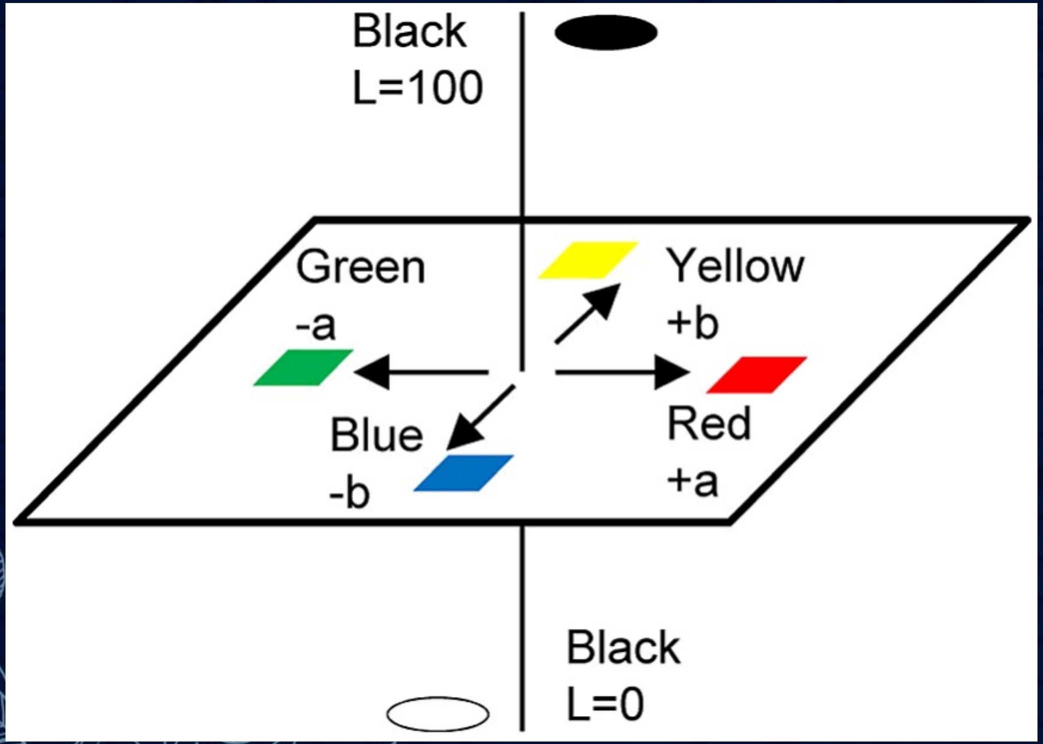


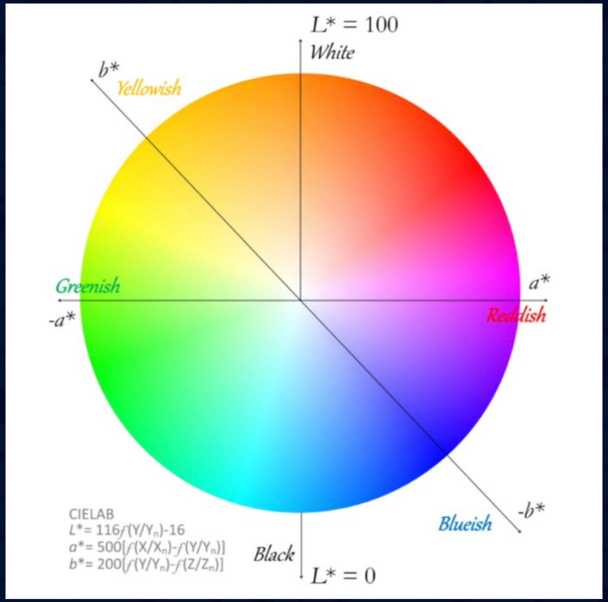
R/G/B

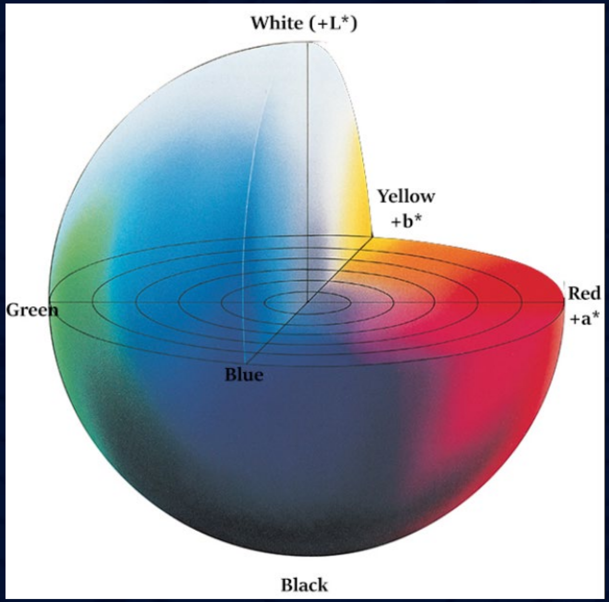
$\hat{Y} \in \mathbb{R}^{H\times W\times 2}$

Method – Architecture

- CIE Lab color space:
  - ✓ CIE: International Commission on Illumination (國際照明委員會)
  - ✓ L channel : Lightness
  - ✓ a channel : where negative values indicate green and positive values indicate red
  - ✓ b channel : where negative values indicate blue and positive values indicate yellow







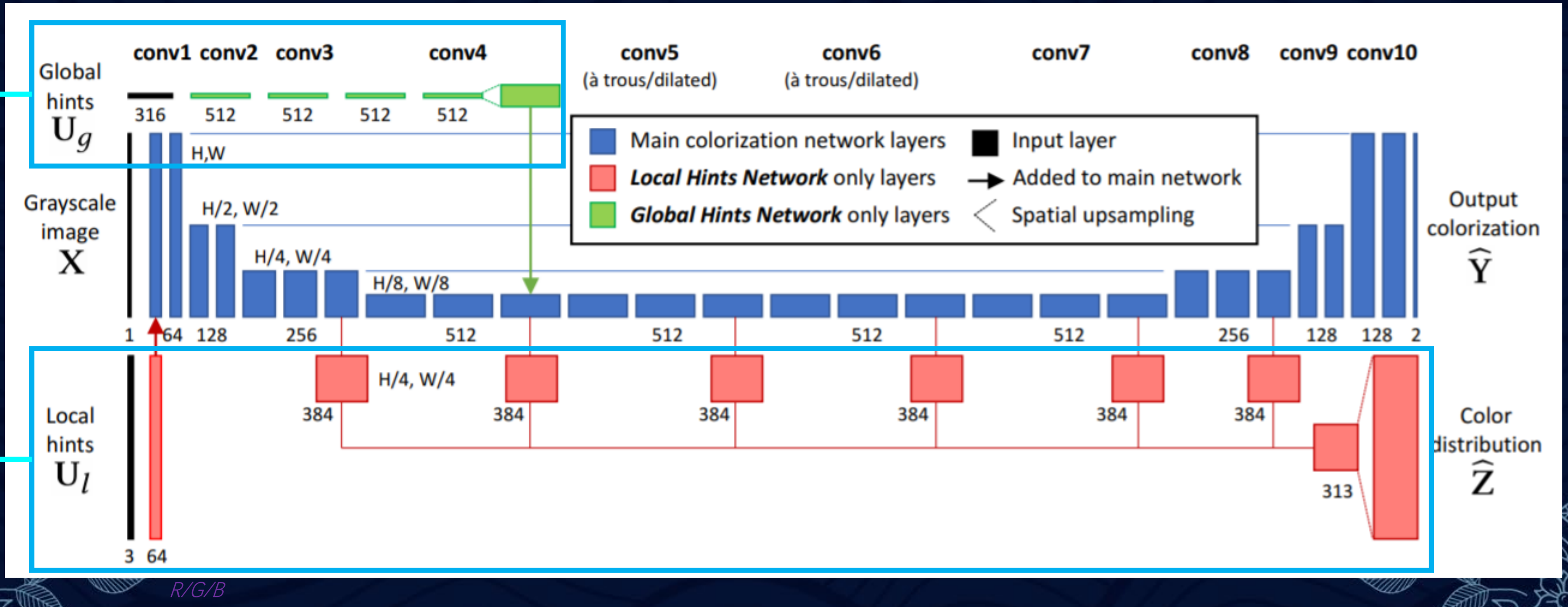
Method – Architecture

- Learning to Colorize:

The minimization problems for the Local and Global Hints Networks are then described below in below Equation. Because they are using functions Pl, Pg to synethtically generate user inputs, our dataset only needs to contain grayscale and color images. They use the 1.3M ImageNet dataset (Russakovsky et al., 2015).

$$\theta_l^* = arg\min_{\theta_l} \mathbb{E}_{X,Y\sim\mathcal{D}}[\mathcal{L}(F_l(X,U_l;\theta_l),Y)] \quad \theta_g^* = arg\min_{\theta_g} \mathbb{E}_{X,Y\sim\mathcal{D}}[\mathcal{L}(F_g(X,U_g;\theta_g),Y)]$$

Across D, argue min Loss Function L, estimate θ of CNN function F with input tensor U



Method – Loss Function

- Smooth L1 Loss:

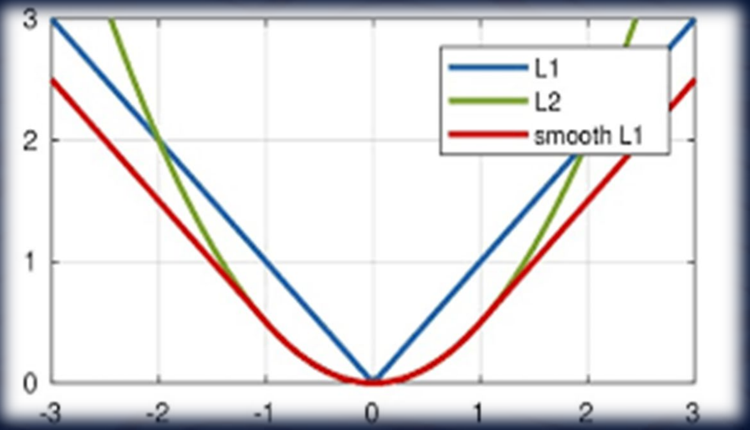
The smooth-ℓ1 is a robust estimator (Huber, 1964), which can help avoid the averaging problem. The loss function ℓδ is evaluated at each pixel and summed together to evaluate the loss L for a whole image. In addition, using a regression loss, described in below Equation with δ=1, enables us to perform end-to-end learning without a fixed inference step.

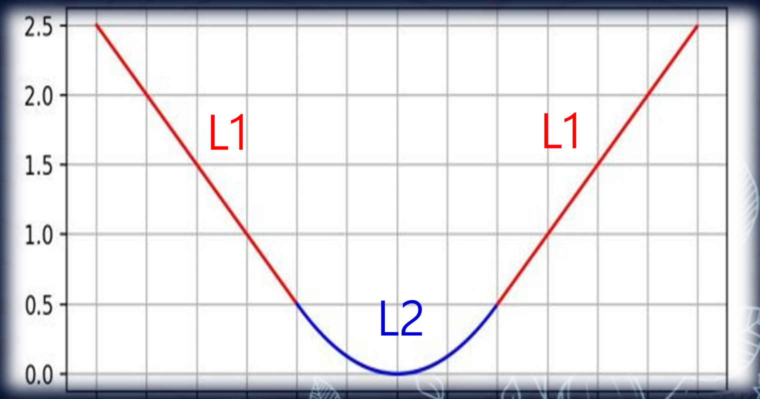
$$\mathcal{L}(\mathcal{F}(X,U;\theta),Y)=\sum_{h,w}\sum_{q}\ell_{\delta}(\mathcal{F}(X,U;\theta)_{h,w,q},Y_{h,w,q})$$

$$\ell_{\delta}(x,y)=\frac{1}{2}(x-y)^2\mathbb{1}_{\{|x-y|<\delta\}}+\delta(|x-y|-\frac{1}{2}\delta)\mathbb{1}_{\{|x-y|\geq\delta\}}$$

δ=1

Loss Function	Derivative of Loss Function
$L_2(x)=x^2$	$\frac{dL_2(x)}{dx}=2x$
$L_1(x)=\lvert x\rvert$	$\frac{dL_1(x)}{dx}=\begin{cases}1, if\ x\geq 0\\ -1, otherwise\end{cases}$
$smoothL_1(x)=\begin{cases}0.5x^2, if\lvert x\rvert<1\\ \lvert x\rvert-0.5, otherwise\end{cases}$	$\frac{dsmooth_{L1}(x)}{dx}=\begin{cases}x, if\ \lvert x\rvert<0\\ \pm 1, otherwise\end{cases}$



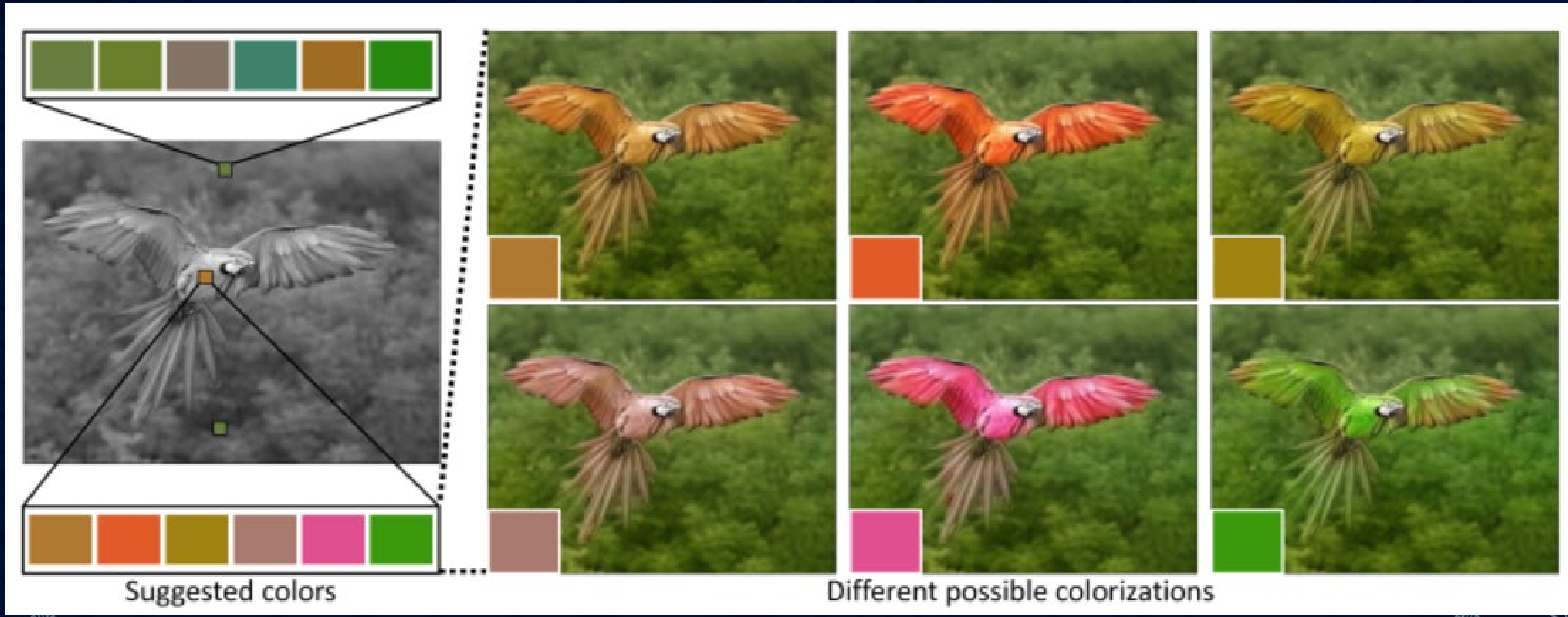


Method – User interface

- Suggested Palette:

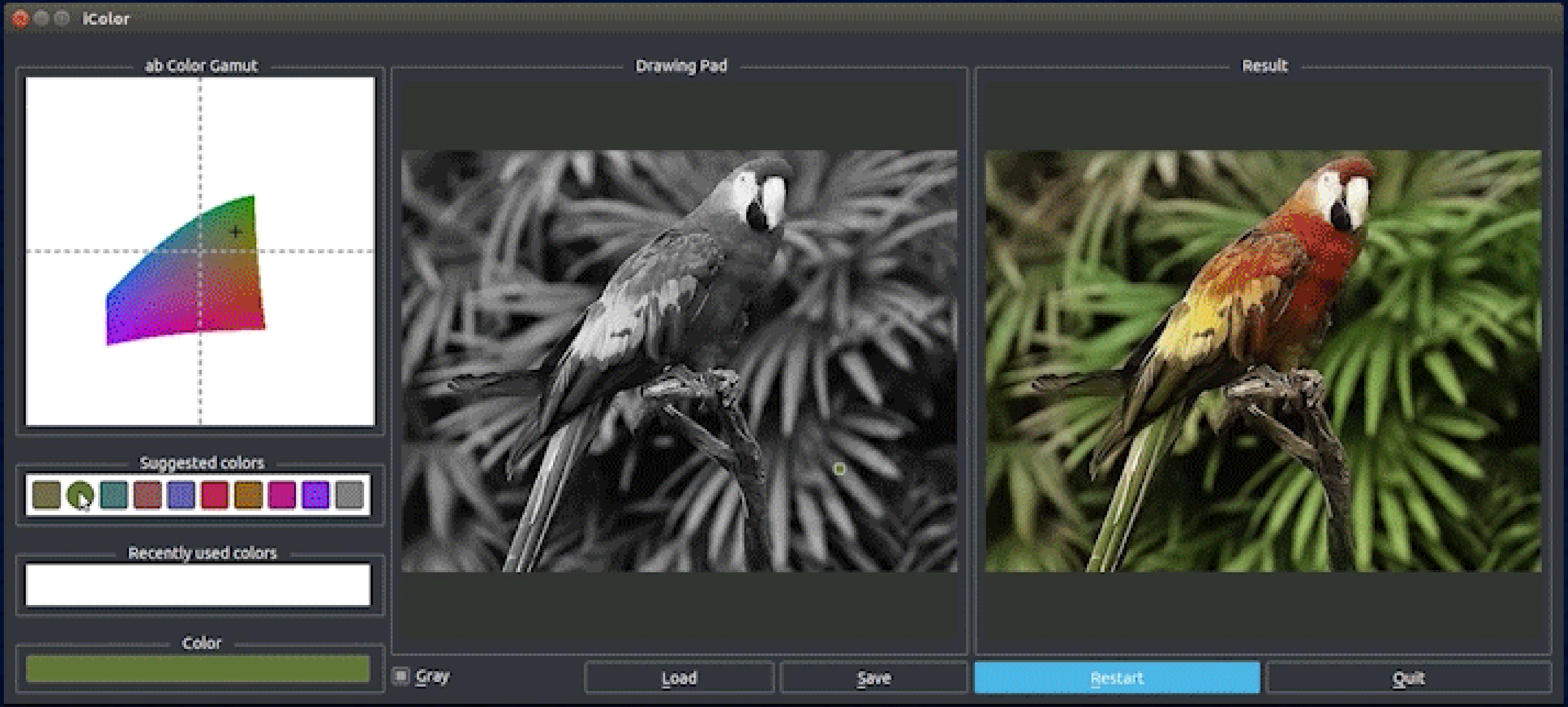
  The interface consists of a drawing pad, showing user points overlaid on the grayscale input image, a display updating the colorization result in real-time, a data-driven color palette that suggests likely color for a given location (as shown in below Figure), and a regular CIE Lab gamut based on the lightness of the current point. A user is always free to add, move, delete, or change the color of any existing points.

  In this example, it show first suggested colors on the background vegetation (top palette), sorted by decreasing likelihood. The suggested colors are common colors for vegetation. it also show the top six suggested colors (bottom palette) of a pixel on the image of the bird. On the right, it show the resulting colorizations, based on the user selecting these top six suggested colors.



Method – User interface

- Suggested Palette:



3 Experiment Results

Detail qualitative and quantitative experiments with proposed method. First, automatically test the Local Hints Network. Second, describe user study. Then, show qualitative examples on unusual colorizations. Finally, evaluate the Global Hints Network.

Experiment Results

- Metric Introduction: PSNR (Peak Signal-to-Noise Ratio)

✓ PSNR (for signal channel):

Given a noise-free m×n monochrome image I and its noisy approximation K, MSE is defined as:

$$\mathrm{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2$$

The PSNR (in dB) is defined as:

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20 \cdot \log_{10}(\frac{MAX_I}{\sqrt{MSE}})$$

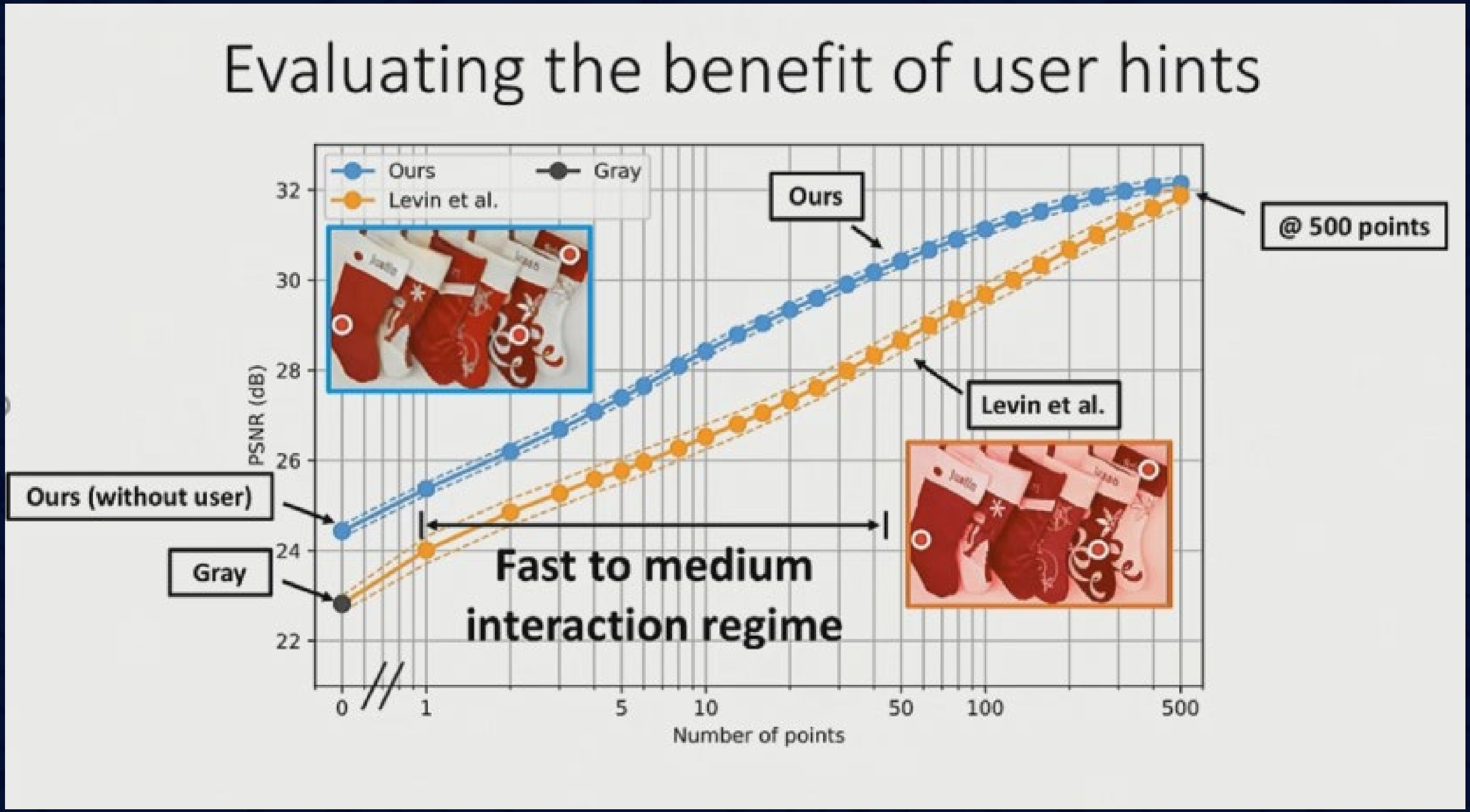
Here, MAXI is the maximum possible pixel value of the image. When the pixels are represented using 8 bits per sample, this is 255. More generally, when samples are represented using linear PCM with B bits per sample, MAXI is 2B−1.

✓ PSNR (for R/G/B channel):

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{\frac{1}{3mn}\sum_{R,G,B}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I_{color}(i,j) - K_{color}(i,j)]^2}\right)$$

Experiment Results

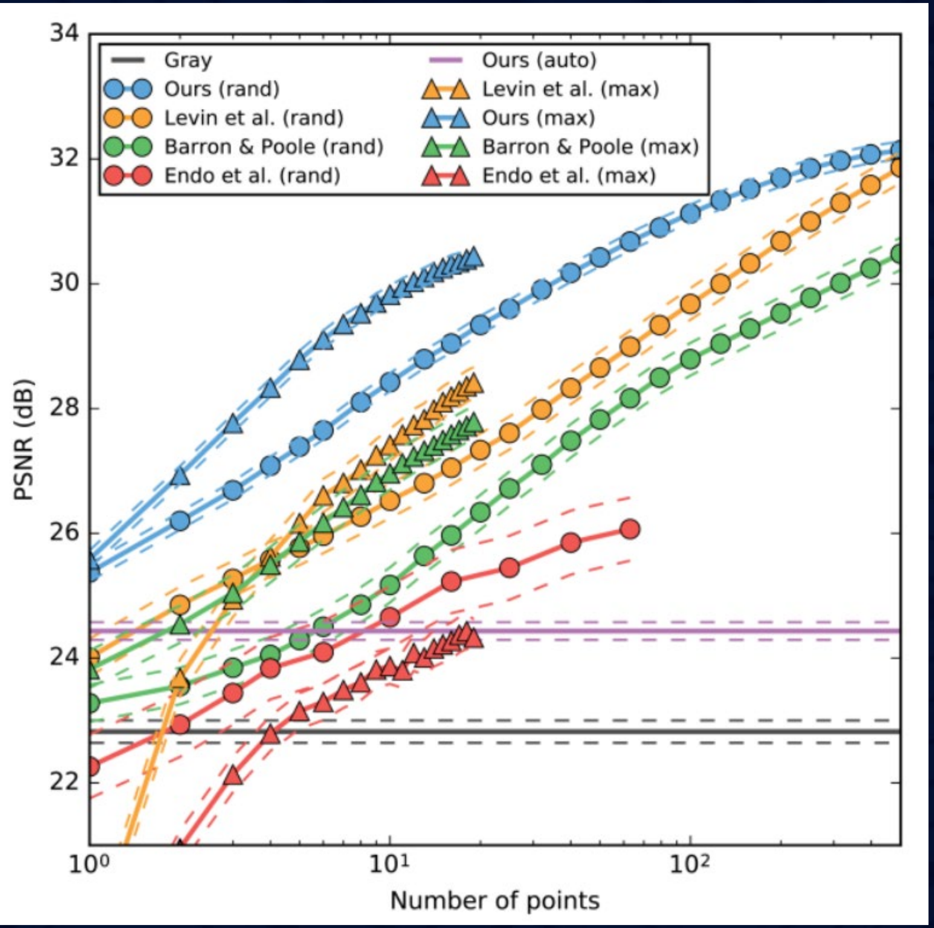
- Ablation Study:



Experiment Results

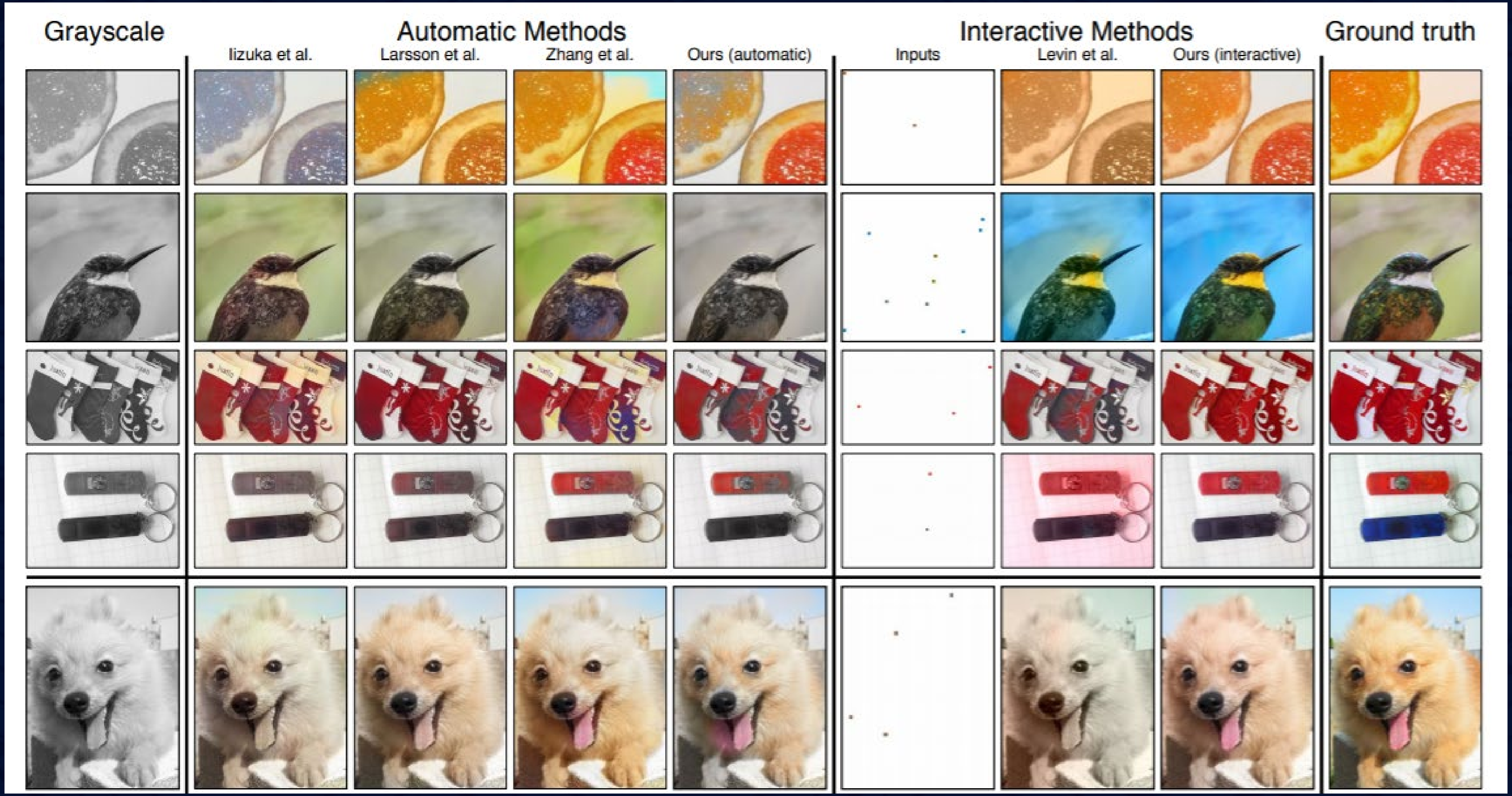
- Performance Comparison of 5 methods:

Method	Added Inputs	PSNR (dB)
Predict gray	–	22.82±0.18
Zhang et al. (2016)	automatic	22.04±0.11
Zhang et al. (2016) (no-rebal)	automatic	24.51±0.15
Larsson et al. (2016)	automatic	24.93±0.14
Iizuka et al. (2016)	automatic	23.69±0.13
Ours (Local)	automatic	24.43±0.14
Ours (Global)	+ global hist	27.85±0.13
Ours (Global)	+ global sat	25.78±0.15
Ours (Local)	+ gt colors	37.70±0.14
Edit propagation	+ gt colors	∞



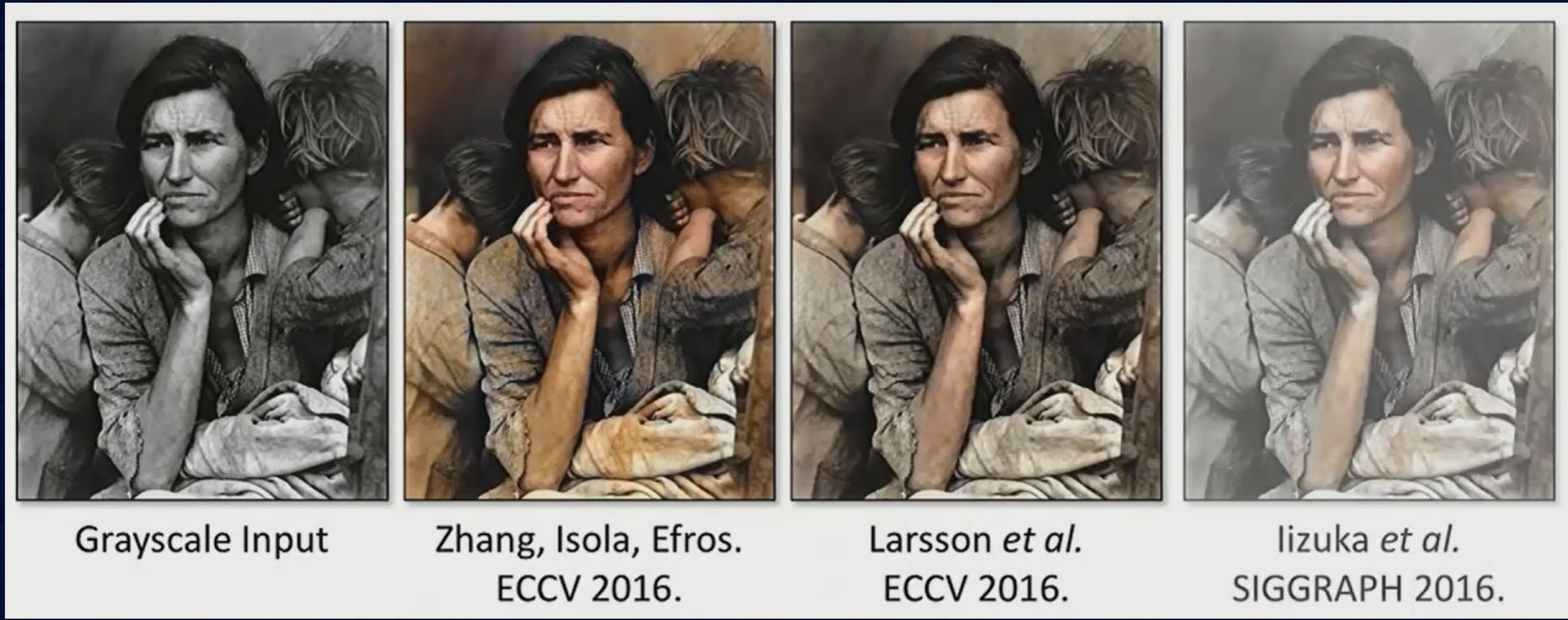
Experiment Results

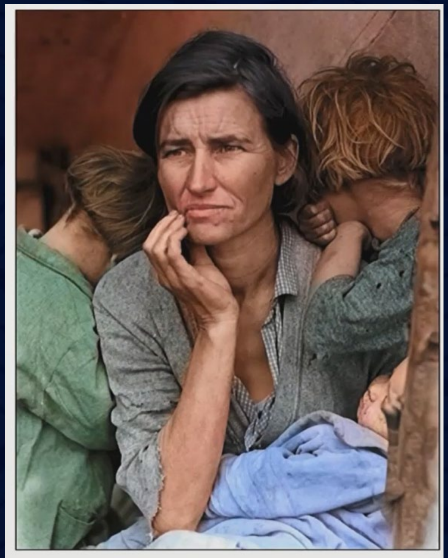
- Performance Comparison of other methods:



Experiment Results

- Performance Comparison of other methods:





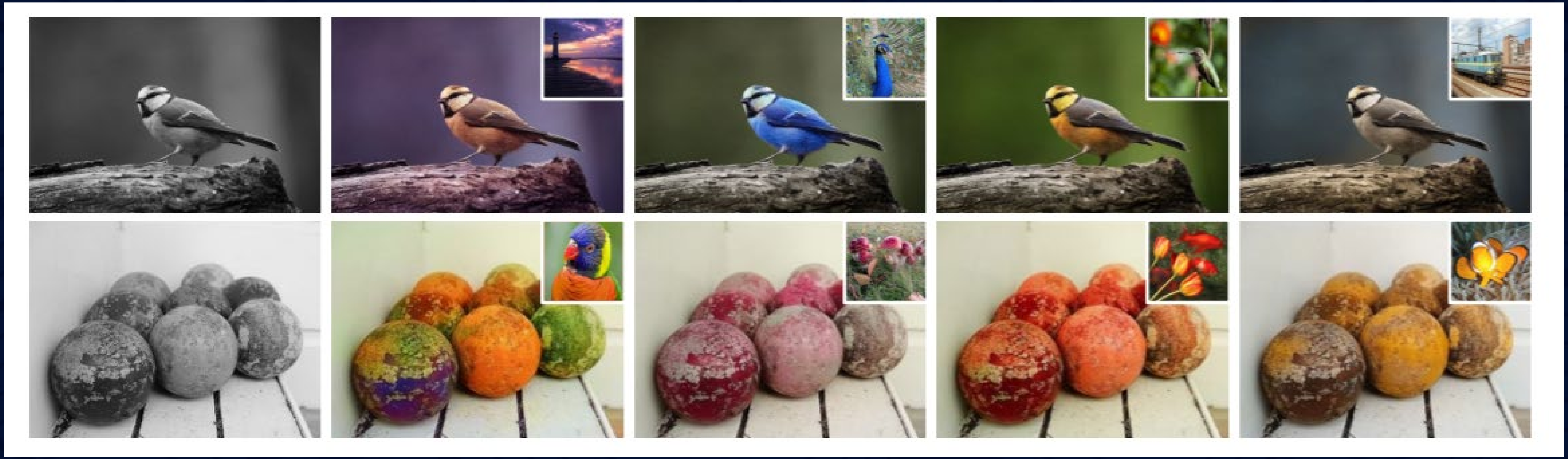
Proposed Method
SIGGRAPH 2017.

Photograph of Migrant Mother by Dorothea Lange, in Nipomo, California, 1936.

Experiment Results

- Global histogram transfer.

  Using our Global Hints Network, we colorize the grayscale version of the image on the left using global histograms from the top-right inset images. Images are from the Imagenet dataset (Russakovsky et al. 2015)



Experiment Results

The proposed method applied to legacy black and white photographs.

- Top left: The Tetons and Snake River, Ansel Adams, 1941 (The National Gallery of Art)
- Bottom left: Muhammad Ali versus Sonny Liston, John Rooney, 1965.
- Right: V-J Day in Times Square, Alfred Eisenstaedt, 1945.

4

Connection & Demo

Sharing of experience, Analysis of advantages and disadvantages, Suggestions for improvement, Demonstration

Connection

A benefit of this method is that the network predicts user-intended actions based on learned semantic similarities. However, the network can also be over-optimistic and produce undesired non-local effects. For example, points added on a foreground object may cause an undesired change in the background.

US-Net is an improved version of U-Net, which does not substantially improve the structure of Encoder-Decoder itself.
➔ " *Implicit Rank-Minimizing Autoencoder* ", LeCun, Oct 04, 2020

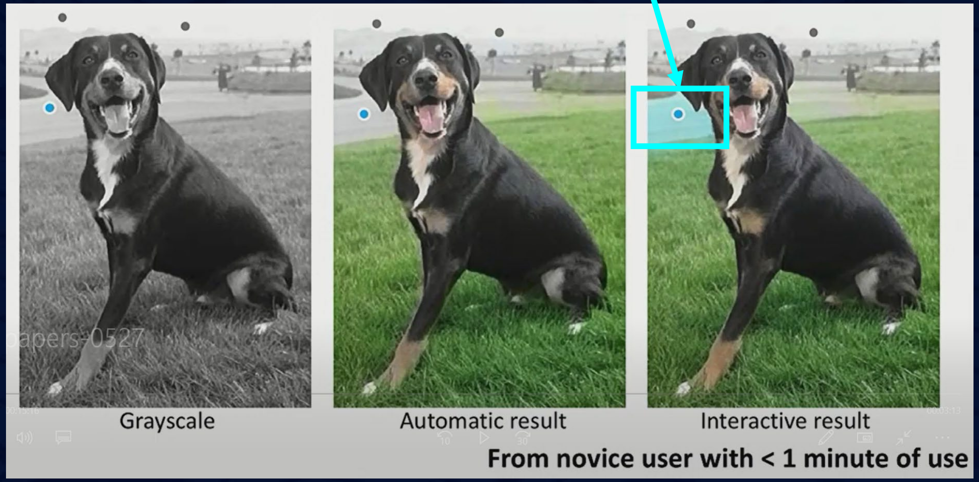
During Convolution, the relationship between Feature maps is not discussed
➔ Channel-wise Convolution

Replace Loss Function by Discriminator (GAN, Generative Adversarial Networks)

Connection

undesired non-local effects

undesired non-local effects



Connection

- *LeCun, " Implicit Rank-Minimizing Autoencoder ", Oct 04, 2020*

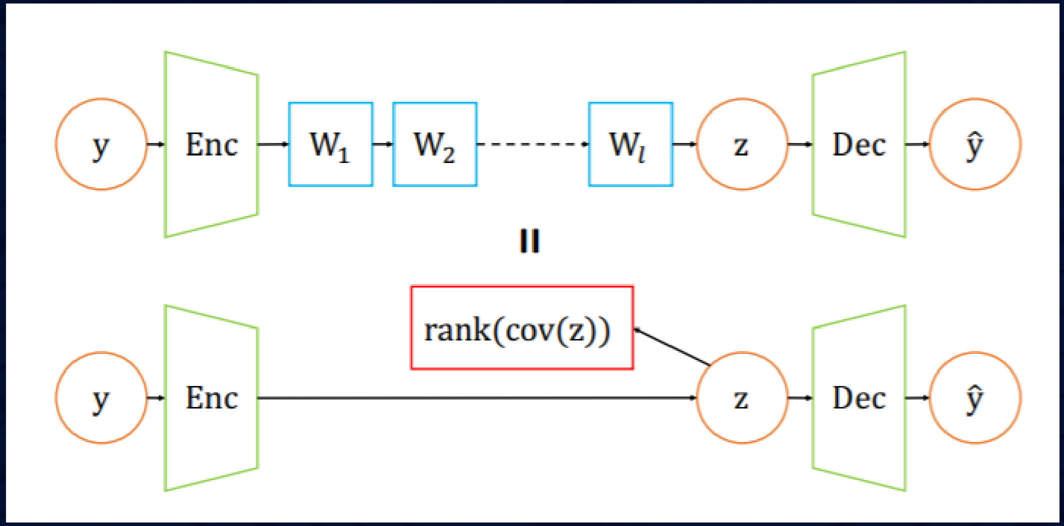


Figure 1: Implicit rank-minimizing autoencoder: a deterministic autoencoder with implicit regularization. The linear matrices that form a linear neural network between the encoder and the decoder are all square matrices. The effect of these matrices is to penalize the rank of the code variable. These matrices are equivalent to a single linear layer at inference time, and thus they do not change the capacity of the autoencoder. In practice, they are absorbed into the last layer of the encoder.

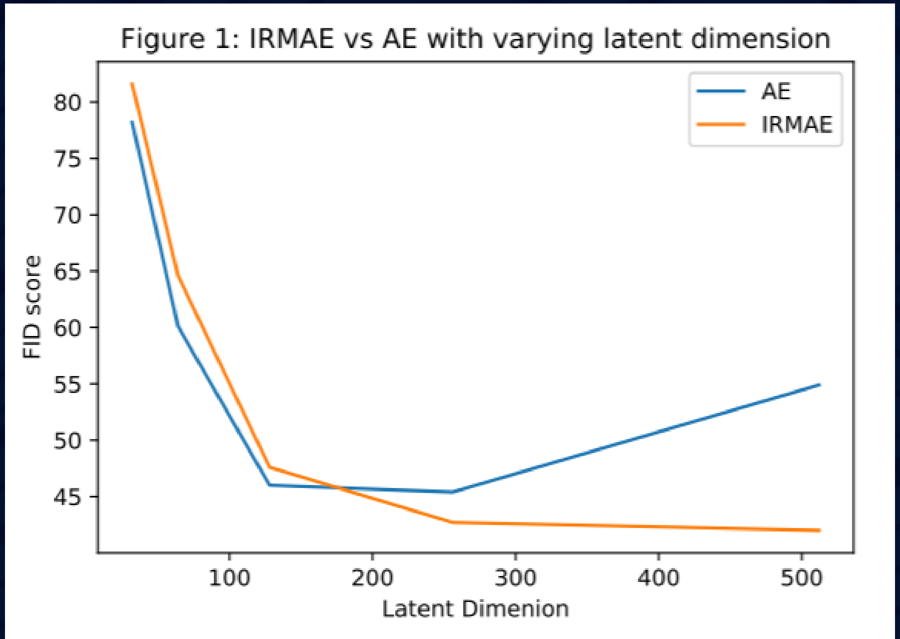
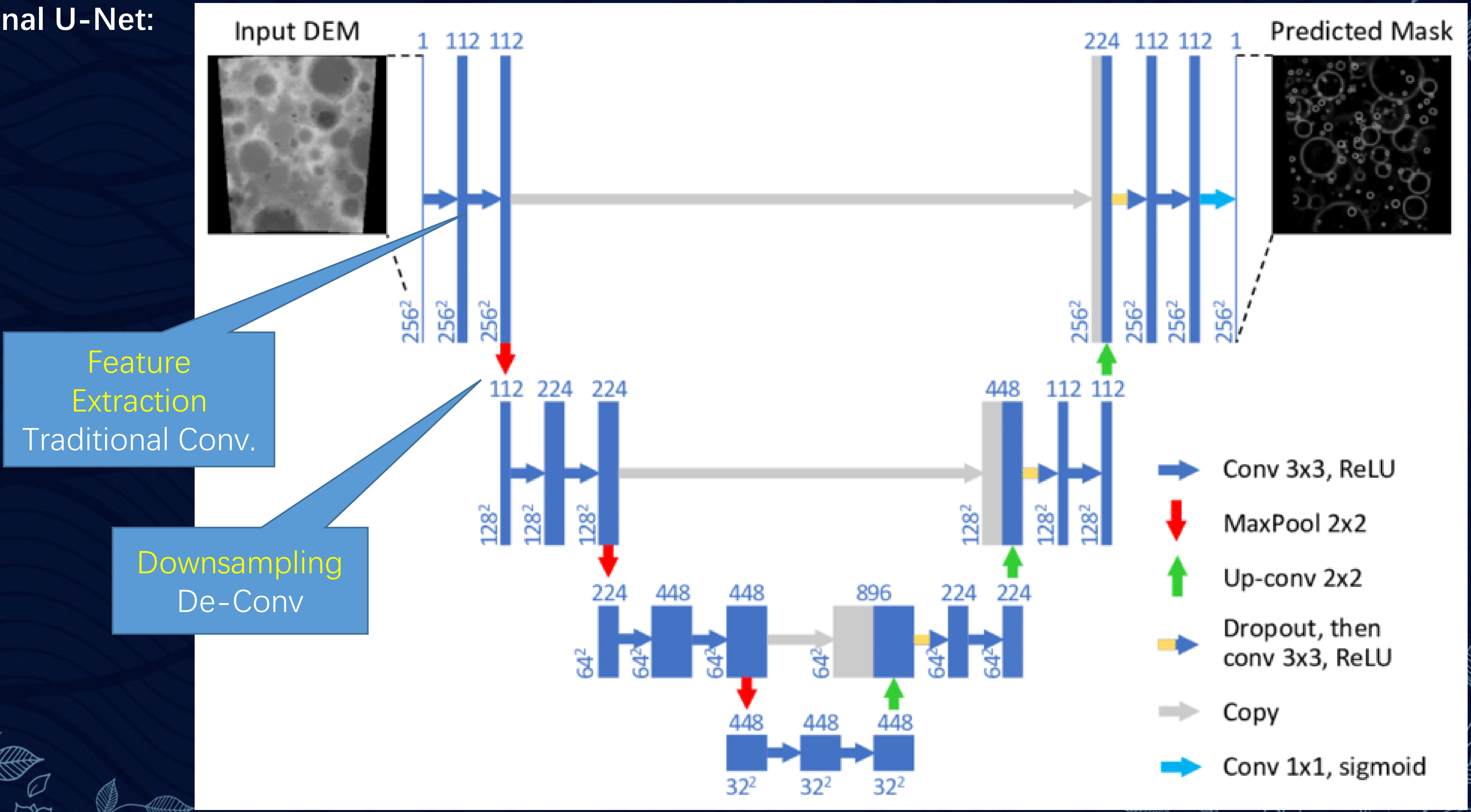


Figure 12: Comparing IRMAE against AEs with different latent dimension. Performed on CelebA dataset

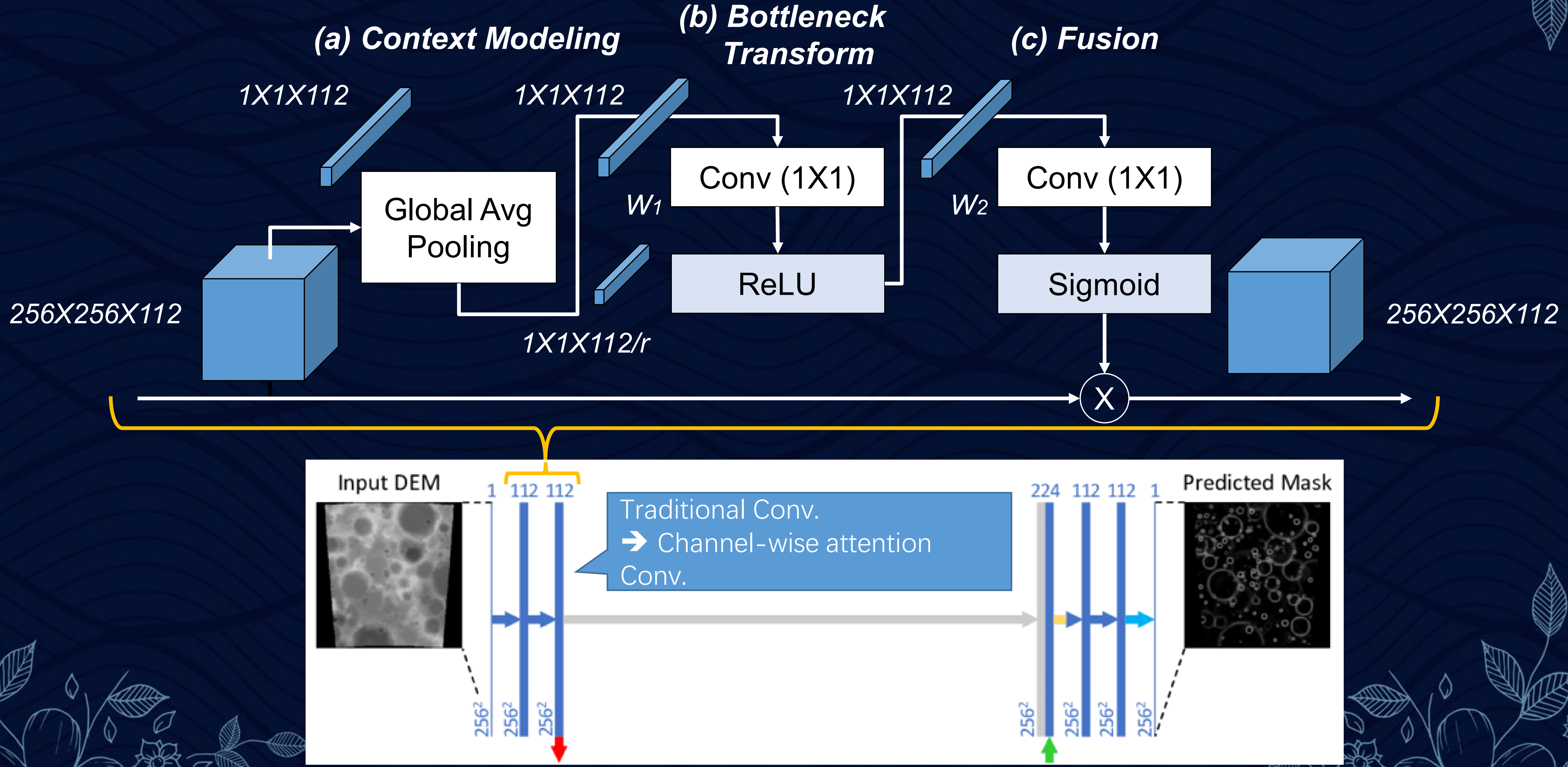
Connection

- **Traditional U-Net:**



Connection

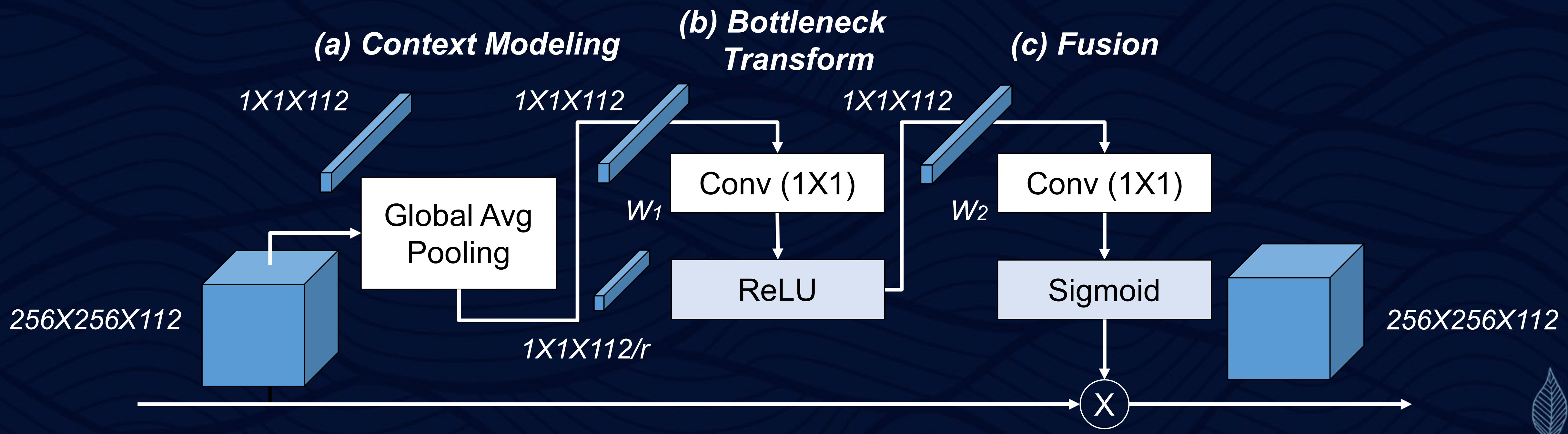
- Channel-wise Attention U-Net:



Connection

- Channel-wise Attention U-Net:

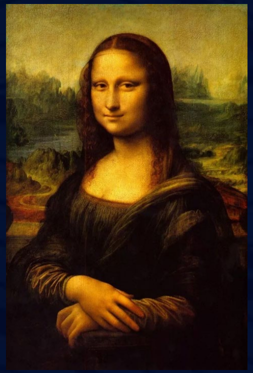
$$z_c = F_{cm} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j) \qquad s = \delta(W_1 z) \qquad s_c = \sigma(W_2 s) = \sigma(W_2 \delta(W_1 z))$$



Connection

- GAN, Generative Adversarial Networks

Training Set (Real)



Generator

Wolfgang Beltracchi

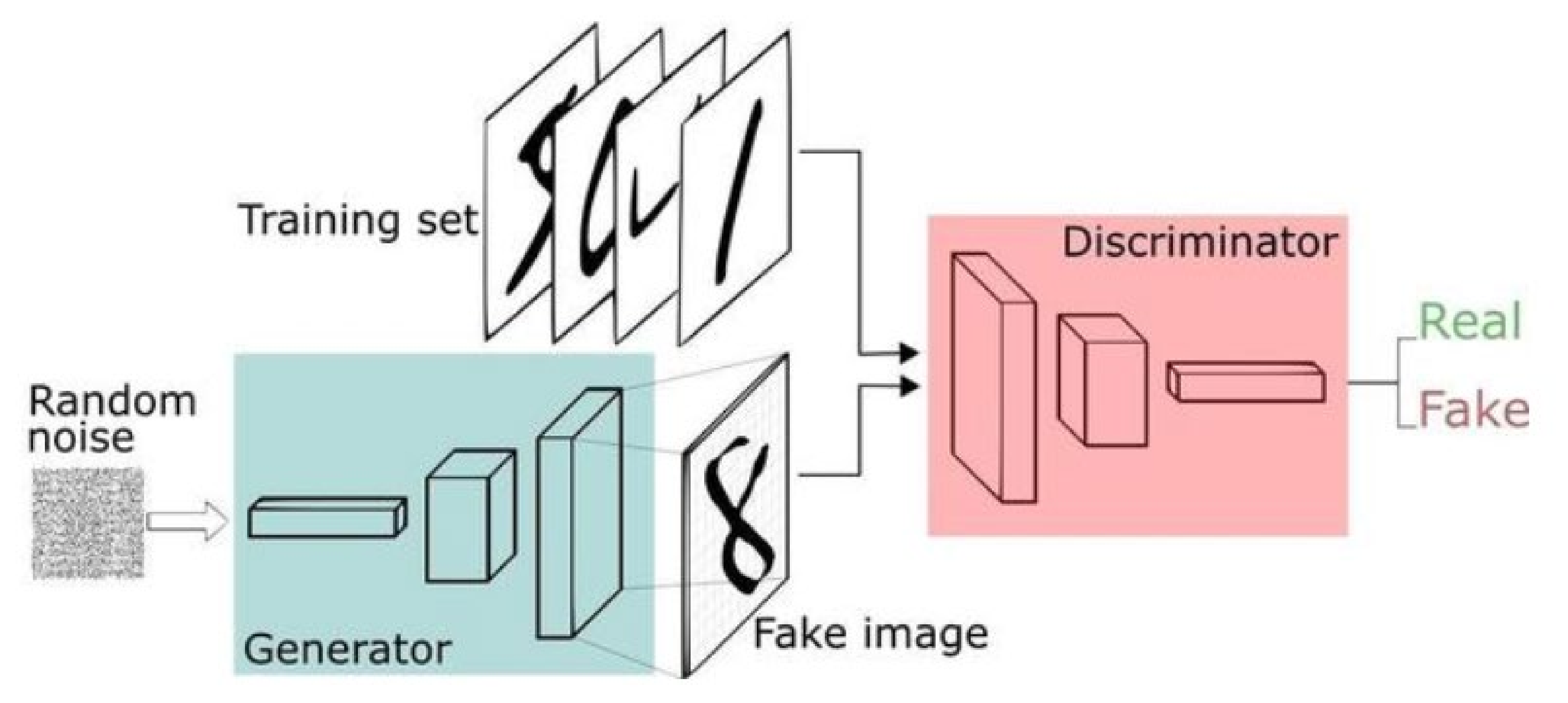
Discriminator

Real

Fake

Connection

- GAN, Generative Adversarial Networks



Real Image

Fake Image

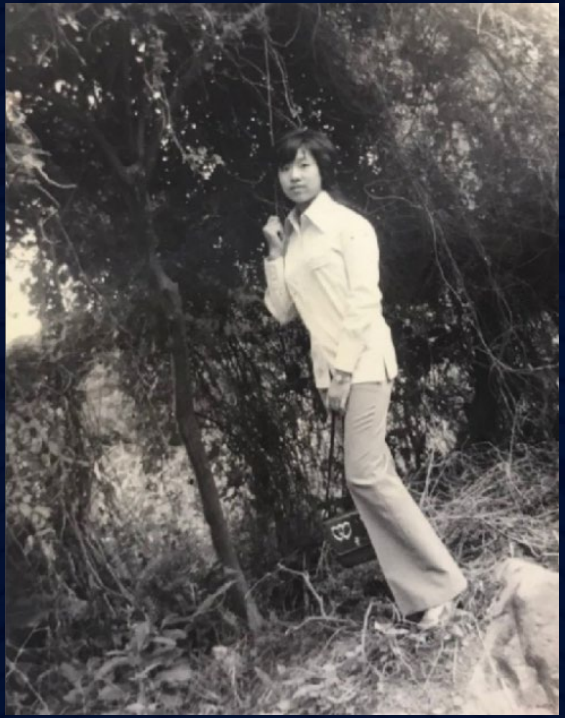
Demo

IPHD Yang, YuanFu

Demo – GAN Application

Phillip Isola et al., " Image-to-Image Translation with Conditional Adversarial Nets (pix2pix) ", CVPR'17

Demo – GAN Application




Phillip Isola et al., " Image-to-Image Translation with Conditional Adversarial Nets (pix2pix) ", CVPR'17

Q & A

IPHD Yang, YuanFu