Slides Available at https://tinyurl.com/usergan

USER-DRIVEN GENERATIVE MODELS

주재걸

고려대학교 컴퓨터학과

Slides partly made by my students, Hyojin Bahng, Wonwoong Cho, Seunghwan Choi, Junsoo Lee, Dongjun Kim, Eungyeop Kim, Yonggyu Kim, Junwoo Park, Yunwon Tae, and Seungjoo Yoo

DAVIAN

Data and Visual Analytics Lab

Jaegul Choo

Associate Professor
Graduate School of AI, KAIST

[Short Bio]

- B.S, 2001, Seoul National University, Electrical Eng.
- M.S, 2009 Georgia Tech, Electrical and Computer Eng.
- Ph.D, 2013 Georgia Tech, Computational Science and Eng.
- Assistant Professor, 2015-2019, Korea University
- Associate Professor, 2019-2020, Korea University

[Data and Visual Analytics Lab (DAVIAN)]

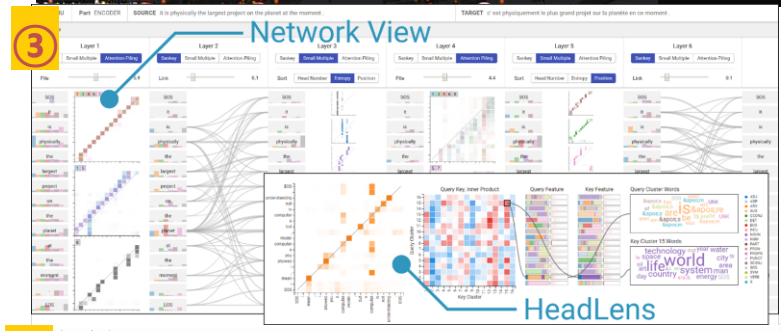
- 13 PhD students, 22 MS students, 10+ Undergrads

[Conference Publication Highlights]

- **2010:** 3 CVPR, 1 ICLR
- **2019:** 2 CVPR (**1 Oral**), 1 AAAI, 1 BMVC (**Oral**), 1 EMNLP, 1 CIKM, 1 CHI, 1 IEEE VIS Short, 1 EuroVIS
- **2018:** 1 CVPR (**Oral**), 1 ECCV, 1 EMNLP, 1 WWW, 1 IJCAI, 1 CHI, 1 IEEE VIS, 1 EuroVIS
- **2017:** 1 AAAI, 2 IJCAI, 1 ICDM, 2 IEEE VIS
- **2016:** 1 ICDM (**Best Student Paper**), 2 IEEE VIS
- **2015:** 1 KDD, 1 ICWSM, 1 IEEE VIS

[Selected Publications by Areas, 2400+ citations]

- **Image Generation and Translation (user-driven, multi-task, multi-domain)**
  - ① Reference-based Sketch Image Colorization using Augmented-Self Reference, **CVPR'20**
  - Image-to-Image Translation via Group-wise Deep Whitening and Coloring, **CVPR'19**, *Oral paper* (*Top 5.5%*)
  - Coloring With Limited Data: Few-Shot Colorization via Memory-Augmented Networks, **CVPR'19**
  - StarGAN: Unified GANs for Multi-Domain Image Translation, **CVPR'18**, *Oral paper* (*Top 2%*)
- **Natural Language Understanding and Generation**
  - NeurQuRI: Neural Question Requirement Inspector for Answerability Prediction in Machine Reading Comprehension, **ICLR'20**
  - Paraphrase Diversification using Counterfactual Debiasing, **AAAI'19**
  - MemoReader: Large-Scale Reading Comprehension through Neural Memory Controller, **EMNLP'18**
- **Text and Social Media Mining**
  - Recommender System via Sequential and Global Preference via Attention Mechanism and Topic Modeling, **CIKM'19**
  - ② TopicOnTiles: Tile-Based Spatio-Temporal Event Analytics via Exclusive Topic Modeling on Social Media, **CHI'18**
  - Short-Text Topic Modeling via Non-negative Matrix Factorization with Local Word-Context Correlations, **WWW'18**
- **Visual Analytics (image, text, sequence analysis)**
  - ③ SANVis: Visual Analytics for Understanding Self-Attention Networks, **IEEE VIS'19 Short**
  - AILA: Attentive Interactive Labeling Assistant for Document Classification through Deep Neural Networks, **CHI'19**
  - Visualizing for the Non-Visual: Deep Learning to Enable Visually Impaired to Use Visualization, **EuroVIS'19**
  - ④ RetainVis: Visual Analytics with Interpretable and Interactive RNNs on Electronic Medical Records, **IEEE VIS'18**



My Current Research

- Image-to-image translation
- Automatic image colorization
- Data augmentation via generative adversarial networks
- Visual analytics for interpreting and interacting with deep neural networks
- Interactive labeling techniques and systems
- Medical image recognition
- Machine reading comprehension
- Time-series prediction

Overview of This Talk

- Intro to conditional generative models [5 min]

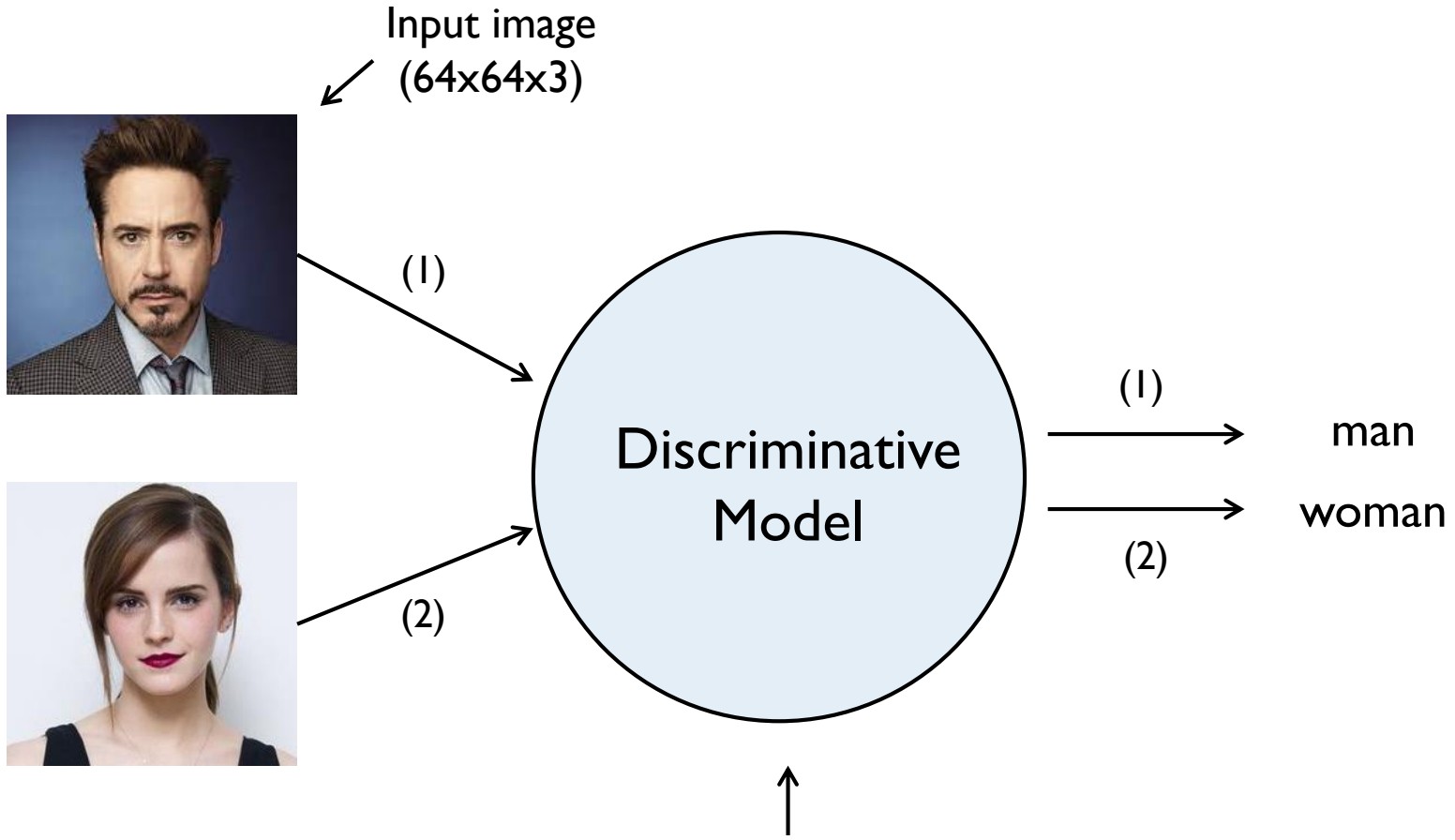
- My own research on interactive automatic colorization [45 min]
  - Colorization using natural language [ECCV'18]
  - Few-shot colorization via memory networks [CVPR'19]
  - Reference-based sketch colorization using augmented self-exemplar [CVPR'20]

- Other work on interactive generative models and future research directions [10 min]

Definition of Generative Model

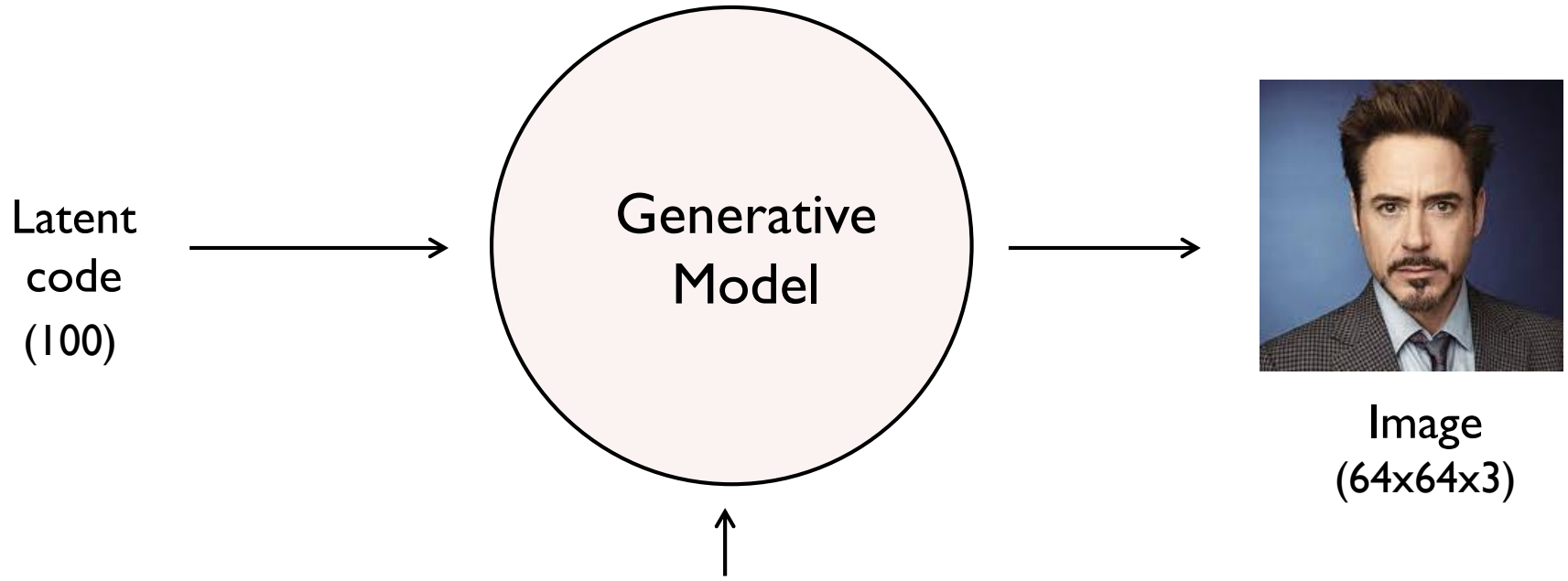
- Recognition vs. Generation (and Translation)
- Recognition: compresses a large number of input values into a small number of output values
- Generation: expands a small number of input values into a large number of output values.
- Translation: transforms a large number of input information into another large number of output values.
- Conditional Generation: additional input is given, which steers the generation processes in a user-driven manner.

Recognition Tasks



The discriminative model learns how to classify input to its class.

Generation Tasks



Translation Tasks

Image
(64x64x3)

Translation
Model

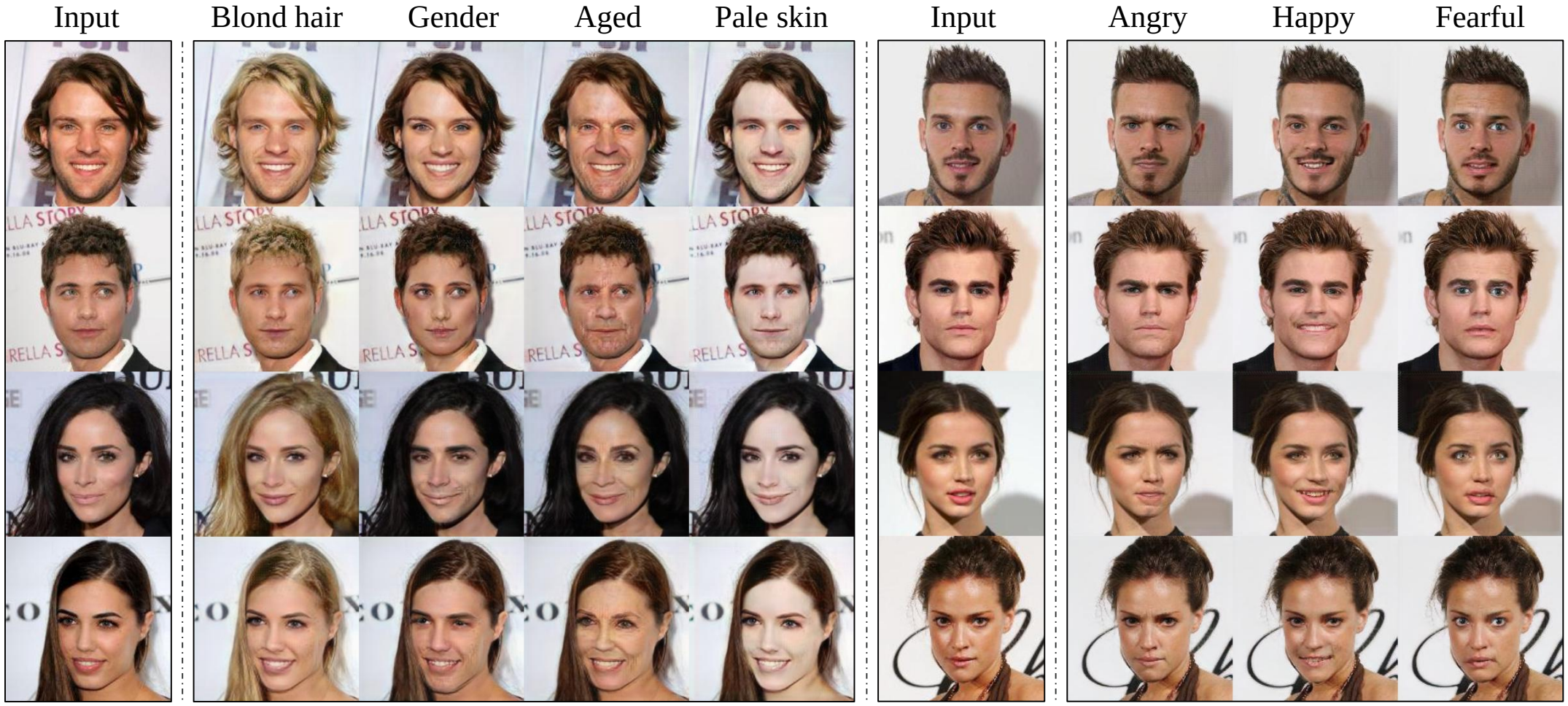
Image
(64x64x3)

CONDITIONAL GENERATION (AND TRANSLATION)

- An additionally given input works as a condition that steers the generation and translation processes in a user-driven manner.

- Two GAN-based models: CGAN and ACGAN

STARGAN: MULTI-DOMAIN IMAGE TRANSLATION



Motivations for Human-in-the-Loop Approach

- As we discussed earlier, unlike recognition tasks, generative models gives the output composed of a large number of values.

- User intent are often too complex to describe as a simple categorical variable.
  -> Flexible, sophisticated forms of user inputs are necessary.

- Some among them may not be satisfactory to users nor aligned with user intent.
  -> Users should be able to partially edit the output in an iterative manner.

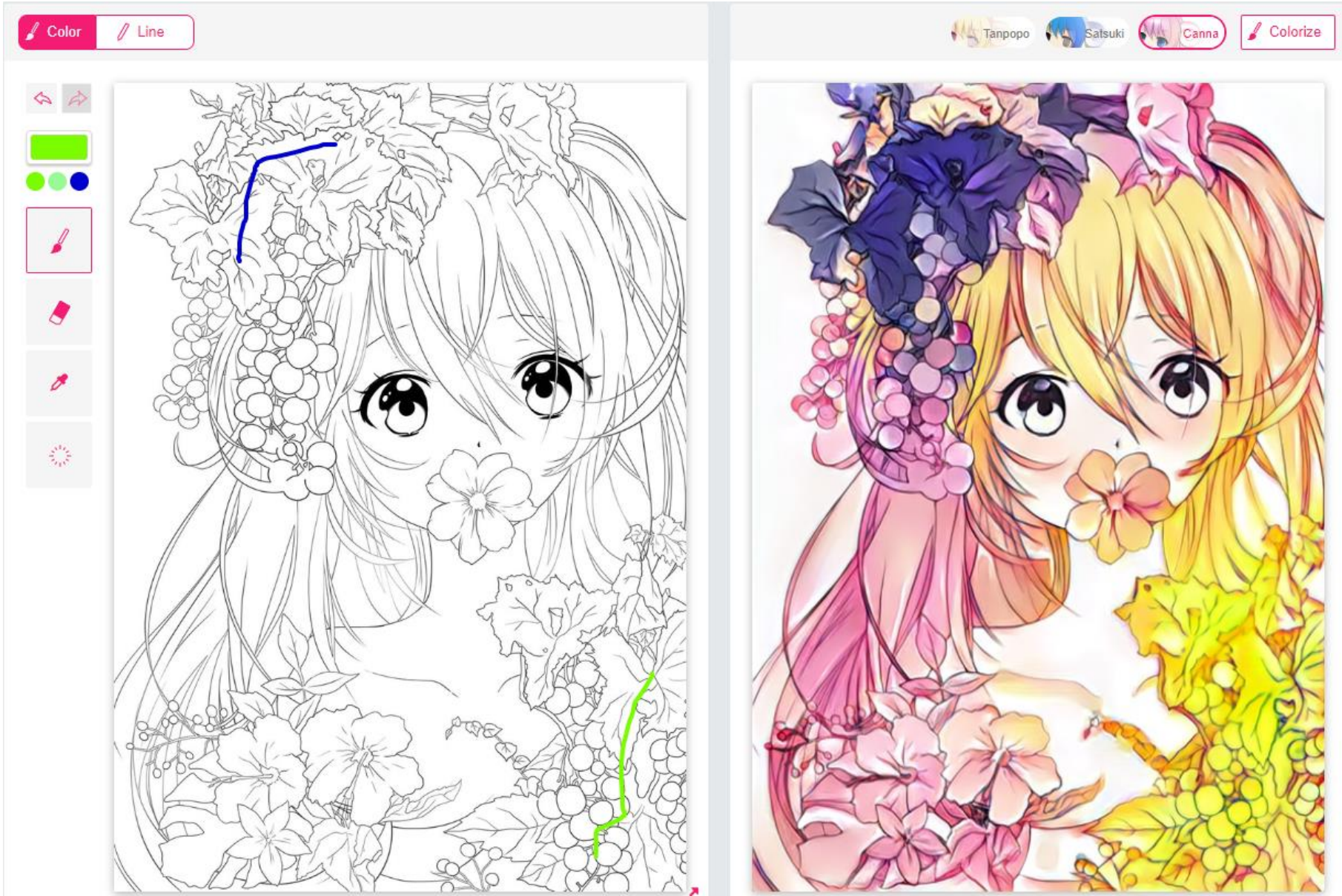
- Machine learning models should facilitate such editing processes by properly propagating user inputs in the generation output.

Taxonomy of User Inputs (or Conditions) in Generative Models

- Global (male or female) vs. Local (strokes and scribbles)
- Reference-based vs. non-reference-based
  - Reference image
    - Users' own vs. one among a pre-given set
- Strokes and scribbles
  - Positive vs. negative clicks (segmentation)
  - Particular colors (colorization)
- Interaction modality
  - Text, voice, AR/VR, …

Strokes and Scribble User-Input

Interactive colorization demo page, https://paintschainer.preferred.tech



(Potentially Interactive) Generation and Translation Tasks

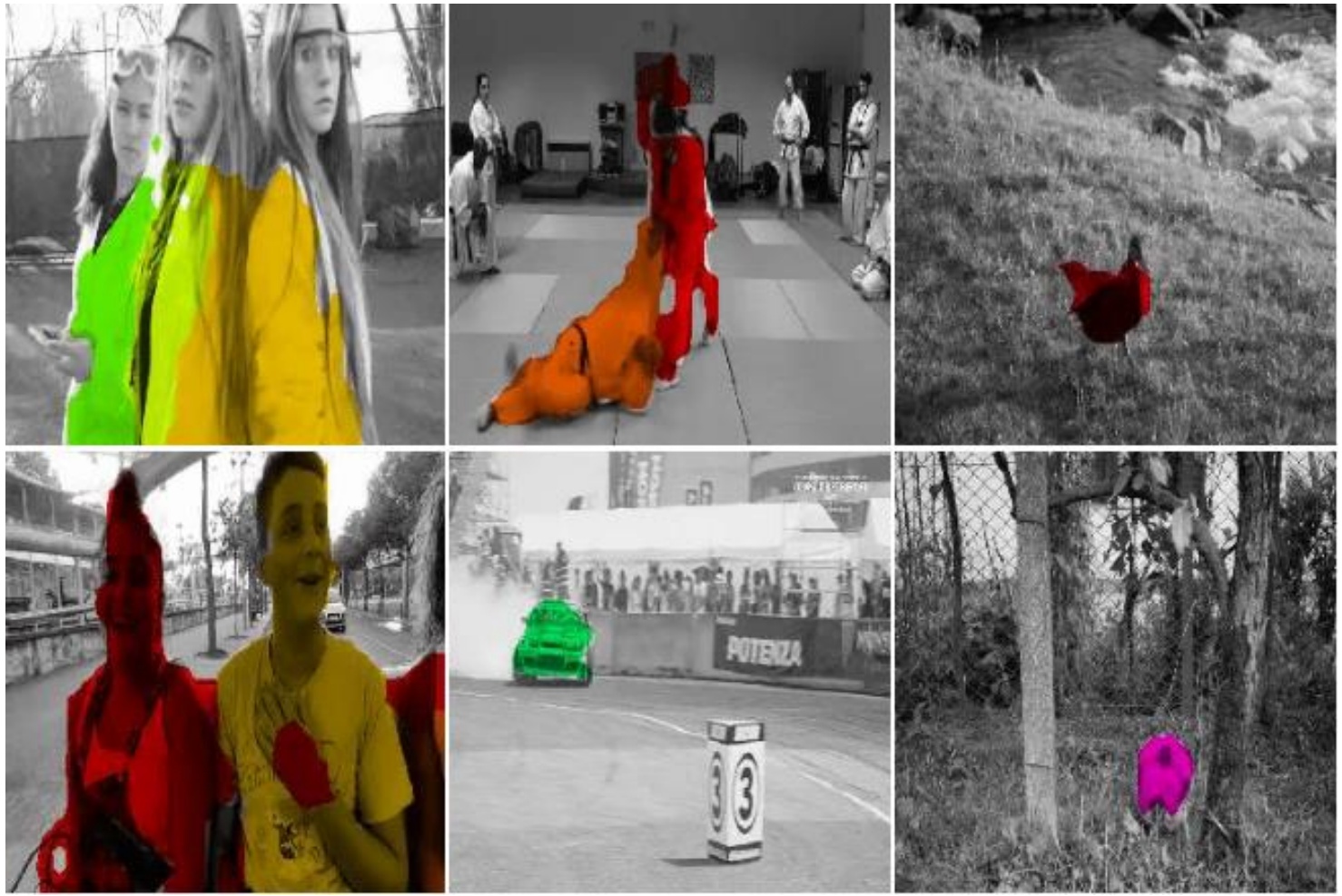
- Computer Vision
  - Image generation and translation
    - Facial attribute transfer, pose transfer, …
  - Interactive instance segmentation for labeling
  - Automatic colorization
  - DeepFashion
  - Video re-targeting
- Natural Language Generation
  - Post-editing in NMT
  - Controllable paraphrasing

Intro to Automatic Image Colorization

- Basically, it is an image-to-image translation task from a grayscale or sketch image into a colorized one.
  - Thus, adversarial learning via an additional discriminator, or simply GAN, is usually adopted.
- Obviously, it has practical impact in content creation, e.g., animation and cartoon.
- It can potentially be used as a general-purpose, self-supervised learning task, which works as a pre-training method for other downstream tasks.
- In general, this task is trained in a paired setting, but as will be seen in the third work I will present, it is not always the case, making the task challenging.

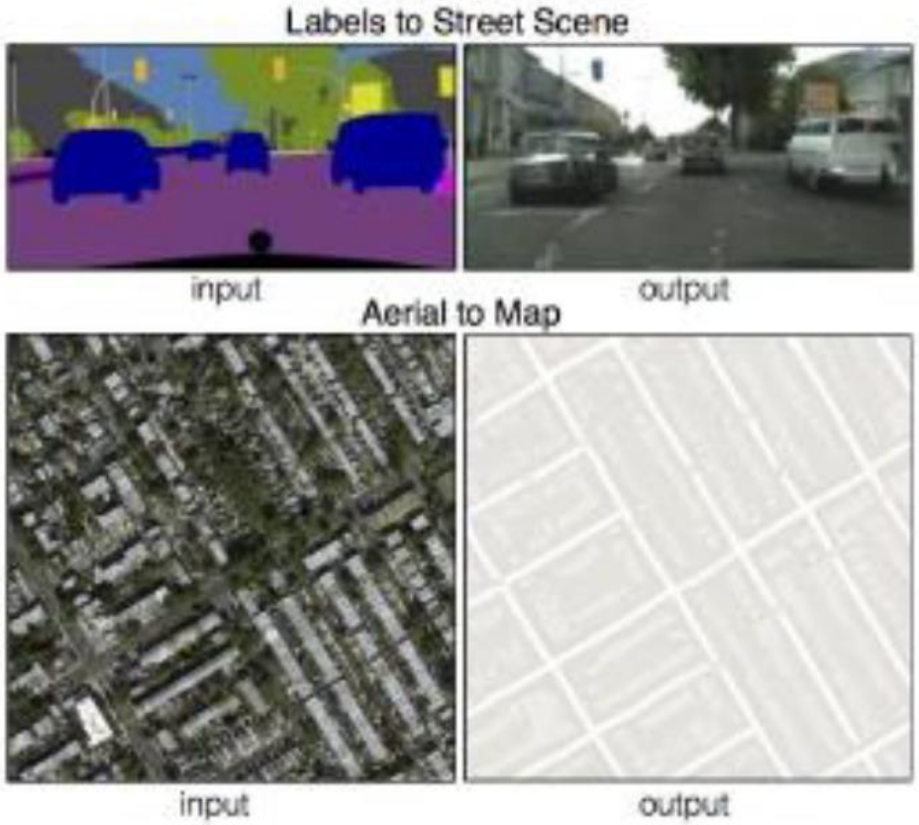
Colorization as Self-Supervised Learning

- Vondrick et al., Tracking Emerges by Colorizing Videos [ECCV'18]
- https://ai.googleblog.com/2018/06/self-supervised-tracking-via-video.html



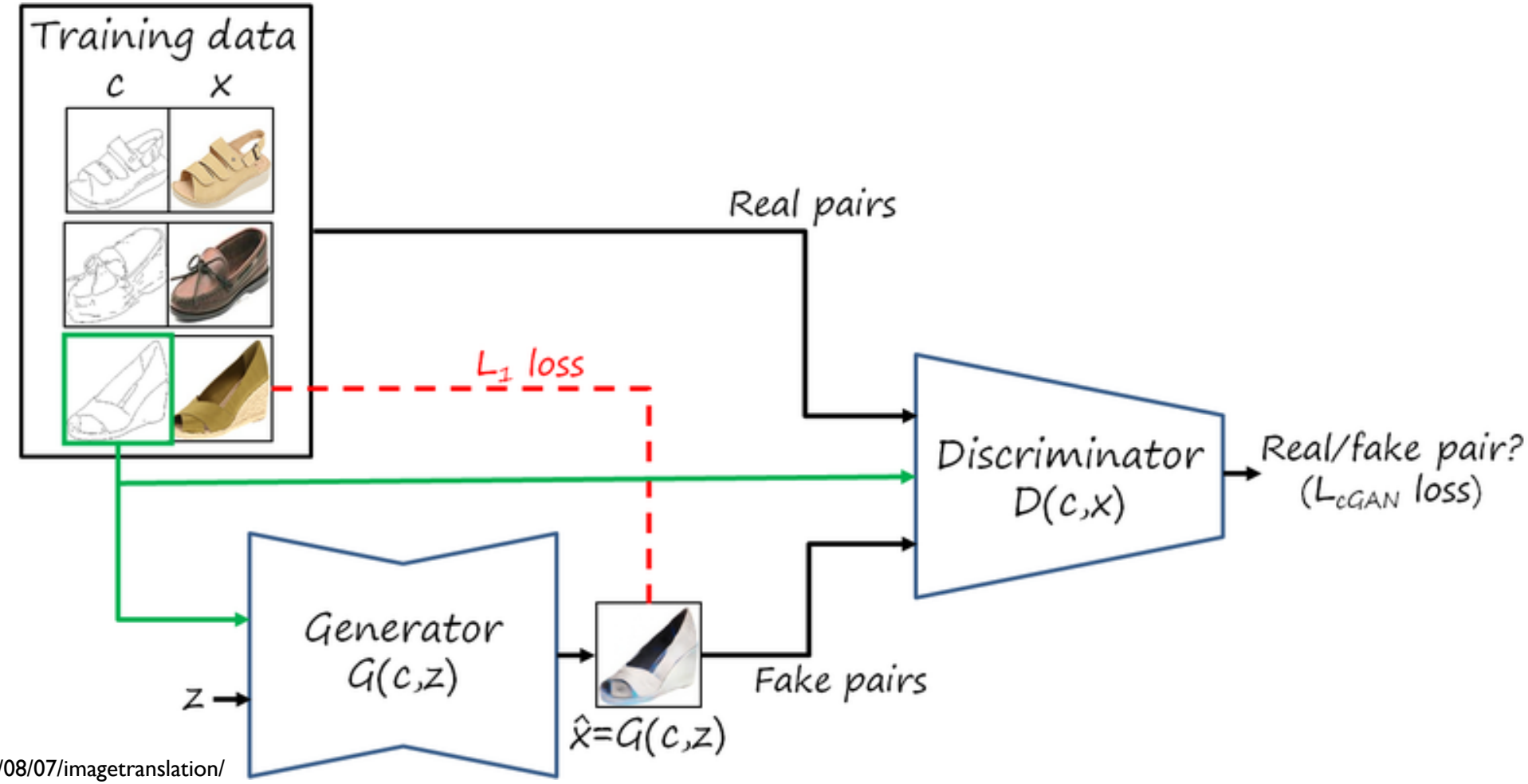
pix2pix: Conditional GAN for Paired Image-to-Image Translation

- Isola et al., Image-to-Image Translation with Conditional Adversarial Networks [CVPR'17]



pix2pix: Conditional GAN for Paired Image-to-Image Translation

- Isola et al., Image-to-Image Translation with Conditional Adversarial Networks [CVPR'17]



Img source: http://www.lherranz.org/2018/08/07/imagetranslation/

Overview of This Talk

- Intro to conditional generative models

- My own research on interactive automatic colorization
  - Colorization using natural language [ECCV'18]
  - Few-shot colorization via memory networks [CVPR'19]
  - Reference-based sketch colorization using augmented self-exemplar [CVPR'20]

- Other work on interactive generative models and future research directions

COLORING WITH WORDS:
GUIDING IMAGE COLORIZATION THROUGH TEXT-BASED PALETTE GENERATION (ECCV 2018)

Hyojin Bahng,* Seungjoo Yoo,* Wonwoong Cho,* David K. Park, Ziming Wu, Xiaojuan Ma, and Jaegul Choo

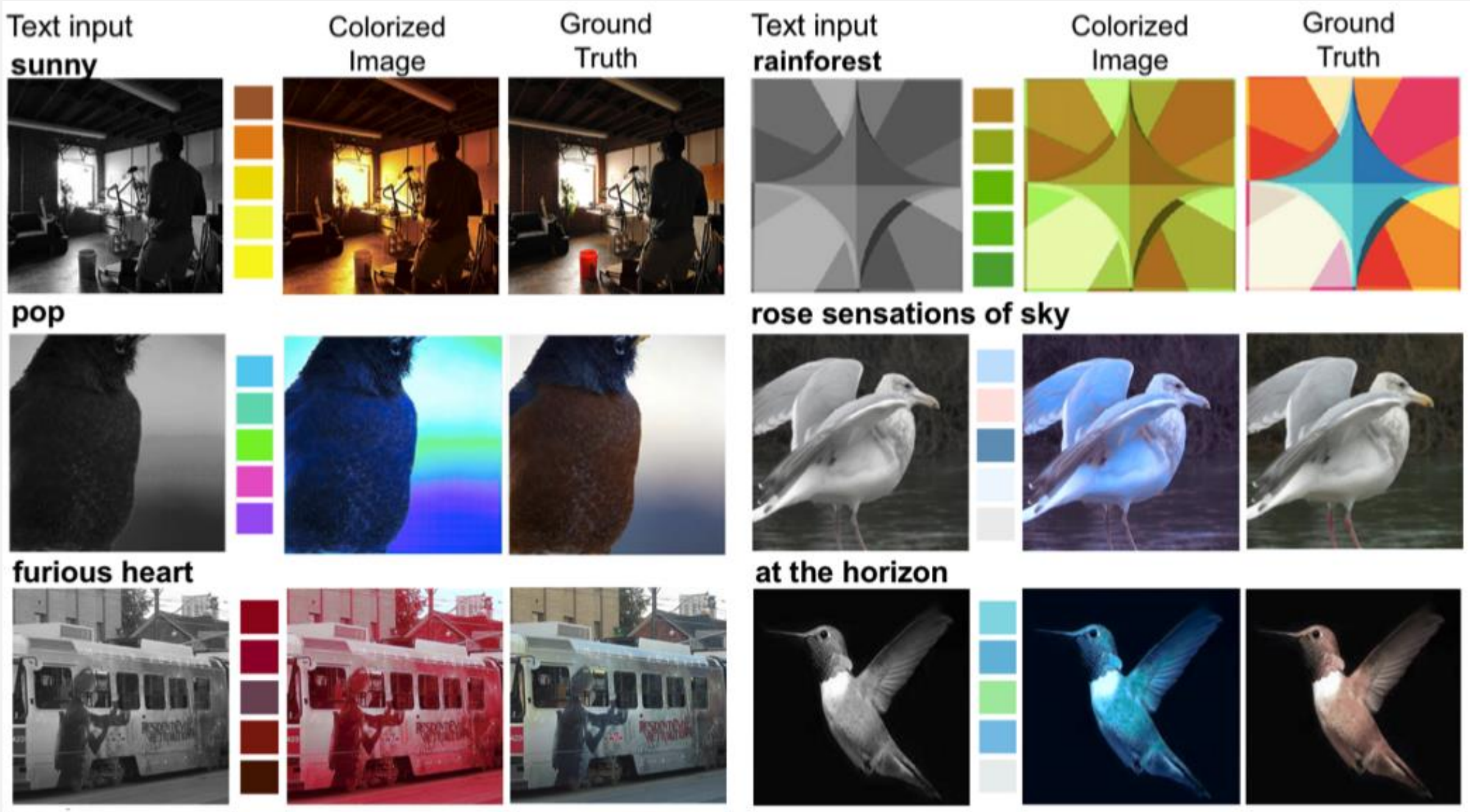
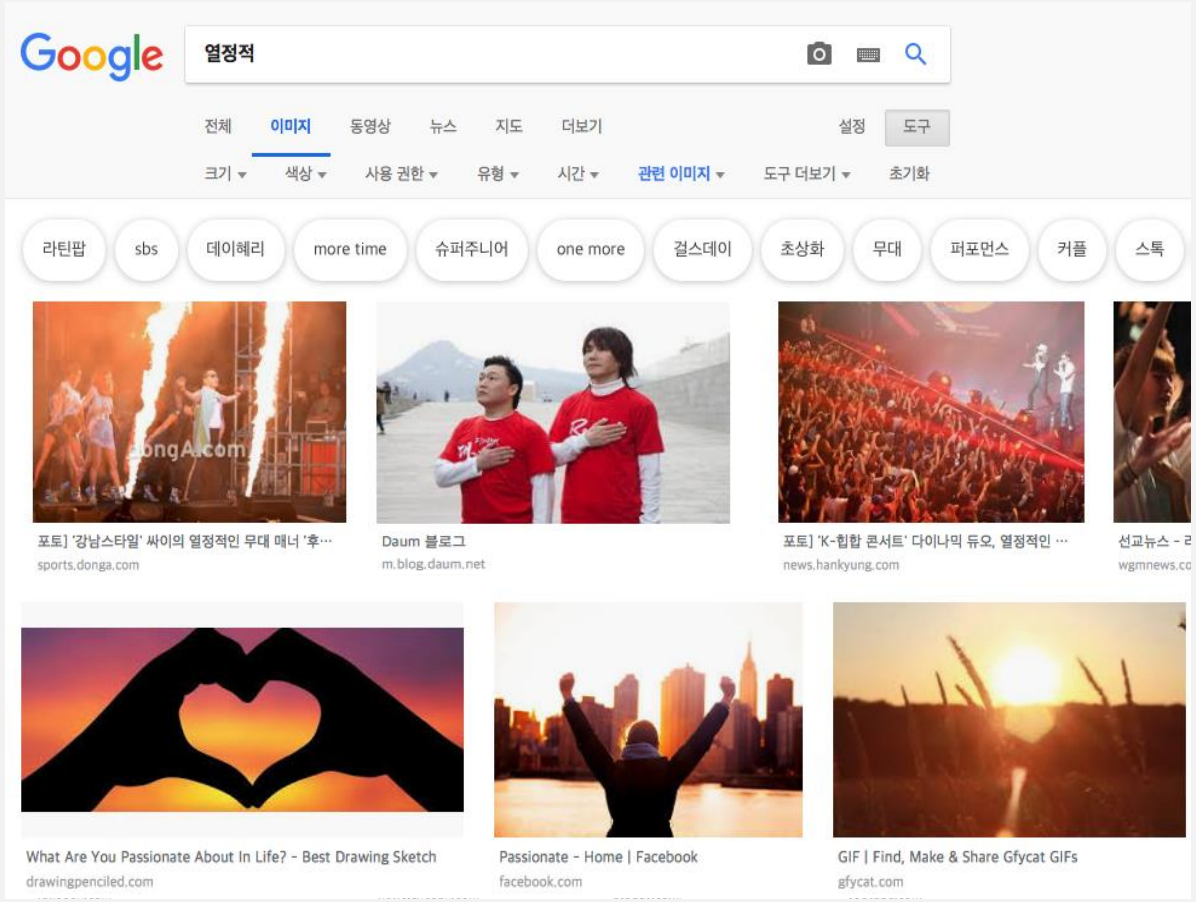
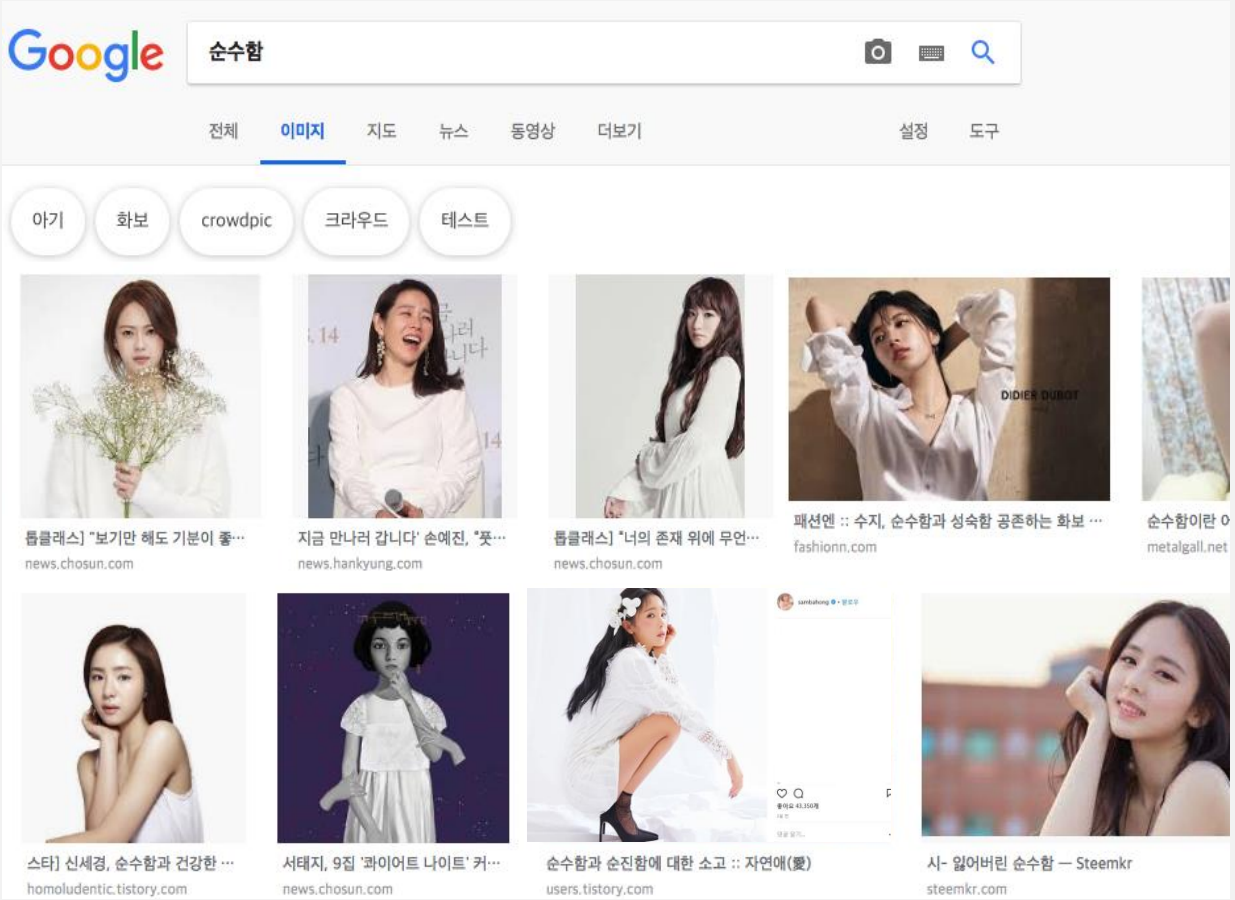

Text input
sunny
Colorized Image
Ground Truth
Text input
rainforest
Colorized Image
Ground Truth
pop
rose sensations of sky
furious heart
at the horizon

TABLE OF CONTENTS

GOAL

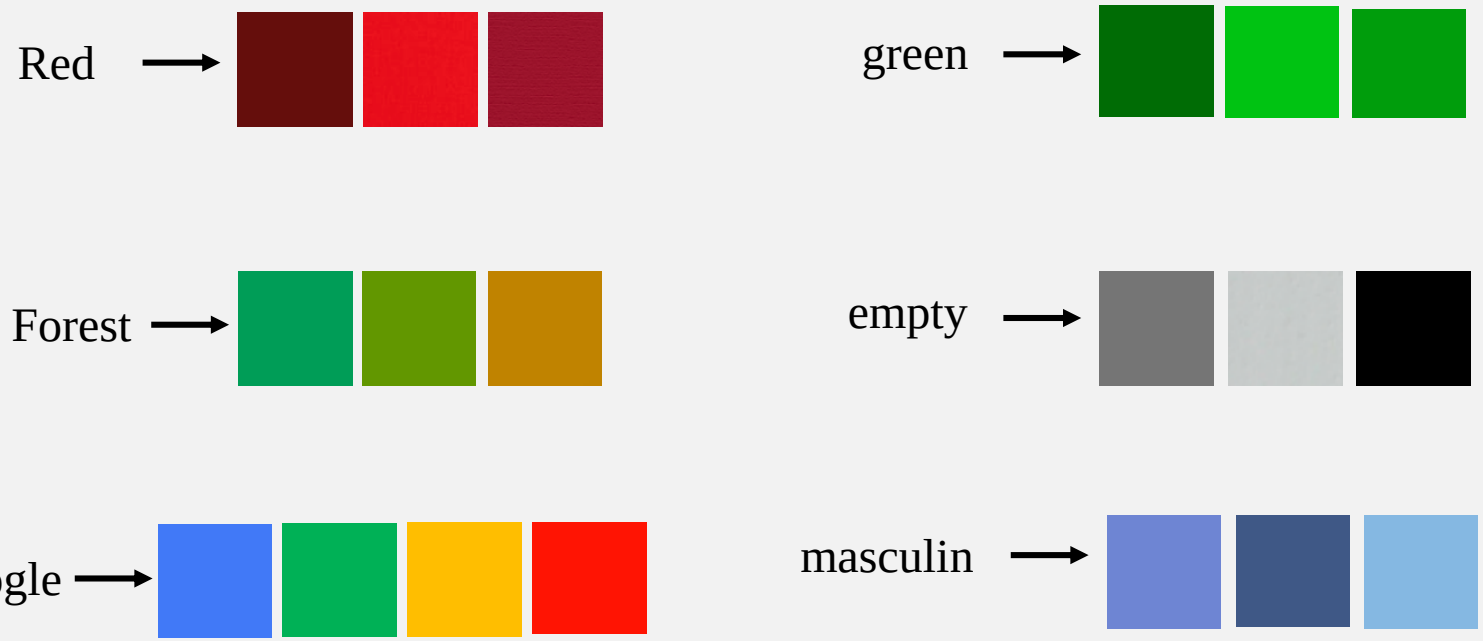
Map text to colors



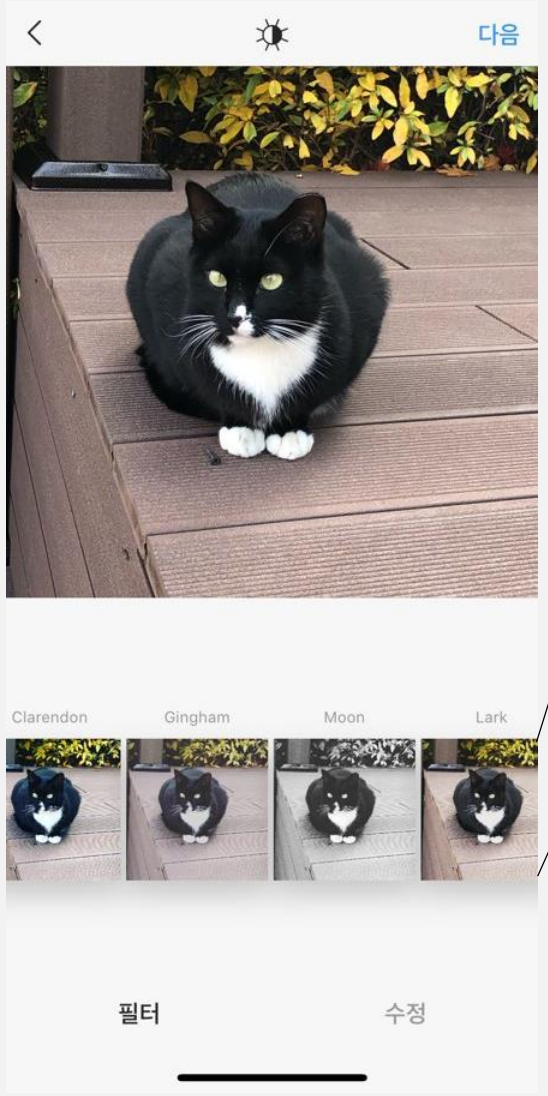


MOTIVATION

- Text can be mapped to multiple colors



MOTIVATION



clarendon gingham moon juno slumber crema ludwig

MOTIVATION


TRUST - DEPENDABLE - STRENGTH
DELL
facebook
OREO
PEACEFUL - GROWTH - HEALTH
Spotify
bp
STARBUCKS COFFEE
Holiday Inn
BALANCE - NEUTRAL - CALM
NIKE
CN
CARTOON NETWORK
Discovery CHANNEL


OPTIMISM - CLARITY - WARMTH
ups
IMDb
Yellow Pages
NATIONAL GEOGRAPHIC
FRIENDLY - CHEERFUL - CONFIDENCE
NICKELODEON
Fanta
orange
EXCITEMENT - YOUTHFUL - BOLD
Coca-Cola
LEGO
Kellogg's
Nintendo

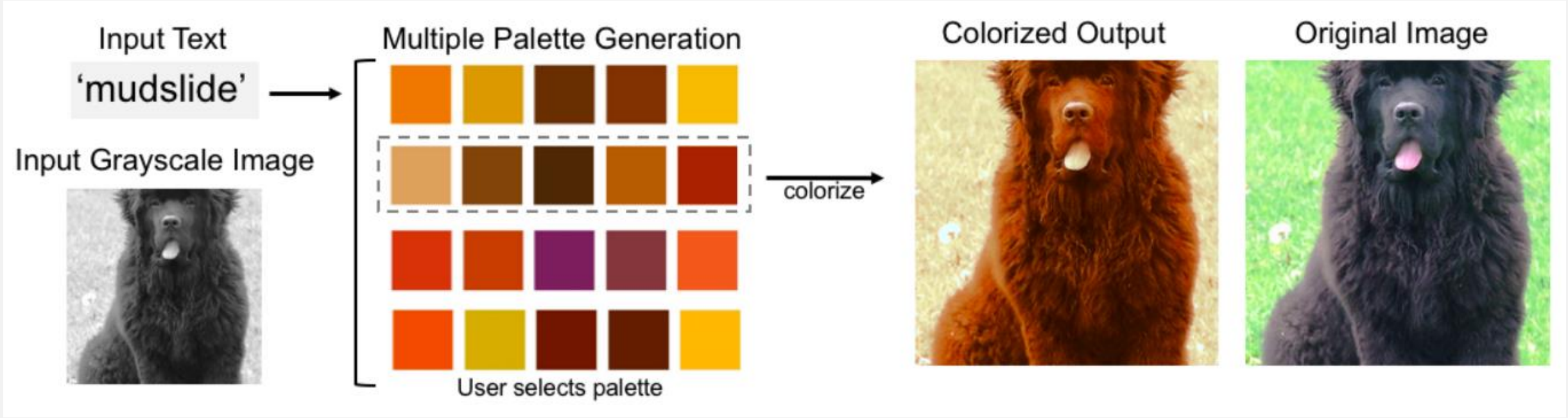
MOTIVATION

- Catch hidden meaning in text

Fig. 14. Handling phrase-level inputs about 'love'.

OVERVIEW OF TEXT2COLORS

OVERVIEW: HOW OUR MODEL WORKS



PAT DATASET

WHY WE MADE OUR OWN DATASET

- Important keywords in writing our research paper

데이터

글쓰기

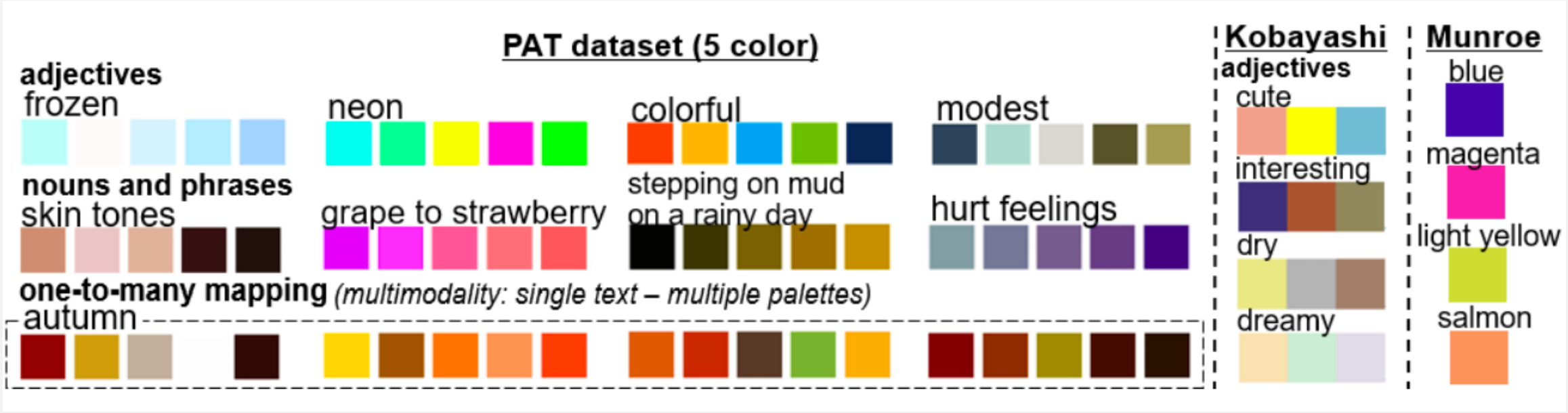
모델설계

학습

디버깅

문제정의

PAT DATASET

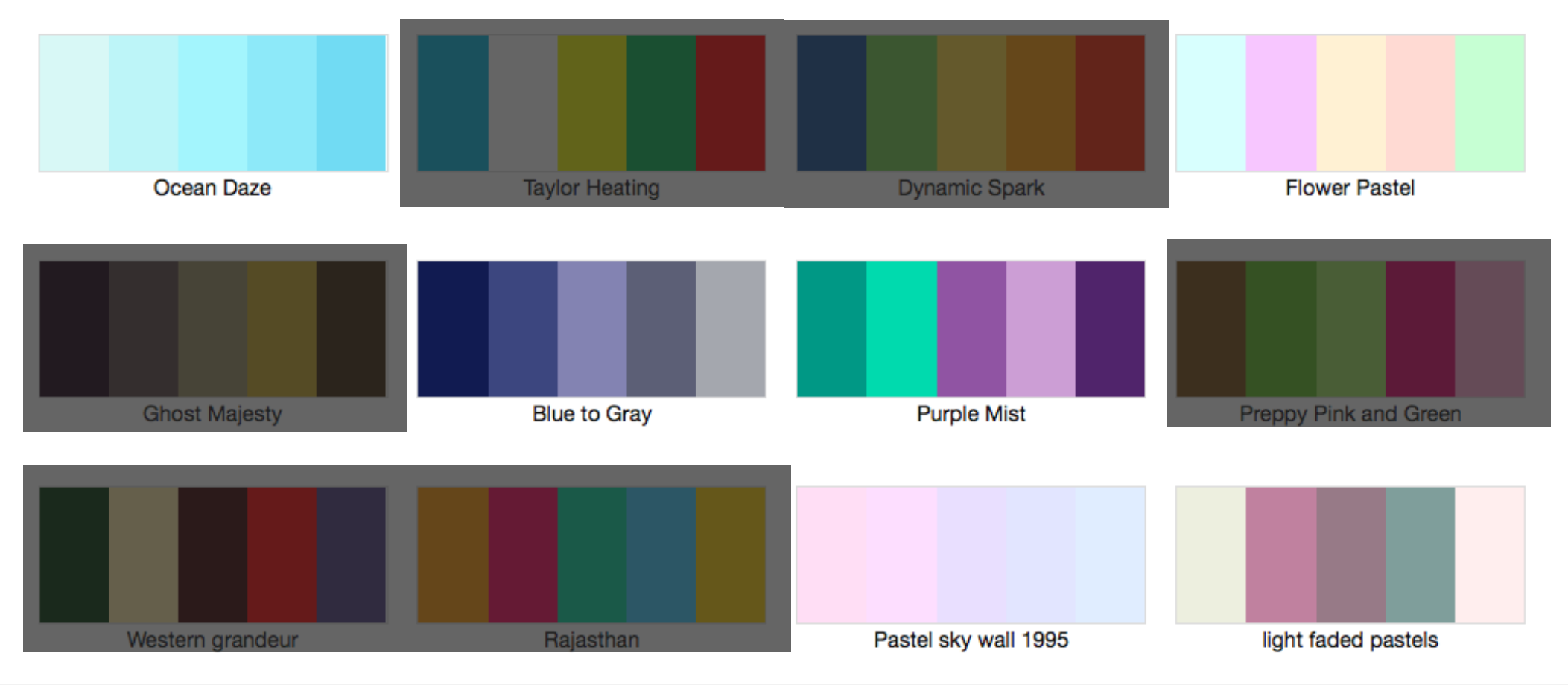


- First dataset to address rich text and multimodality
- First large-scale color dataset matched with words

PAT DATASET

- Data is crawled from color-hex.com
- 4 annotators manually refined the data

PAT DATASET

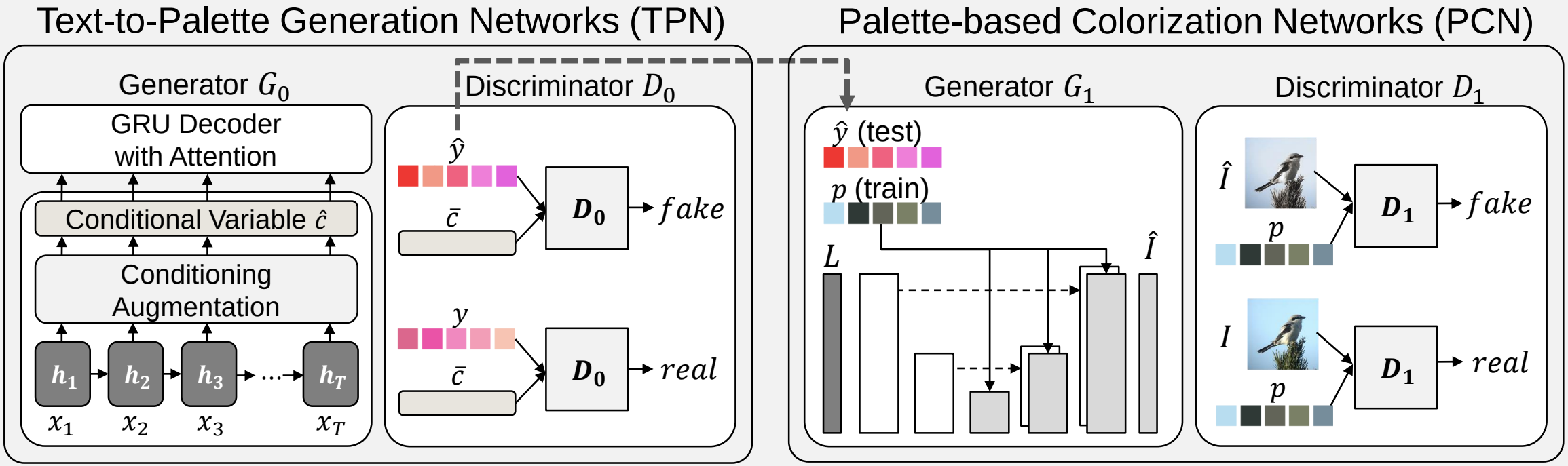


- Data is crawled from color-hex.com
- 4 annotators manually refined the data

PROPOSED METHOD

PROPOSED METHOD (OVERVIEW)

Two Conditional GANs

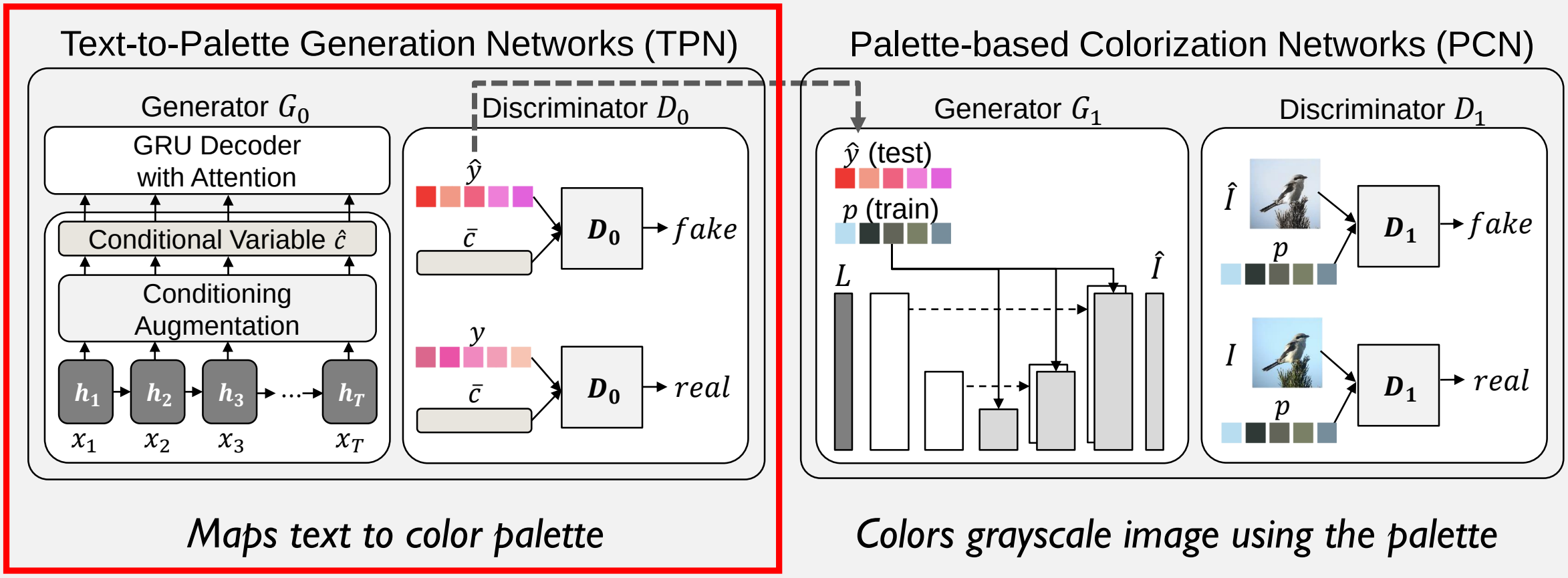


Maps text to color palette

Colors grayscale image using the palette

PROPOSED METHOD (OVERVIEW)

Two Conditional GANs

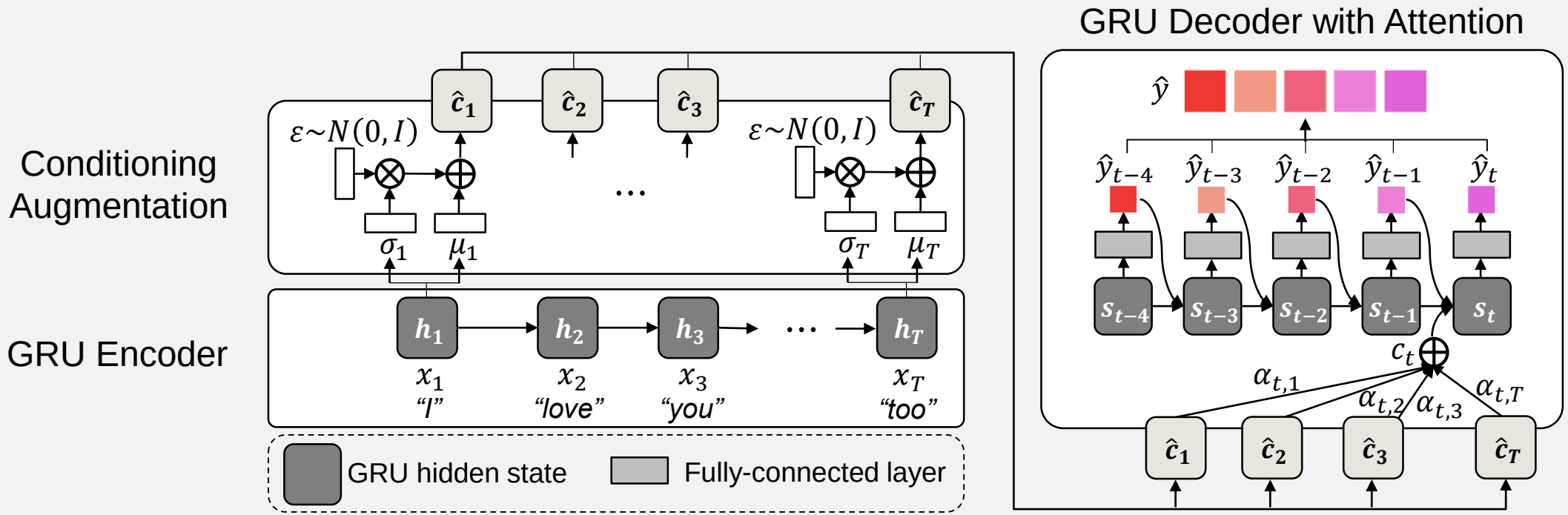


Maps text to color palette

Colors grayscale image using the palette

TEXT-TO-PALETTE GENERATION NETWORKS (TPN)

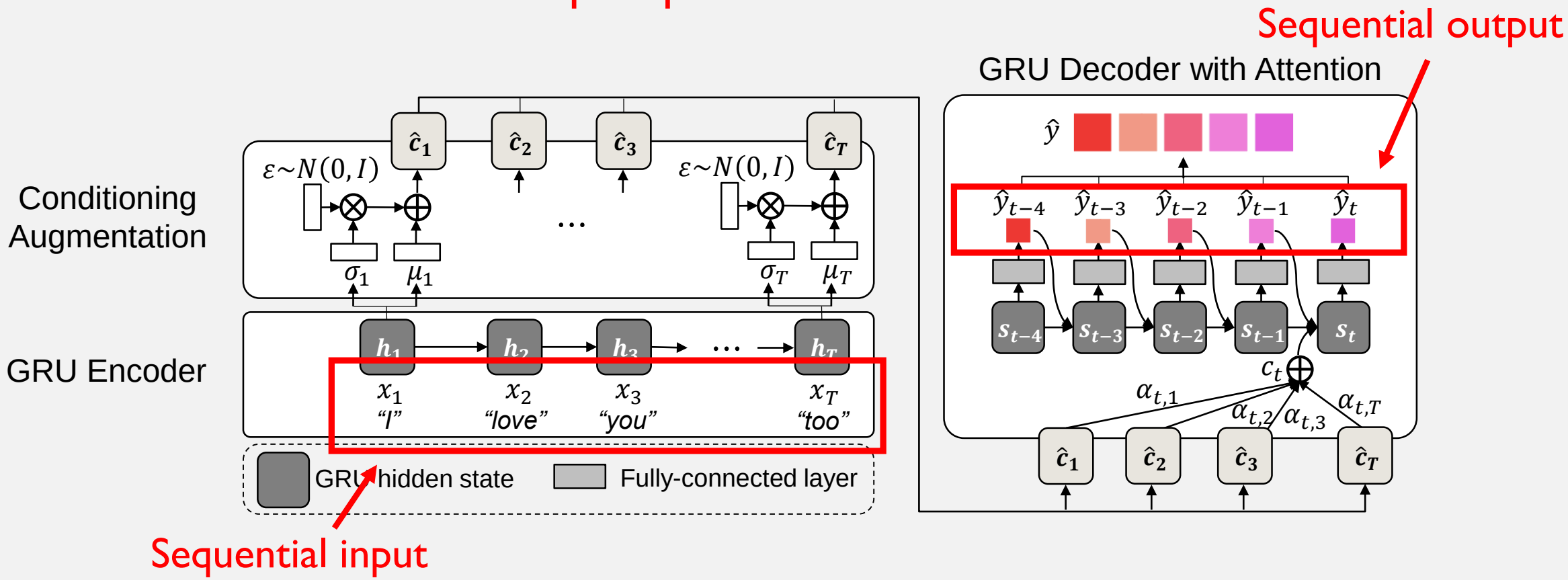
Generator Architecture



TEXT-TO-PALETTE GENERATION NETWORKS (TPN)

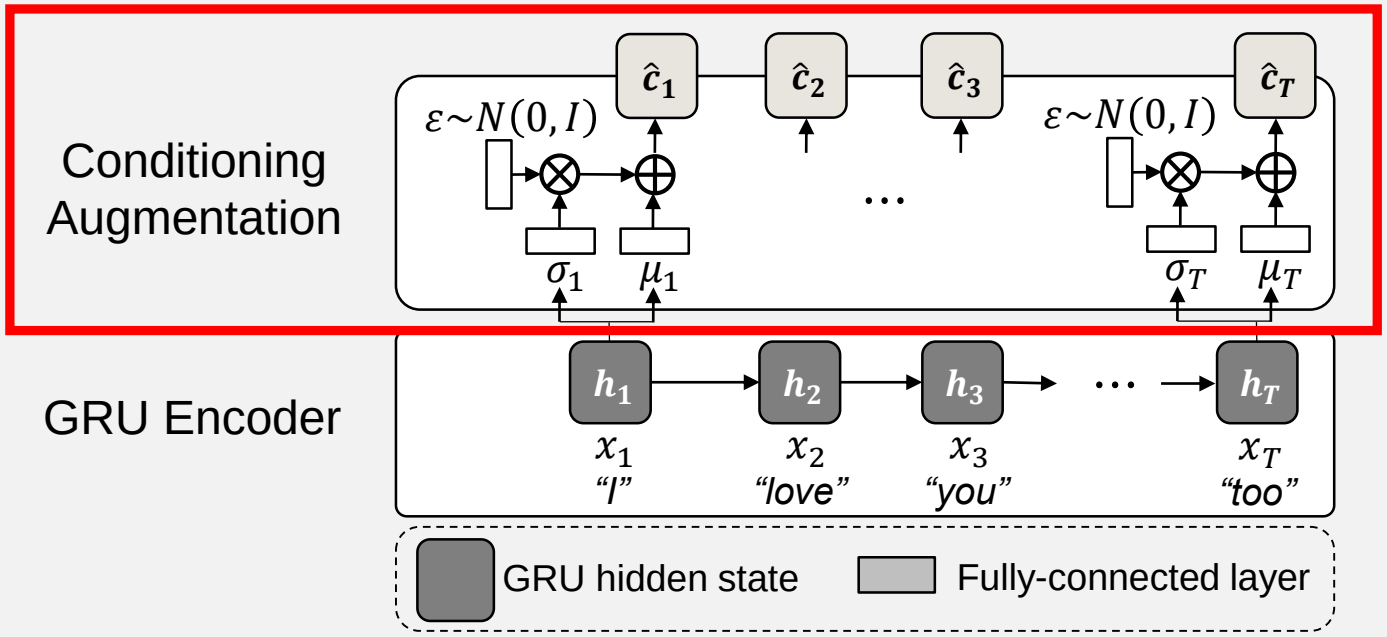
Generator Architecture

"Seq2seq with Attention"

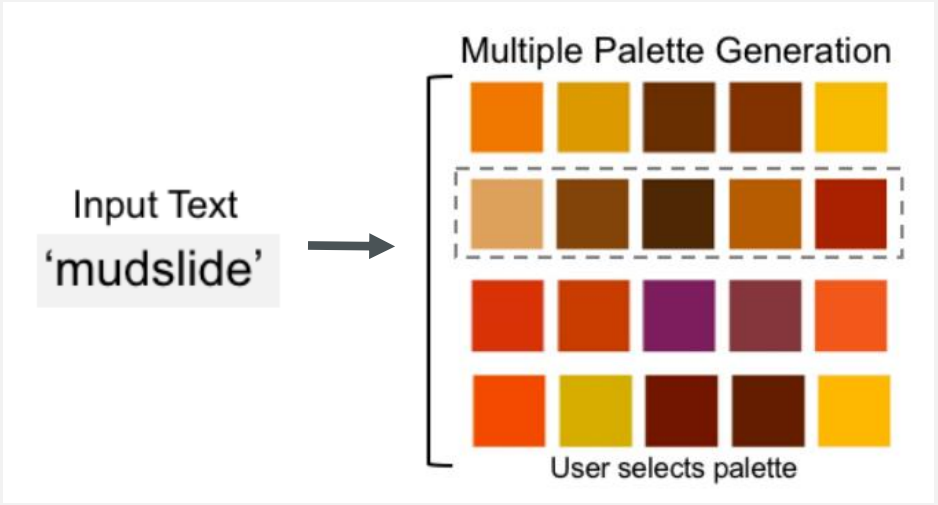


TEXT-TO-PALETTE GENERATION NETWORKS (TPN)

Generator Architecture

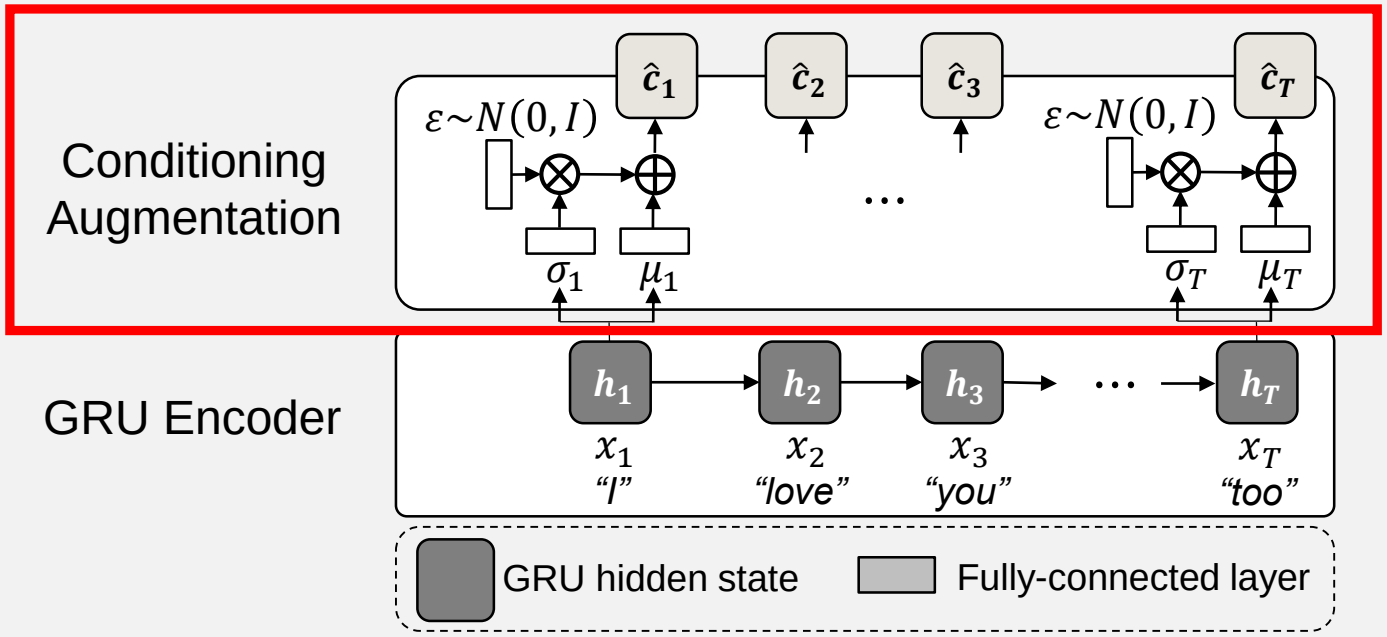


Our Goal



TEXT-TO-PALETTE GENERATION NETWORKS (TPN)

Generator Architecture



Fixed encoder outputs h

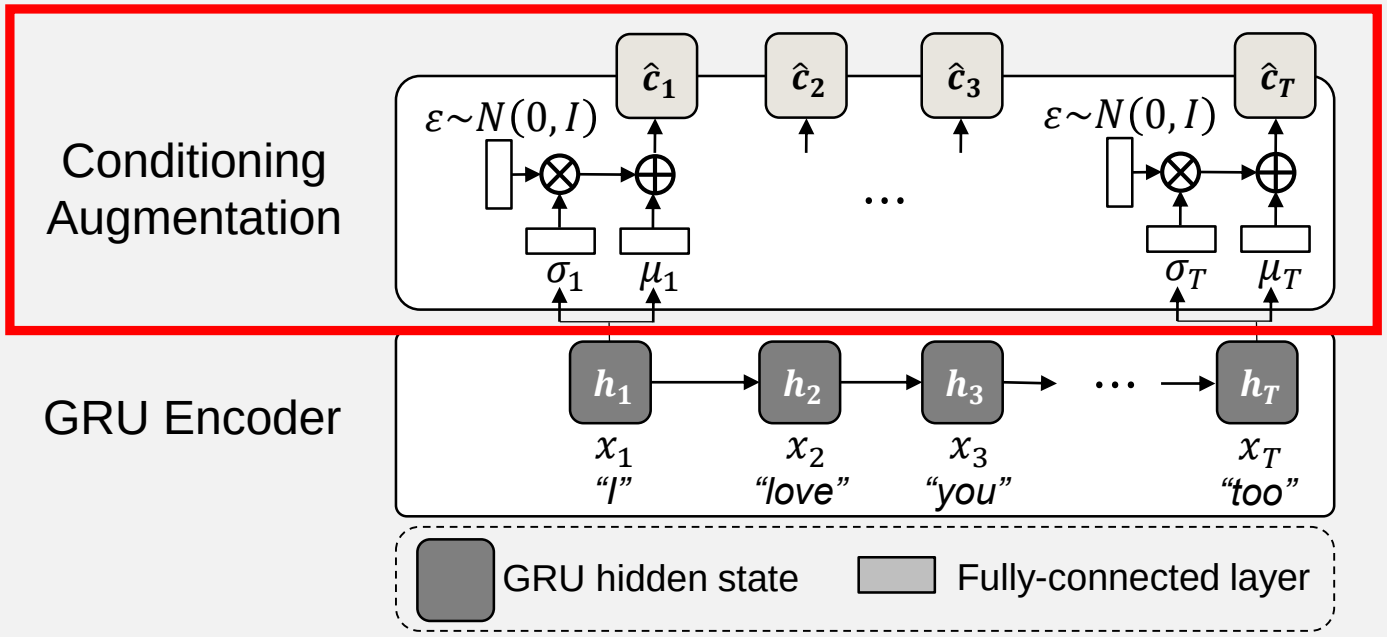
Sample conditional variable

$\hat{c} \sim N(\mu(h), \Sigma(h))$

Adding randomness!

TEXT-TO-PALETTE GENERATION NETWORKS (TPN)

Generator Architecture



Fixed encoder outputs h

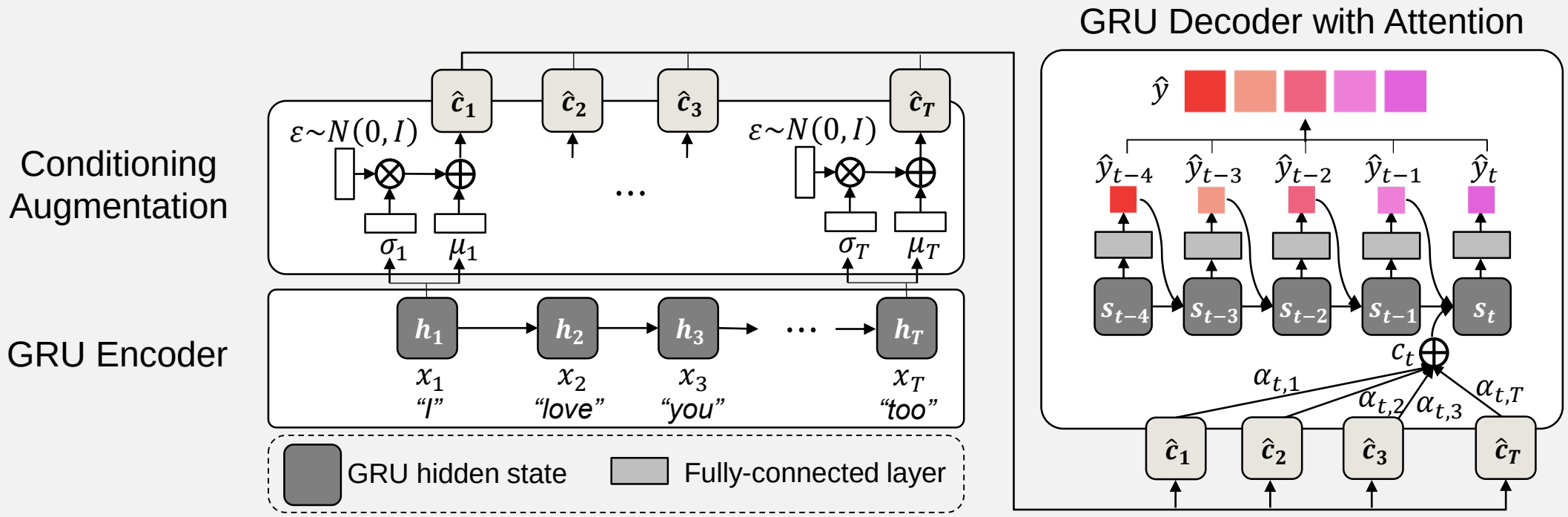
Sample conditional variable

$$\hat{c} = \mu + \sigma \odot \epsilon, \quad \epsilon \sim N(0, I)$$

Adding randomness!

TEXT-TO-PALETTE GENERATION NETWORKS (TPN)

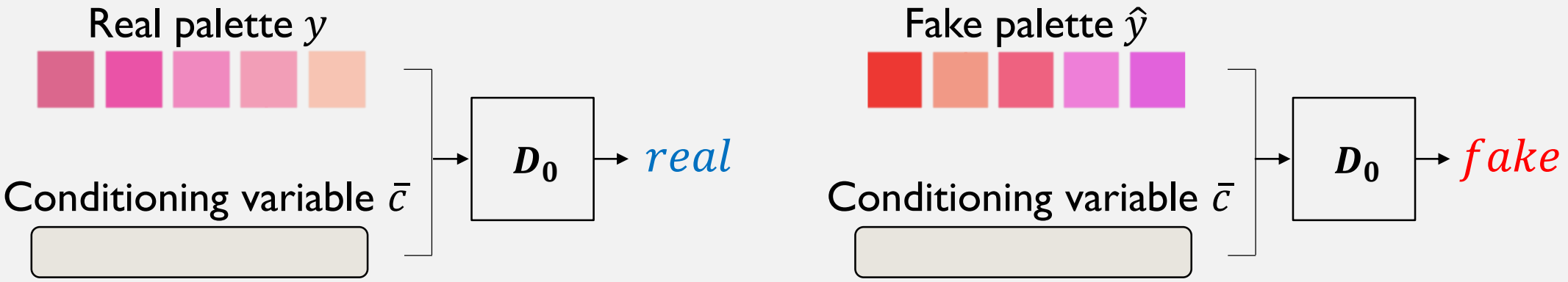
Generator Architecture



TEXT-TO-PALETTE GENERATION NETWORKS (TPN)

Training the Discriminator

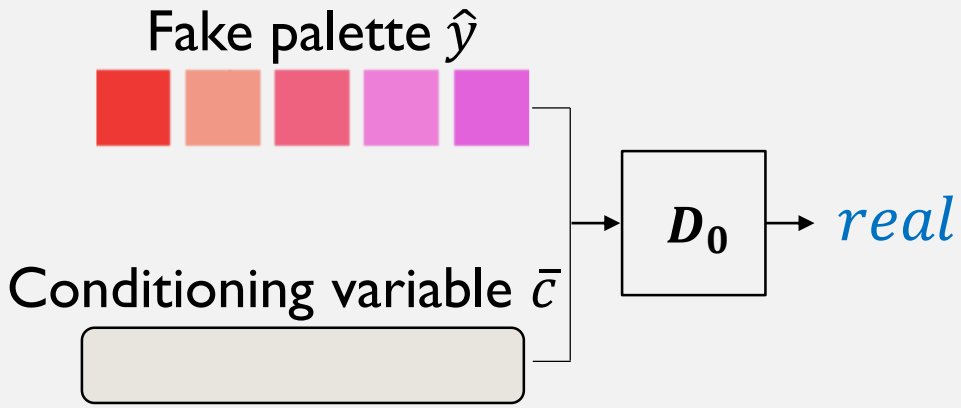
$$L_{D_0} = \mathbb{E}_{y \sim P_{data}}[\log D_0(\bar{c}, y)] + \mathbb{E}_{x \sim P_{data}}[\log(1 - D_0(\bar{c}, \hat{y}))]$$

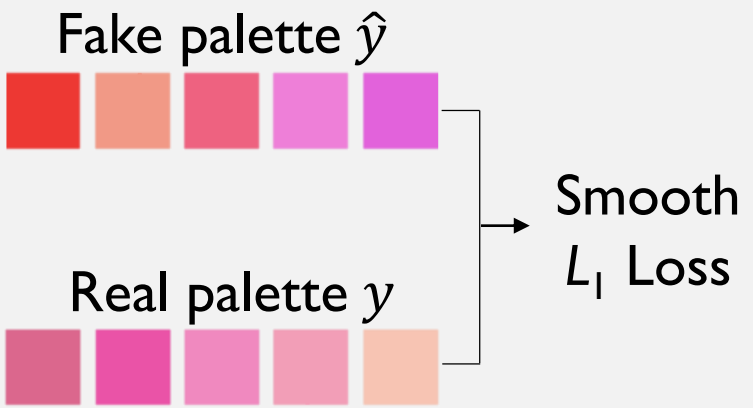


TEXT-TO-PALETTE GENERATION NETWORKS (TPN)

Training the Generator

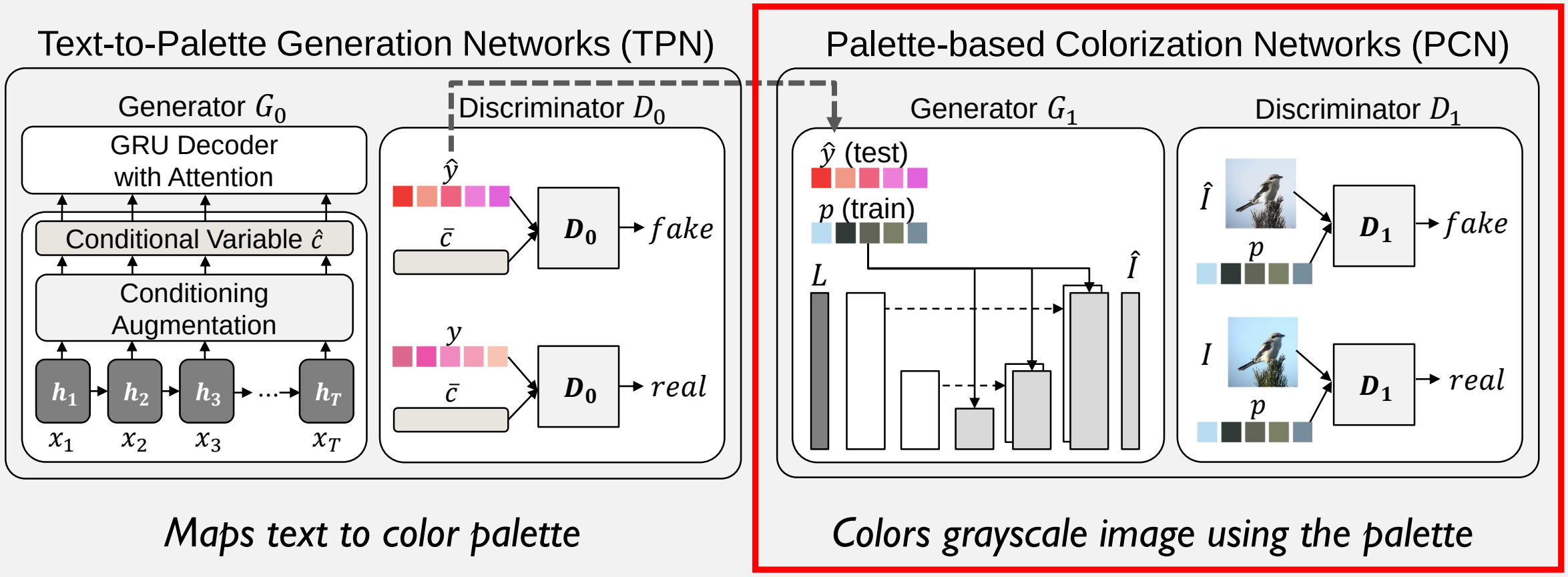
$$L_{G_0} = \mathbb{E}_{x \sim P_{data}}[\log(1 - D_0(\bar{c}, \hat{y}))] + \lambda_H L_H(\hat{y}, y) + \lambda_{KL} D_{KL}(\mathcal{N}(\mu(h), \Sigma(h)) \parallel \mathcal{N}(0, I))$$





PROPOSED METHOD (OVERVIEW)

Two Conditional GANs

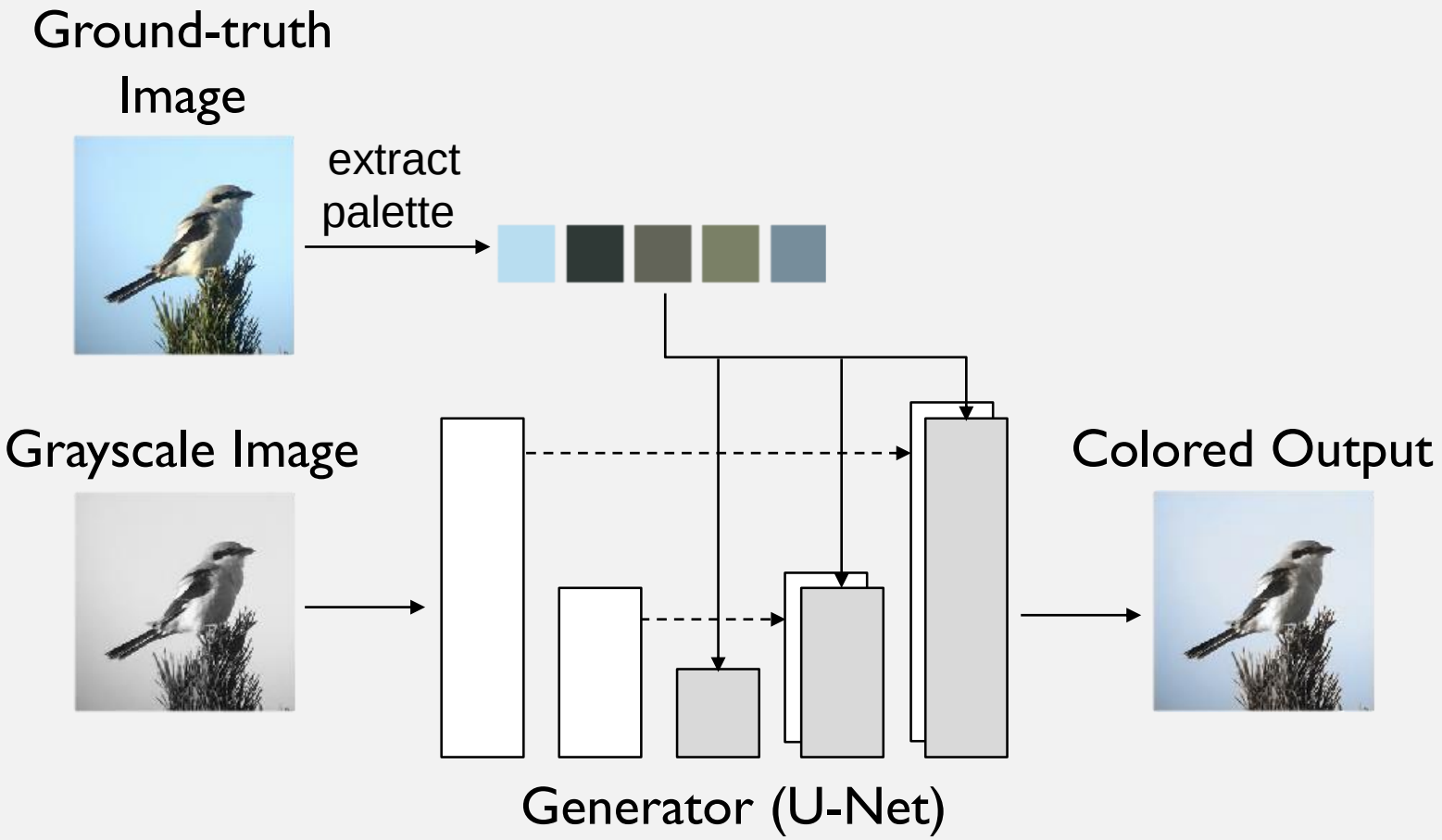


Maps text to color palette

Colors grayscale image using the palette

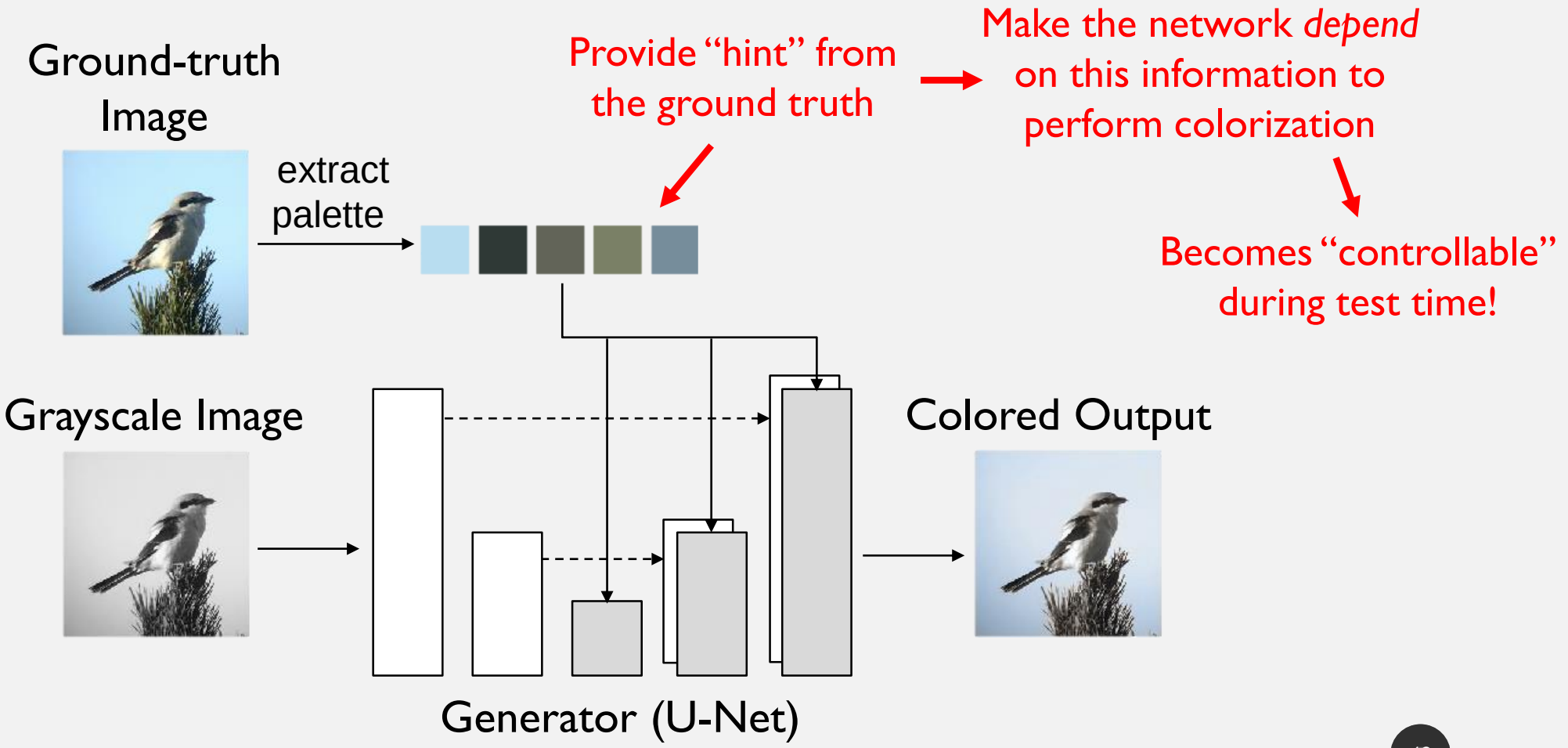
PALETTE-BASED COLORIZATION NETWORKS (PCN)

Generator Architecture (Training)



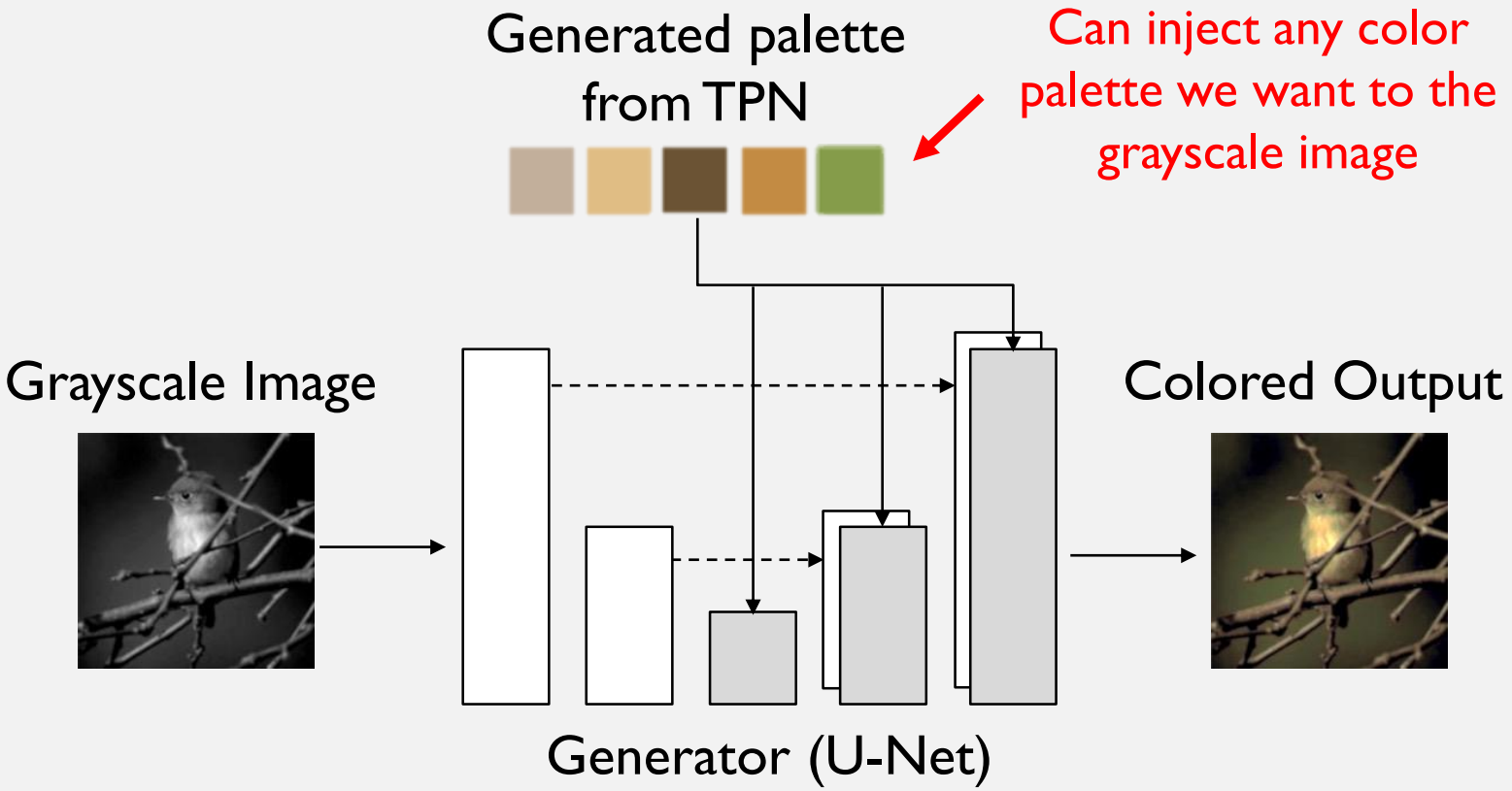
PALETTE-BASED COLORIZATION NETWORKS (PCN)

Generator Architecture (Training)



PALETTE-BASED COLORIZATION NETWORKS (PCN)

Generator Architecture (Test time)



PALETTE-BASED COLORIZATION NETWORKS (PCN)

Training the Discriminator

$$L_{D_1} = \mathbb{E}_{I \sim P_{data}}[\log D_1(p, I)] + \mathbb{E}_{\hat{I} \sim P_{G_1}}[\log(1 - D_1(p, \hat{I}))]$$

Ground-truth palette

Real image I

D_1 → $real$

Ground-truth palette

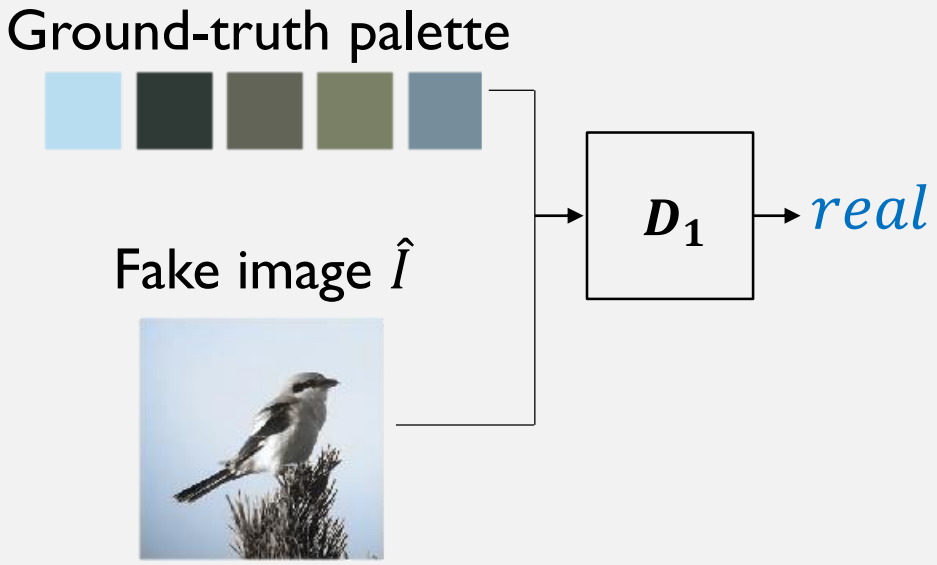
Fake image $\hat{I}$

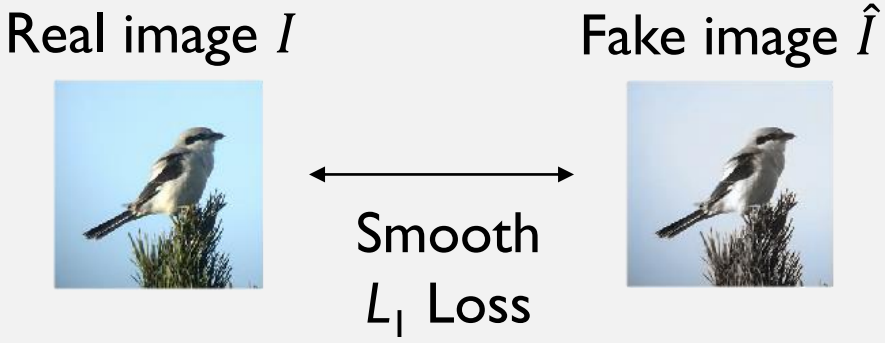
D_1 → $fake$

PALETTE-BASED COLORIZATION NETWORKS (PCN)

Training the Generator

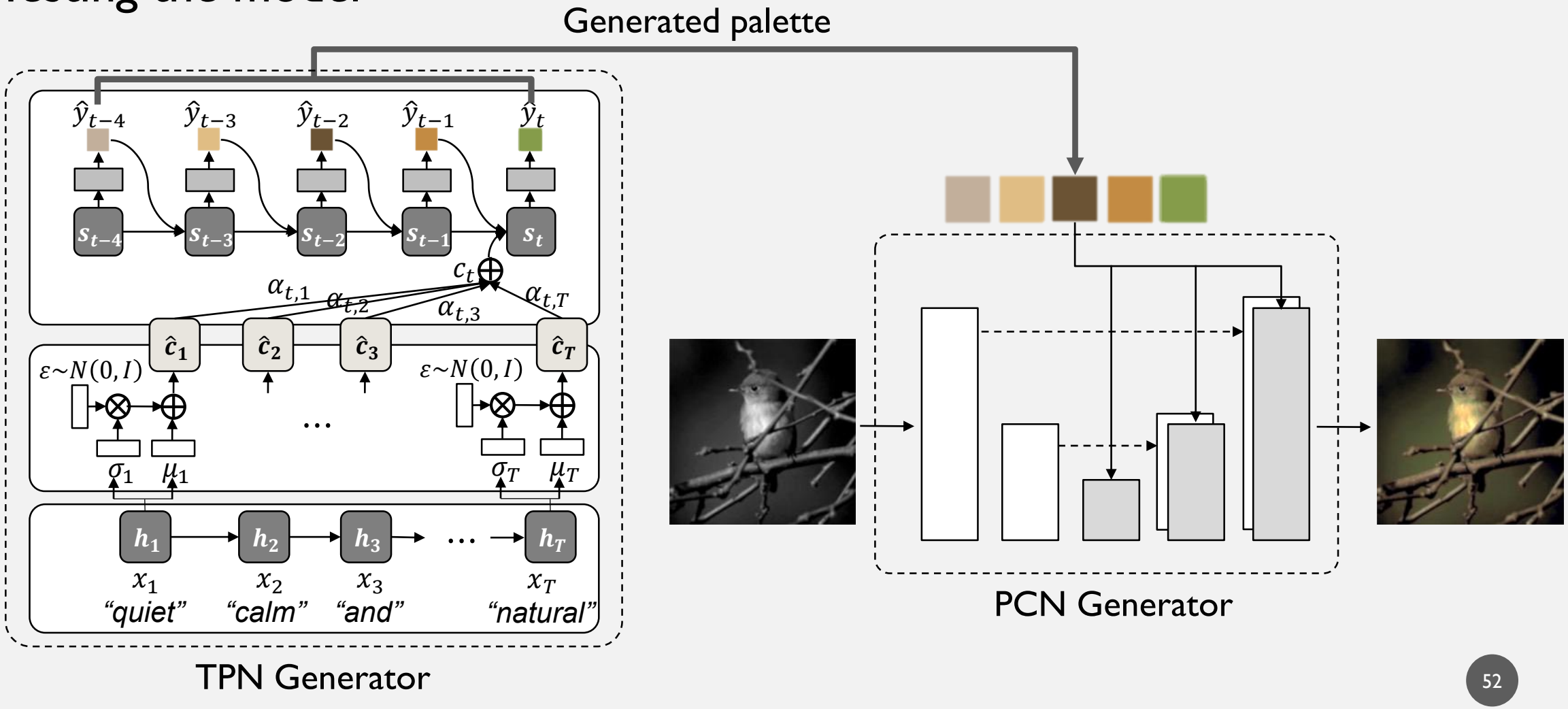
$$L_{G_1} = \mathbb{E}_{\hat{I} \sim P_{G_1}} [\log(1 - D_1(p, \hat{I}))] + \lambda_H L_H(\hat{I}, I)$$





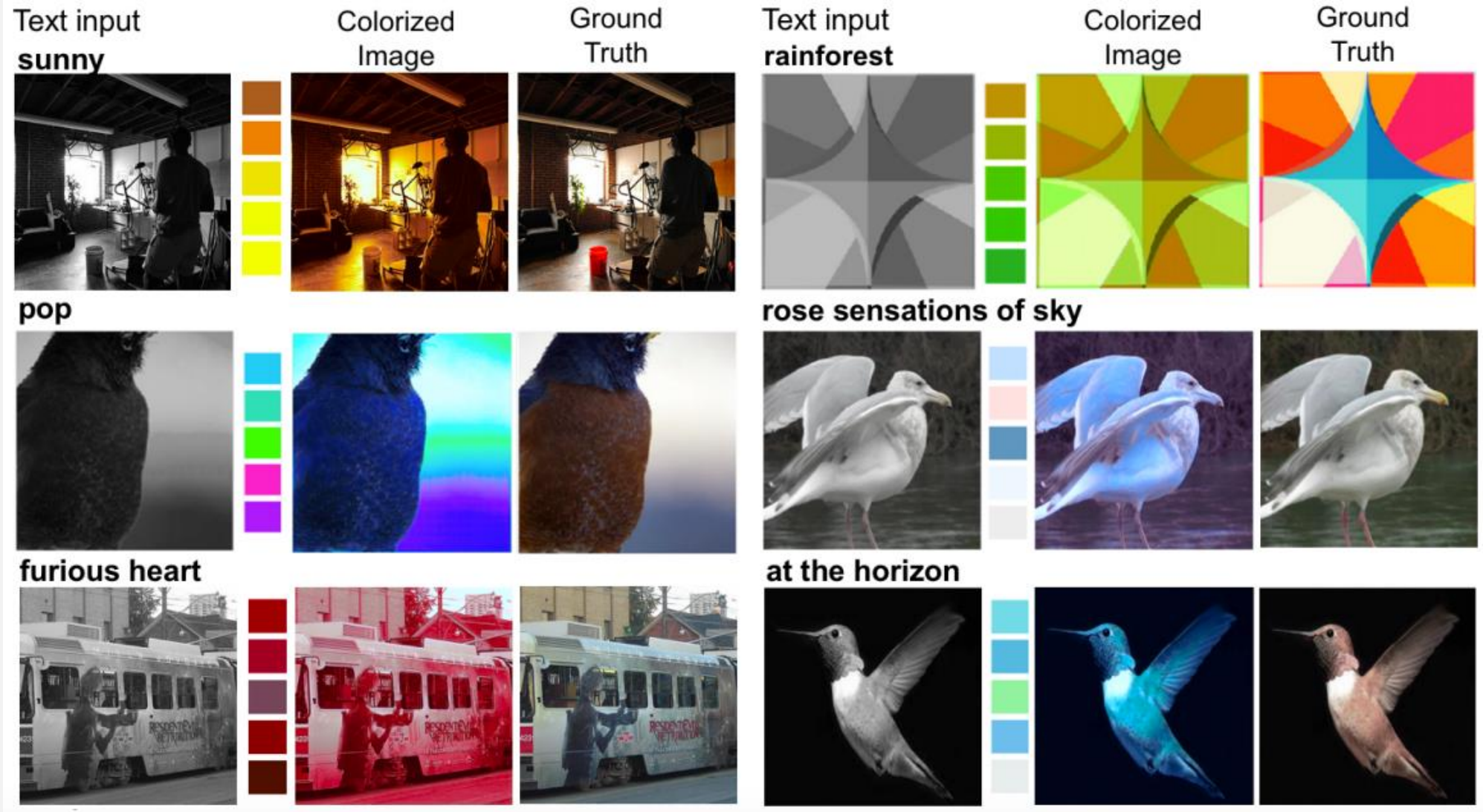
TEXT2COLORS

Testing the model



EXPERIMENTS

EXPERIMENTS: QUALITATIVE RESULTS

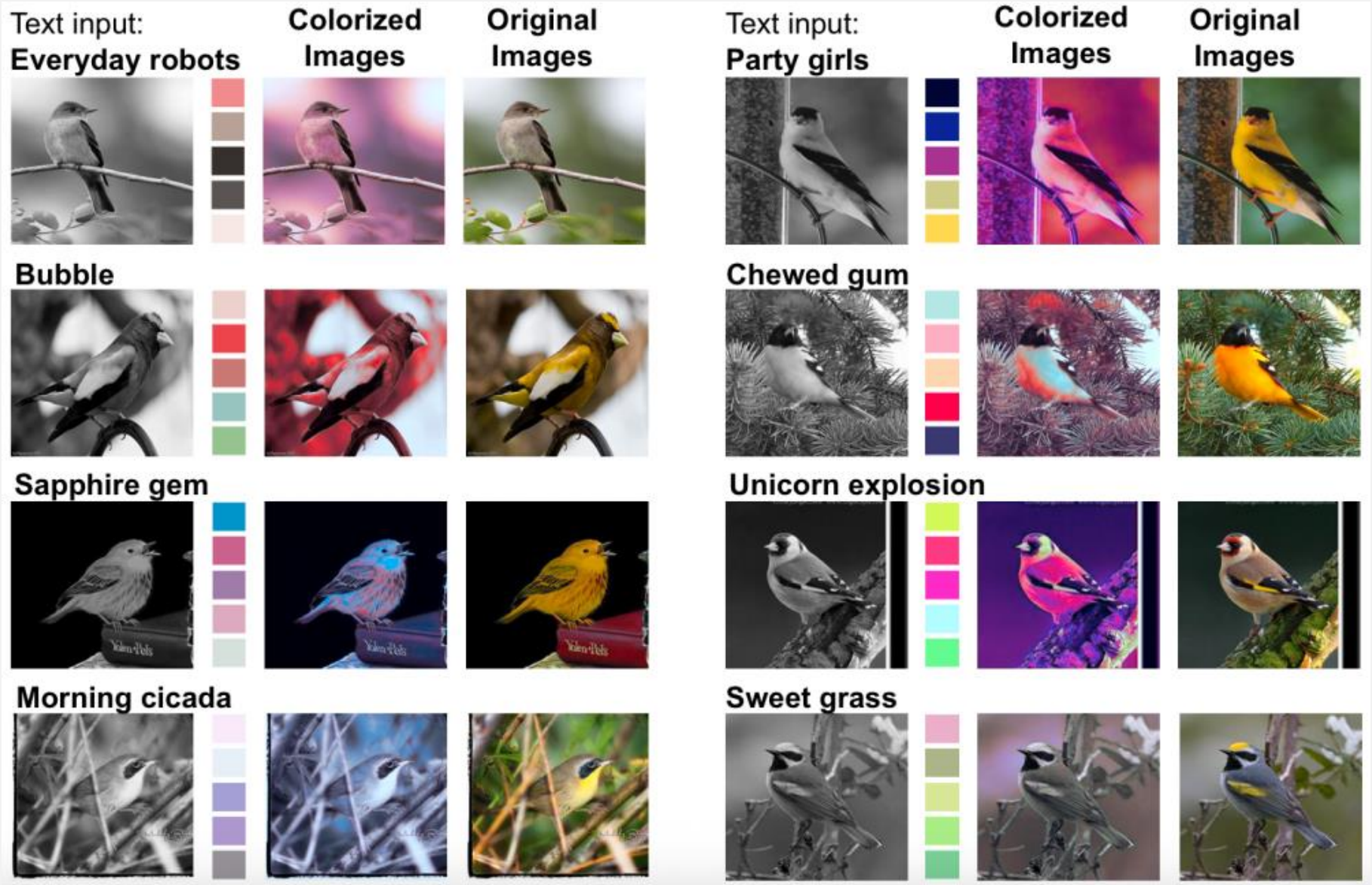


EXPERIMENTS: QUALITATIVE RESULTS

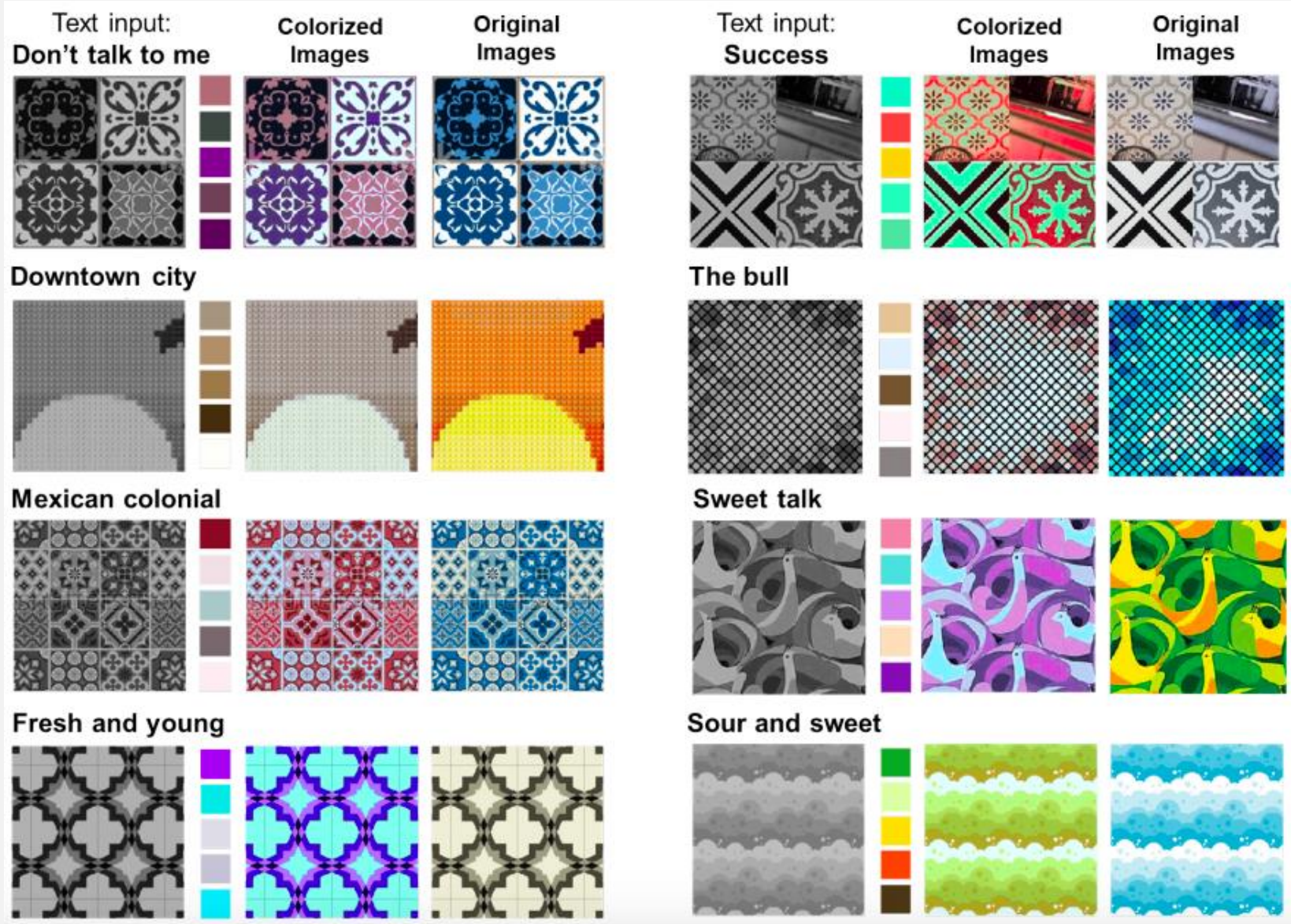
Fig. 14. Handling phrase-level inputs about 'love'.

EXPERIMENTS: QUALITATIVE RESULTS

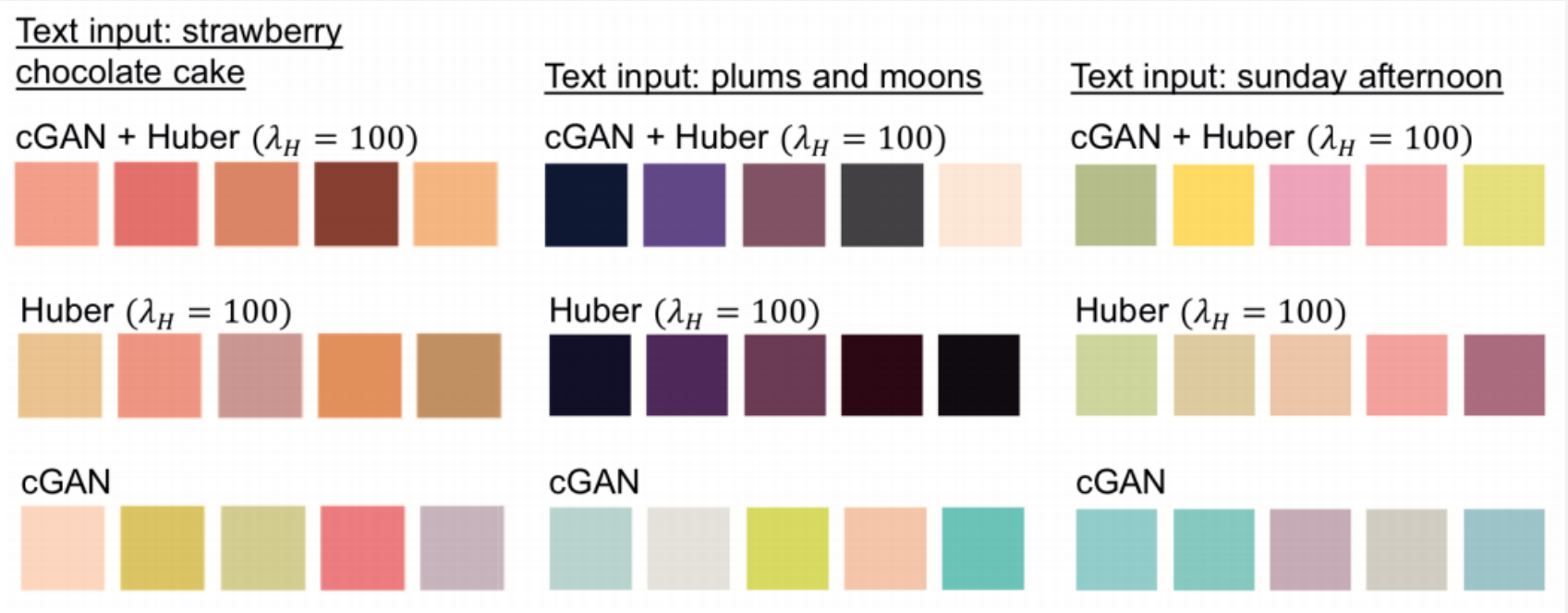
EXPERIMENTS: QUALITATIVE RESULTS

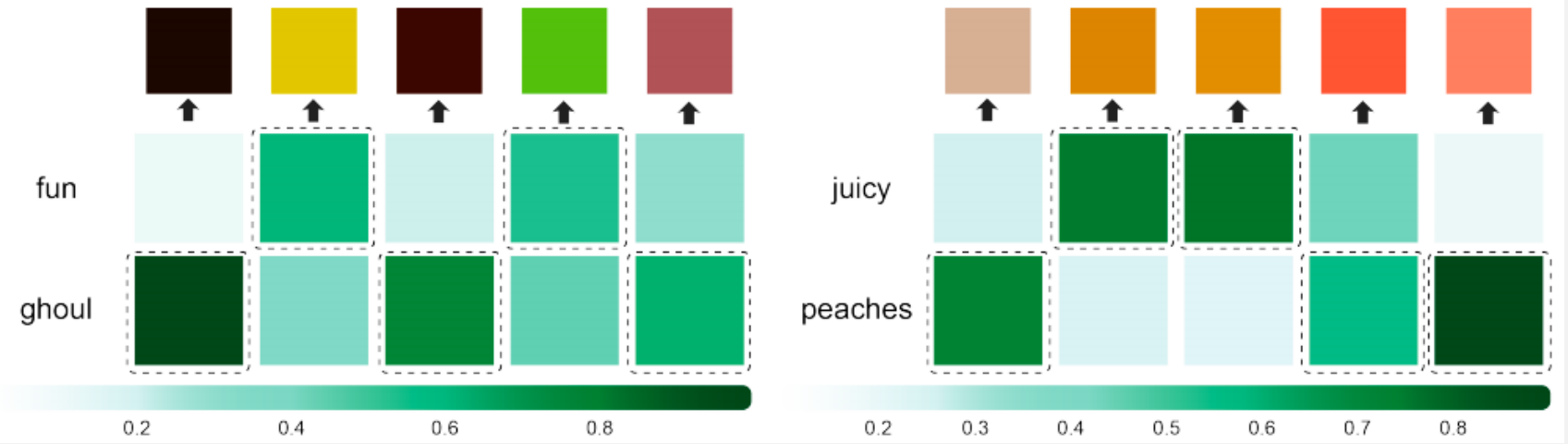


EXPERIMENTS: QUALITATIVE RESULTS



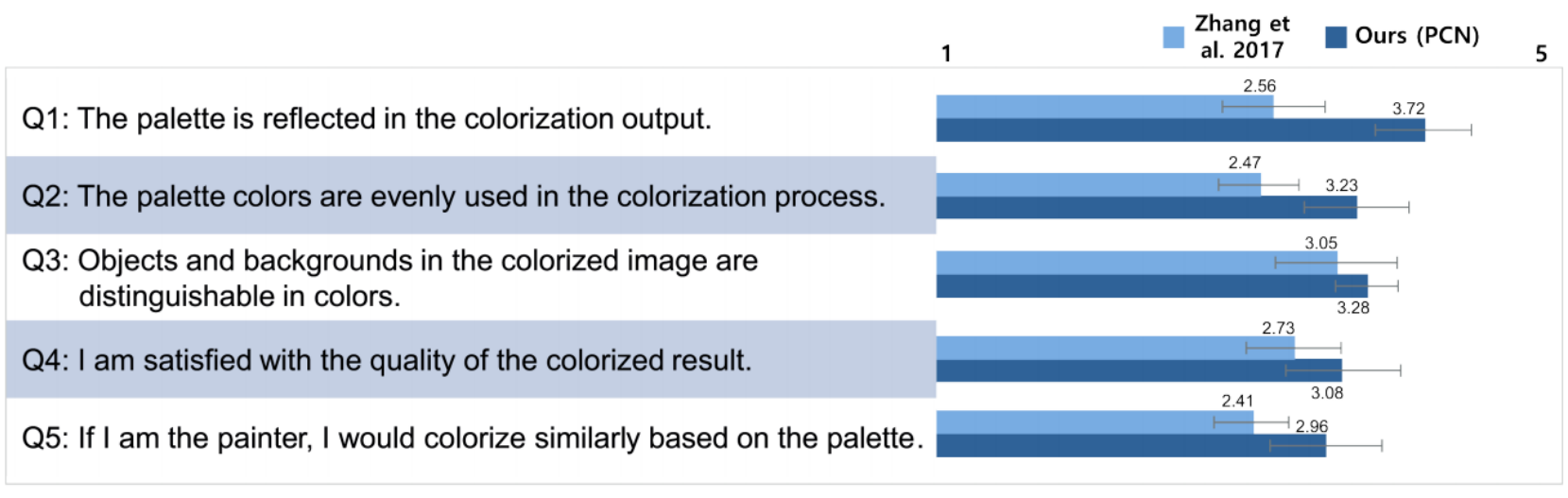
EXPERIMENTS: ABLATION STUDY & ATTENTION MECHANISM



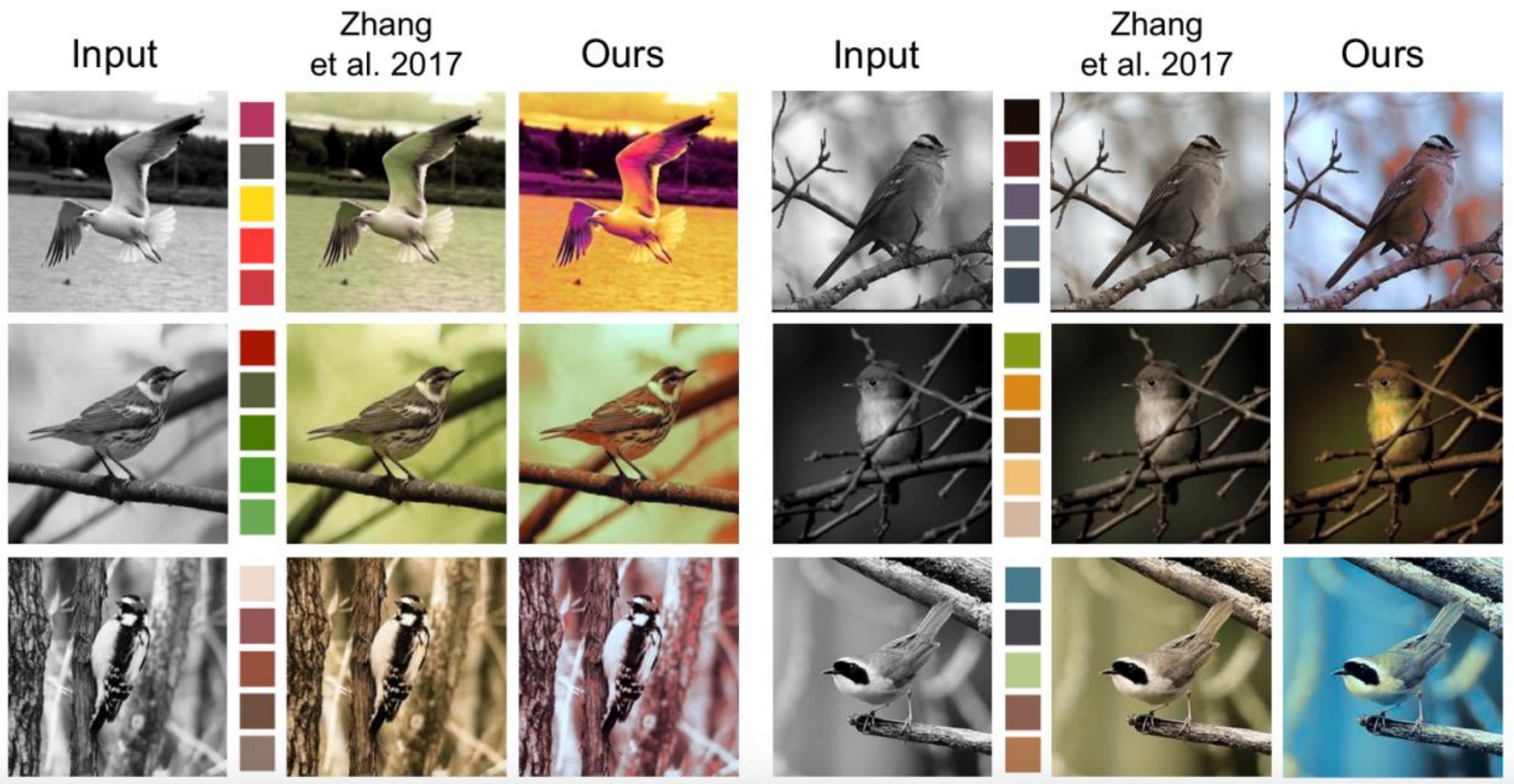


EXPERIMENTS: QUANTITATIVE COMPARISON

		Palette Evaluation				User Study: Part I			
Model Variations		**Diversity**		**Multimodality**		**Fooling Rate (%)**			
Objective Function	CA	Mean	Std	Mean	Std	Mean	Std	Max	Min
Ours (TPN)	X	19.36	8.74	0.0	0.0	-	-	-	-
Ours (TPN)	O	20.82	7.43	5.43	8.11	**56.2**	12.7	**76.7**	37.1
Heer and Stone	-	35.92	12.66	0.0	0.0	39.6	10.8	58.2	25.8
Ground truth palette	-	32.60	21.84	-	-	-	-	-	-



EXPERIMENTS: QUALITATIVE COMPARISON



FAILURE CASES

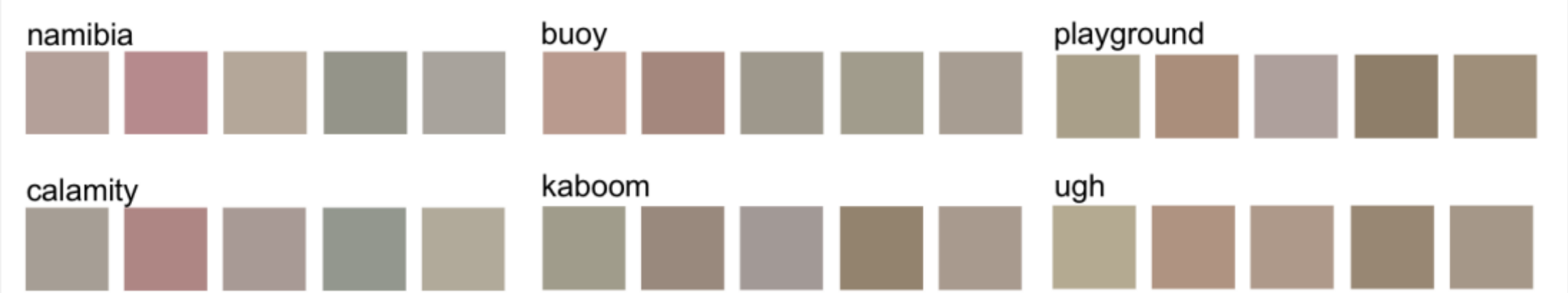


Fig. 16. Failed results of TPN. Our model fails and outputs the same washed-out grayish-brown color palettes for unknown tokens.

Overview of This Talk

- Intro to conditional generative models

- My own research on interactive automatic colorization
  - Colorization using natural language [ECCV'18]
  - Few-shot colorization via memory networks [CVPR'19]
  - Reference-based sketch colorization using augmented self-exemplar [CVPR'20]

- Other work on interactive generative models and future research directions

COLORING WITH LIMITED DATA :
FEW-SHOT COLORIZATION VIA MEMORY-AUGMENTED NETWORKS (CVPR 2019)

Seungjoo Yoo, Hyojin Bahng, Sunghyo Chung,
Junsoo Lee, Jaehyuk Chang, and Jaegul Choo

MOTIVATION

- Data Scarcity
  - Existing deep learning colorization models require a significant amount of training data
  - Data for animations and cartoons are often small in number

ImageNet

# of images: 14,197,122

유미의 세포들

# of images: 9600

MOTIVATION

- Data Scarcity in cartoons and animations

Requires significant cost for creating animations

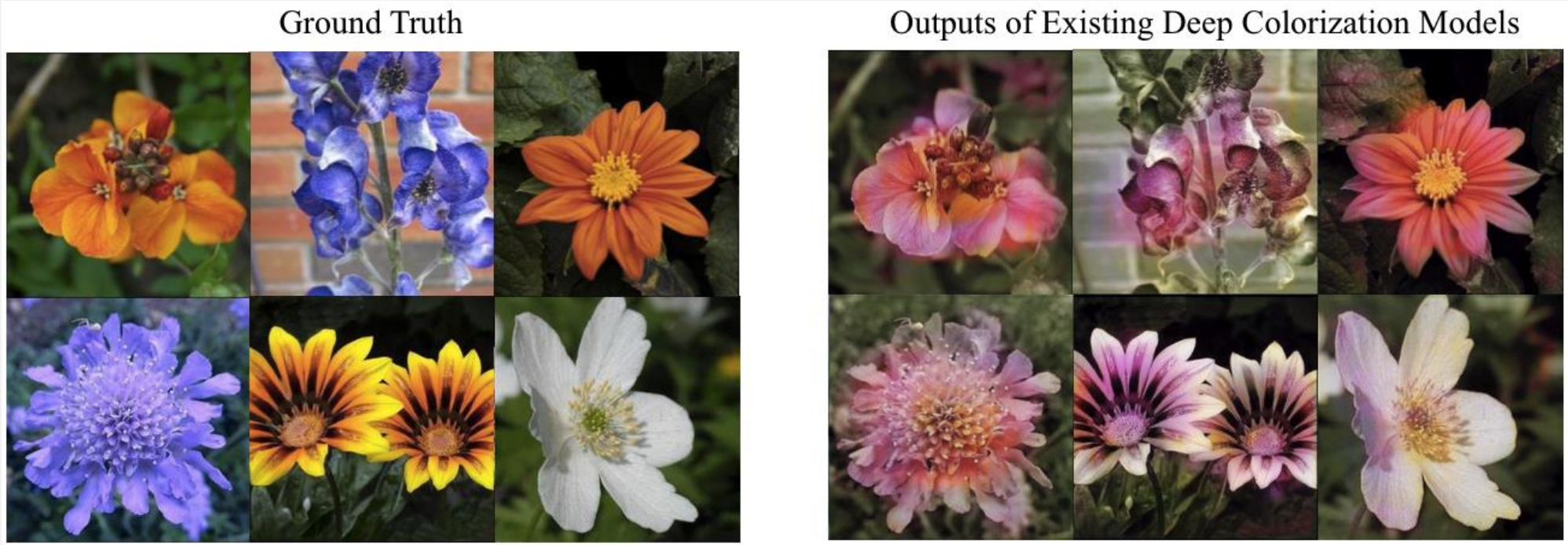
→

Black-and-White 3D modeling

Colorization

MOTIVATION

- Dominant Color Effect
  - Deep colorization models tend to ignore diverse colors present in a training set and learn only a few dominant colors. This can minimize overall loss, but has unsatisfactory results.
  - This can minimize the overall loss, but has unsatisfactory results



MOTIVATION

- Dominant Color Effect

Real-world image

Ground Truth

Output

Dominant color: brown, green

Cartoon

Ground Truth

Output

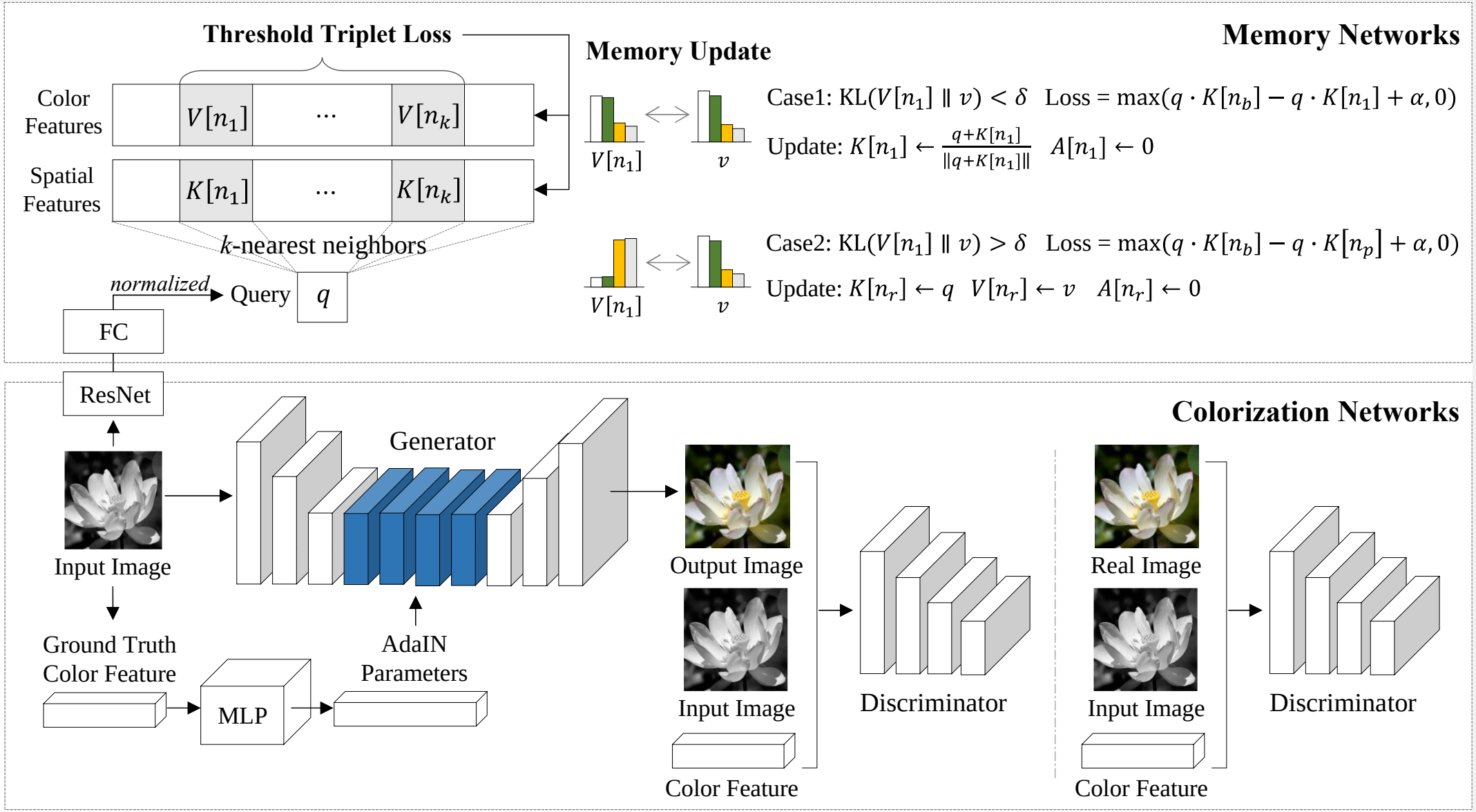
Dominant color: yellow, blue

MODEL OVERVIEW

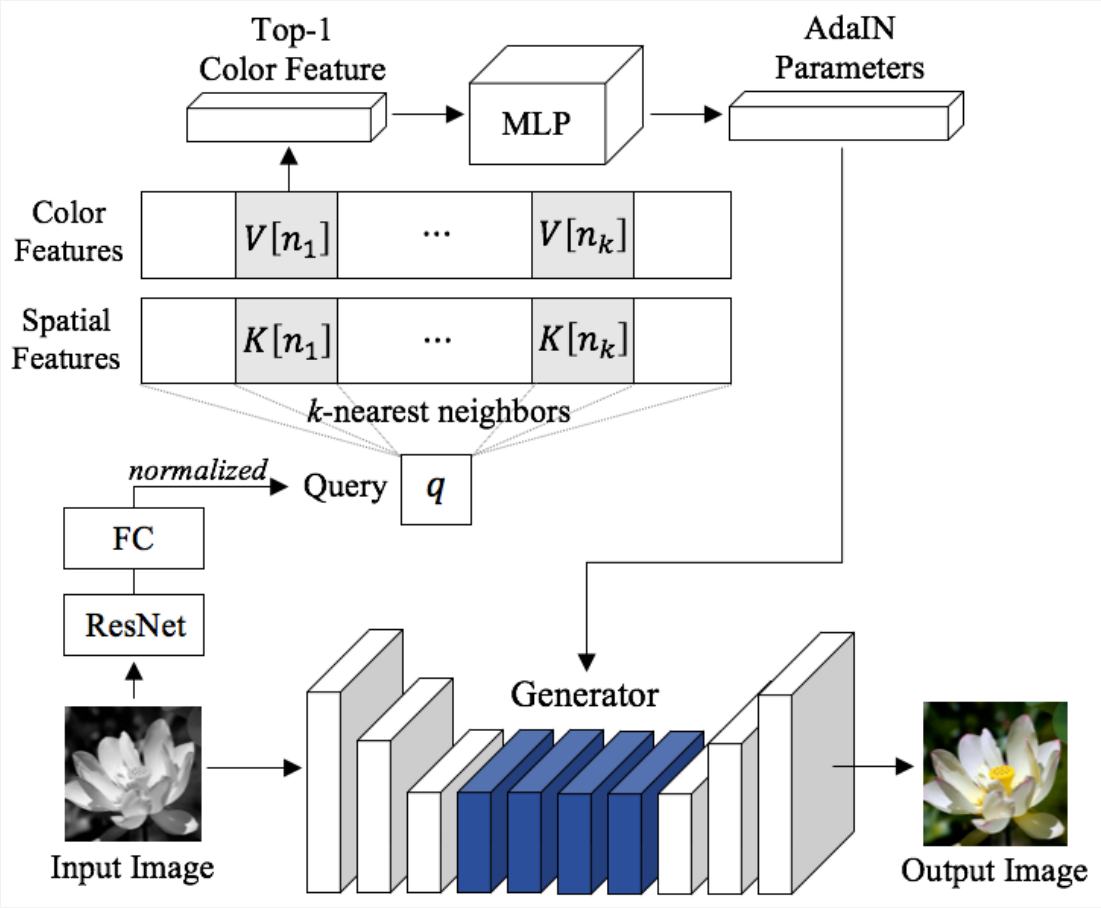
Our memory-augmented colorization model MemoPainter:

- Can color rare instances and suffers less from Dominant Color Effect
- Can be trained with very little data, even one-shot and few-shot learning
- Introduce a novel Threshold Triplet Loss for unsupervised training of memory networks.

MODEL ARCHITECTURE

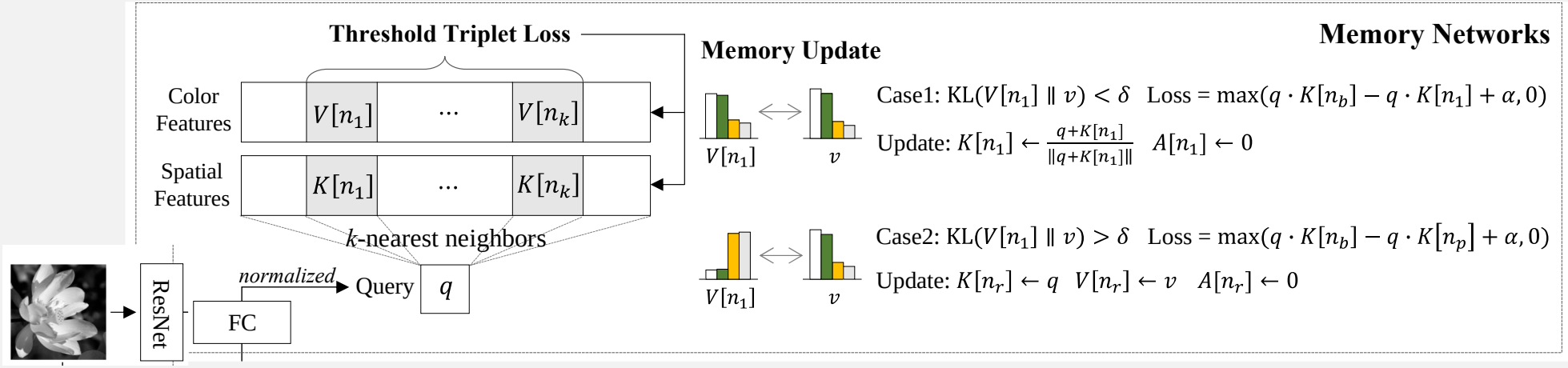


MODEL ARCHITECTURE



- During test time, we retrieve the top-1 color feature from our memory and give it as a condition to the trained generator

MODEL ARCHITECTURE



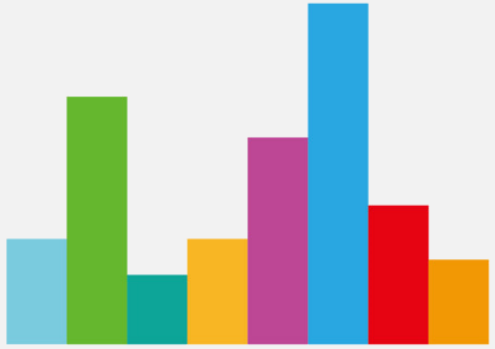
- Stores three different types of information: key memory, value memory, and age.

$$M = (K_1, V_1, A_1), (K_2, V_2, A_2), ..., (K_m, V_m, A_m)$$

  - Key: spatial features
  - Value: color features
  - Age: age of memory slot

MODEL ARCHITECTURE

Color features



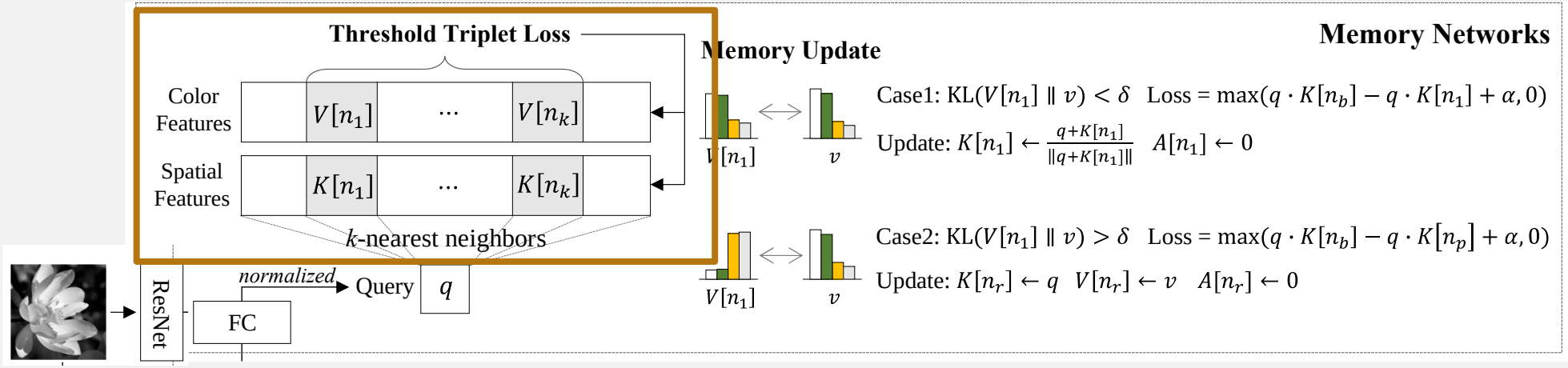
Color dist

Top 10 dominant RGB colors (taken from color thief)

- We leverage two variants to represent color information stored in value memory:
  - Color distributions: color distributions over 313 quantized color values
  - RGB color values: set of ten dominant RGB color values of an image

$$V = C_{dist} \text{ or } C_{RGB}.$$

MODEL ARCHITECTURE



- Query computation

$$q = WX_{rp5} + b,\ q = \frac{q}{\|q\|},$$

- Nearest-neighbor computation

$$\mathrm{NN}(q, M) = \mathrm{argmax}_i\ q \cdot K\ [i],$$
$$(n_1, ..., n_k) = \mathrm{NN}_k(q, M),$$

THRESHOLD TRIPLET LOSS

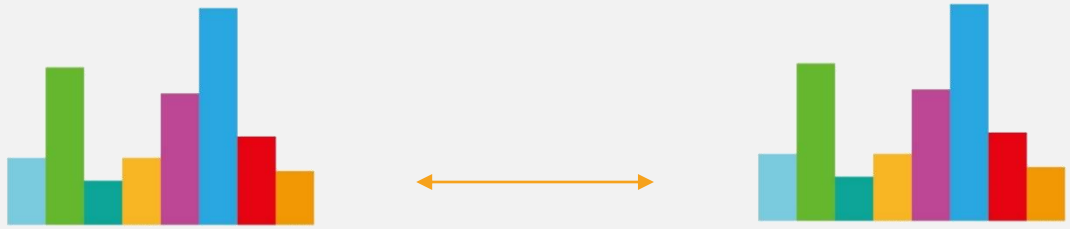
- Goal of triplet loss: making those images of a specific class closer to each other (positive neighbors) than it is to images of any other class (negative neighbors).

Example label: Girl, restaurant, blue clothes, yellow hair

Blue background, two people, talking

- Class label information is not available in most data for colorization tasks.

THRESHOLD TRIPLET LOSS



- We assume that given two images, if they **have similar spatial features** and the **distance between their color distribution** are within a certain threshold, they are likely to be in the same class.

Positive neighbor

$$\mathrm{KL}(V[n_p] \parallel v) < \delta.$$

Negative neighbor

$$\mathrm{KL}(V[n_b] \parallel v) > \delta.$$

Threshold Triplet Loss

$$L_t(q, M, \delta) = \max(q \cdot K[n_b] - q \cdot K[n_p] + \alpha, 0).$$

Memory Update

- Our memory *M* is updated after a new query *q* is introduced to the network.

(i) If the distance between $V[n_1]$ and *v* is within the color threshold, we update the key by averaging $K[n_1]$ and *q* and normalizing it. Age is also reset to zero.

$$\mathrm{KL}(V[n_p] \parallel v) < \delta.$$

$$K[n_1] \leftarrow \frac{q + K[n_1]}{\parallel q + K[n_1] \parallel}, \quad A[n_1] \leftarrow 0.$$

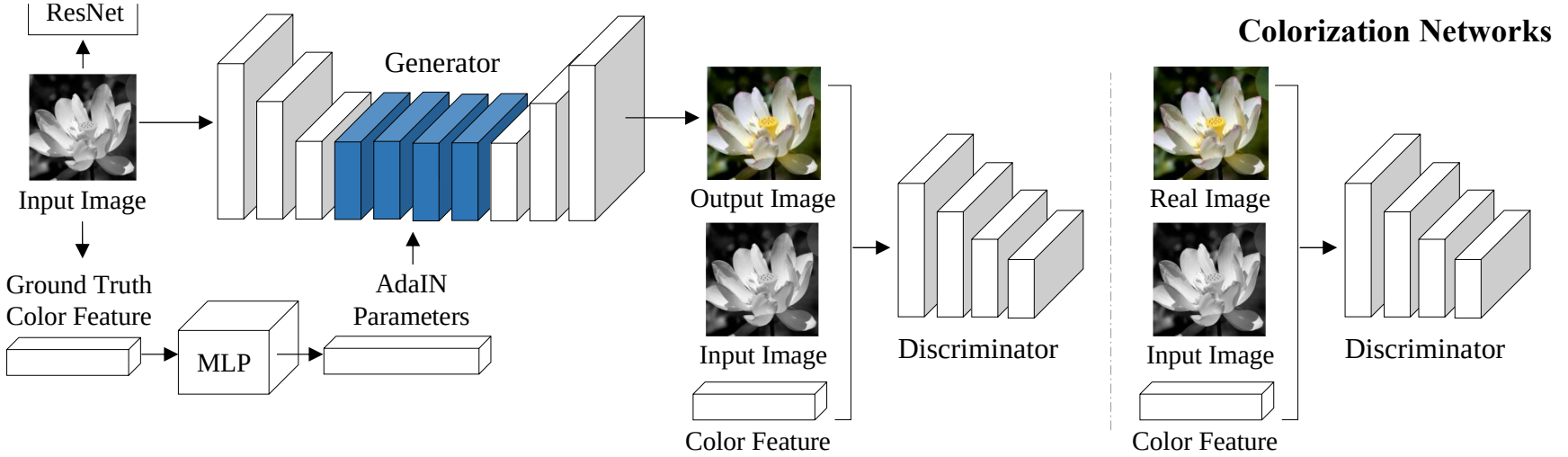
(ii) If the distance between $V[n_1]$ and *v* exceeds color threshold δ, this indicates that there exists no memory slot available in our current memory that matches v. Thus, (*q*, *v*) will be newly written into the memory.

$$\mathrm{KL}(V[n_b] \parallel v) > \delta.$$

$$K[n_r] \leftarrow q, V[n_r] \leftarrow v_q, A[n_r] \leftarrow 0.$$

Model architecture

Colorization networks



- Conditional GAN

$$L_D = \mathbb{E}_{x \sim P_{data}}[\log D(x, C, y)] + \mathbb{E}_{x \sim P_{data}}[(1 - \log D(x, C, G(x, C)))],$$

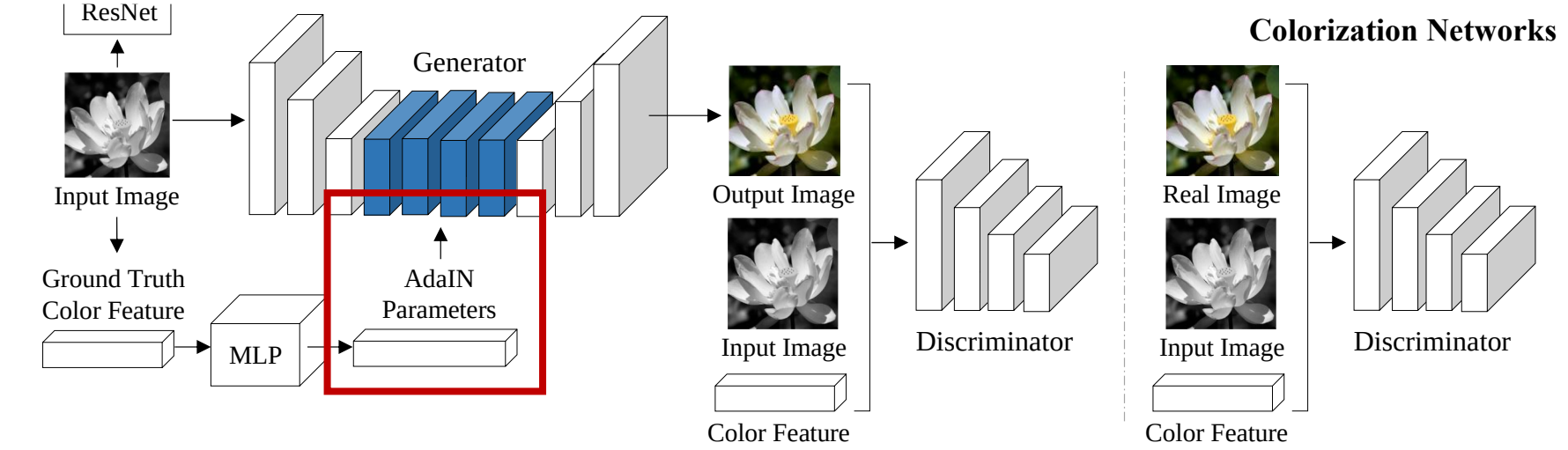
$$L_G = E_{x \sim P_{data}}[(1 - \log D(x, C, G(x, C)))] + L_{sL1}(y, G(x, C)).$$

- Smooth L1 Loss

$$L_{sL1}(y, \hat{y}) = \begin{cases} \frac{1}{2}(y - \hat{y})^2 & \text{for } |y - \hat{y}| \leq \delta \\ \delta |y - \hat{y}| - \frac{1}{2}\delta^2 & \text{otherwise.} \end{cases}$$

Colorization networks

Colorization networks



- Coloring with Adaptive Instance Normalization (AdaIN)

$$\text{AdaIN}(z, C) = \sigma(C)\left(\frac{z - \mu(z)}{\sigma(z)}\right) + \mu(C),$$

Colorization networks

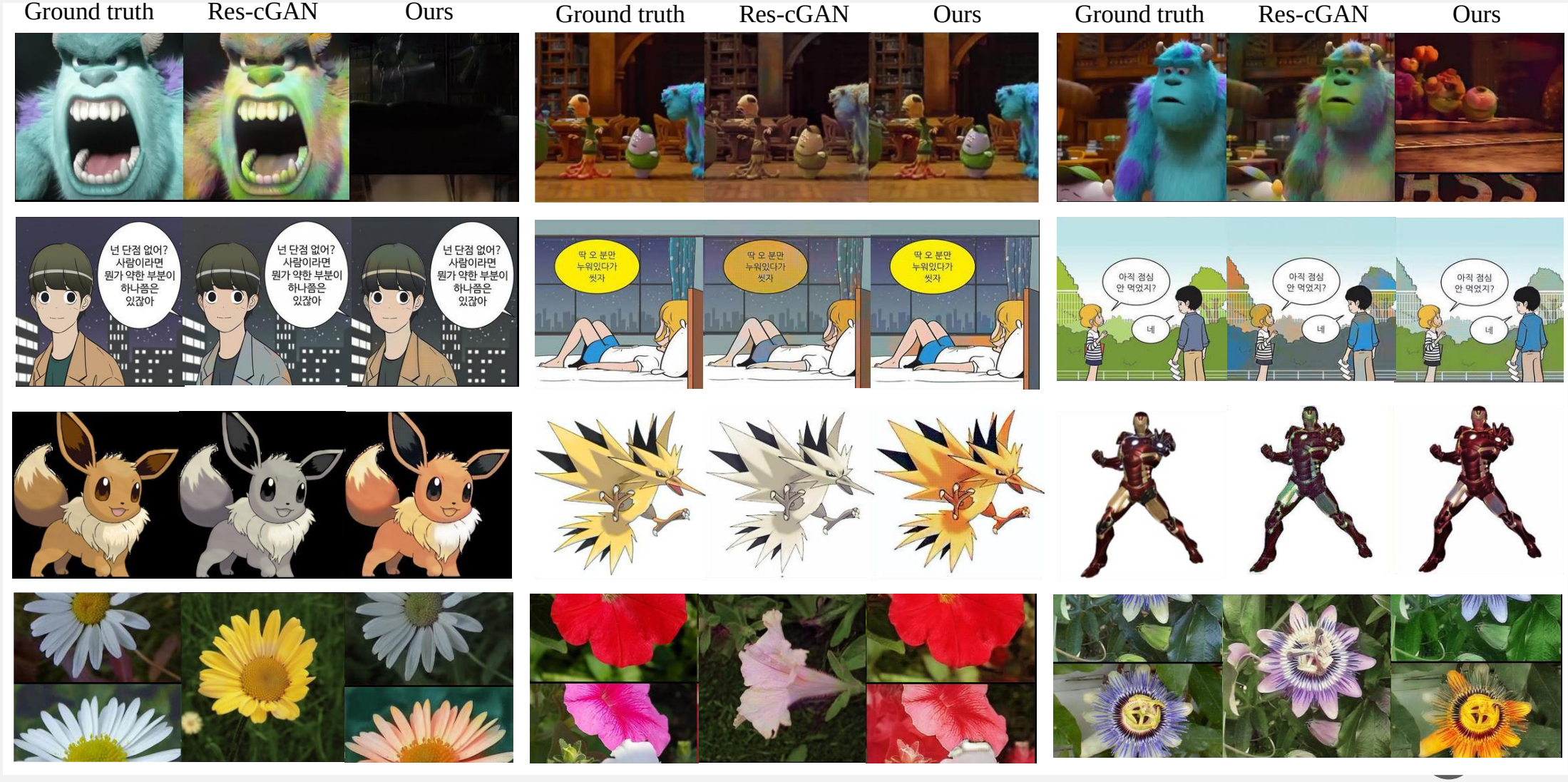
- Using **color distribution** like style information

Color distribution

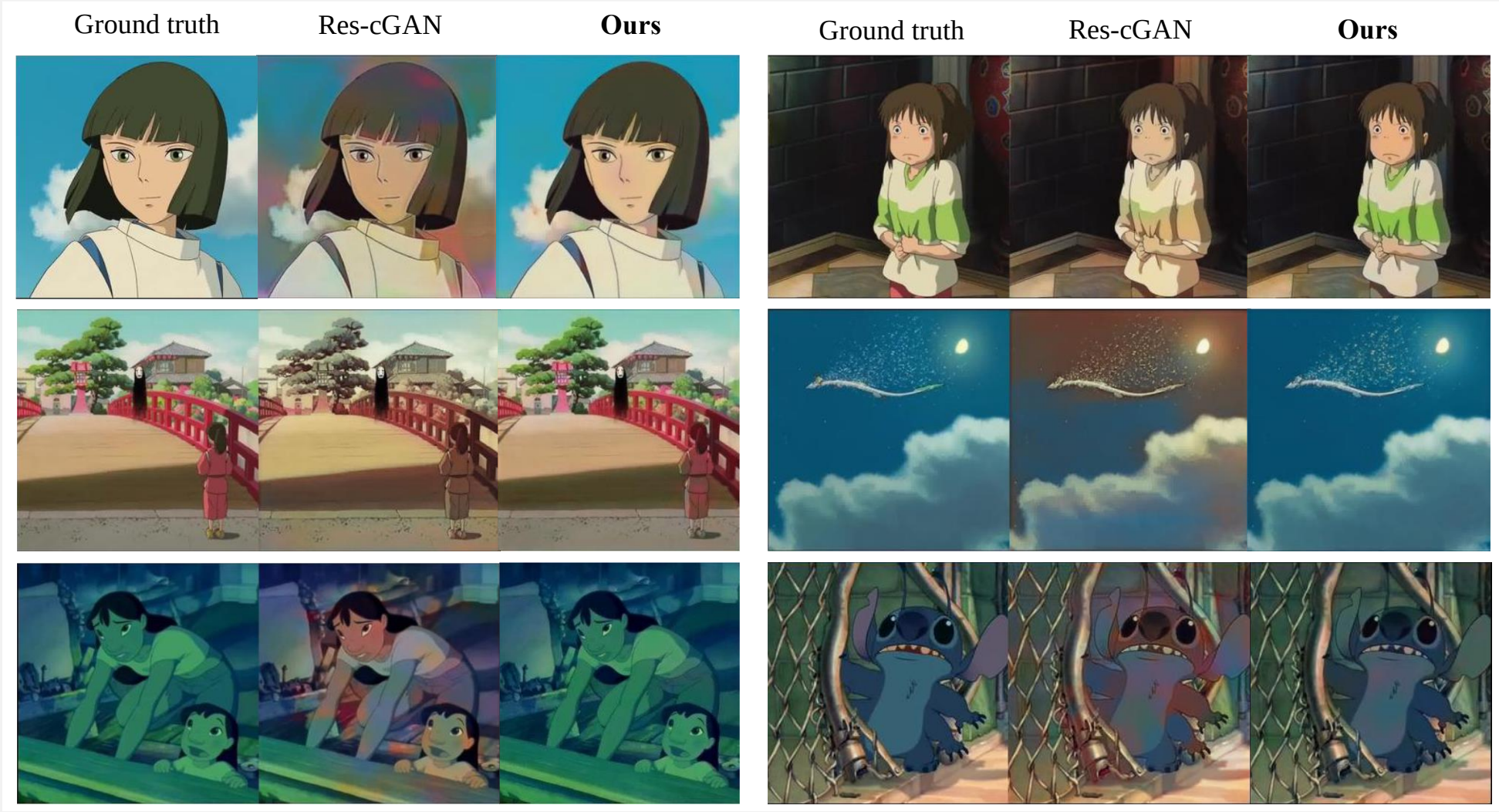
Instead of

Image

RESULTS

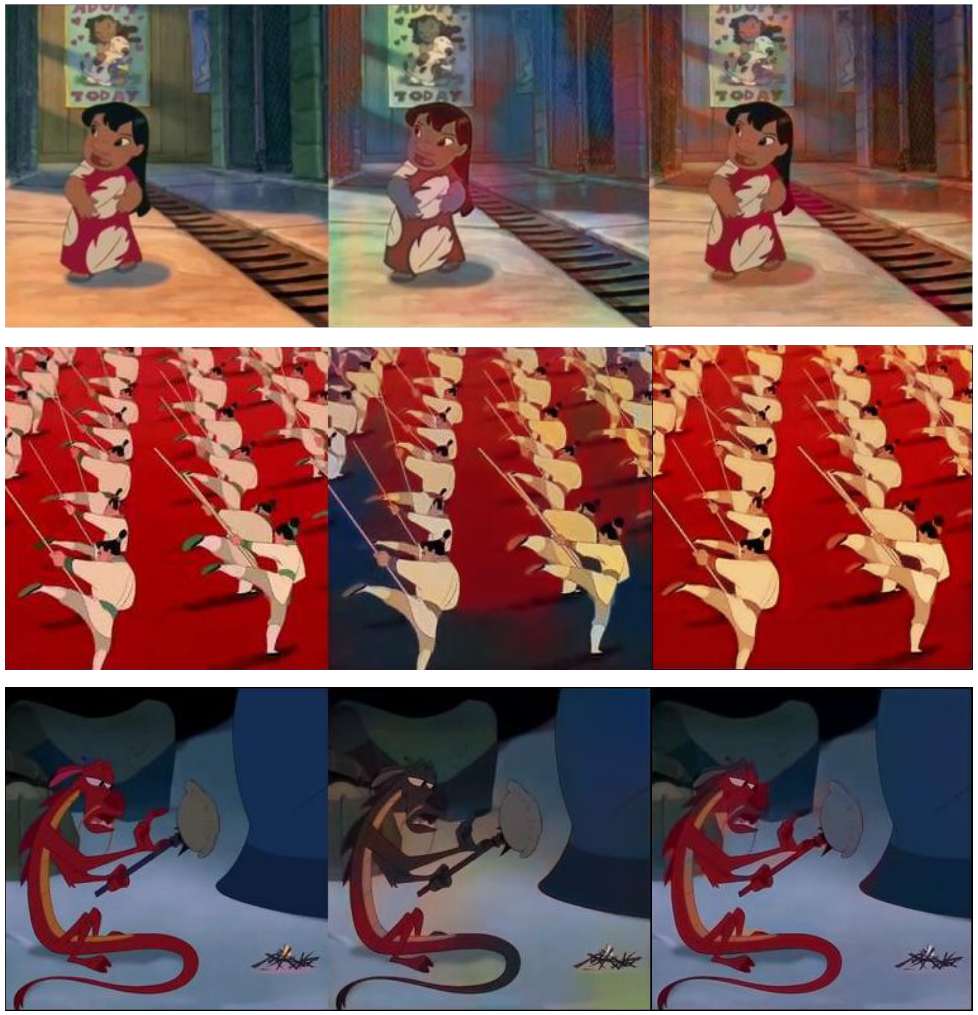

Ground truth
Res-cGAN
Ours
Ground truth
Res-cGAN
Ours
Ground truth
Res-cGAN
Ours
animation (few-shot)
cartoons (few-shot)
cartoons (one-shot)
real-images (few-shot)
넌 단점 없어? 사람이라면 뭔가 약한 부분이 하나쯤은 있잖아
딱 오 분만 누워있다가 씻자
아직 점심 안 먹었지?
네

RESULTS



RESULTS

Ground truth Res-cGAN **Ours**



Ground truth Res-cGAN **Ours**

EXPERIMENTS

	5-way		15-way	
	5-shot	10-shot	5-shot	10-shot
Ours (Unsup.)	87.50%	87.50%	69.44%	70.83%
Ours (Sup.)	91.66%	87.50%	72.22%	73.61%

Table 2. **Classification accuracy of the threshold triplet loss.**

- We show corresponding images of the top-3 color features retrieved from our memory networks.
- Our memory networks are trained to retrieve color features highly relevant to the content of the query image.
- We additionally show high classification accuracy of our memory networks.

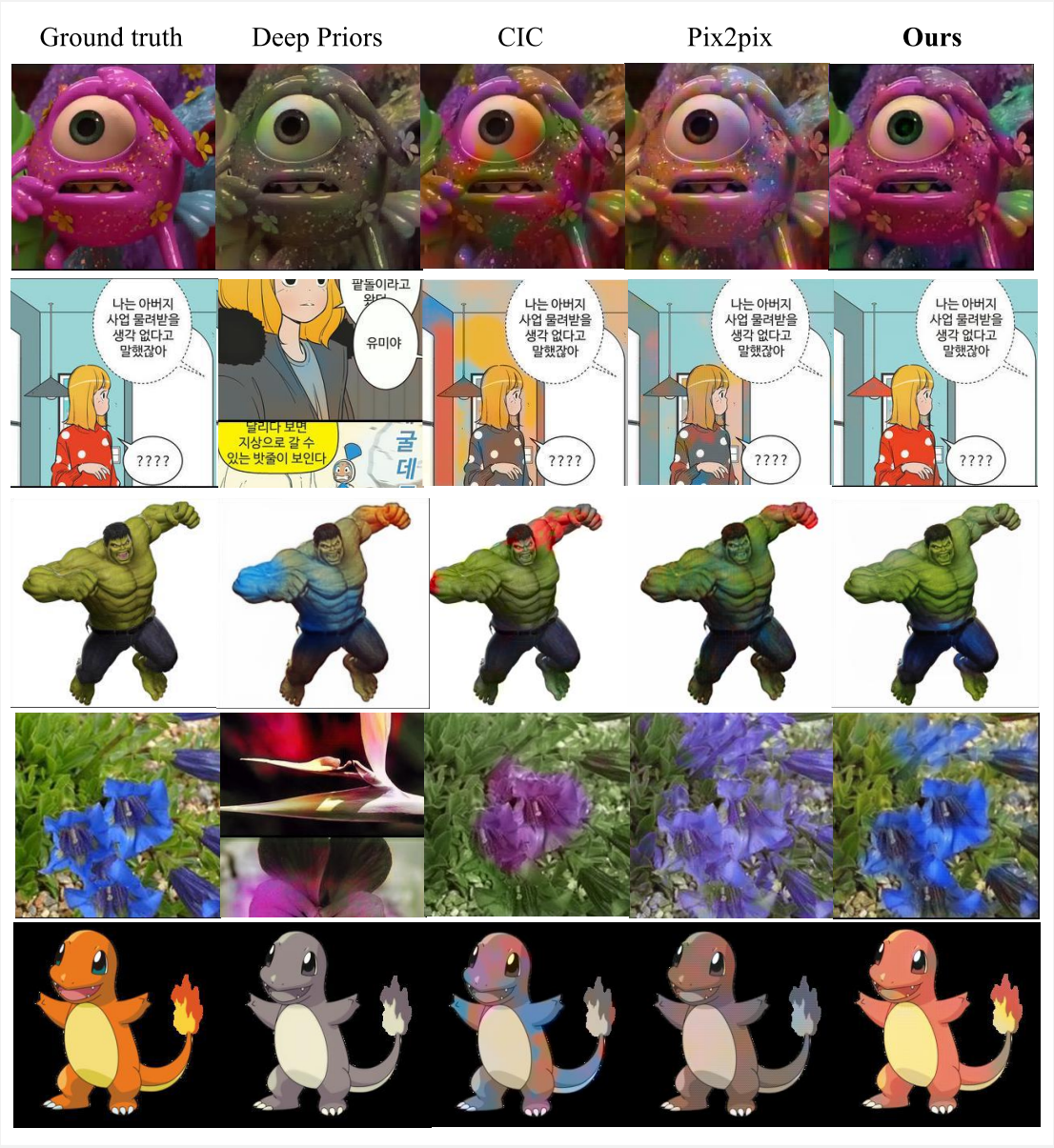
EXPERIMENTS

Comparison to baselines

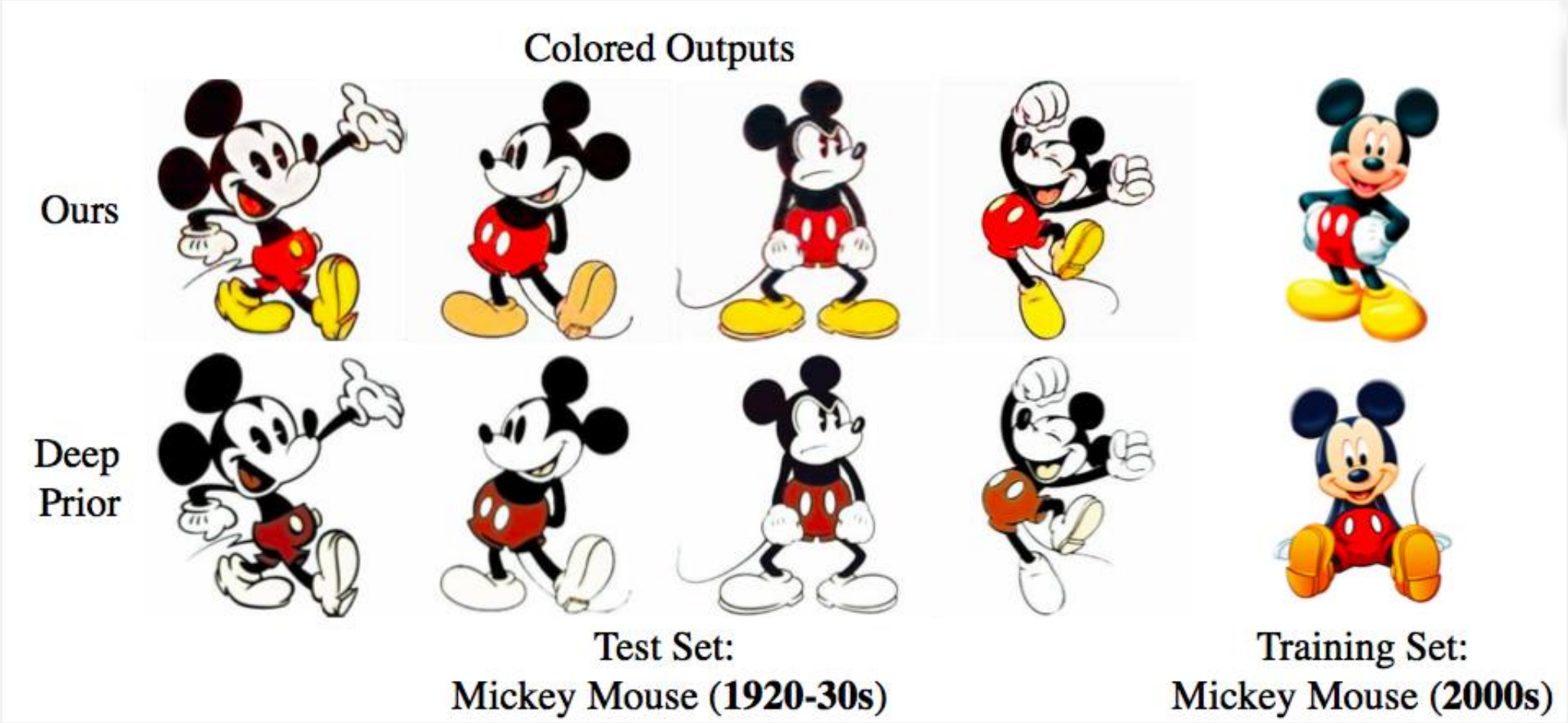
	CIC	Deep Priors	Pix2pix	**Ours**
PSNR	22.64	22.14	14.63	21.78

(a) MSE-based (b) GAN-based

	One-shot		**Few-shot**	
	User-study	LPIPS	User-study	LPIPS
Ours	**75%**	**8.48**	**71%**	**1.34**
CIC	10%	9.89	7%	1.80
Pix2pix	5%	13.47	16%	2.34
Deep Prior	10%	19.26	4%	2.03



PRACTICAL USE CASE



EXPERIMENTS ON MEMORY NETWORKS

Memory networks require a lot of memory?

- Storing 10,000 features for our memory network only requires an additional 10, 000 x 512 x 32 = 19.53MB of memory

Memory networks require a lot of parameters?

	Ours	CIC	Pix2pix	Deep Priors
Parameters	**12m**	32m	14m	35m

- Even with external memory networks, our model has the least amount of trainable parameters
- Our actual memory networks only have **262k parameters**

Robust to hyperparameters?

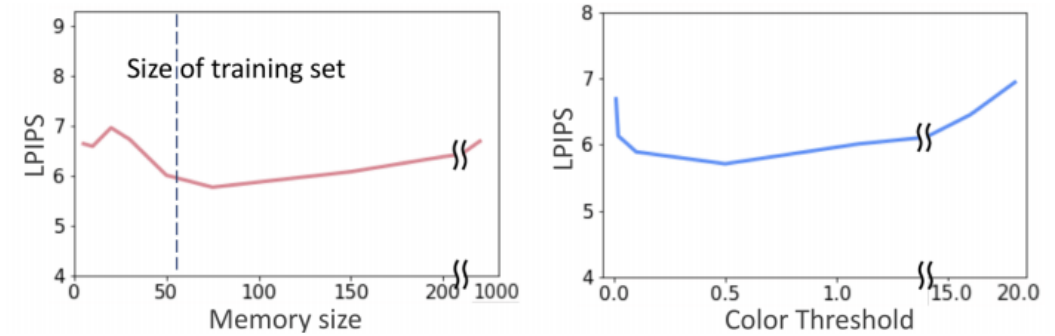


Figure 9. **Analysis of memory size and color threshold.** LPIPS scores are similar across various hyperparameters of the memory networks. Quality drops (high LPIPS) only with excessively small or large hyperparameters.

FAILURE CASES

- When our memory networks are insufficiently trained, they can retrieve an irrelevant memory slot
- Proper optimization of memory networks is key

FAILURE CASES

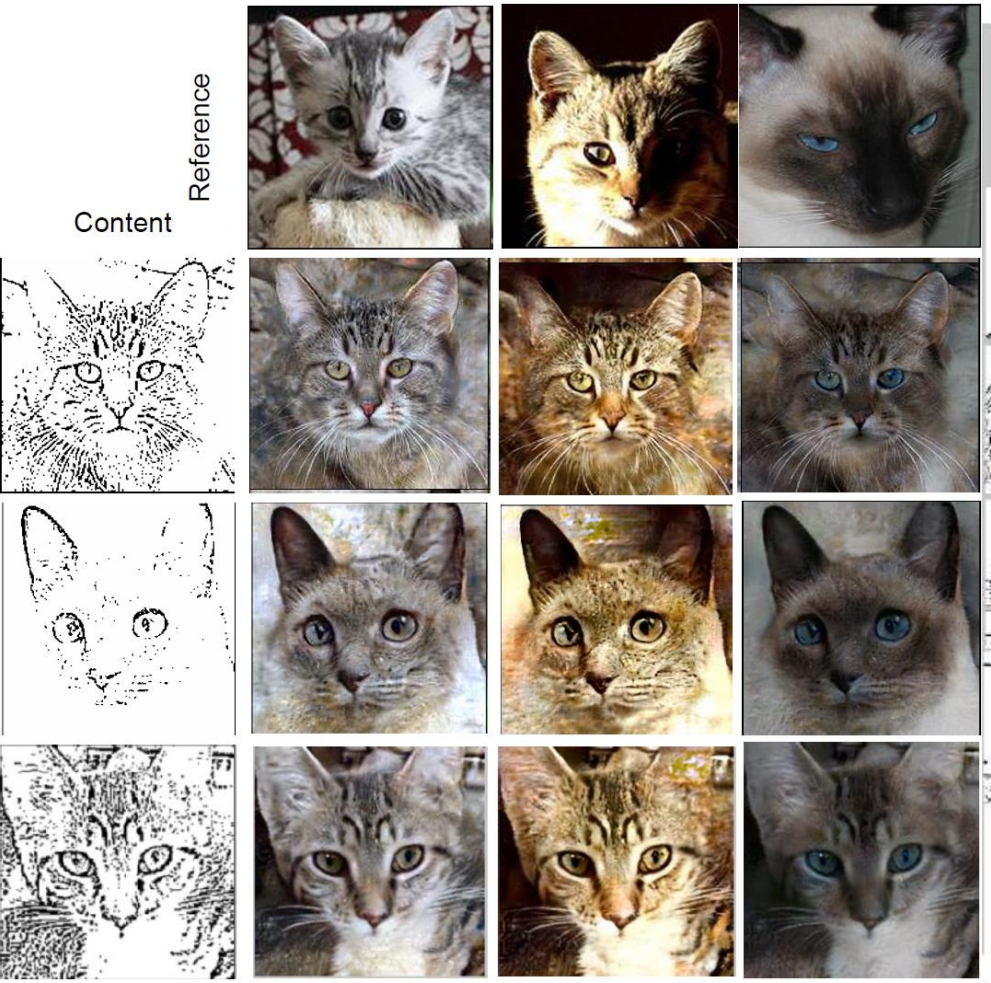
Figure 6. **Colorization results using the top-3 memory slots.** We show that our memory networks can retrieve appropriate color features for a given input. Different memory slots may be used to produce diverse results. All other samples in the paper are colored using the top-1 memory slot.

- Users can utilize top-k slots instead of top-1 memory slot

Overview of This Talk

- Intro to conditional generative models
- My own research on interactive automatic colorization
  - Colorization using natural language [ECCV'18]
  - Few-shot colorization via memory networks [CVPR'19]
  - Reference-based sketch colorization using augmented self-exemplar [CVPR'20]
- Other work on interactive generative models and future research directions

Reference-based Sketch Colorization



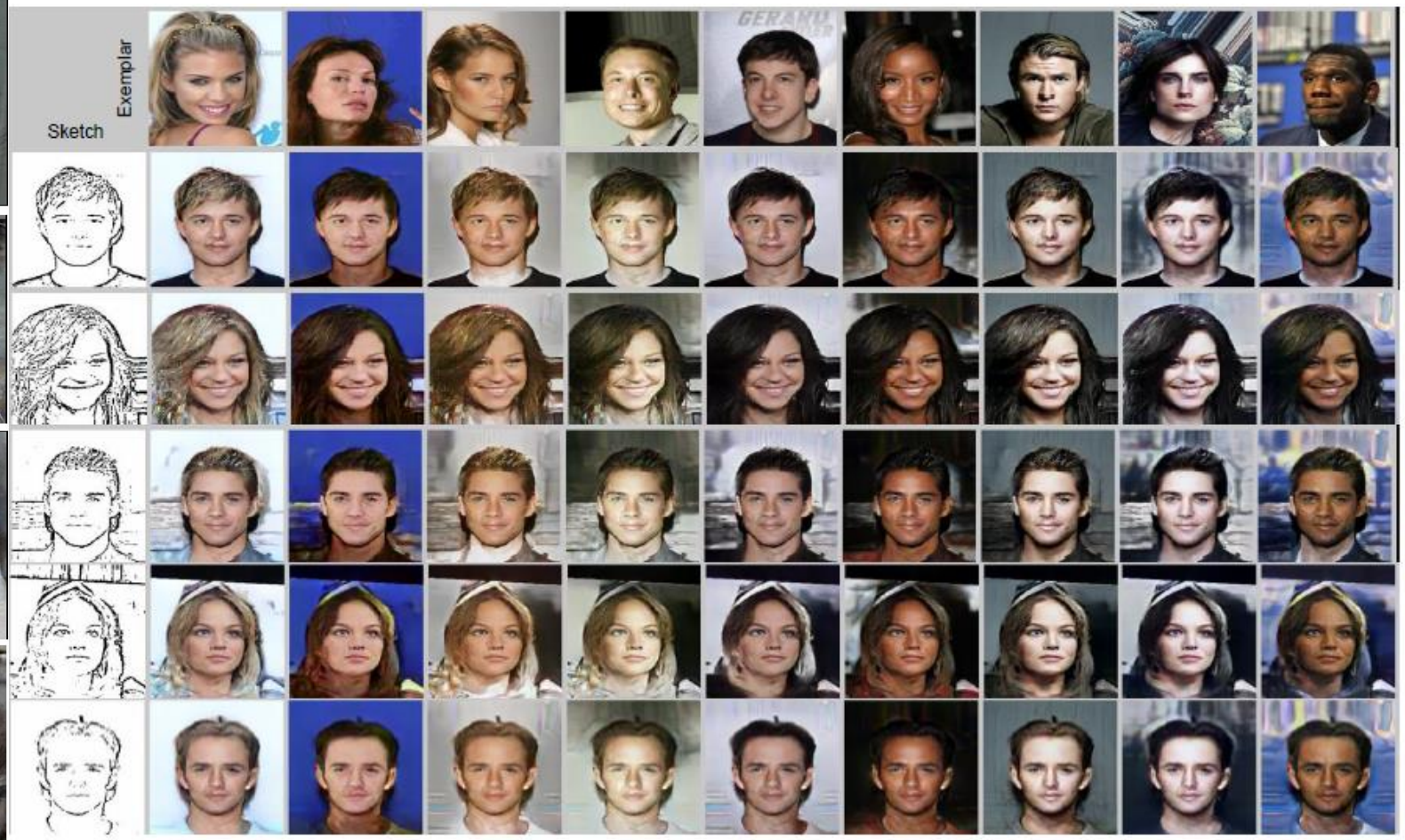


Figure 1: Qualitative results of our method on the CelebA [10] dataset. Each row has the same content while each column has the same reference.

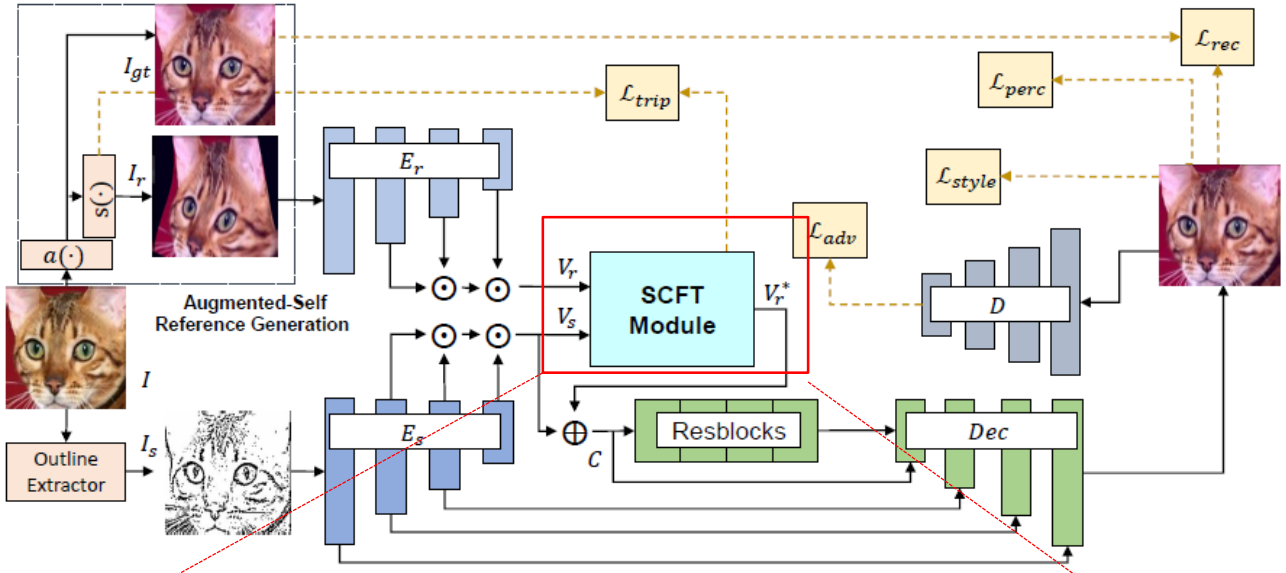
Challenges in Reference-based Sketch Colorization

- Sketch images are information-scarce, so there is little cue for their colorization.

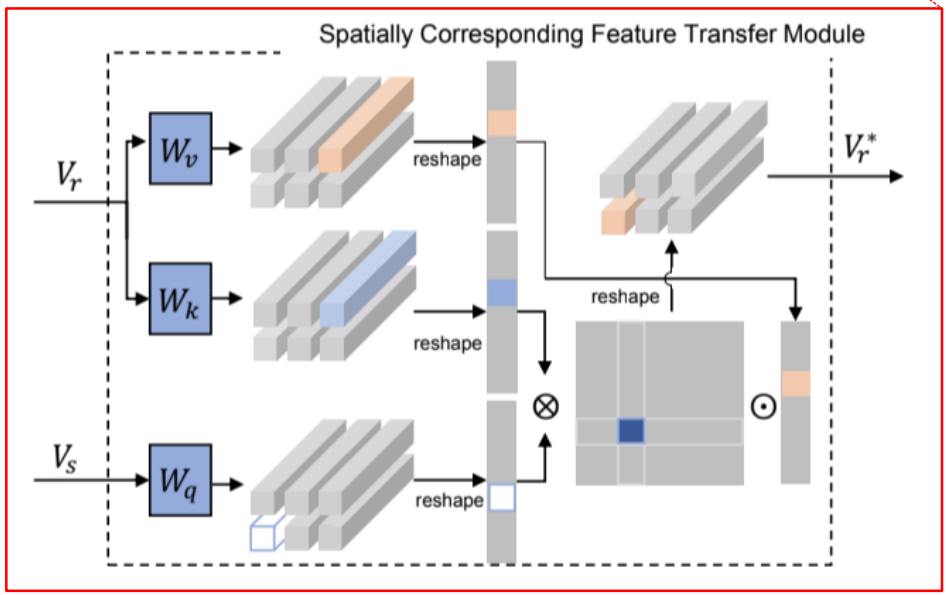
- Reference-based colorization can no longer be trained in a paired setting.

Model Architecture

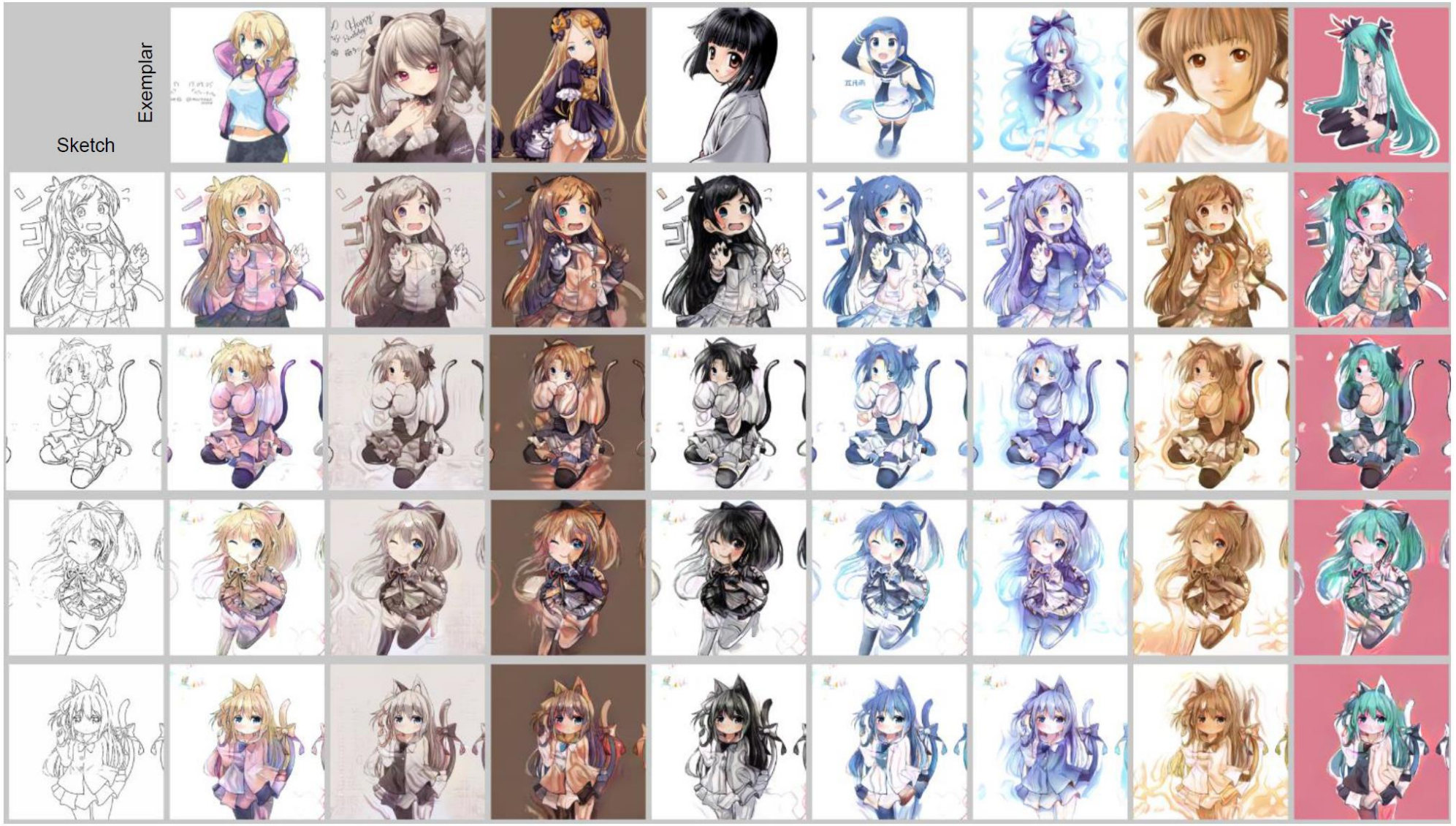
- We utilize a geometrically-transformed self-images as pseudo reference, which allows model training in a paired setting.



- Through pixel-wise correspondence naturally obtained, we directly supervise which pixel colors in the reference to transfer to which pixel in the target.



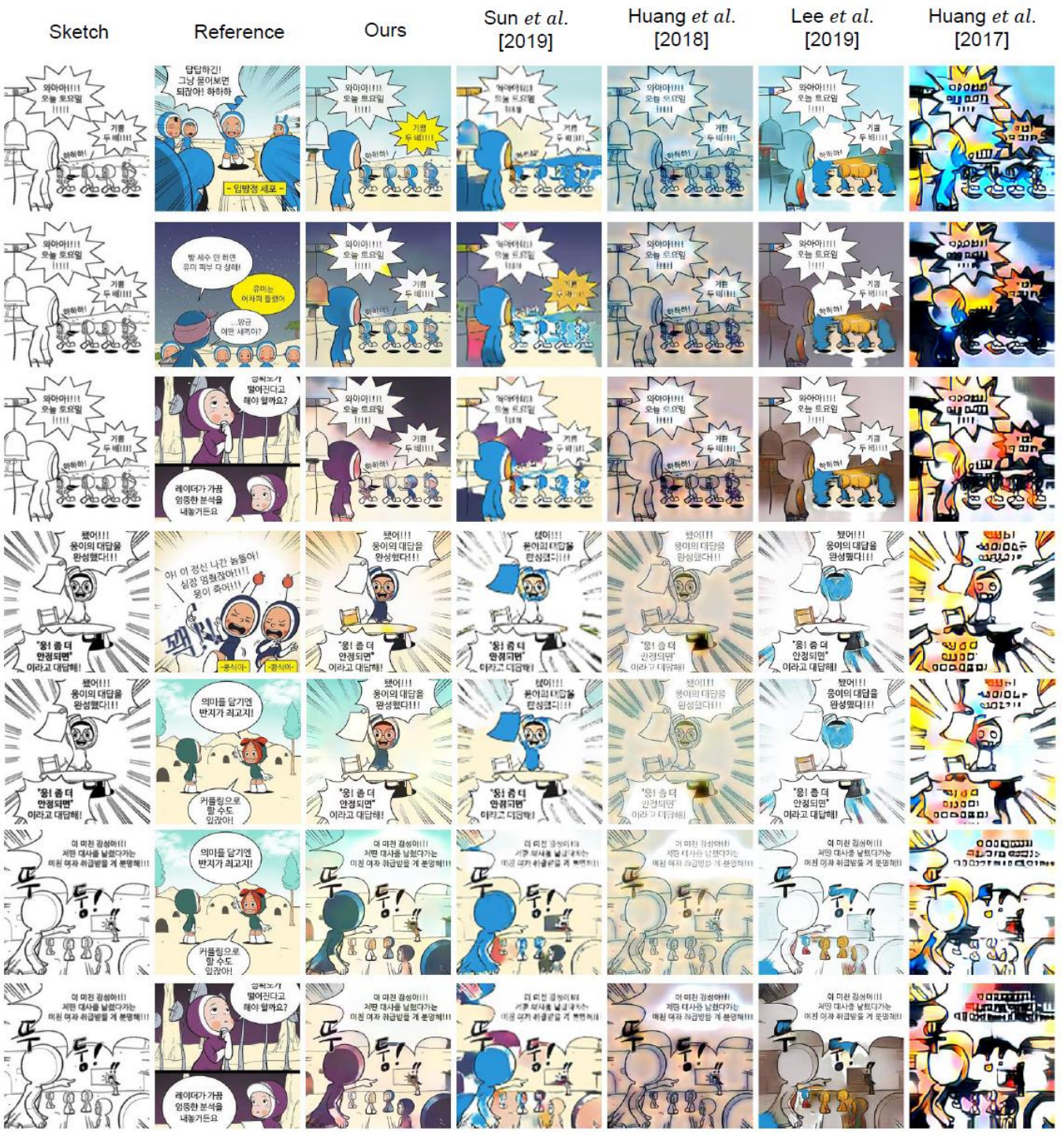
Qualitative Results



Qualitative Results

(a) Sketch (b) Reference (c) Synthesized image

Figure 2: Visualization of our attention mechanism.



Overview of This Talk

- Intro to conditional generative models

- My own research on interactive automatic colorization
  - Colorization using natural language [ECCV'18]
  - Few-shot colorization via memory networks [CVPR'19]
  - Reference-based sketch colorization using augmented self-exemplar [CVPR'20]

- Other work on interactive generative models and future research directions

Future Research Directions

- Support for real-time, multiple iterative, maybe local interactions
  - Reflecting higher-order user intent in multiple sequential interactions
- Revealing inner-workings and interaction handle
  - E.g., explicitly using (interpretation-friendly) attention module
- Better simulating user inputs in the training stage
- Incorporating data visualization and advanced user interfaces
- Leveraging hard rule-based approaches, e.g. ,following sharp edges
- Incorporating users' implicit feedback and online learning

GauGAN: Interactive Tool of SPADE

http://nvidia-research-mingyuliu.com/gaugan/

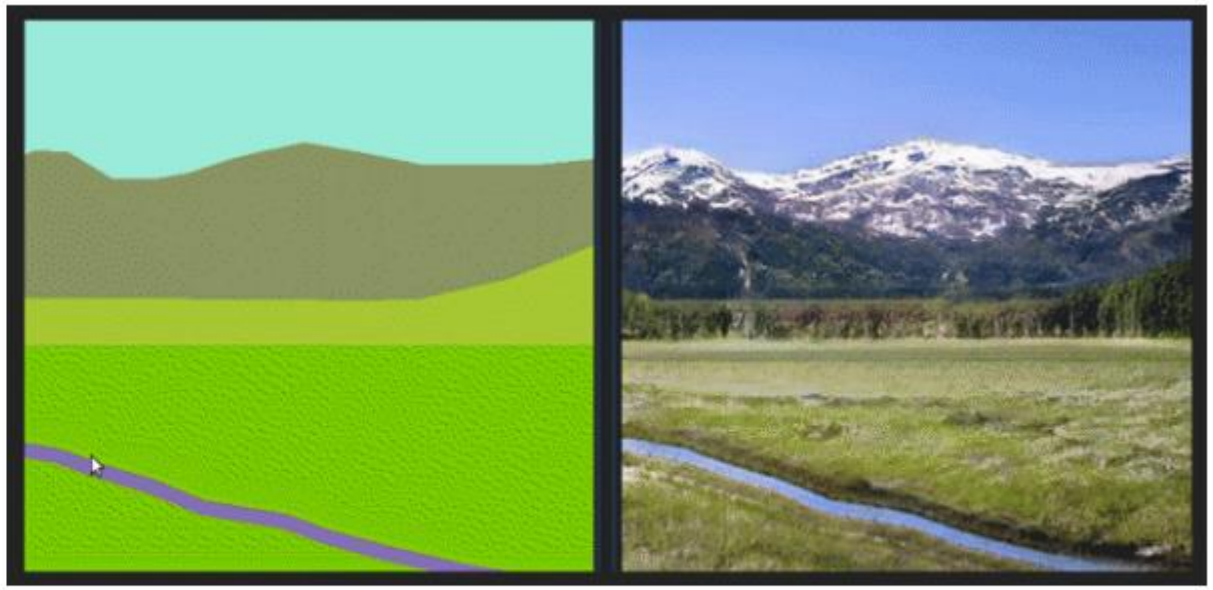
GAUGAN

GauGAN, named after post-Impressionist painter Paul Gauguin, creates photorealistic images from segmentation maps, which are labeled sketches that depict the layout of a scene.

Artists can use paintbrush and paint bucket tools to design their own landscapes with labels like river, rock and cloud. A style transfer algorithm allows creators to apply filters — changing a daytime scene to sunset, or a photorealistic image to a painting. Users can even upload their own filters to layer onto their masterpieces, or upload custom segmentation maps and landscape images as a foundation for their artwork

View Research Paper > | Read Blog > | Resources >

LAUNCH INTERACTIVE DEMO



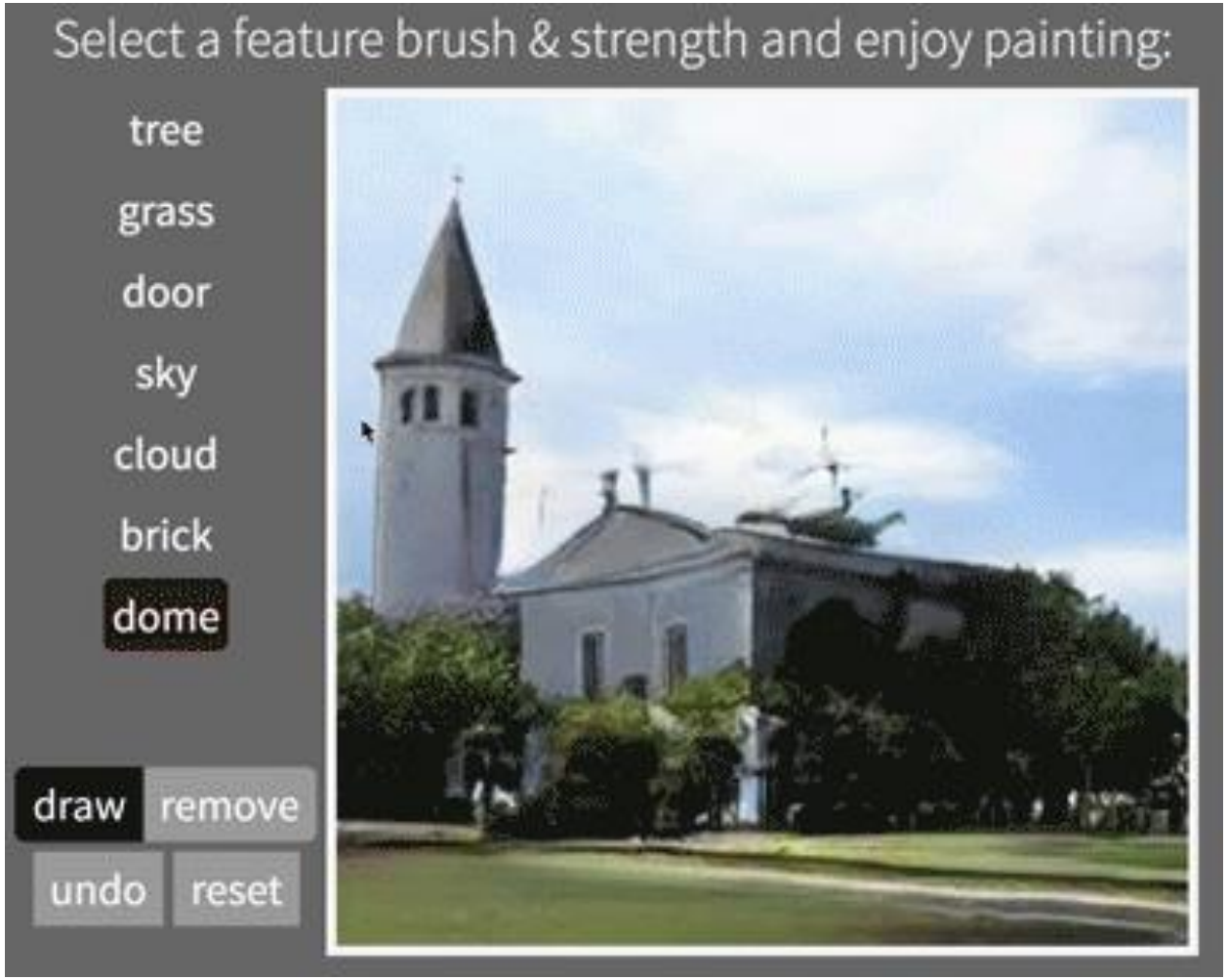
Park et al., Semantic Image Synthesis with Spatially-Adaptive Normalization, CVPR'19 (Oral)

GANPaint: Interactive Image Generation

Bau et al., GAN Dissection:Visualizing and Understanding Generative Adversarial Networks, ICLR'19
https://youtu.be/yVCgUYe4JTM

Goal

- A user edits a generated image or a photograph with high-level concepts rather than pixel colors
- A software manipulates an image to achieve a user-specified goal while keeping the result photorealistic



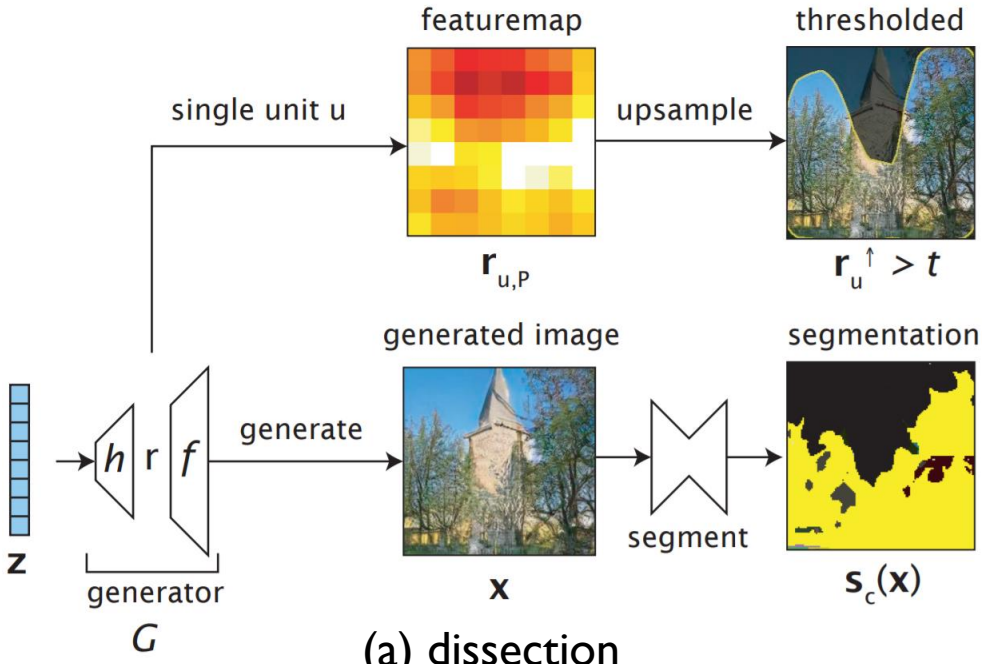
Interactive image generation demo page:
http://gandissect.res.ibm.com/ganpaint.html

GANPaint: Interactive Image Generation

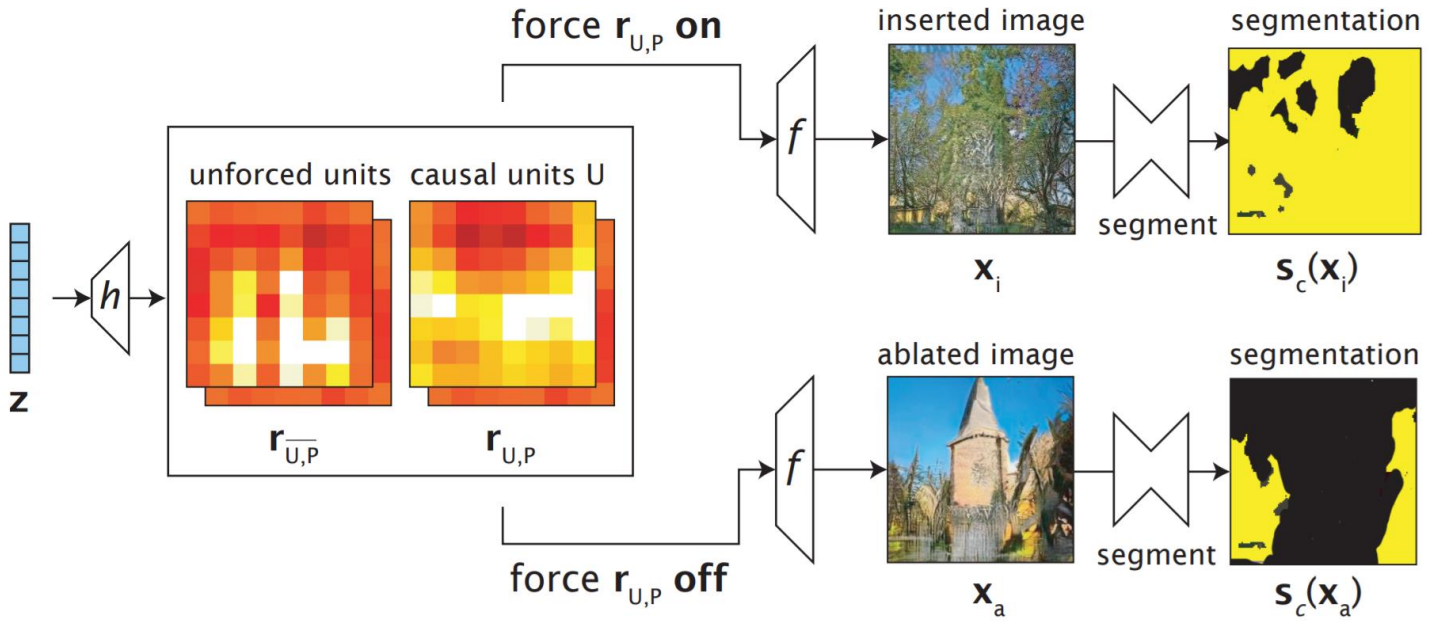
Bau et al., GAN Dissection: Visualizing and Understanding Generative Adversarial Networks, ICLR'19
https://youtu.be/yVCgUYe4JTM

Contributions

- Provide the first systematic analysis for understanding the internal representations of GANs
- Show several practical applications enabled by the analytic framework
- Provide open source interpretation tools: https://github.com/CSAILVision/gandissect

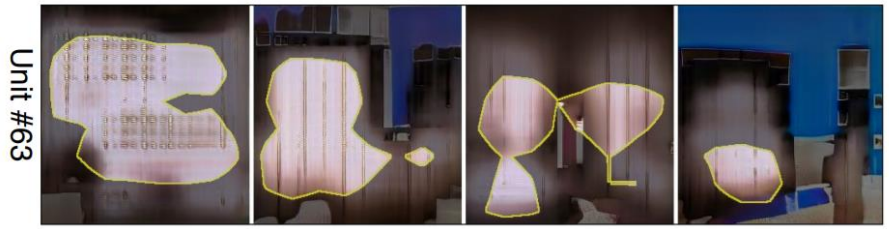


(a) dissection



(b) intervention

GANPaint: Interactive Image Generation



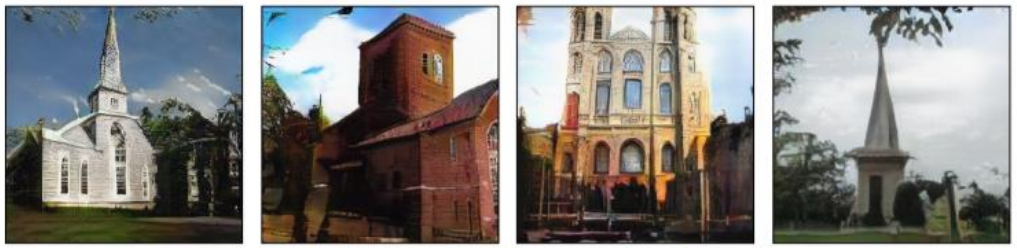
(a) Example artifact-causing units

(b) Bedroom images with artifacts

(c) Ablating “artifact” units improves results

(a) Generate images of churches

(b) Identify GAN units that match trees



(c) Ablating units removes trees

(d) Activating units adds trees

Interactive Data Labeling

Papadopoulos et al.,
Extreme Clicking for Efficient Object Annotation,
CVPR'17

Different User Interfaces should also be considered.

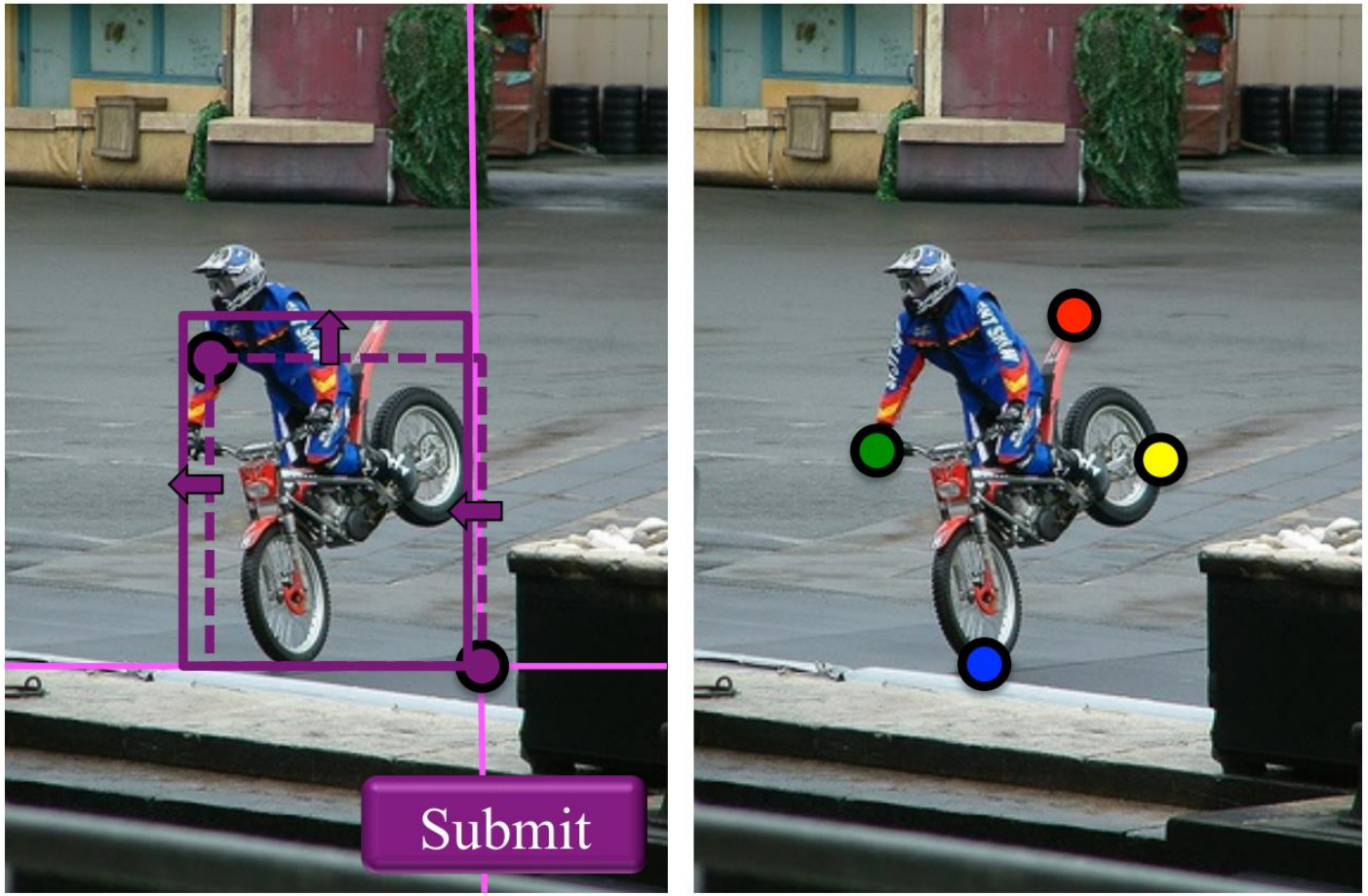
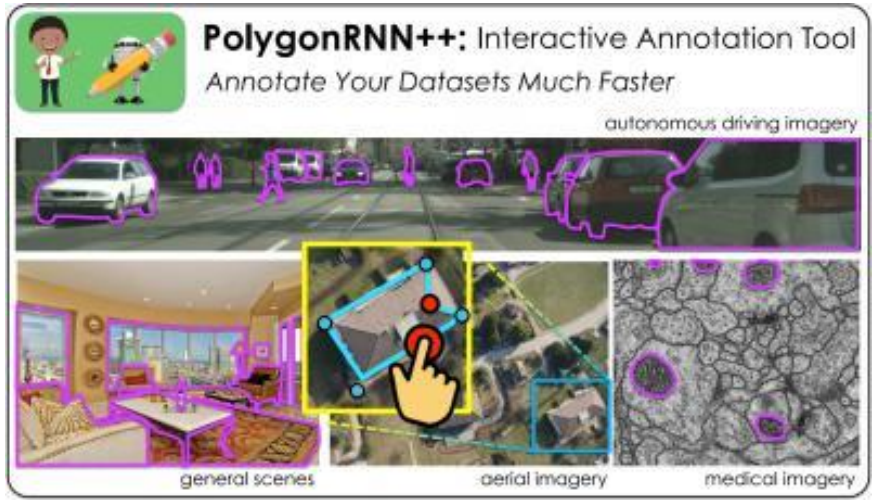


Figure 1. **Annotating an instance of motorbike:** (a) The conventional way of drawing a bounding box. (b) Our proposed extreme clicking scheme.

Interactive Segmentation

Acuna et al., Efficient Annotation of Segmentation Datasets with PolygonRNN++, CVPR'18
https://youtu.be/evGqMnL4P3E



Contributions
(a) Design a new CNN encoder architecture
(b) Show how to effectively train the model with Reinforcement Learning
(c) Significantly increase the output resolution using Graph Neural Network

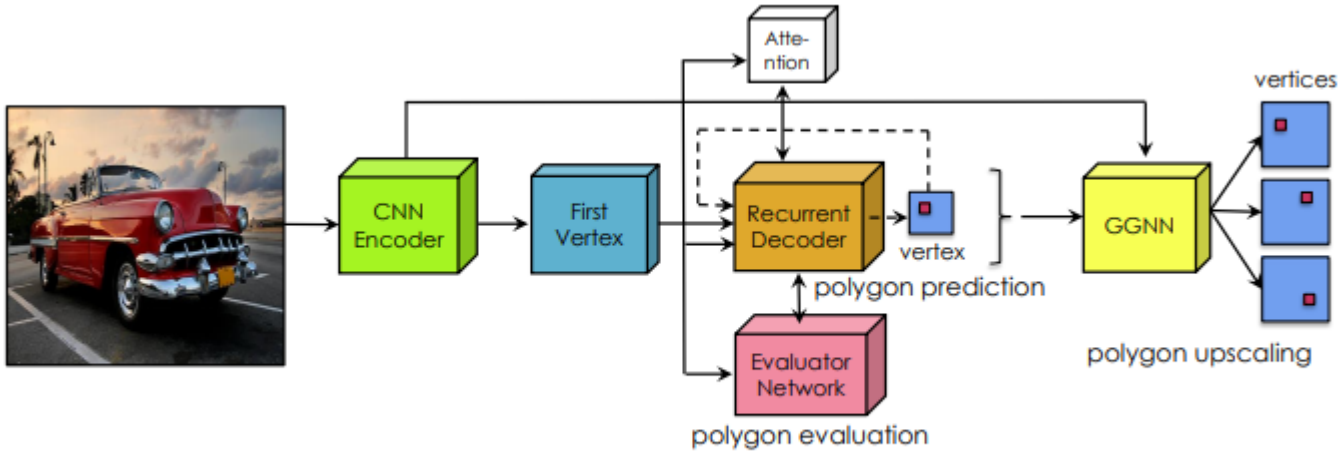


Figure 2: Polygon-RNN++ model (figures best viewed in color)

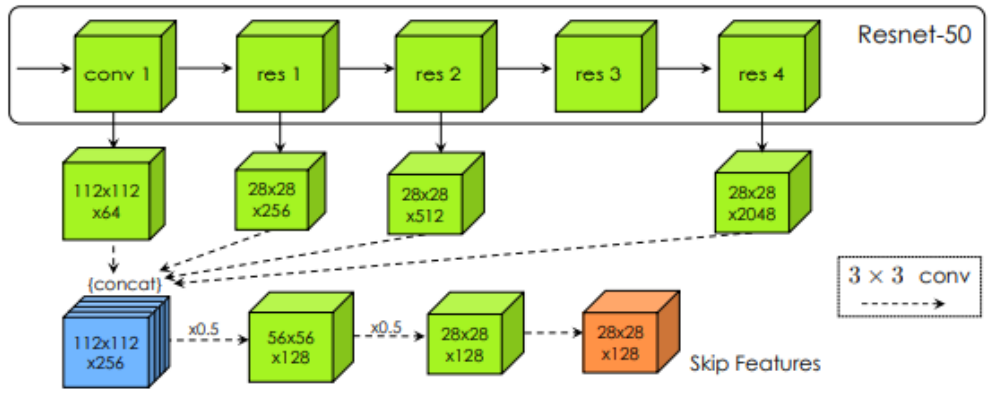
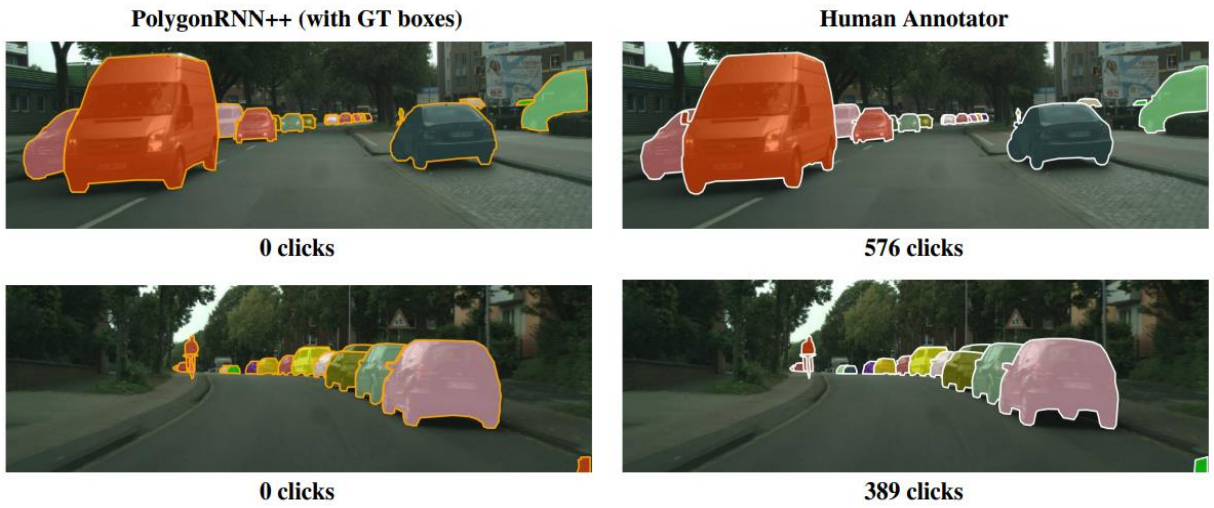


Figure 4: Residual Encoder architecture. Blue tensor is fed to GNN, while the orange tensor is input to the RNN decoder.

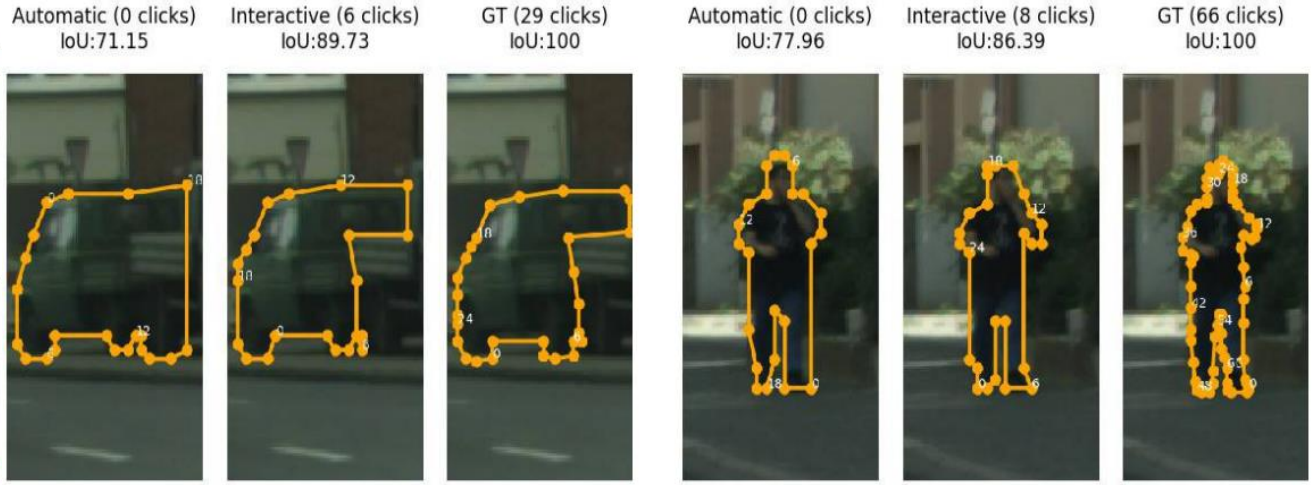
(a)

Interactive Segmentation

- Interactive object segmentation is a well studied problem with the aim to reduce the time and cost of annotation
- Human-level performance was achieved with only a few user clicks per object



(a) Automatic annotation



(b) Semi-automatic annotation

(a), (b): Acuna et al., Efficient Annotation of Segmentation Datasets with PolygonRNN++, CVPR'18

Interactive Segmentation

Ling et al., Fast Interactive Object Annotation with Curve-GCN, CVPR'19
https://youtu.be/ycD2BtO-QzU

Contributions

- Alleviate the sequential nature of Polygon RNN, by predicting all vertices simultaneously using a Graph Convolutional Network (GCN)
- Run at 29.3ms in automatic, and 2.6ms in interactive mode, making it 10x and 100x faster than Polygon-RNN++

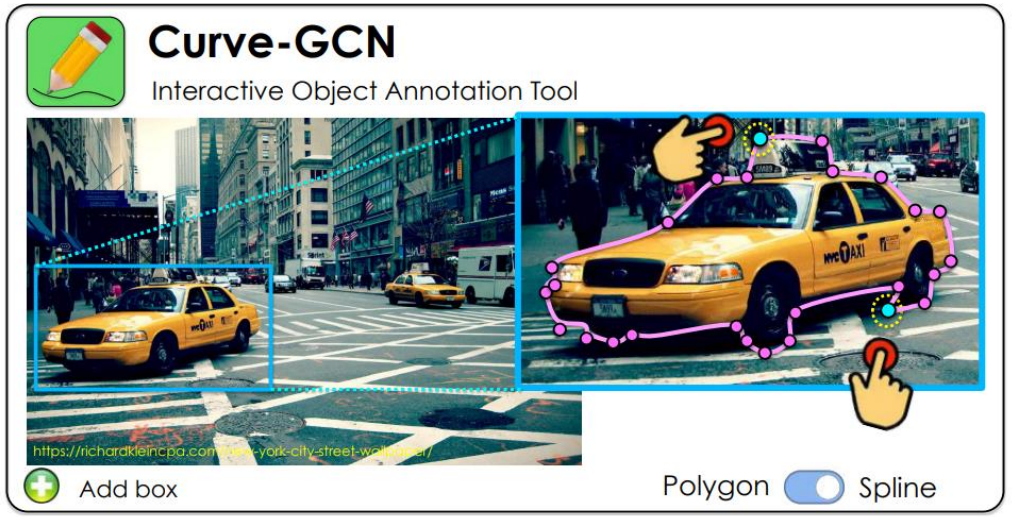


Figure 1: We propose Curve-GCN for interactive object annotation. In contrast to Polygon-RNN [7, 2], our model parametrizes

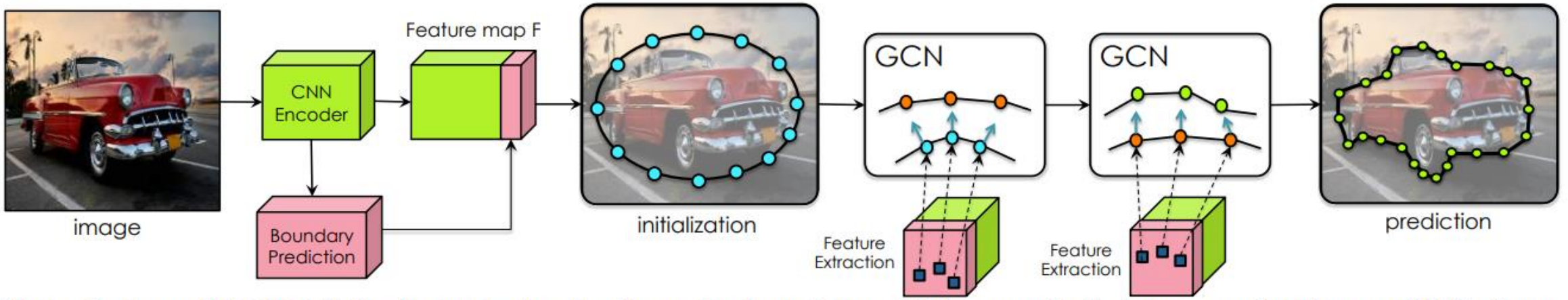


Figure 2: **Curve-GCN**: We initialize N control points (that form a closed curve) along a circle centered in the image crop with a diameter of 70% of image height. We form a graph and propagate messages via a Graph Convolutional Network (GCN) to predict a location shift for each node. This is done iteratively (3 times in our work). At each iteration we extract a feature vector for each node from the CNN's features F, using a bilinear interpolation kernel.

Interactive Neural Machine Translation

- Word-Based

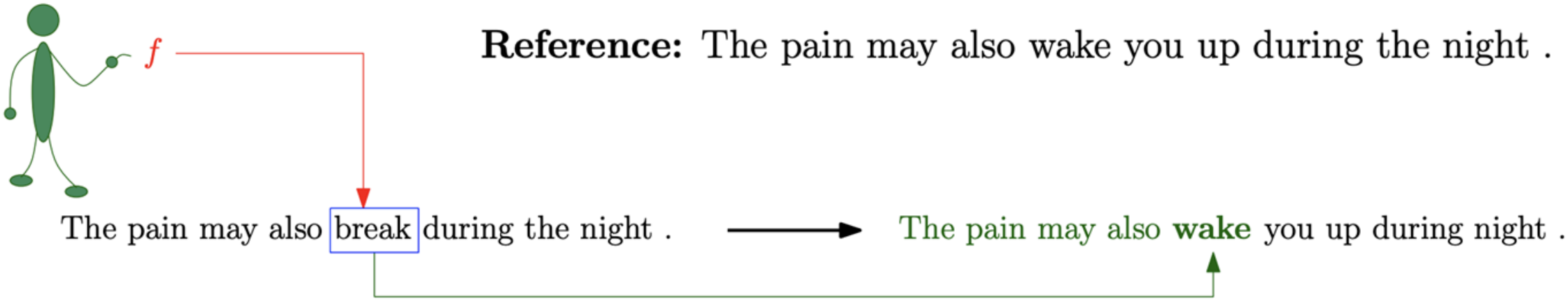


Figure 1: Single iteration of prefix-based IMT. The user wants to translate the French sentence "La doleur peut également vous réveiller pendant la nuit ." into English. The user corrects the first wrong word from the hypothesis provided by the system, introducing the word "wake" at position 5. Next, the system generates a new hypothesis, that contains the validated prefix together with the corrected word. Note that, although the system generates a partially correct suffix, in this new hypothesis it is also introduced a new error ("during night" instead of "during the night"). This behaviour is intended to be solved with the segment-based approach.

[Álvaro Peris, Miguel Domingo and Francisco Casacuberta. 2017. Interactive neural machine translation. In *Computer Speech and Language*.]

Interactive Neural Machine Translation

- Segment-Based

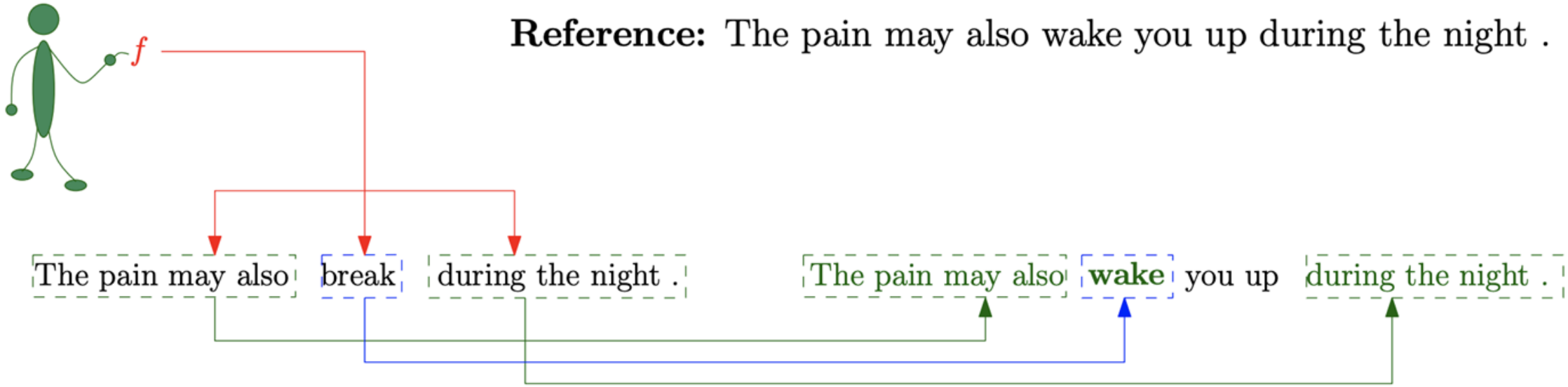


Figure 2: Segment-based IMT iteration for the same example than in Fig. 1. In this case, the user validates two segments and introduces a word correction. The system generates a new hypothesis that contains the word correction and keeps the validated segments. The user feedback is $f =$ "The pain may also", "wake", "during the night .". The reaction of the system is to generate the sequence of non-validated segments $\tilde{g} = \lambda$, "you up", λ; being λ the empty string. The hypothesis offered by the system consists in the combination of the validated and non-validated segments.

[Álvaro Peris, Miguel Domingo and Francisco Casacuberta. 2017. Interactive neural machine translation. In *Computer Speech and Language*.]

Interactive Neural Machine Translation

- Interactive-predictive System for Multimodal Sequence to Sequence Tasks

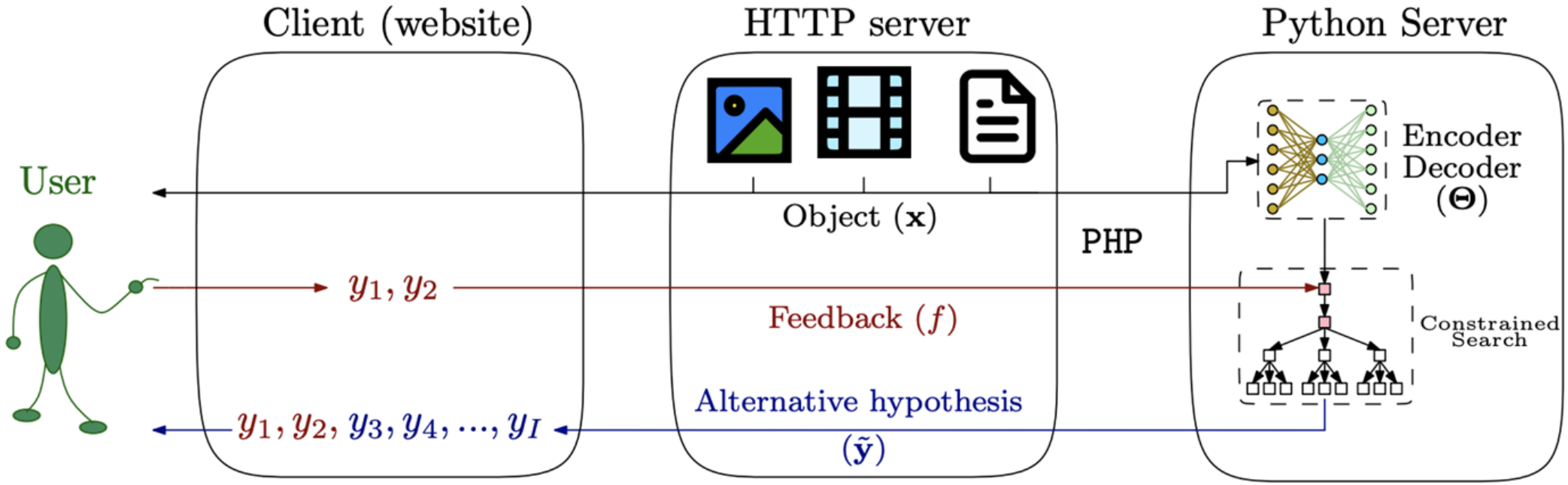


Figure 1: System architecture. The client, a website, presents the user several input objects (images, videos or texts) and a prediction. The user then introduces a feedback signal, for correcting this prediction. After being introduced, the feedback signal is sent to the server—together with the input object—for generating an alternative hypothesis, which takes into account the user corrections.

[Álvaro Peris and Francisco Casacuberta. 2019. A Neural, Interactive-predictive System for Multimodal Sequence to Sequence Tasks. *ACL 2019.*]

Interactive Neural Machine Translation

- Interactive-predictive System for Multimodal Sequence to Sequence Tasks

0	**System**	A group of football players in red uniforms.
1	**User**	*A* **f** group of football players in red uniforms.
	System	*A f*ootball player in a red uniform is holding a football.
2	**User**	*A football player in a red uniform is* **w** holding a football.
	System	*A football player in a red uniform is w*earing a football.
3	**User**	*A football player in a red uniform is wearing a* **h** football.
	System	*A football player in a red uniform is w*earing a helmet.
4	**User**	*A football player in a red uniform is wearing a helmet.*

Figure 3: Interactive-predictive session for correcting the caption generated in Fig. 2. At each iteration, the user introduces a character correction (boxed). The system modifies its hypothesis, taking into account this feedback: keeping the correct prefix (green) and generating a compatible suffix.

Interactive Neural Machine Translation

- Self-Regulated interactive learning guides to choose a certain type of feedback
- Four different types of feedback
  - 1. Full correction
  - 2. Error marking
  - 3. Self-supervision
  - 4. None

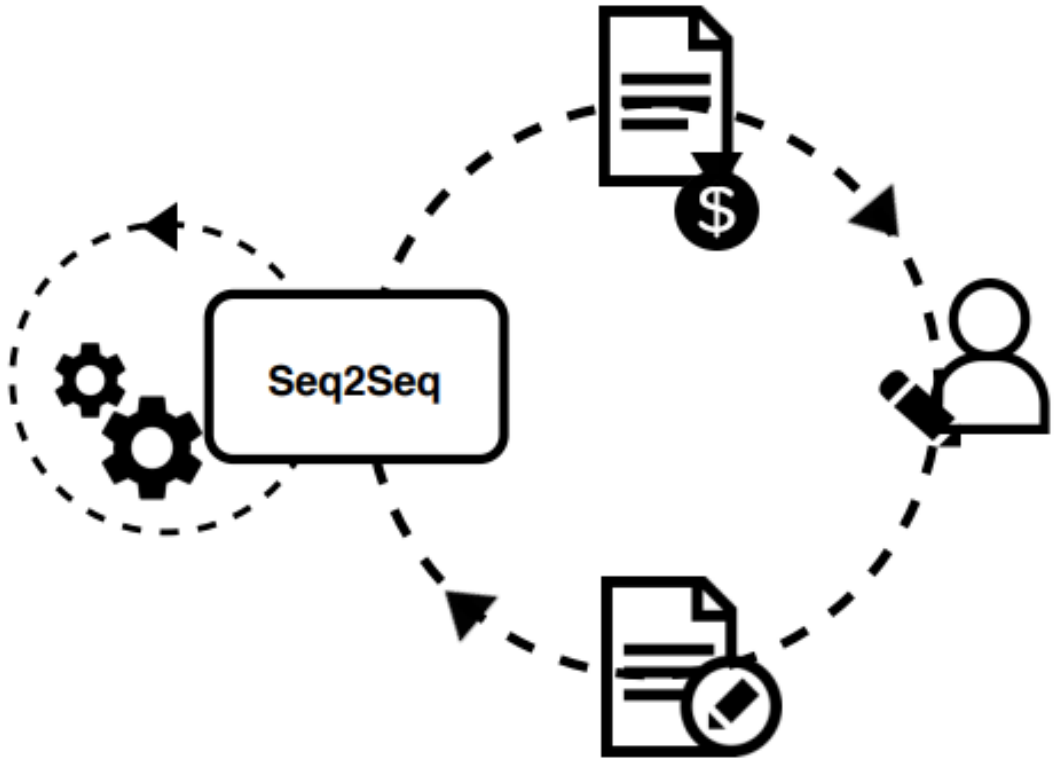


Figure 1: Human-in-the-loop self-regulated learning.

[Julia Kreutzer and Stefan Riezler. 2019. Self-Regulated Interactive Sequence-to-Sequence Learning. *ACL 2019*.]

Interactive Neural Machine Translation

SELF	0	x	Sie greift in ihre Geldbörse und gibt ihm einen Zwanziger .
		$\hat{y}$	It attacks their wallets and gives him a twist .
		y^*	She reaches into her purse and hands him a 20 .
WEAK	9	x	Und als ihr Vater sie sah und sah , wer sie geworden ist , in ihrem vollen Mädchen-Sein , schlang er seine Arme um sie und brach in Tränen aus .
		$\hat{y}$	And when her father saw them and saw who became them , in their full girl 's , he swallowed his arms around them and broke out in tears .
		y^*	When her father saw her and saw who she had become , in her full girl self , he threw his arms around her and broke down crying .
FULL	59	x	Und durch diese zwei Eigenschaften war es mir möglich , die Bilder zu erschaffen , die Sie jetzt sehen .
		$\hat{y}$	And ~~through~~ these two ~~features ,~~ I was able to create the images you now ~~see~~ .
		y^*	And it was with those two properties that I was able to create the images that you 're seeing right now .

Table 1: Examples from the IWSLT17 training set, cost (2nd column) and feedback decisions made by *Reg3*. For weak feedback, marked parts are underlined, for full feedback, the corrections are marked by underlining the parts of the reference that got inserted and the parts of the hypothesis that got deleted.

Interactive Neural Machine Translation

- Original human-interactive NMT requires many efforts, such as editing or deleting
- To reduce the human efforts, it employs the **reinforcement learning idea** of humans providing reward signals in form of judgments on the quality of the machine translation

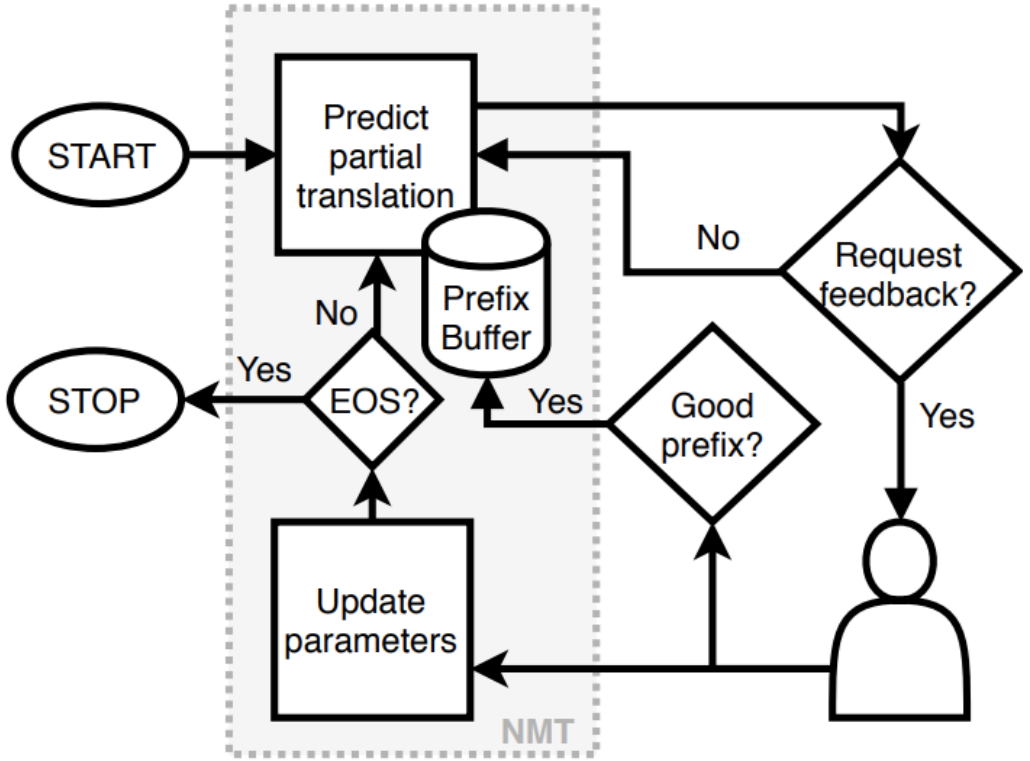


Figure 1: Interaction of the NMT system with the human during learning for a single translation.

[Khanh Nguyen, Hal Daumé III, and Jordan Boyd-Graber. 2017. Reinforcement Learning for Bandit Neural Machine Translation with Simulated Human Feedback. *EMNLP 2017*.]

Interactive Neural Machine Translation

SRC	la réponse que nous , en tant qu' individus , acceptons est que nous sommes libres parce que nous nous gouvernons nous-mêmes en commun plutôt que d' être dirigés par une organisation qui n' a nul besoin de tenir compte de notre existence .
REF	the answer that we as individuals accept is that we are free because we rule ourselves in common , rather than being ruled by some agency that need not take account of us . < /s>

Partial sampled translation	**Feedback**
the	1
the answer	1
the answer we	0.6964
the answer we ,	0.6246
the answer we as individuals allow to 14 are	0.6008
the answer we , as individuals , go down to speak 8 , are being free because we govern ourselves , rather from being based together	0.5155
the answer we , as people , accepts is that we principle are free because we govern ourselves , rather than being led by a organisation which has absolutely no need to take our standards . < /s>	0.5722

Table 4: Interaction protocol for three translations. These translations were sampled from the model when the algorithm decided to request human feedback (line 10 in Algorithm 1). Tokens that get an overall negative reward (in combination with the critic), are marked in red, the remaining tokens receive a positive reward. When a prefix is good (i.e. $\geq \mu$, here $\mu = 0.8$) it is stored in the buffer and used for forced decoding for later samples (underlined).

Interactive Neural Machine Translation

- Other interesting Interactive NMT papers are listed in the following link: https://github.com/THUNLP-MT/MT-Reading-List#interactive-nmt

Interactive NMT

- Joern Wuebker, Spence Green, John DeNero, Saša Hasan and Minh-Thang Luong. 2016. Models and Inference for Prefix-Constrained Machine Translation. In *Proceedings of ACL 2016*. (Citation: 14)
- Rebecca Knowles and Philipp Koehn. 2017. Neural Interactive Translation Prediction. In *Proceedings of AMTA 2016*. (Citation: 24)
- Álvaro Peris, Miguel Domingo and Francisco Casacuberta. 2017. Interactive neural machine translation. In *Computer Speech and Language*. (Citation: 21)
- Khanh Nguyen, Hal Daumé III, and Jordan Boyd-Graber. 2017. Reinforcement Learning for Bandit Neural Machine Translation with Simulated Human Feedback. In *Proceedings of EMNLP 2017*. (Citation: 11)
- Álvaro Peris and Francisco Casacuberta. 2018. Active Learning for Interactive Neural Machine Translation of Data Streams. In *Proceedings of CoNLL 2018*. (Citation: 1)
- Tsz Kin Lam, Julia Kreutzer, and Stefan Riezler. 2018. A Reinforcement Learning Approach to Interactive-Predictive Neural Machine Translation. In *Proceedings of EAMT 2018*.
- Julia Kreutzer, Shahram Khadivi, Evgeny Matusov, Stefan Riezler. 2018. Can Neural Machine Translation be Improved with User Feedback?. In *Proceedings of NAACL 2018*. (Citation: 3).
- Pavel Petrushkov, Shahram Khadivi and Evgeny Matusov. 2018. Learning from Chunk-based Feedback in Neural Machine Translation. In *Proceedings of ACL 2018*. (Citation: 1)
- Julia Kreutzer, Joshua Uyheng, and Stefan Riezler. 2018. Reliability and Learnability of Human Bandit Feedback for Sequence-to-Sequence Reinforcement Learning. In *Proceedings of ACL 2018*. (Citation: 2)
- Álvaro Peris and Francisco Casacuberta. 2019. A Neural, Interactive-predictive System for Multimodal Sequence to Sequence Tasks. In *Proceedings of ACL 2019*.
- Miguel Domingo, Mercedes García-Martínez, Amando Estela, Laurent Bié, Alexandre Helle, Álvaro Peris, Francisco Casacuberta, and Manuerl Herranz. 2019. Demonstration of a Neural Machine Translation System with Online Learning for Translators. In *Proceedings of ACL 2019*.
- Julia Kreutzer and Stefan Riezler. 2019. Self-Regulated Interactive Sequence-to-Sequence Learning. In *Proceedings of ACL 2019*.

Future Research Directions

- Support for real-time, multiple iterative interactions
  - Reflecting higher-order user intent in multiple sequential interactions
- Revealing inner-workings and interaction handle
  - E.g., explicitly using (interpretation-friendly) attention module
- Better simulating user inputs in the training stage
- Incorporating data visualization and advanced user interfaces
- Leveraging hard rule-based approaches
- Incorporating users' implicit feedback and online learning

Useful Links

- 2019 ICML Workshop on Human In the Loop Learning (HILL)
  - https://sites.google.com/view/hill2019
  - Videos: https://icml.cc/Conferences/2019/ScheduleMultitrack?event=3511
- 2020 IUI 2020 Workshop on Human-AI Co-Creation with Generative Models (programs are not yet updated)
  - https://hai-gen2020.github.io/
- Key researchers
  - David Bau: https://people.csail.mit.edu/davidbau/home/
  - Sanja Fidler: https://www.cs.utoronto.ca/~fidler/
  - Richard Zhang: https://richzhang.github.io/
  - Jun-Yan Zhu: https://people.csail.mit.edu/junyanz/

Novel conditional generative models and their user interfaces from a user-centric perspective can open up new research directions in further advancing artificial intelligence techniques and applications.

Thank you!

Overview of This Talk

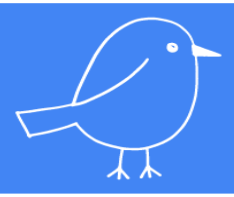
- Intro to generative adversarial networks (GANs) and conditional GANs
- Motivations of User-Interactive Generative Models

- Interactive Generative Tasks
- Taxonomy of User Input

Generative Adversarial Networks

- Generative: It is a model for generation.
- Networks: The model is formed as neural networks.
- Adversarial: Improves the generation quality via adversarial training (using an additional discriminator).

Progressive Growing of GANs



- Generated Images

Karras et al. Progressive Growing of GANs for Improved Quality, Stability, and Variation, ICLR'18

StyleGAN

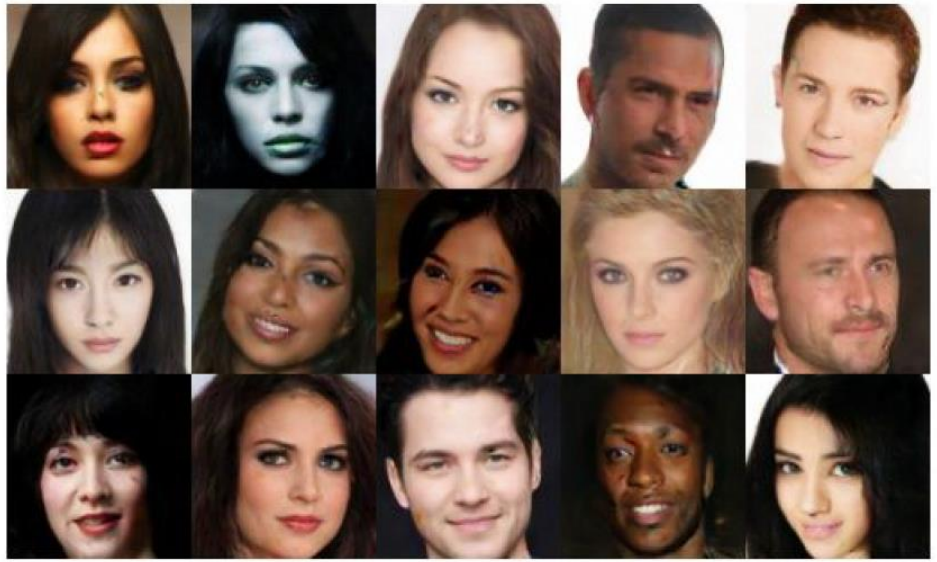
Karras et al. A Style-Based Generator Architecture for Generative Adversarial Networks, CVPR'19

ThisPersonDoesNotExist.com

Karras et al. A Style-Based Generator Architecture for Generative Adversarial Networks, C

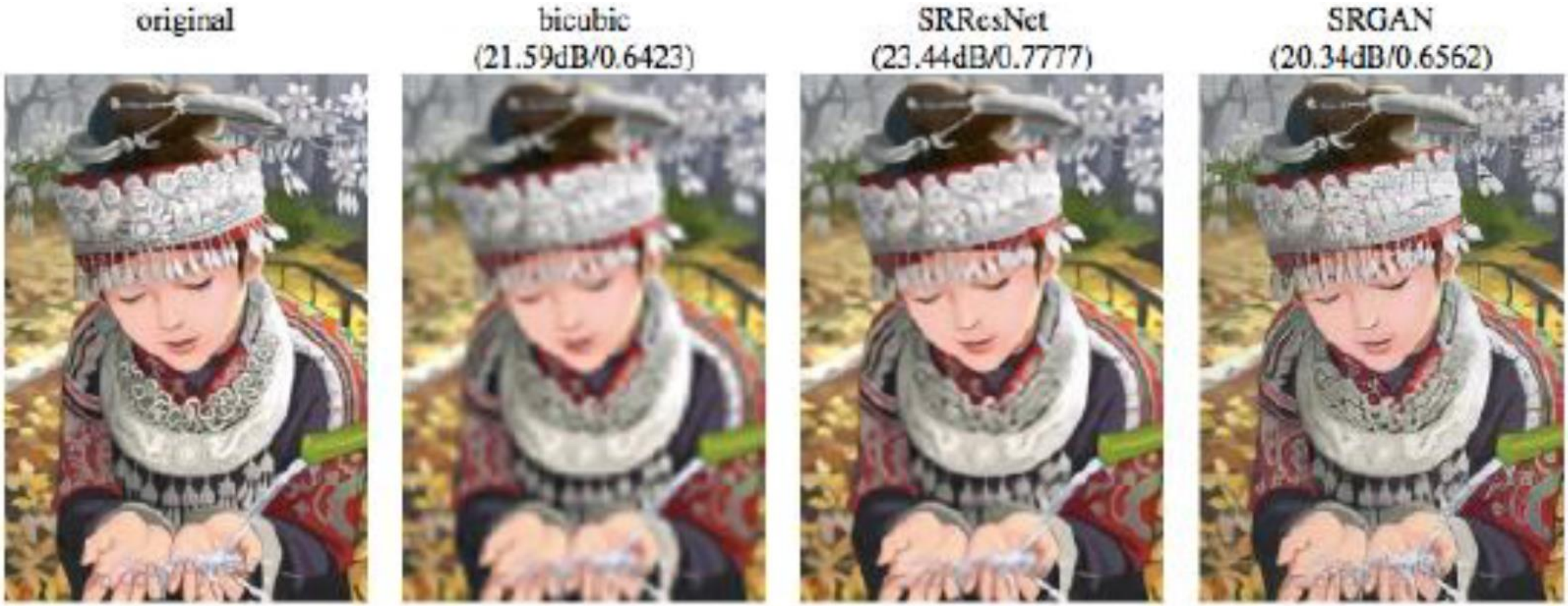
Applications of Generative Models

- Realistic samples for artwork, super-resolution, colorization, etc.



Applications of Generative Models

- Super Resolution
  - C. Ledig, et al., "**Photo-realistic Single Image Super-Resolution using a Generative Adversarial Network**", CoRR, abs/1609.04802.



Applications of Generative Models

- In-Painting
  - Raymond Yeh, et al., "**Semantic Image Inpainting with Perceptual and Contextual Losses**", arXiv 1607.07539

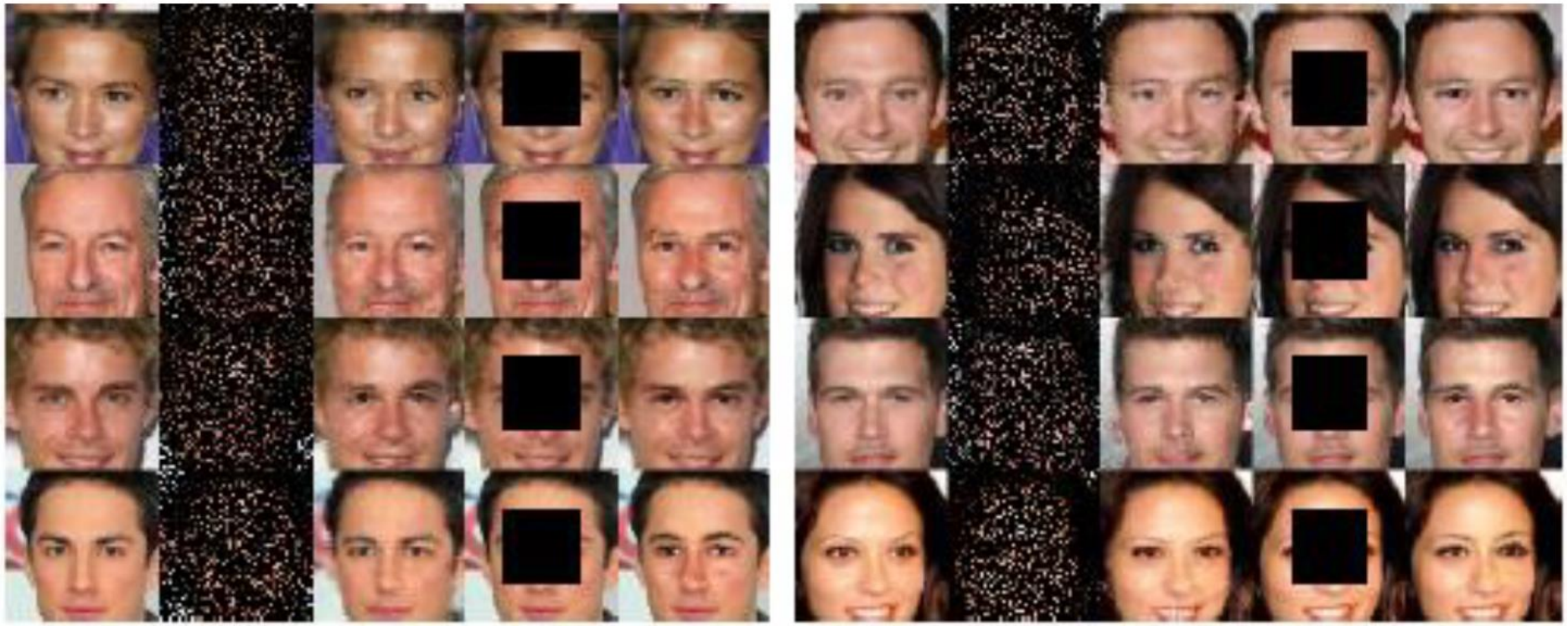
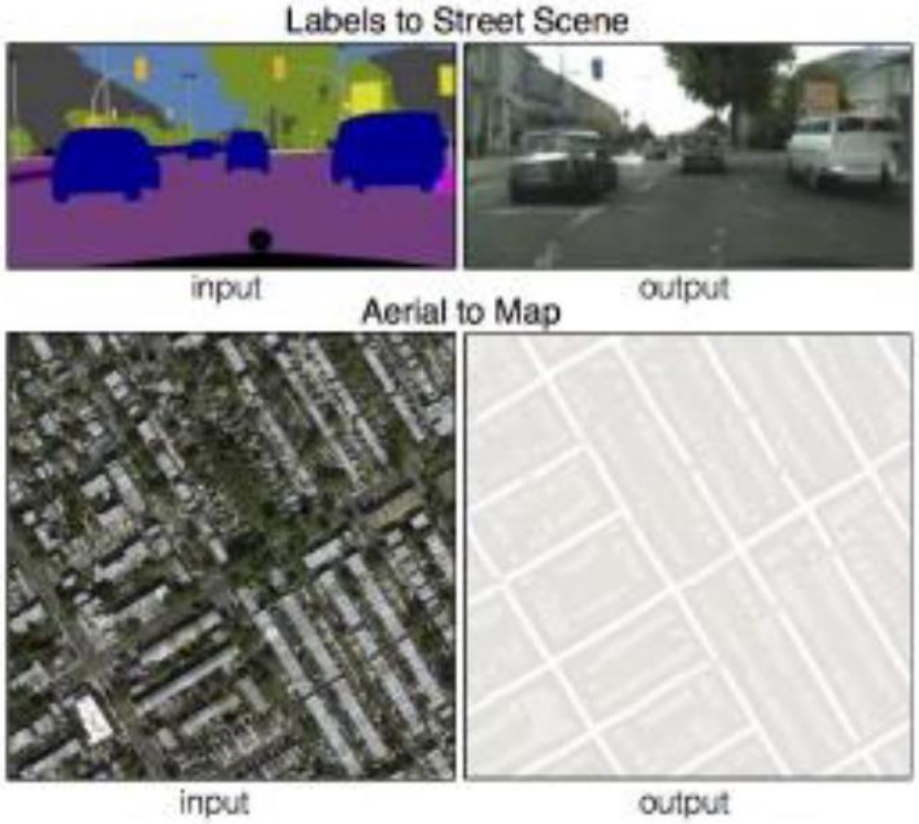


Figure 4: For each example, **Column 1:** Original images from the dataset. **Column 2:** Images with 80% random missing pixels. **Column 3:** Inpainting of column 2 by our method. **Column 4:** Image with large central region missing. **Column 5:** Inpainting of column 4 by our method.

(Paired) Image-to-Image Translation

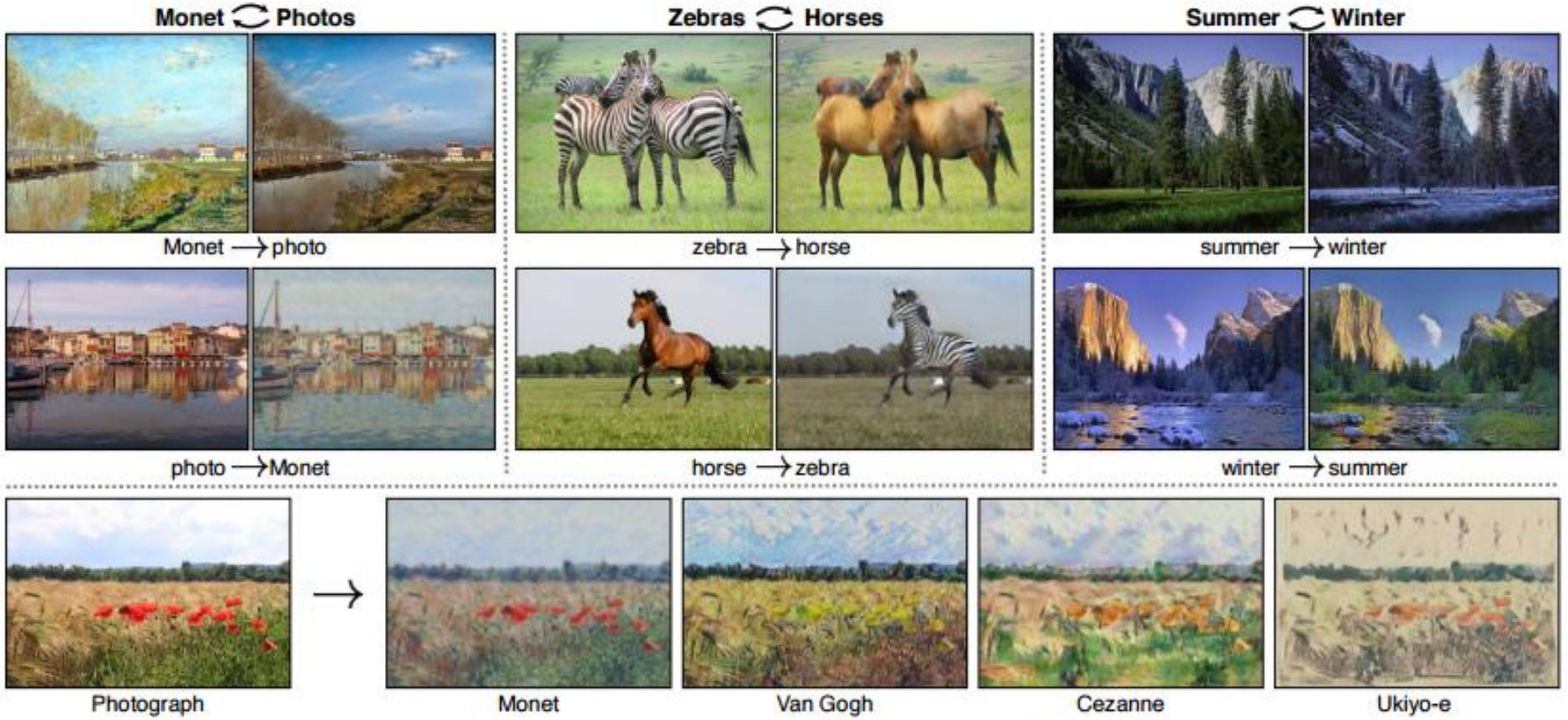
- pix2pix: Paired Image-to-Image Translation



Phillip Isola et al. Image-to-Image Translation with Conditional Adversarial Networks, CVPR'17

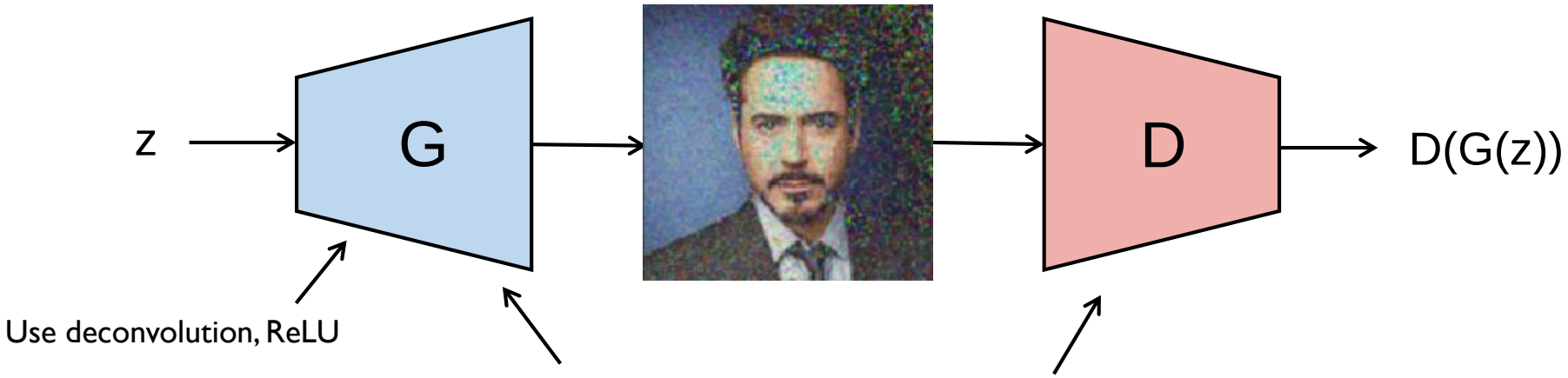
- CycleGAN: Unpaired Image-to-Image Translation

presents a GAN model that transfer an image from a source domain A to a target domain B in the absence of paired examples.

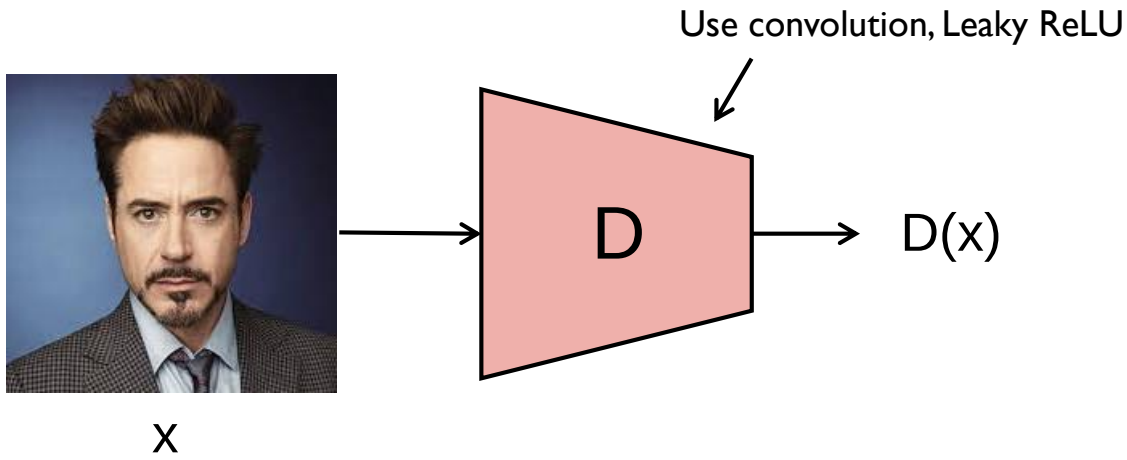


Jun-Yan Zhu et al. Unpaired Image-to-Image Translation using Cycle Consistent Adversarial Networks, 2017

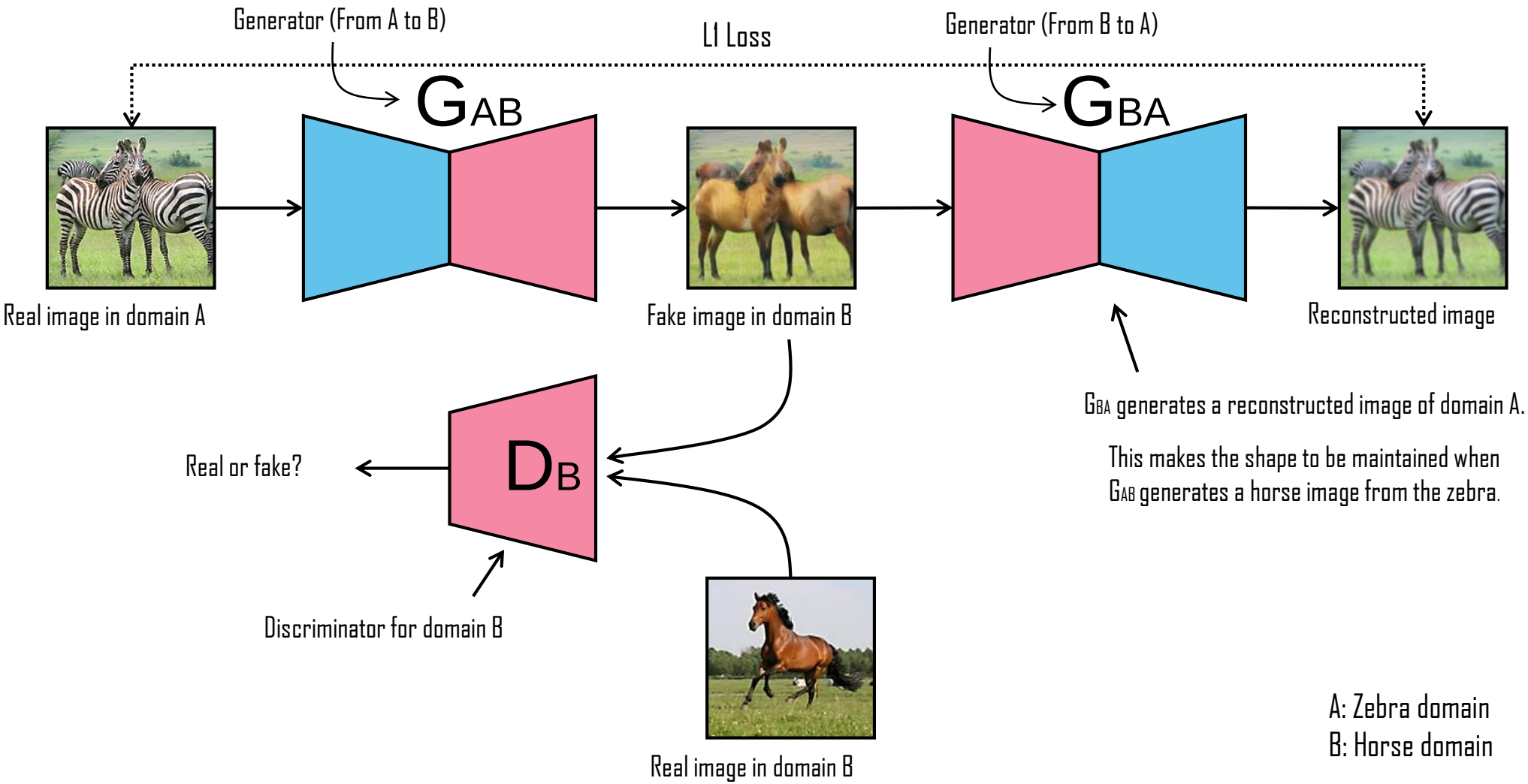
DCGAN (Generative Model)



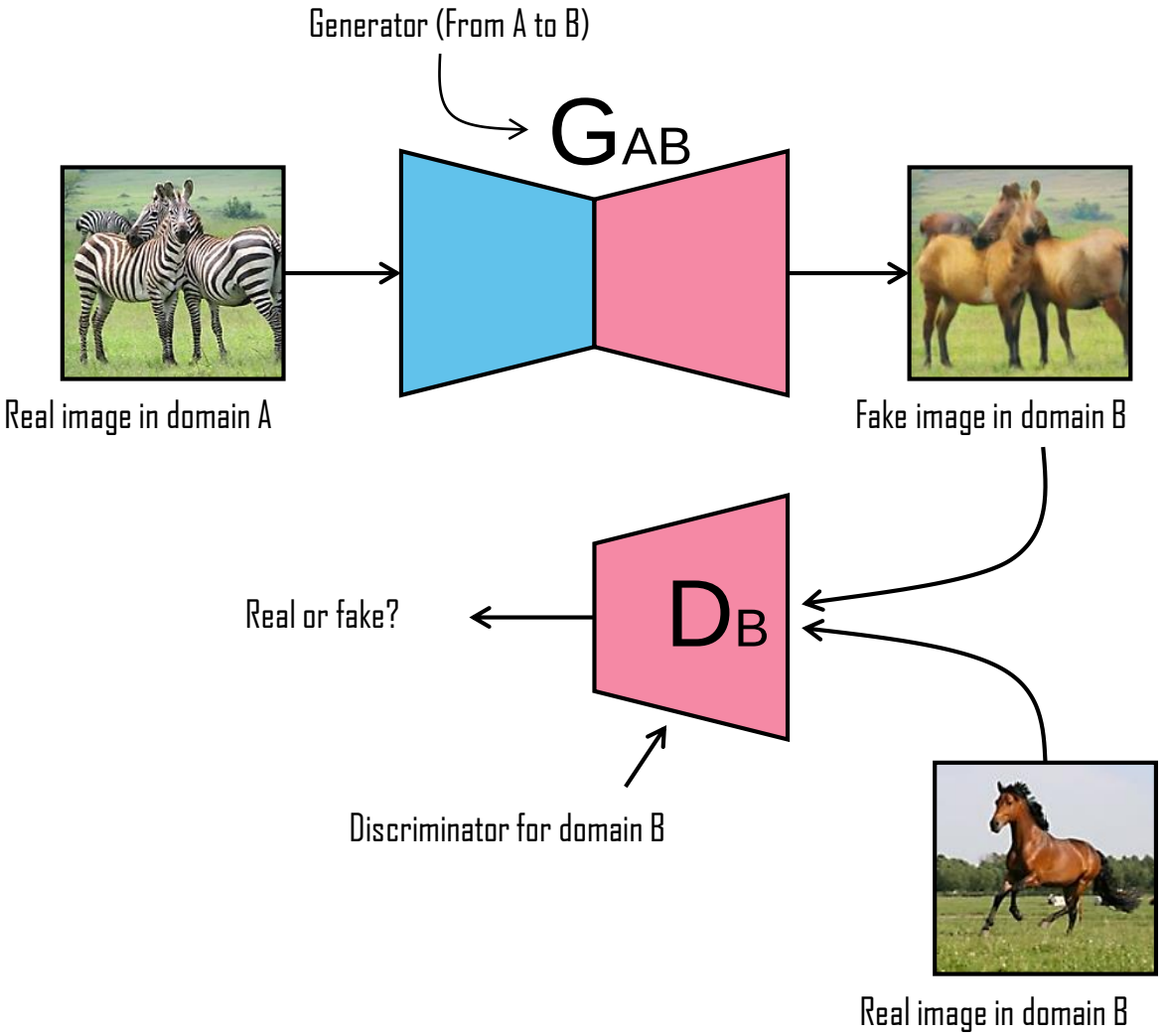
- No pooling layer (Instead strided convolution)
- Use batch normalization
- Adam optimizer(lr=0.0002, beta1=0.5, beta2=0.999)



CycleGAN

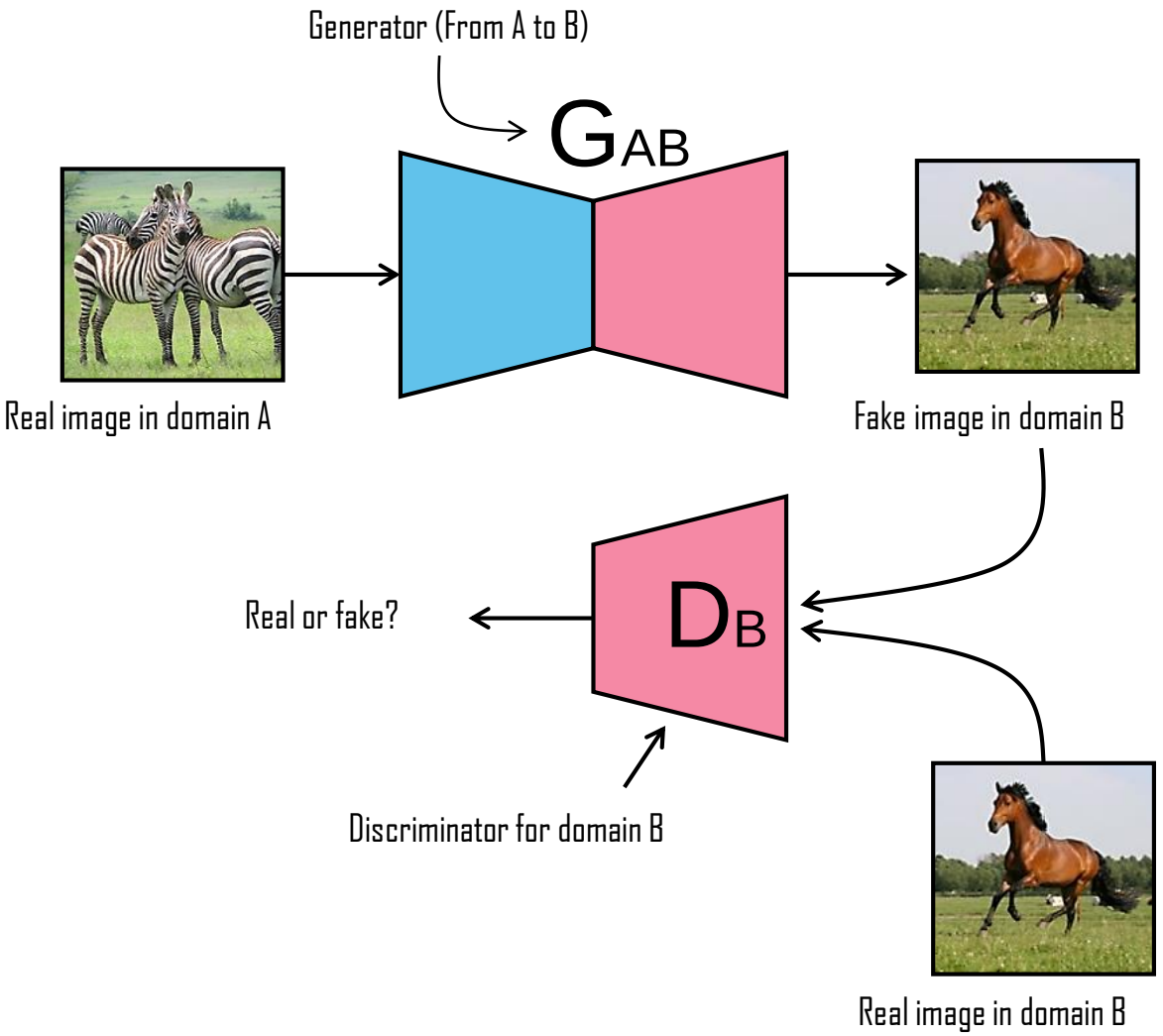

Generator (From A to B)
L1 Loss
Generator (From B to A)
GAB
GBA
Real image in domain A
Fake image in domain B
Reconstructed image
GBA generates a reconstructed image of domain A.
This makes the shape to be maintained when GAB generates a horse image from the zebra.
Real or fake?
DB
Discriminator for domain B
Real image in domain B
A: Zebra domain
B: Horse domain

CycleGAN



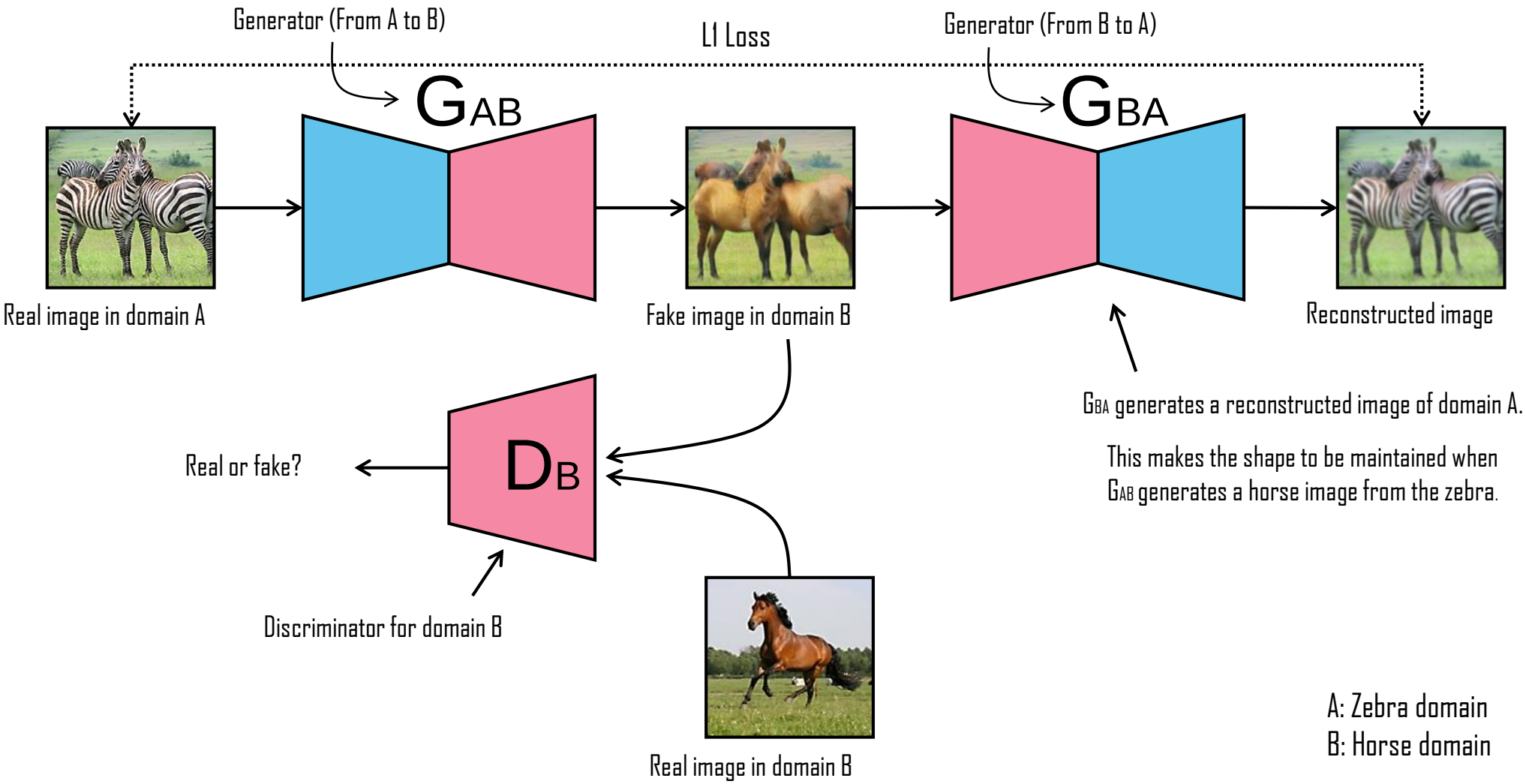
A: Zebra domain
B: Horse domain

CycleGAN

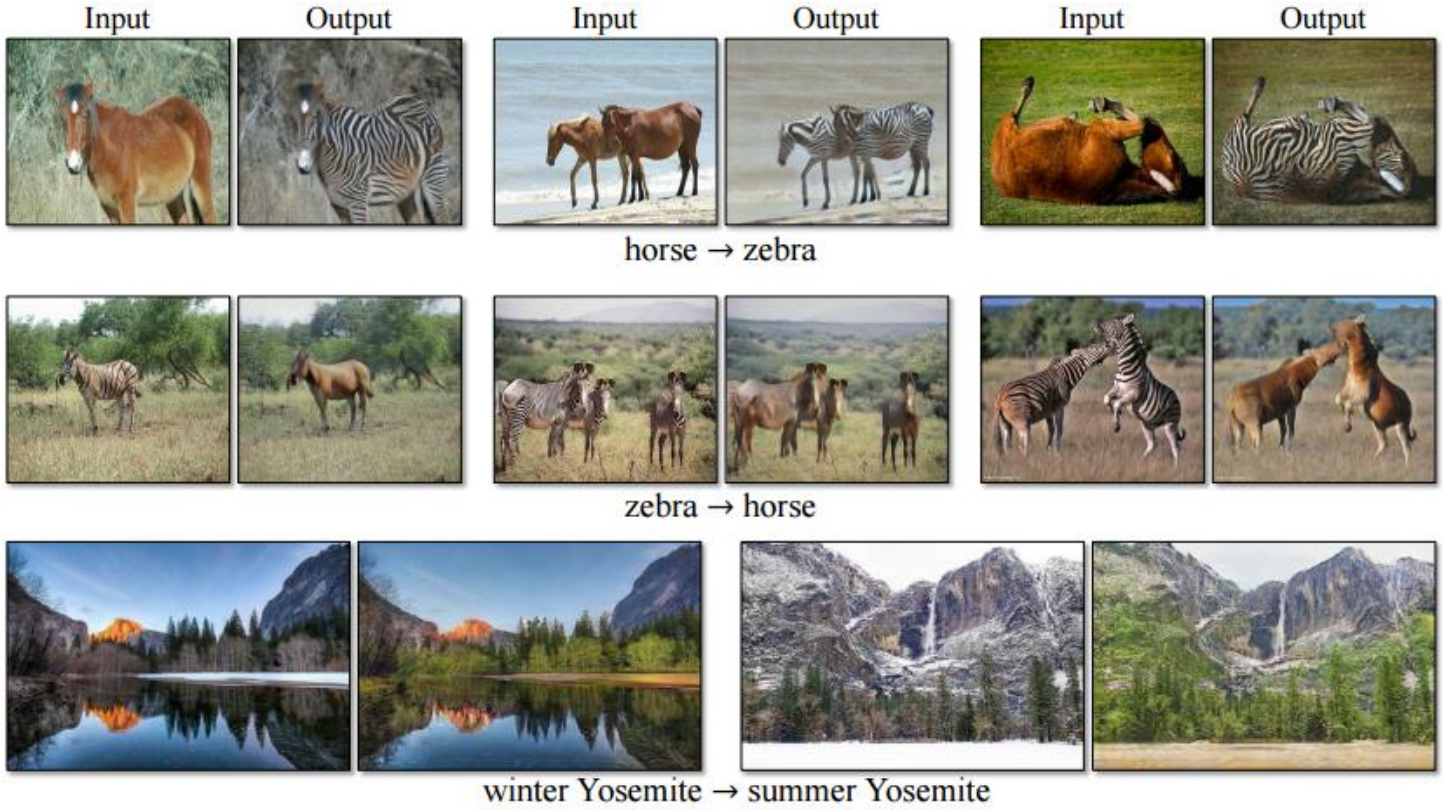


A: Zebra domain
B: Horse domain

CycleGAN


Generator (From A to B)
L1 Loss
Generator (From B to A)
GAB
GBA
Real image in domain A
Fake image in domain B
Reconstructed image
GBA generates a reconstructed image of domain A.
This makes the shape to be maintained when GAB generates a horse image from the zebra.
Real or fake?
DB
Discriminator for domain B
Real image in domain B
A: Zebra domain
B: Horse domain

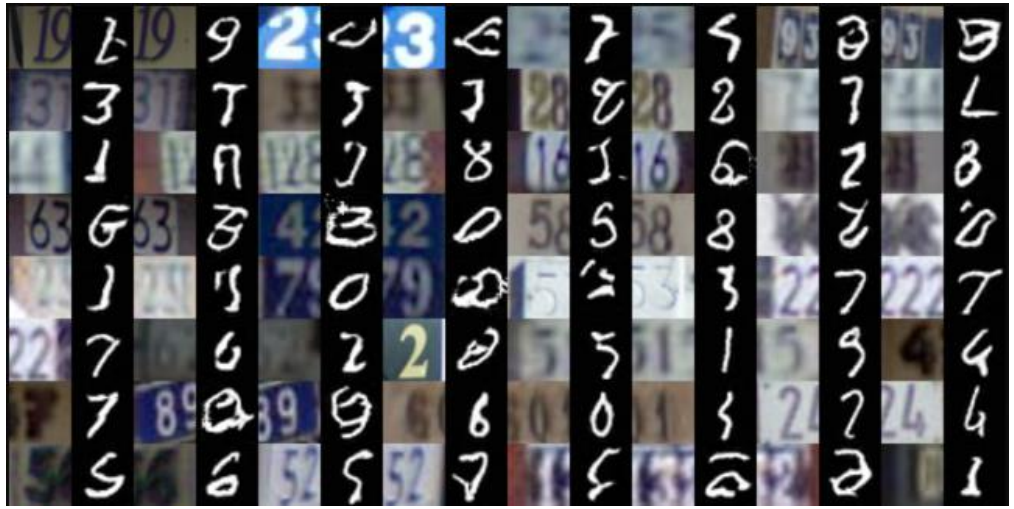
- Results



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

- Results

  Odd columns contain real images and even columns contain generated images.



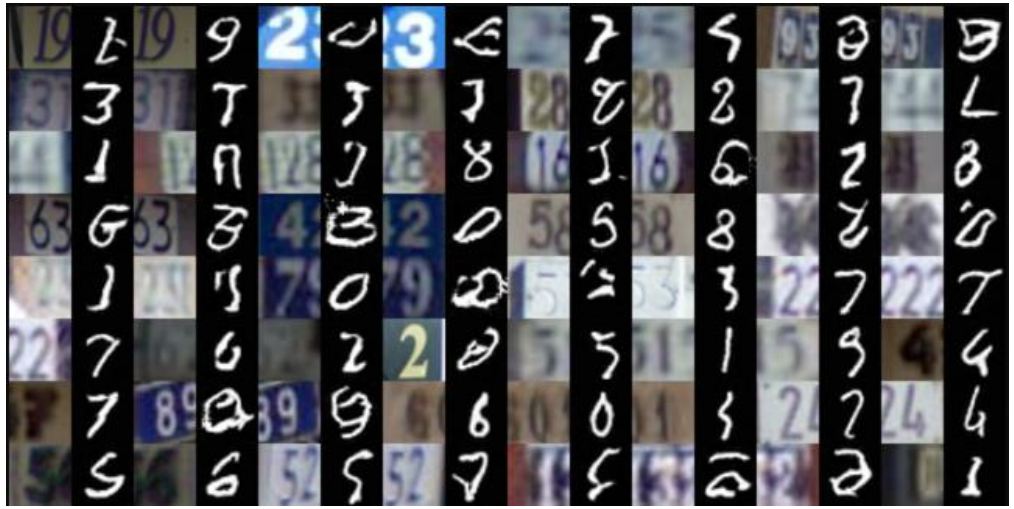
SVHN-to-MNIST

MNIST-to-SVHN

https://github.com/yunjey/mnist-svhn-transfer

- Results

  Odd columns contain real images and even columns contain generated images.



SVHN-to-MNIST

MNIST-to-SVHN

https://github.com/yunjey/mnist-svhn-transfer

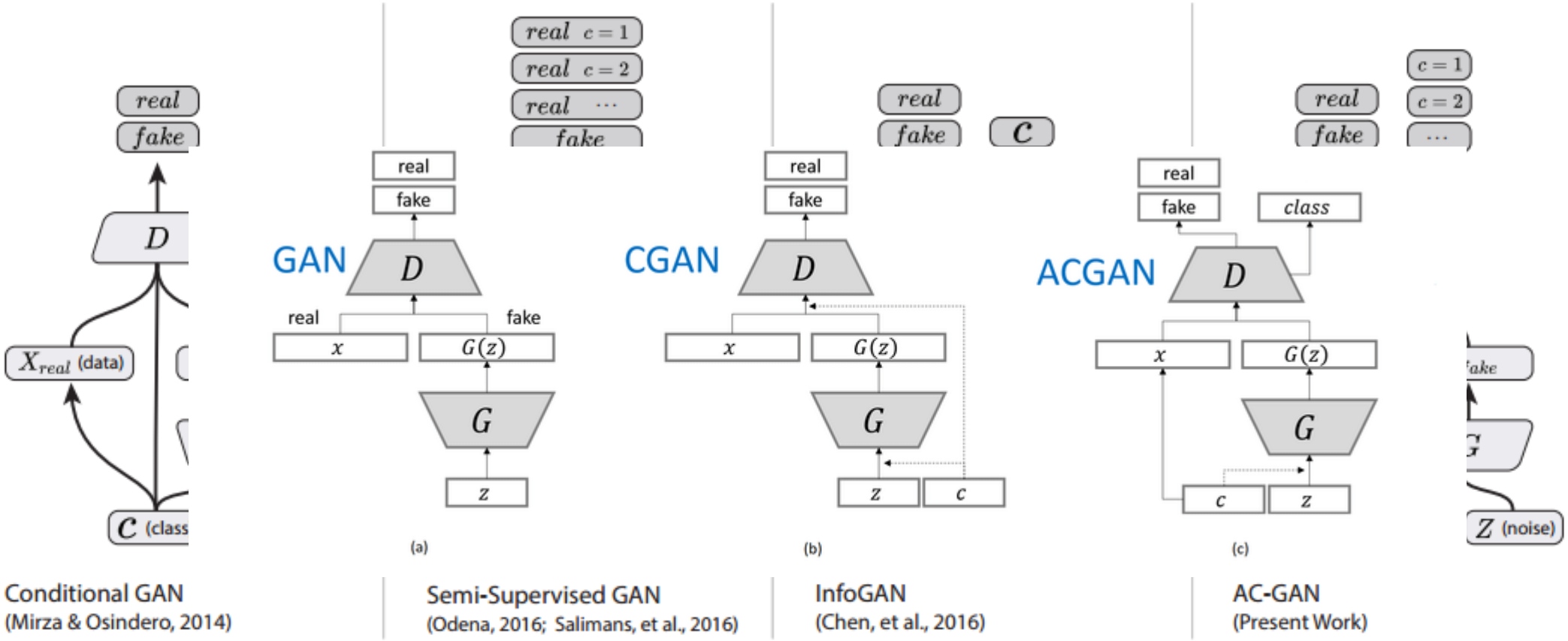
Conditional Generation (and Translation)

- An additionally given input works as a condition that steers the generation and translation processes in a user-driven manner.

- Two GAN-based models: CGAN and ACGAN

Conditional GAN and ACGAN

- Auxiliary Classifier GAN (ACGAN), 2016
  - Improves the training of GANs using class labels



Conditional GAN
(Mirza & Osindero, 2014)

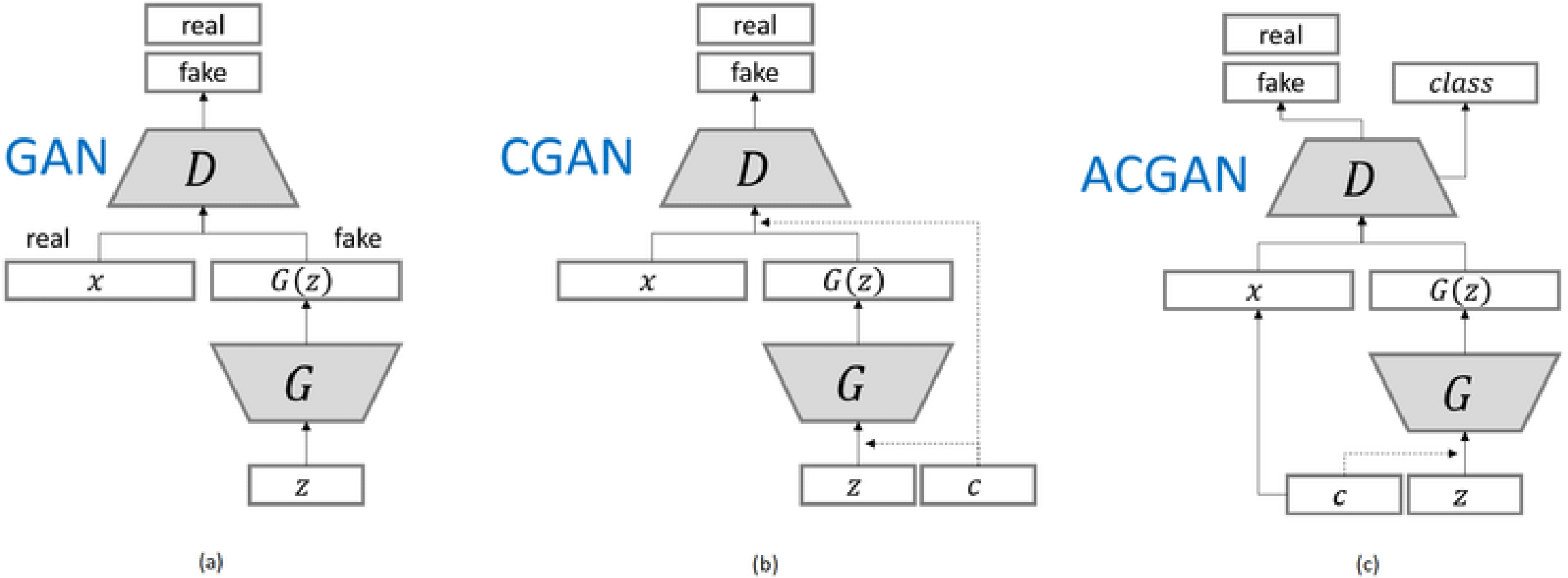
Semi-Supervised GAN
(Odena, 2016; Salimans, et al., 2016)

InfoGAN
(Chen, et al., 2016)

AC-GAN
(Present Work)

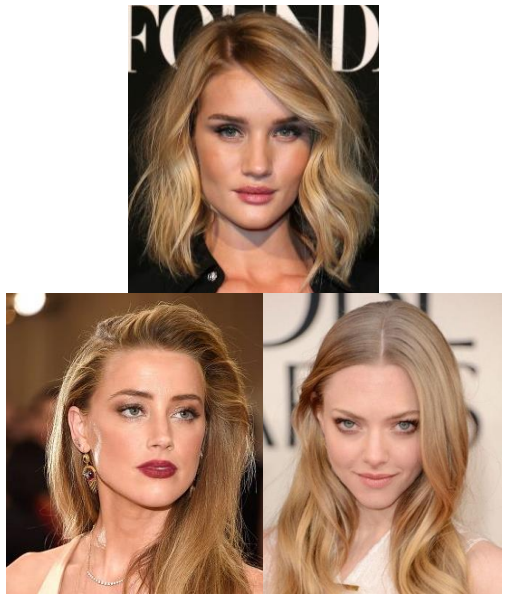
Conditional GAN and ACGAN

- Auxiliary Classifier GAN (ACGAN), 2016
  - Improves the training of GANs using class labels



MULTI-DOMAIN IMAGE-TO-IMAGE TRANSLATION

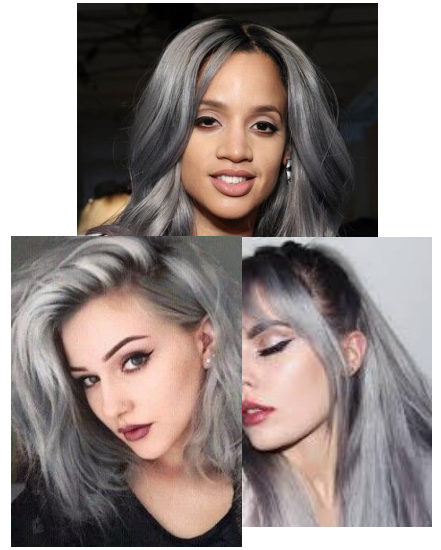
Domain A
(blond hair)



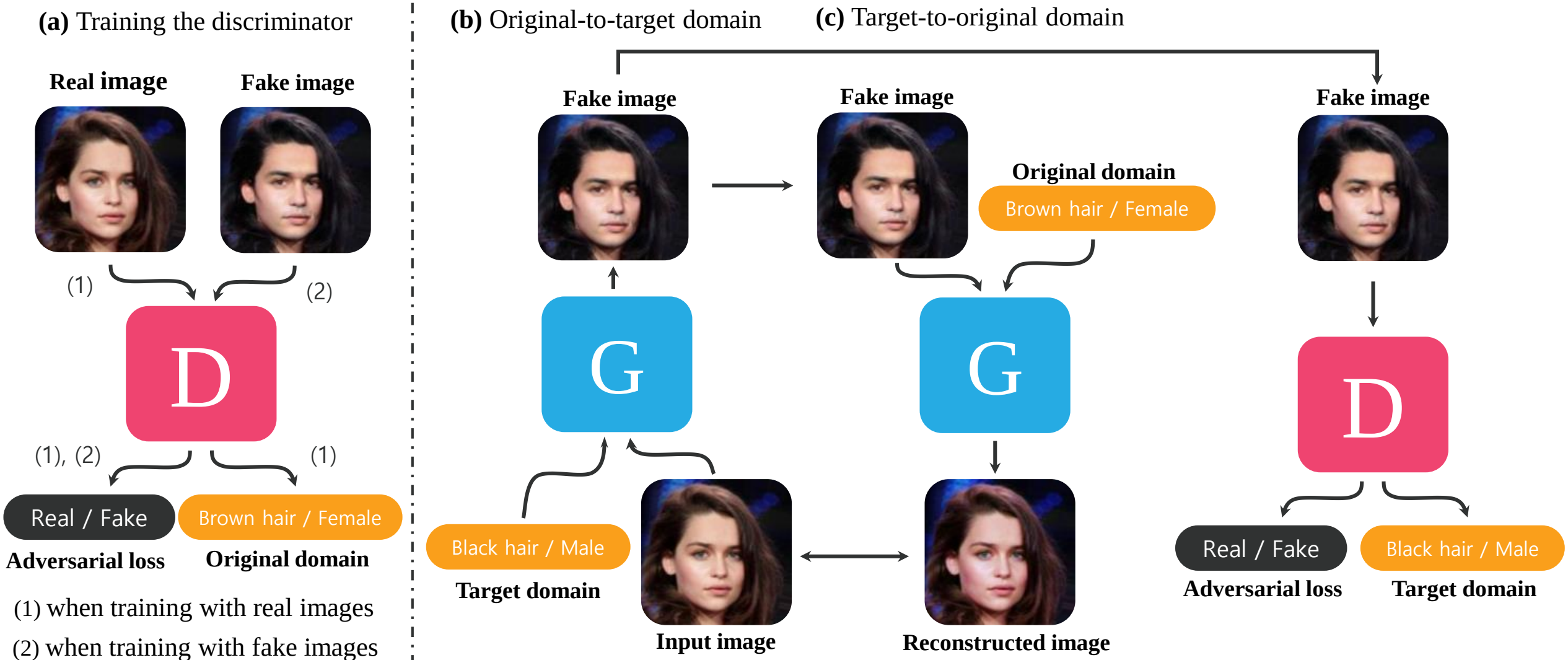
Domain B
(black hair)

Domain C
(brown hair)

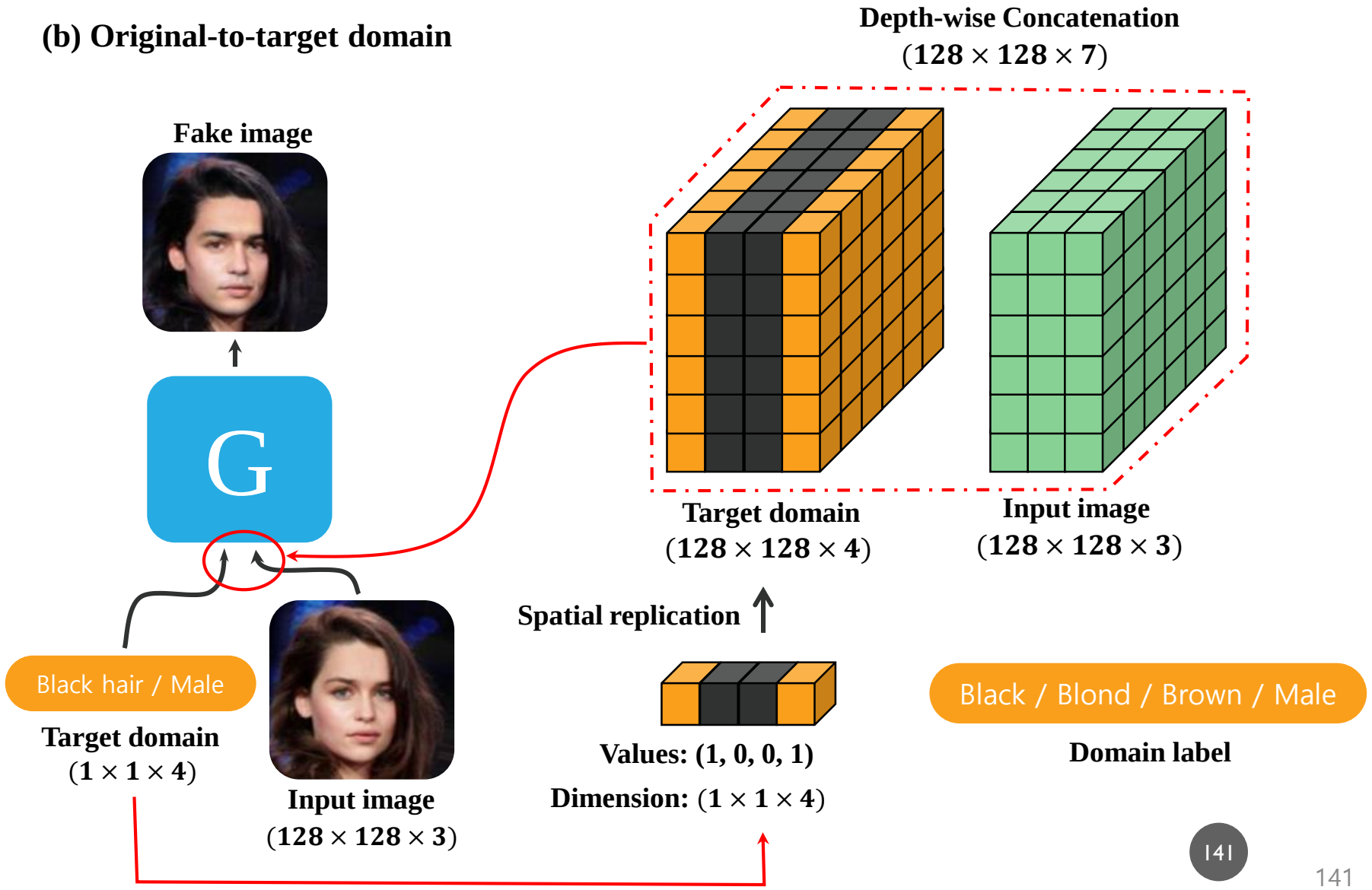
Domain D
(gray hair)



STARGAN


(a) Training the discriminator
Real image
Fake image
(1)
(2)
D
(1), (2)
(1)
Real / Fake
Brown hair / Female
Adversarial loss
Original domain
(1) when training with real images
(2) when training with fake images
(b) Original-to-target domain
(c) Target-to-original domain
Fake image
Fake image
Fake image
Original domain
Brown hair / Female
G
G
D
Black hair / Male
Target domain
Input image
Reconstructed image
Real / Fake
Black hair / Male
Adversarial loss
Target domain

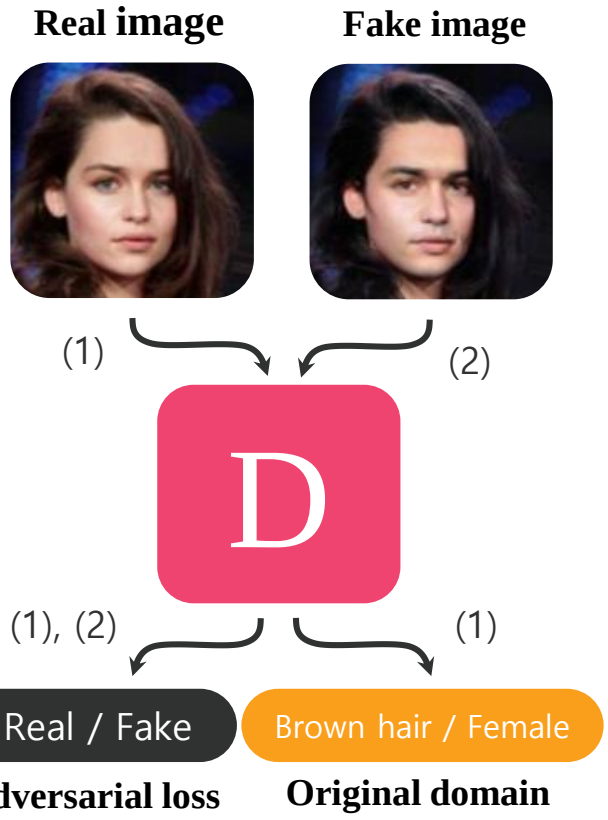
STARGAN


(b) Original-to-target domain
Depth-wise Concatenation
(128 × 128 × 7)
Fake image
G
Target domain
(128 × 128 × 4)
Input image
(128 × 128 × 3)
Spatial replication
Black hair / Male
Target domain
(1 × 1 × 4)
Input image
(128 × 128 × 3)
Values: (1, 0, 0, 1)
Dimension: (1 × 1 × 4)
Black / Blond / Brown / Male
Domain label

141

STARGAN

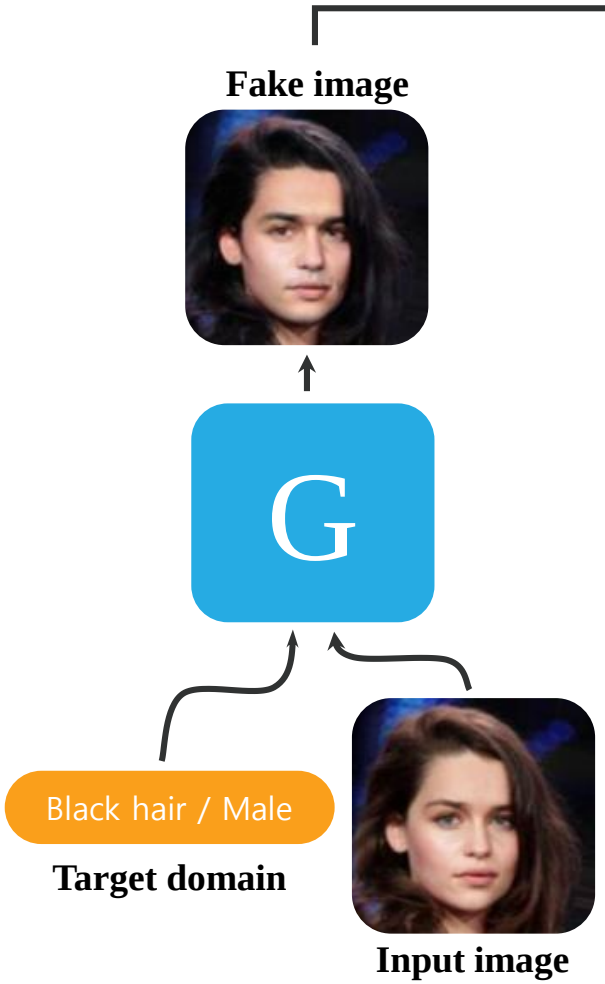
(a) Training the discriminator



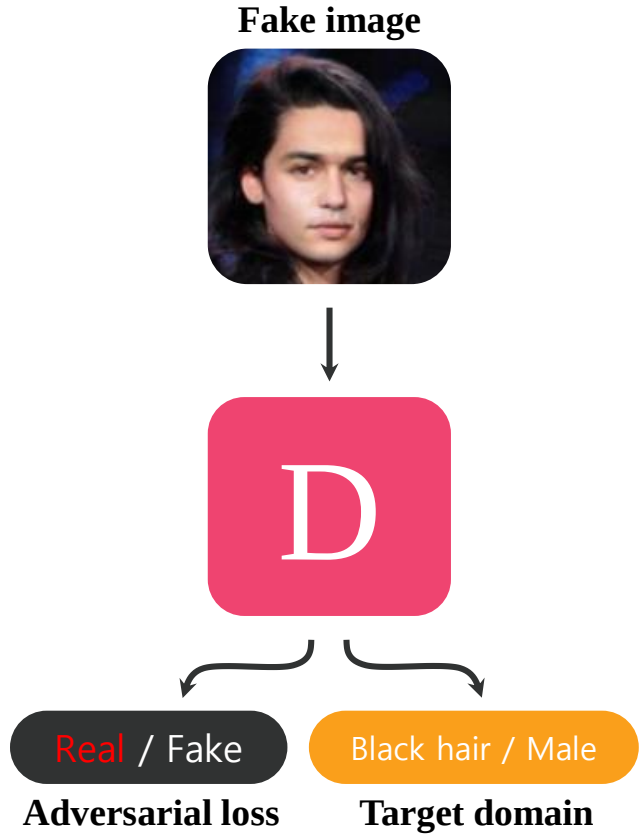
(1) when training with real images

(2) when training with fake images

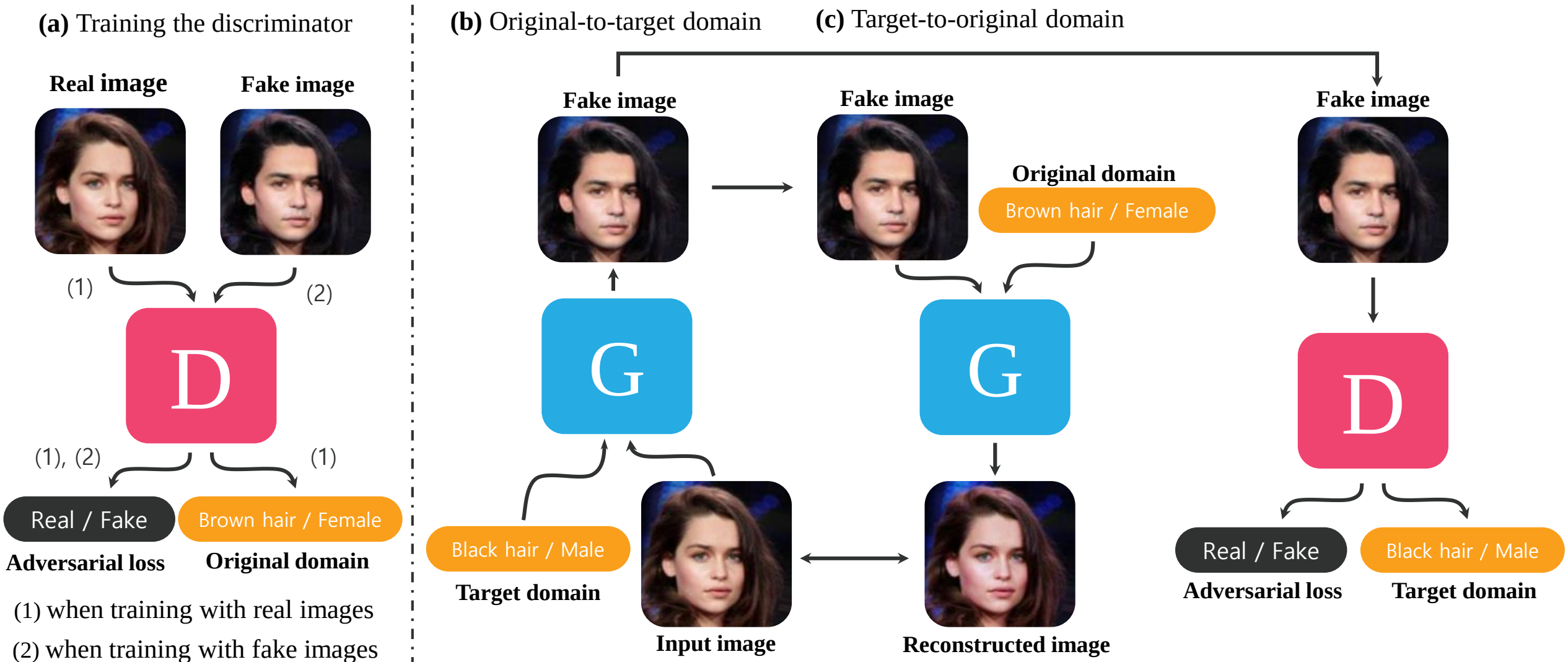
(b) Original-to-target domain



(d) Fooling the discriminator



STARGAN


(a) Training the discriminator
Real image
Fake image
(1)
(2)
D
(1), (2)
(1)
Real / Fake
Brown hair / Female
Adversarial loss
Original domain
(1) when training with real images
(2) when training with fake images
(b) Original-to-target domain
(c) Target-to-original domain
Fake image
Fake image
Fake image
Original domain
Brown hair / Female
G
G
D
Black hair / Male
Target domain
Input image
Reconstructed image
Real / Fake
Black hair / Male
Adversarial loss
Target domain