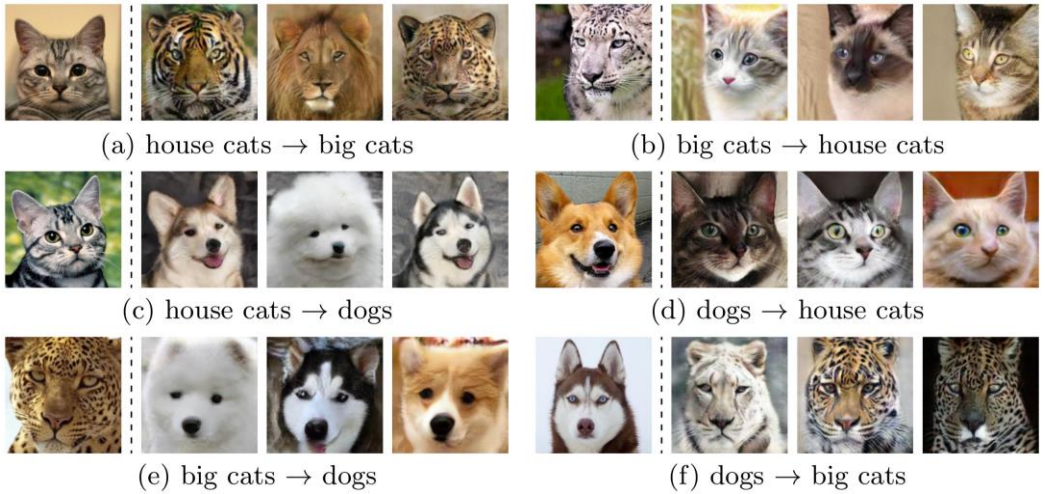
03
Image-to-Image Translation

Ch 3. Diverse Image-to-Image Translation

Clip 1. MUNIT, DRIT

Multimodal Image-to-Image Translation

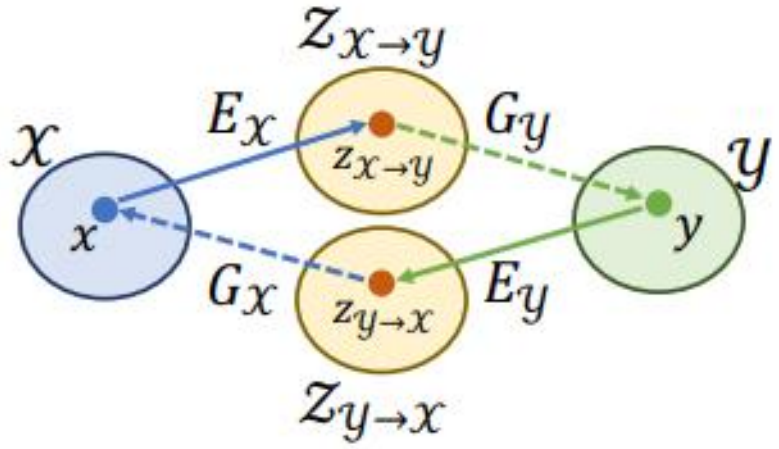
같은 Domain 내에도 다양한 Style을 가진 이미지들이 존재함.
(예: 검은 고양이, 하얀 고양이, 줄무늬 고양이)
같은 Domain 내에서도 다양한 Style을 반영한 Image-to-Image Translation을 하고자 함.



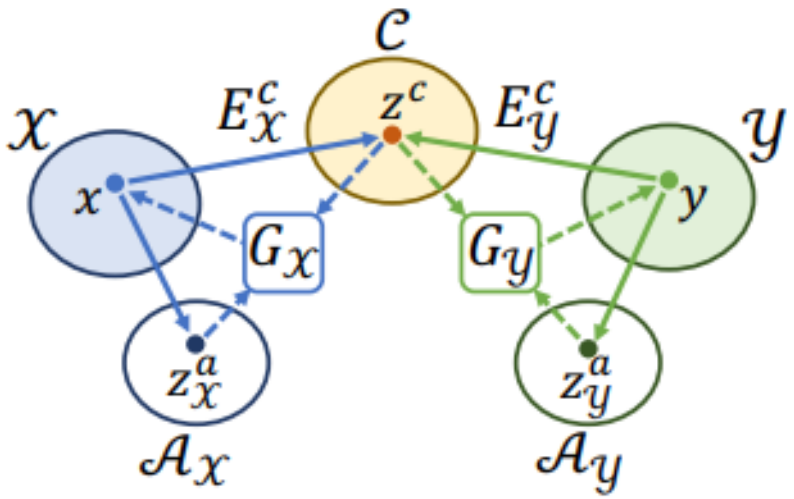
Huang et al., "Multimodal Unsupervised Image-to-Image Translation", ECCV 2018

CycleGAN vs. MUNIT, DRIT

CycleGAN은 각 Domain 별로 서로 다른 Latent space로 이미지를 mapping함.
반면, MUNIT과 DRIT은 Shared Content Space와 각 Domain의 Attribute Space로 Disentangle함.



[CycleGAN]

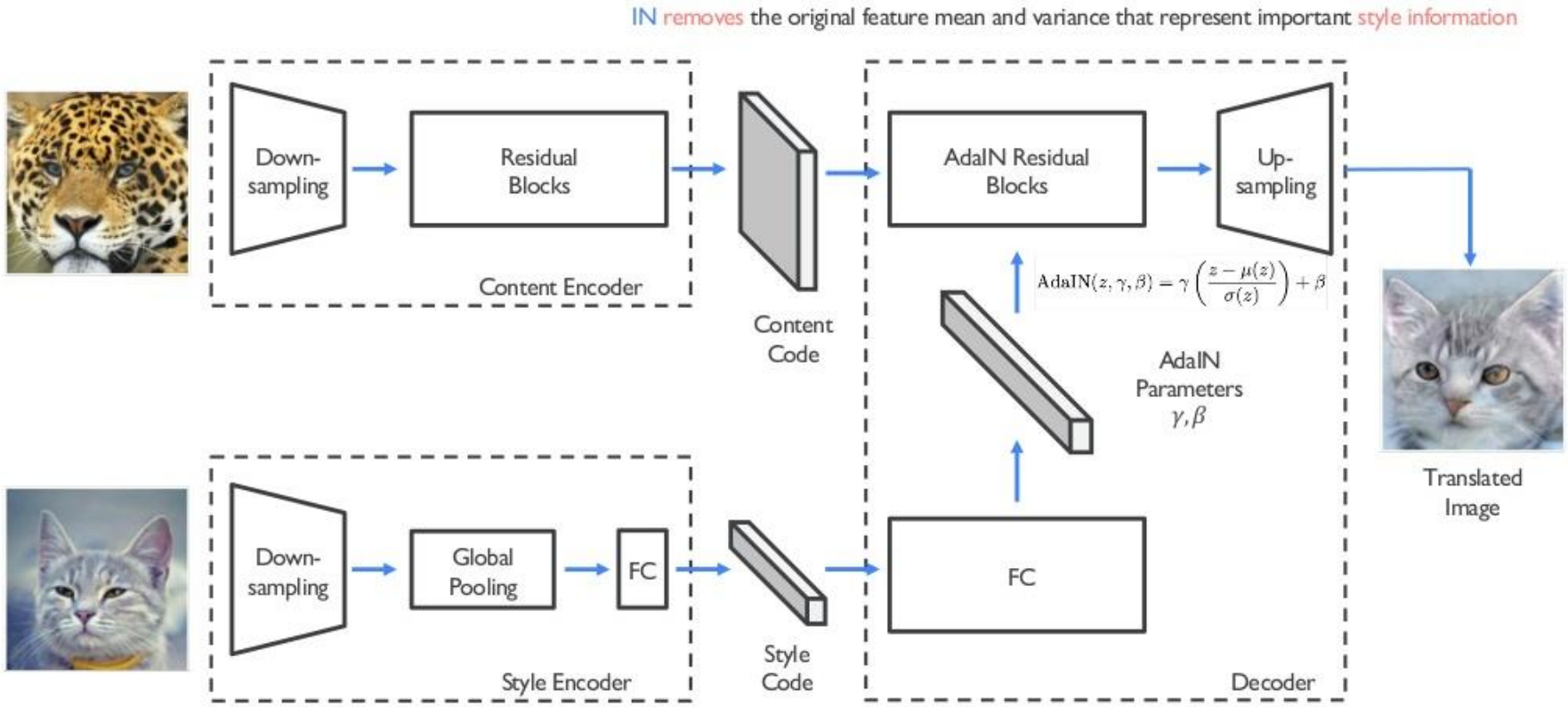


[MUNIT, DRIT]

Huang et al., "Multimodal Unsupervised Image-to-Image Translation", ECCV 2018

MUNIT의 모델 구조

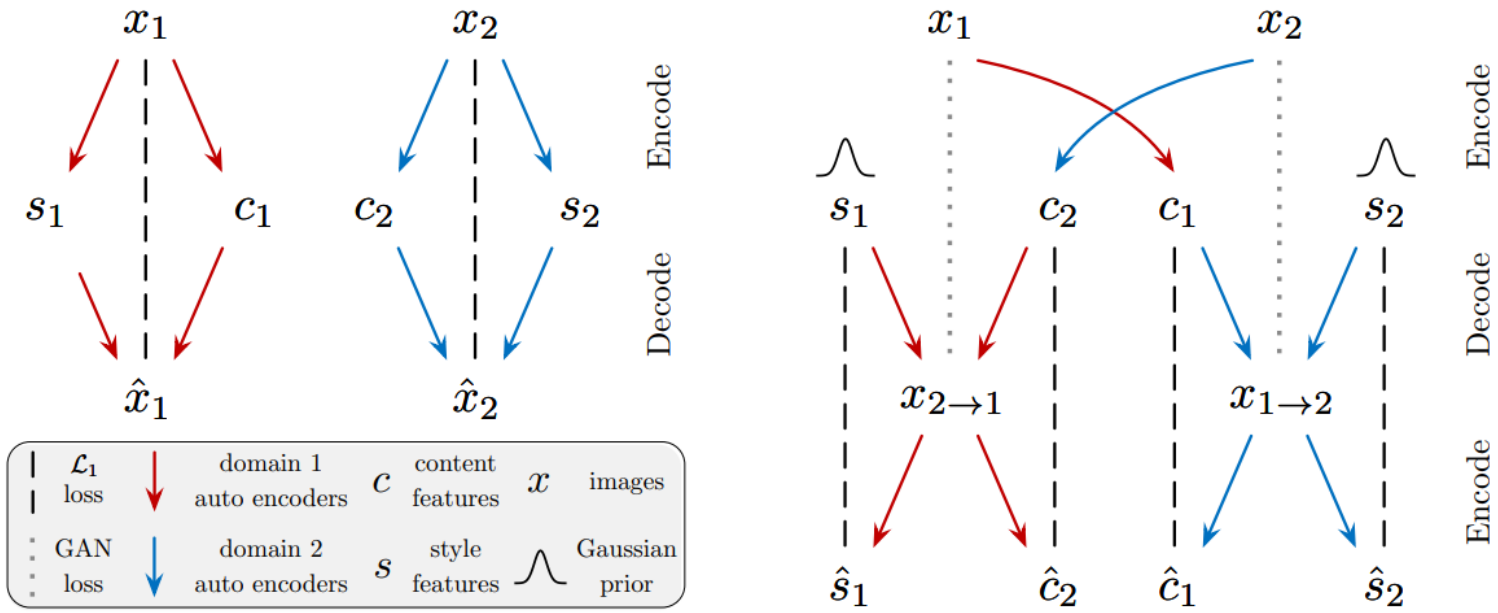
Style 이미지를 vector 형태로 encoding한 후,
AdaIN으로 Content Code에 Style을 반영시킴.



Huang et al., "Multimodal Unsupervised Image-to-Image Translation", ECCV 2018

MUNIT의 손실 함수

MUNIT은 각 Domain별로 Reconstruction Loss (L_1 Loss)와
Cross-domain Translation (GAN Loss & Cycle Consistency Loss)을 통해 학습을 진행함.



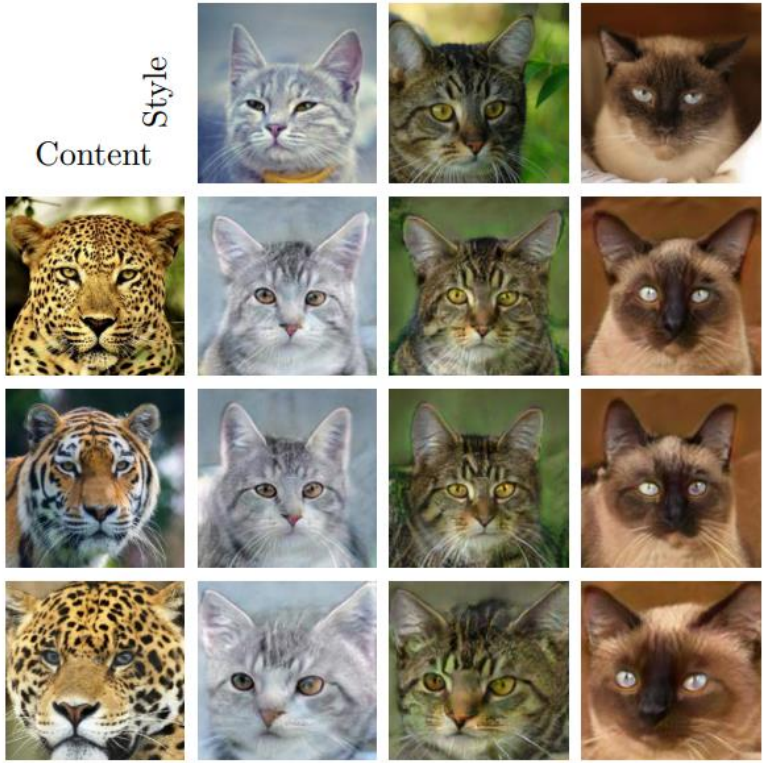
(a) Within-domain reconstruction (b) Cross-domain translation

Huang et al., "Multimodal Unsupervised Image-to-Image Translation", ECCV 2018

MUNIT의 이미지 변환 결과

MUNIT은 아래 결과와 같이 다양한 Style 이미지를 반영해서 이미지를 변환할 수 있음.

(a) edges → shoes

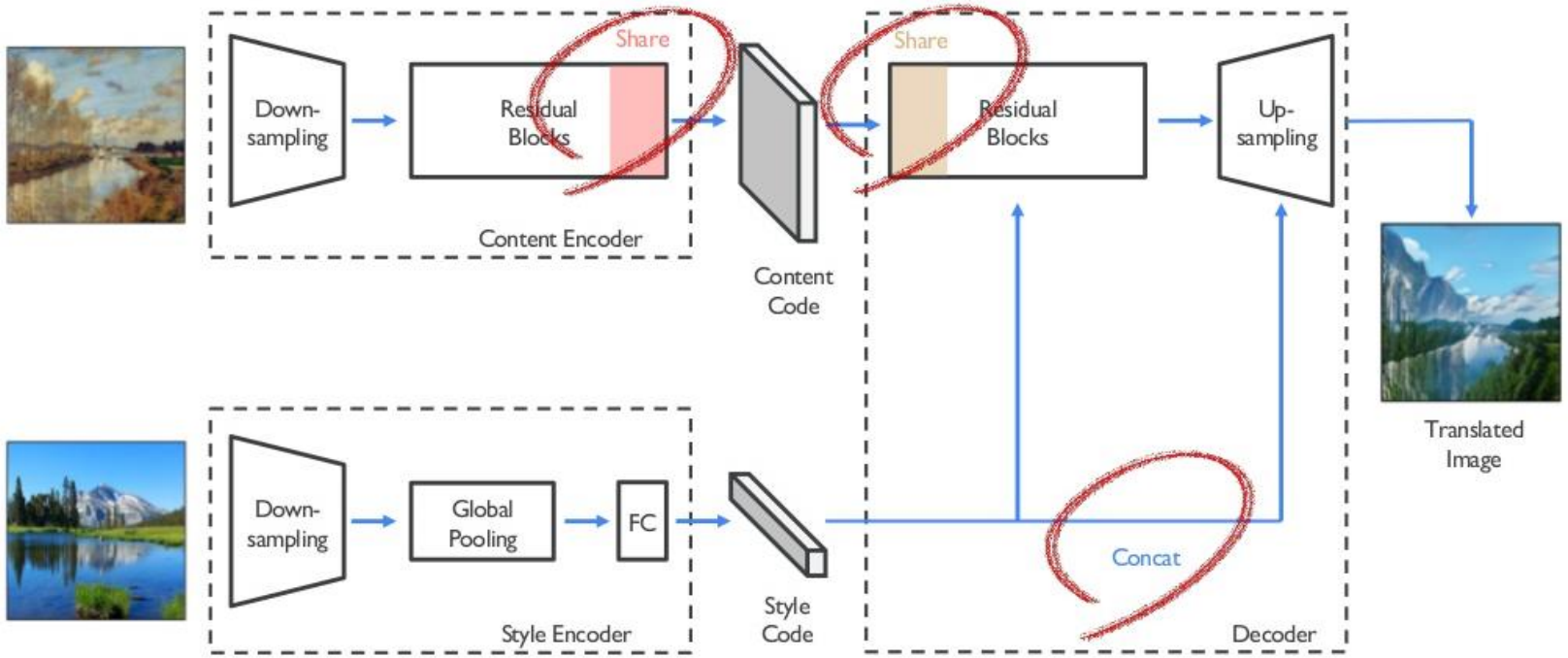


(b) big cats → house cats

Huang et al., "Multimodal Unsupervised Image-to-Image Translation", ECCV 2018

DRIT

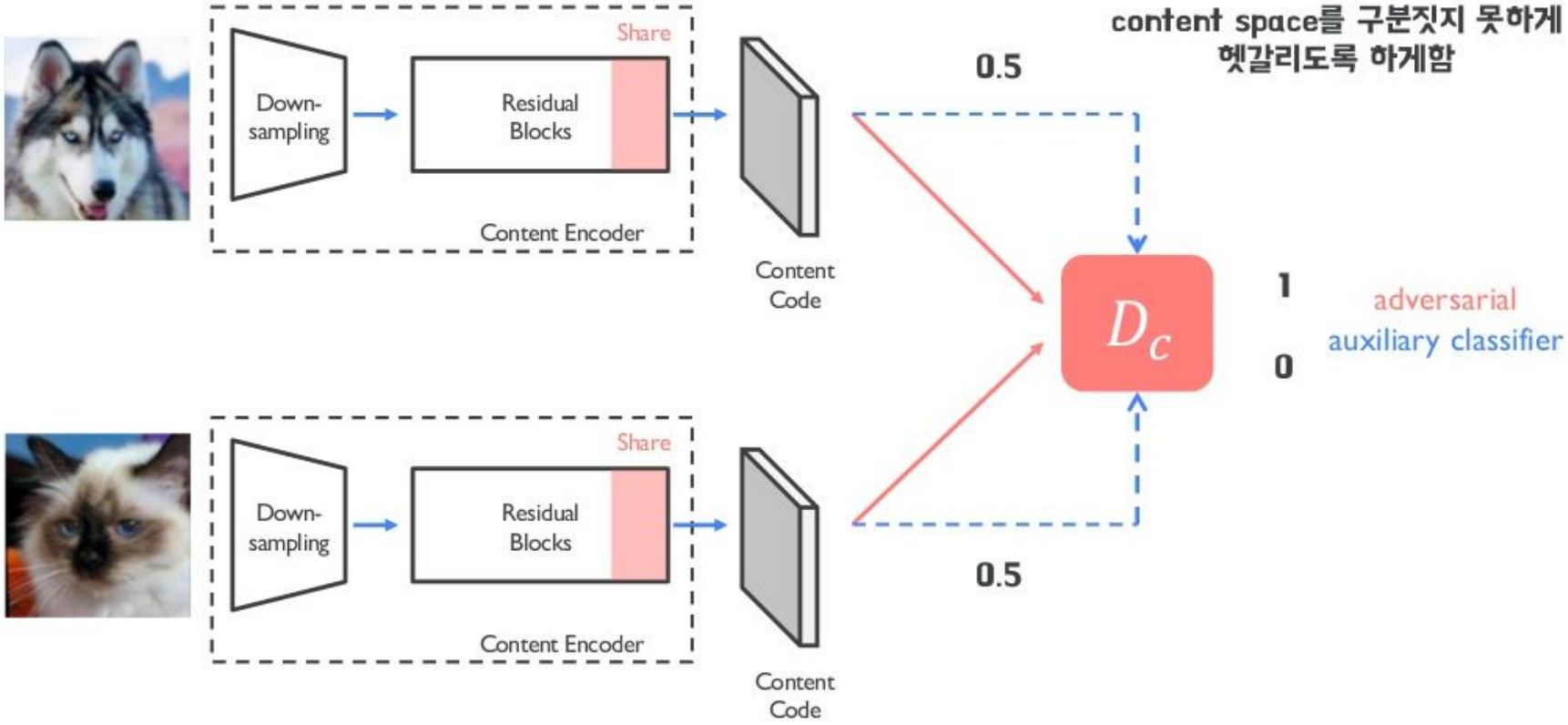
DRIT이 MUNIT과 다른 점은 크게 세 가지임.
(1) Content Encoder의 weights를 sharing하는 점.
(2) Style Code를 AdaIN 대신 Concat 연산으로 주입하는 점.



Lee et al., "Diverse Image-to-Image Translation via Disentangled Representations", ECCV 2018

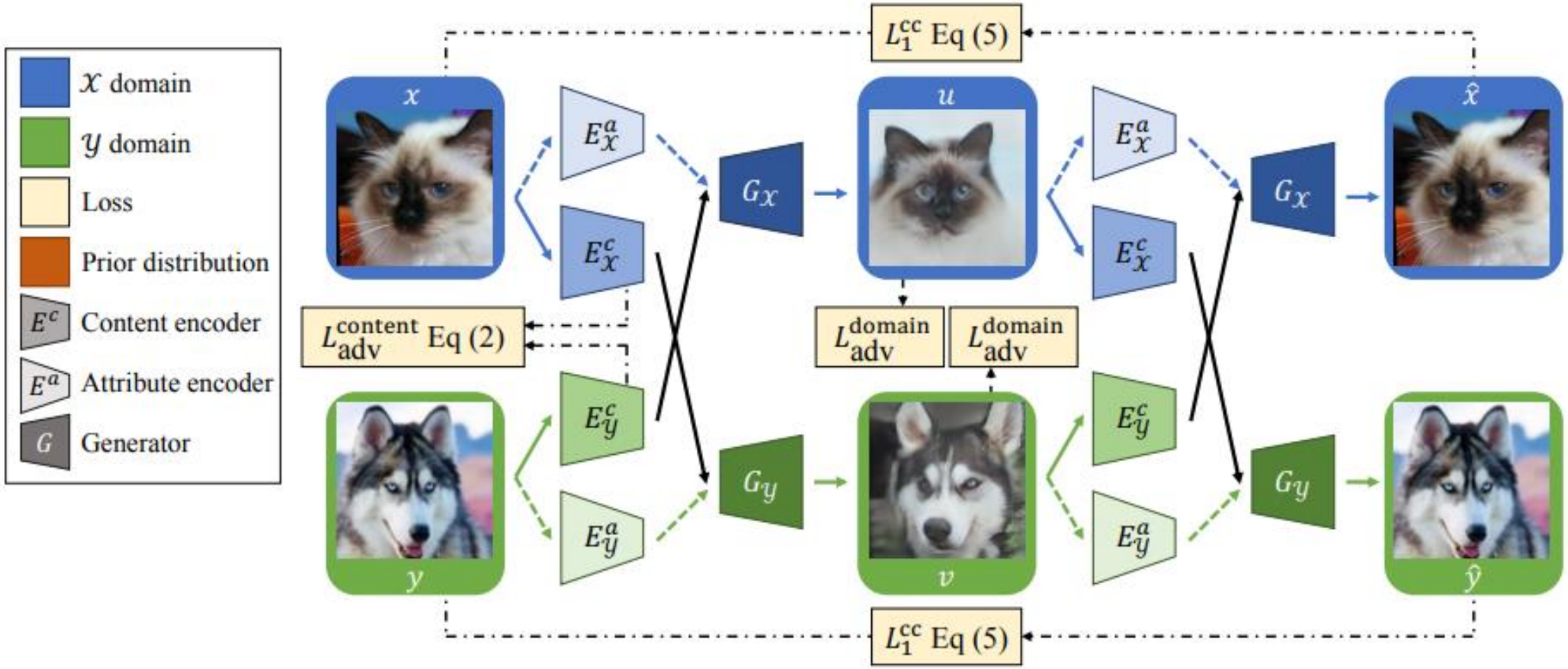
DRIT

DRIT이 MUNIT과 다른 점은 크게 3가지임.
(3) Content Discriminator를 사용함.



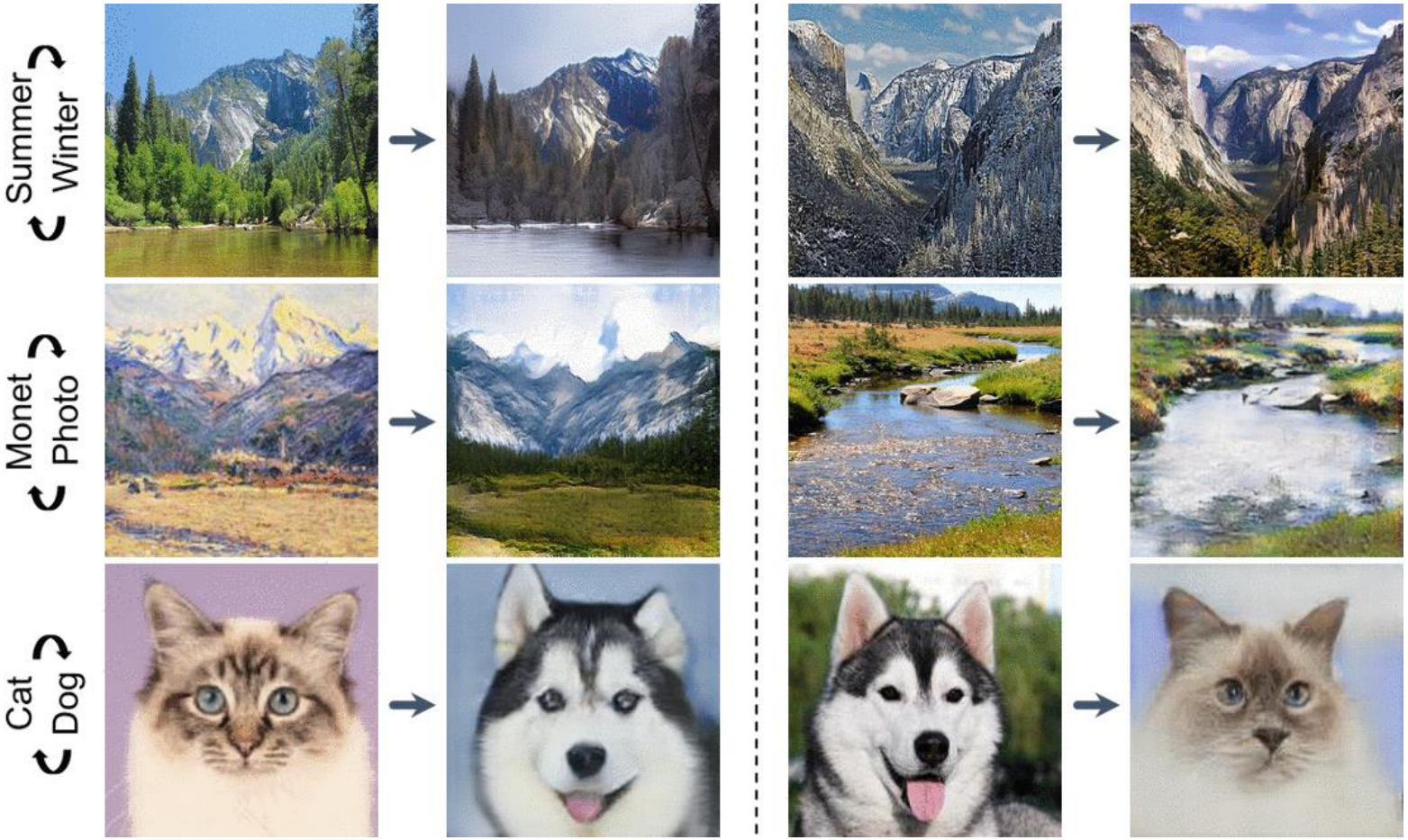
Lee et al., "Diverse Image-to-Image Translation via Disentangled Representations", ECCV 2018

DRIT의 손실함수



Lee et al., "Diverse Image-to-Image Translation via Disentangled Representations", ECCV 2018

DRIT의 이미지 변환 결과



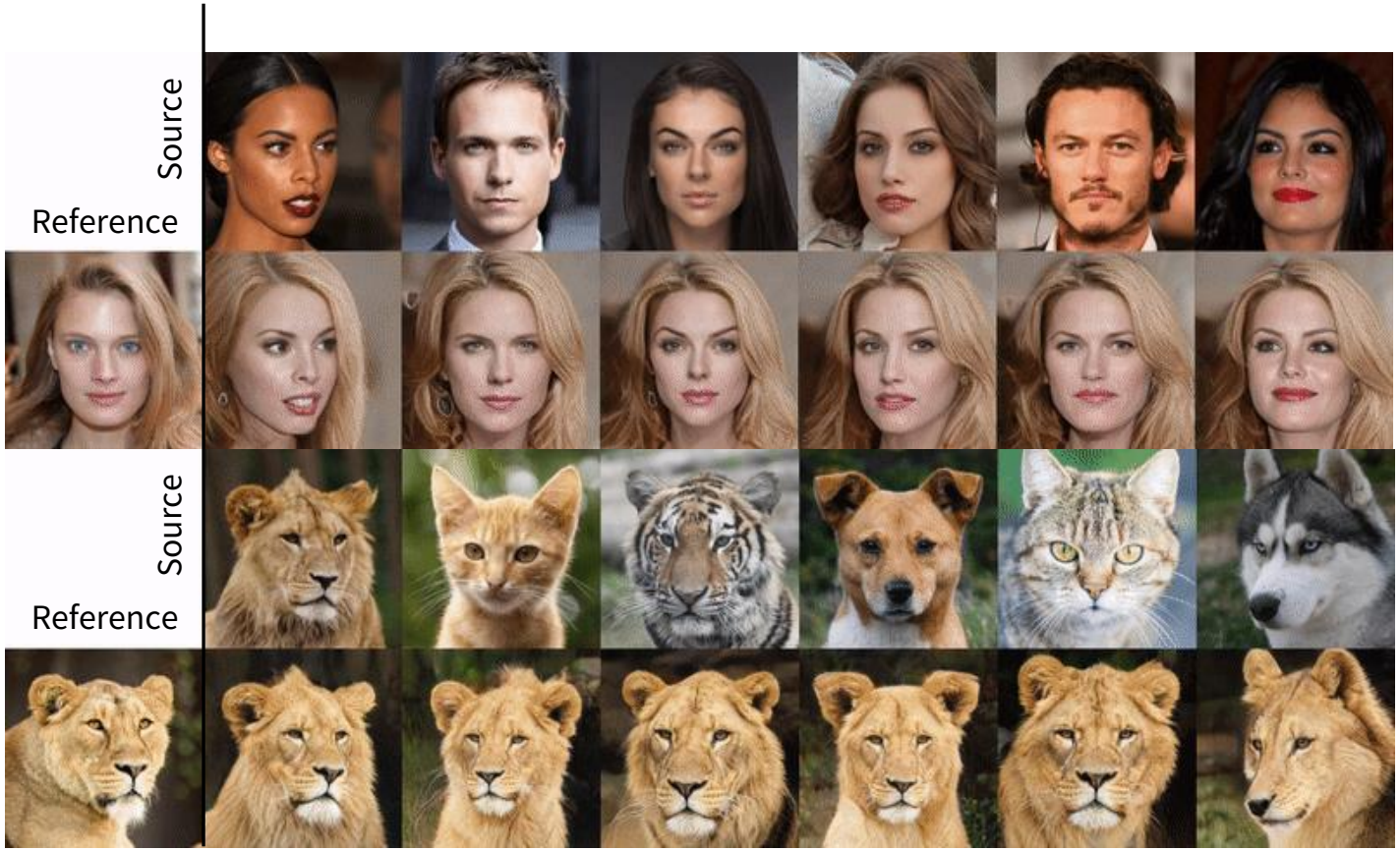
Lee et al., "Diverse Image-to-Image Translation via Disentangled Representations", ECCV 2018

Ch 3. Diverse Image-to-Image Translation

Clip 2. StarGAN-v2

StarGAN-v2

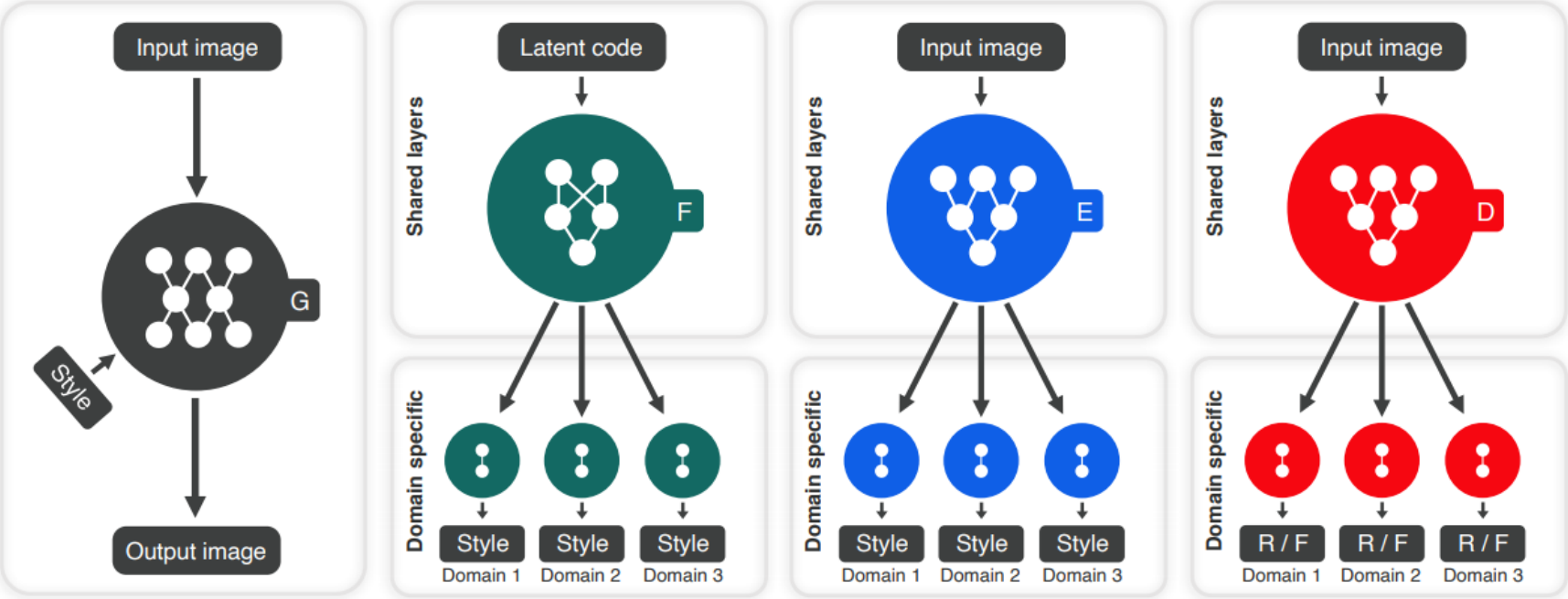
StarGAN-v2는 하나의 모델로 multi-domain 및 multi-modal 모델을 수행함



Choi et al., "StarGAN v2: Diverse Image Synthesis for Multiple Domains", CVPR 2020

StarGAN-v2 의 구성 모듈

StarGAN-v2는 크게 4가지 모듈로 구성되어 있음.

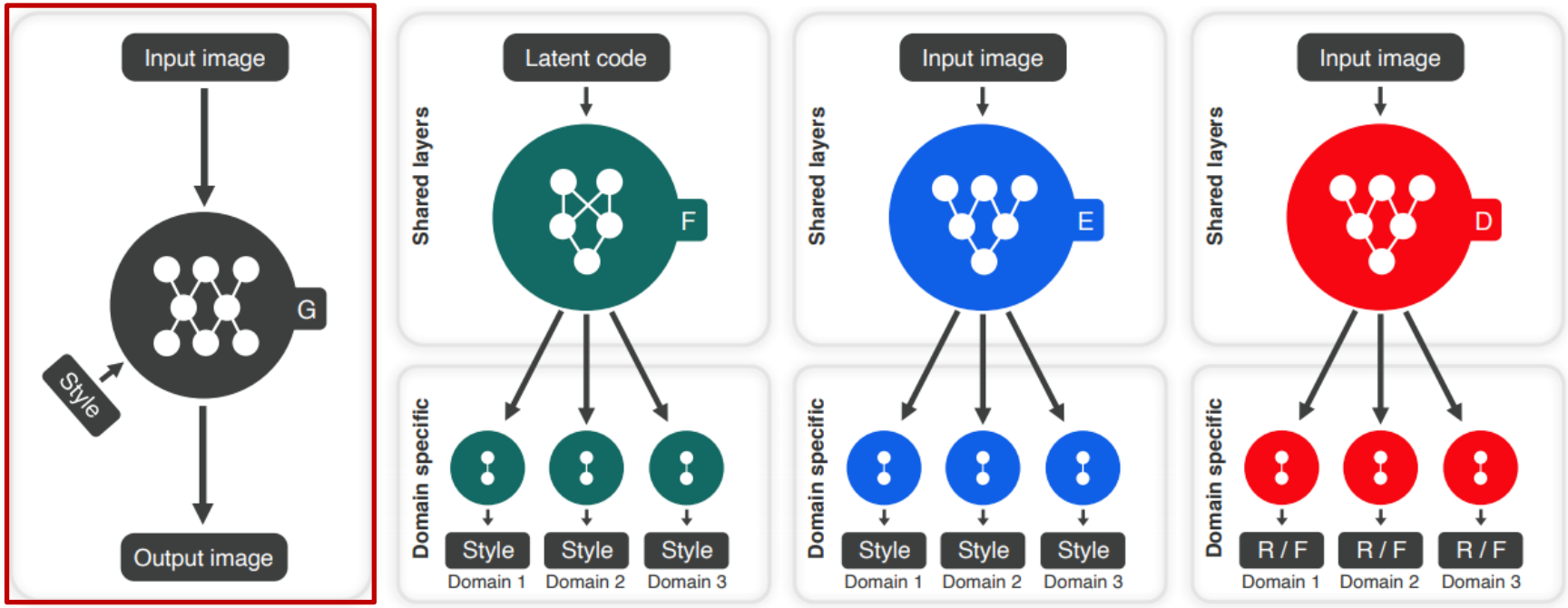


(a) Generator (b) Mapping network (c) Style encoder (d) Discriminator

Choi et al., "StarGAN v2: Diverse Image Synthesis for Multiple Domains", CVPR 2020

StarGAN-v2 의 구성 모듈

Generator는 input 이미지와 style code s를 받아 output 이미지를 생성함.
s는 mapping network 혹은 style encoder에서 생성되며, AdaIN을 통해 Generator에 주입됨.

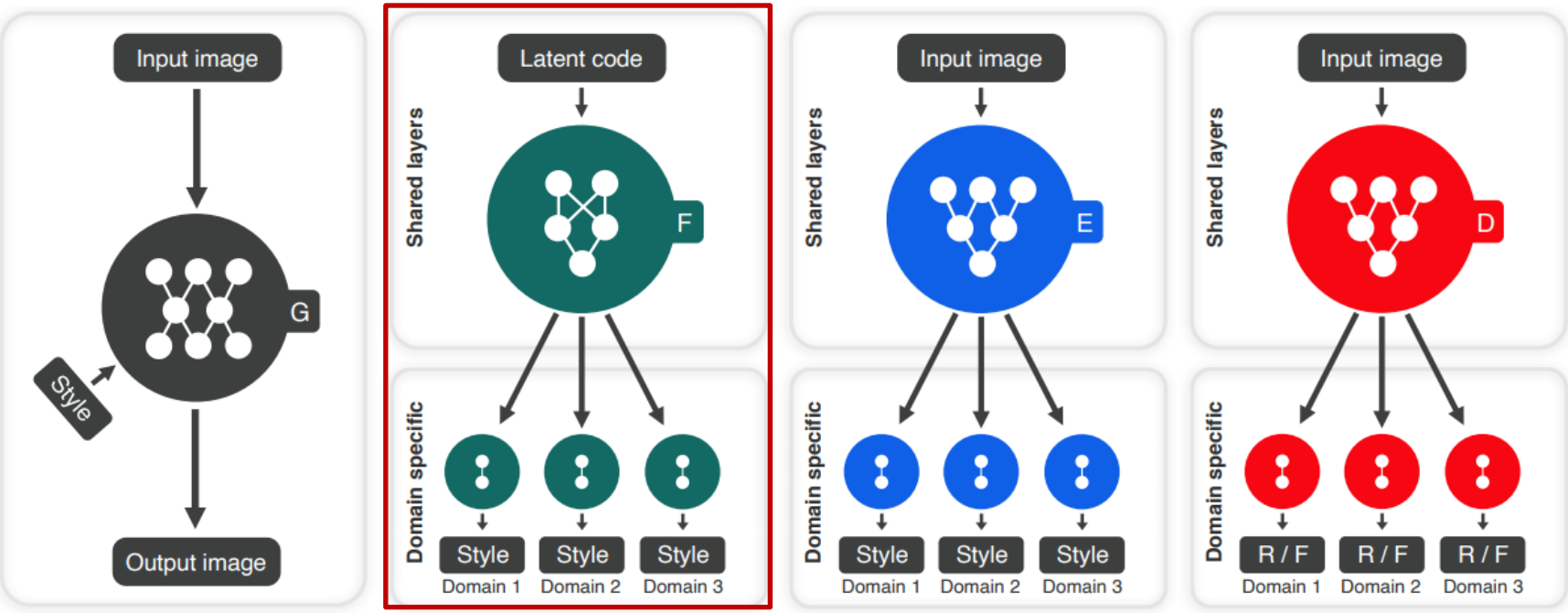


(a) Generator (b) Mapping network (c) Style encoder (d) Discriminator

Choi et al., "StarGAN v2: Diverse Image Synthesis for Multiple Domains", CVPR 2020

StarGAN-v2 의 구성 모듈

Mapping network는 latent code z와 domain label y 만으로 style code를 생성하는 MLP임.
각 domain의 다양한 style code를 생성할 수 있음.

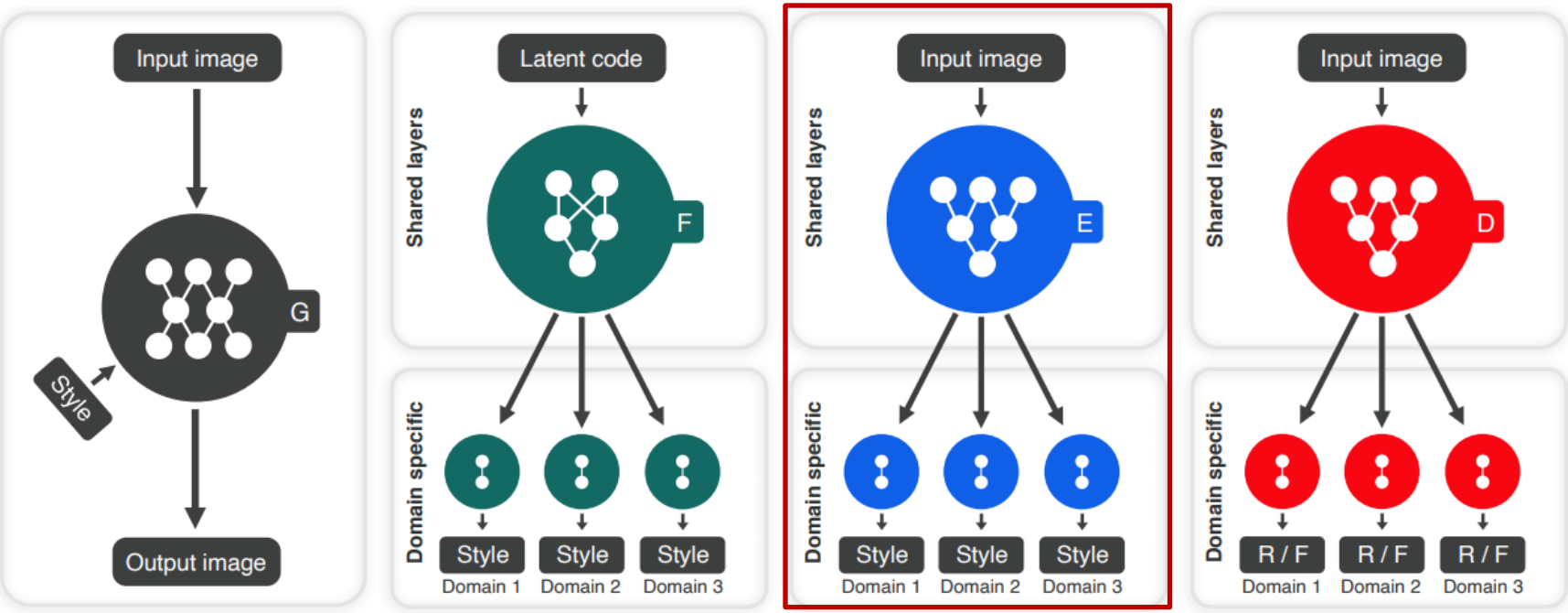


(a) Generator (b) Mapping network (c) Style encoder (d) Discriminator

Choi et al., "StarGAN v2: Diverse Image Synthesis for Multiple Domains", CVPR 2020

StarGAN-v2 의 구성 모듈

Style encoder는 reference 이미지에서 style code s를 추출함.
Style encoder를 통해 다양한 style 이미지를 반영한 이미지를 합성할 수 있음.

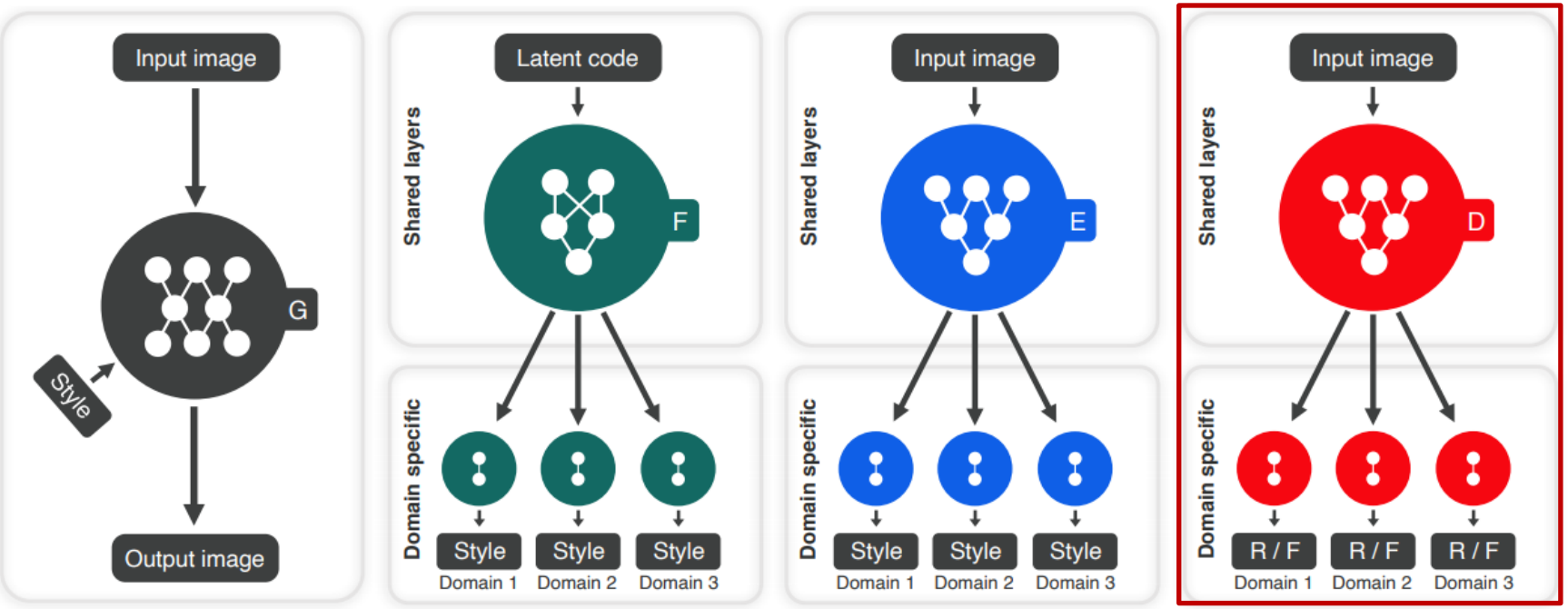


(a) Generator (b) Mapping network (c) Style encoder (d) Discriminator

Choi et al., "StarGAN v2: Diverse Image Synthesis for Multiple Domains", CVPR 2020

StarGAN-v2 의 구성 모듈

Discriminator는 각 domain y에 대해 이미지가 진짜인지 가짜인지 구분하는 역할을 함.



(a) Generator (b) Mapping network (c) Style encoder (d) Discriminator

Choi et al., "StarGAN v2: Diverse Image Synthesis for Multiple Domains", CVPR 2020

StarGAN-v2 의 손실함수

1. Adversarial loss

$$\mathcal{L}_{adv} = \mathbb{E}_{\mathbf{x},y}\left[\log D_y(\mathbf{x})\right] + \mathbb{E}_{\mathbf{x},\tilde{y},\mathbf{z}}\left[\log\left(1 - D_{\tilde{y}}(G(\mathbf{x},\tilde{\mathbf{s}}))\right)\right]$$

2. Style reconstruction loss

$$\mathcal{L}_{sty} = \mathbb{E}_{\mathbf{x},\tilde{y},\mathbf{z}}\left[\|\tilde{\mathbf{s}} - E_{\tilde{y}}(G(\mathbf{x},\tilde{\mathbf{s}}))\|_1\right]$$

3. Diversity sensitive loss

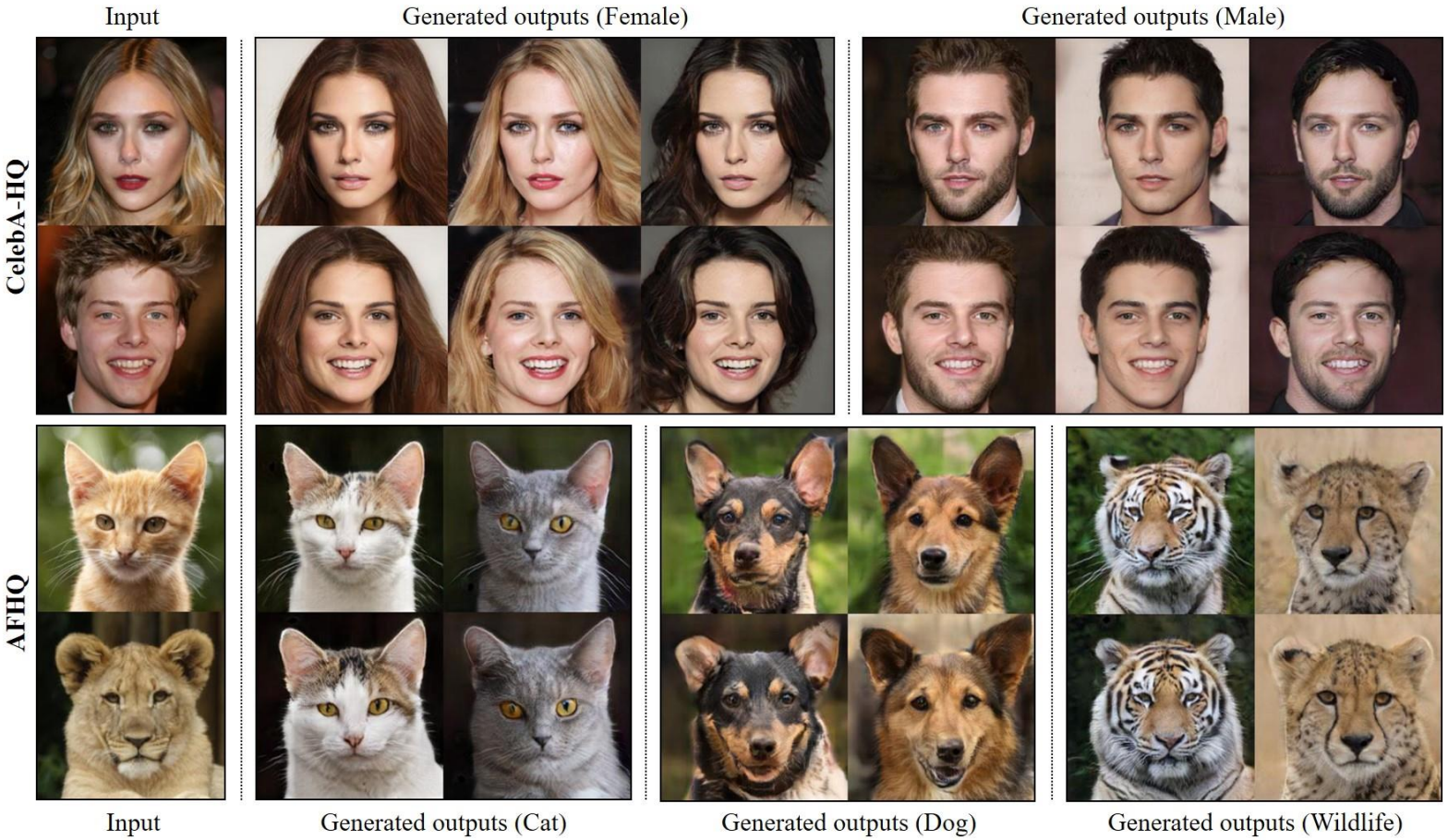
$$\mathcal{L}_{ds} = \mathbb{E}_{\mathbf{x},\tilde{y},\mathbf{z}_1,\mathbf{z}_2}\left[\|G(\mathbf{x},\tilde{\mathbf{s}}_1) - G(\mathbf{x},\tilde{\mathbf{s}}_2)\|_1\right]$$

4. Cycle consistency loss

$$\mathcal{L}_{cyc} = \mathbb{E}_{\mathbf{x},y,\tilde{y},\mathbf{z}}\left[\|\mathbf{x} - G(G(\mathbf{x},\tilde{\mathbf{s}}),\hat{\mathbf{s}})\|_1\right]$$

Choi et al., "StarGAN v2: Diverse Image Synthesis for Multiple Domains", CVPR 2020

StarGAN-v2의 이미지 변환 결과



Choi et al., "StarGAN v2: Diverse Image Synthesis for Multiple Domains", CVPR 2020

04
Advanced Image-to-Image Translation Applications

Ch 4. Advanced Image-to-Image Translation Applications

Clip 1. Neural Talking Head

Neural Talking Head

Neural Talking Head (혹은 Face Reenactment)는
특정 비디오의 얼굴 표정을 따라하는 비디오를 생성해내는 태스크임.

유사하게 전신 동작(예: 춤 동작)을 따라하는 비디오를 생성해내는 태스크는 Motion Transfer라고 부름.

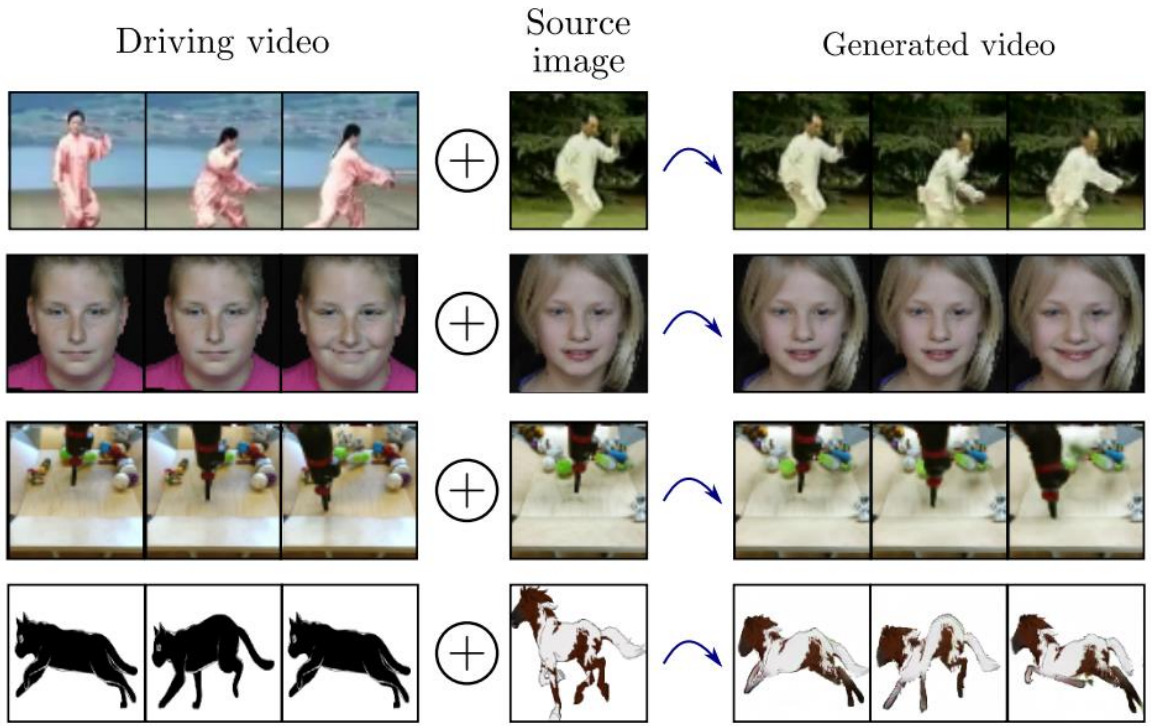
Driving Video

Source Image

Aliaksandr et al., "First Order Motion Model for Image Animation", NeurIPS 2019

MonkeyNet

Motion Transfer를 수행할 수 있는 대표적인 모델로 MonkeyNet이 있음.
모델은 Driving Video의 움직임을 따라하도록 Source Image를 변환하는 것을 목표로 함.



Aliaksandr et al., "Animating Arbitrary Objects via Deep Motion Transfer", CVPR 2019

Optical Flow

Optical Flow는 비디오 내 물체의 움직임 패턴을 의미함.
Optical Flow는 2차원의 Vector Field로 표현됨.
일부 pixel들의 움직임만을 계산한 Optical Flow를 Sparse Optical Flow,
모든 pixel들의 움직임을 계산한 Optical Flow를 Dense Optical Flow라고 부름.

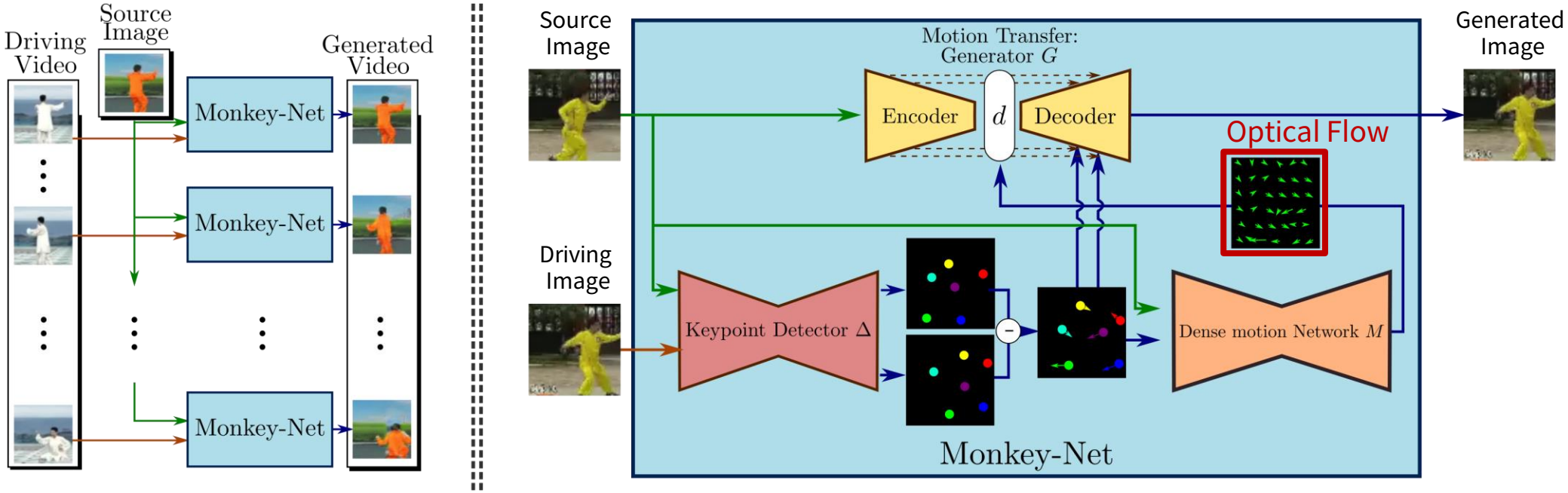
Sparse Optical Flow

Dense Optical Flow

https://nanonets.com/blog/optical-flow/

MonkeyNet의 Overview

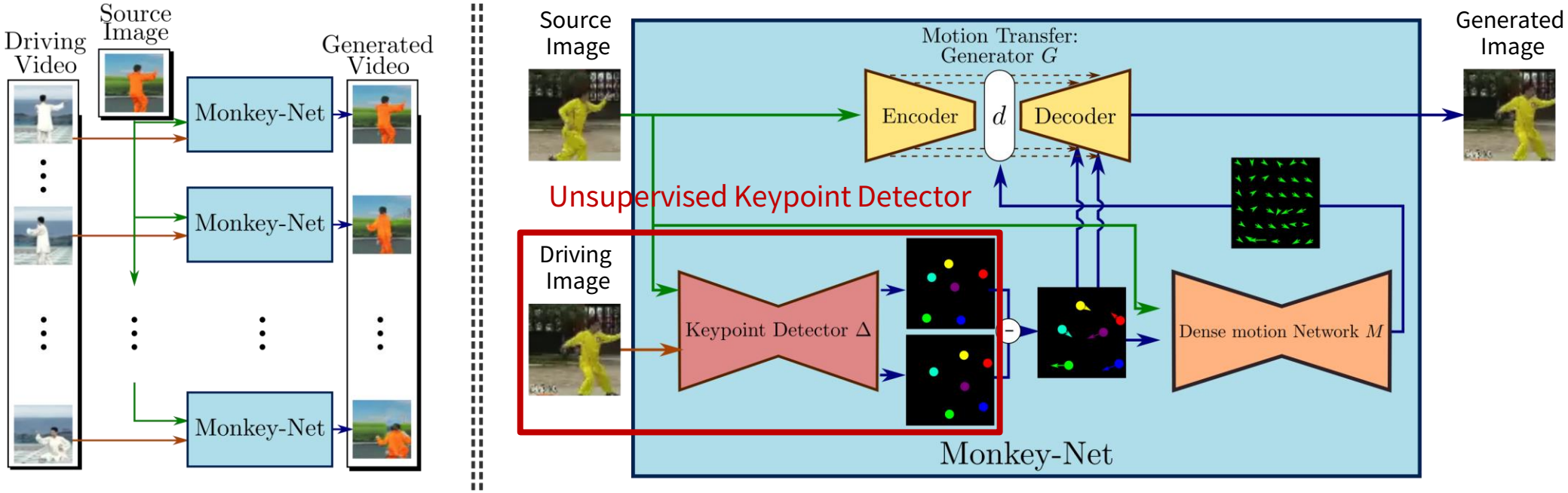
MonkeyNet에서는 Source Image를 변환할 때, Optical Flow를 사용함.



Aliaksandr et al., "Animating Arbitrary Objects via Deep Motion Transfer", CVPR 2019

Unsupervised Keypoint Detector

MonkeyNet에서는 Optical Flow를 계산하기 위해 이미지에서 Keypoint를 추출하는 과정을 거침.
이 때, Unsupervised 방식으로 Keypoint를 추출할 수 있는 방법을 활용함.

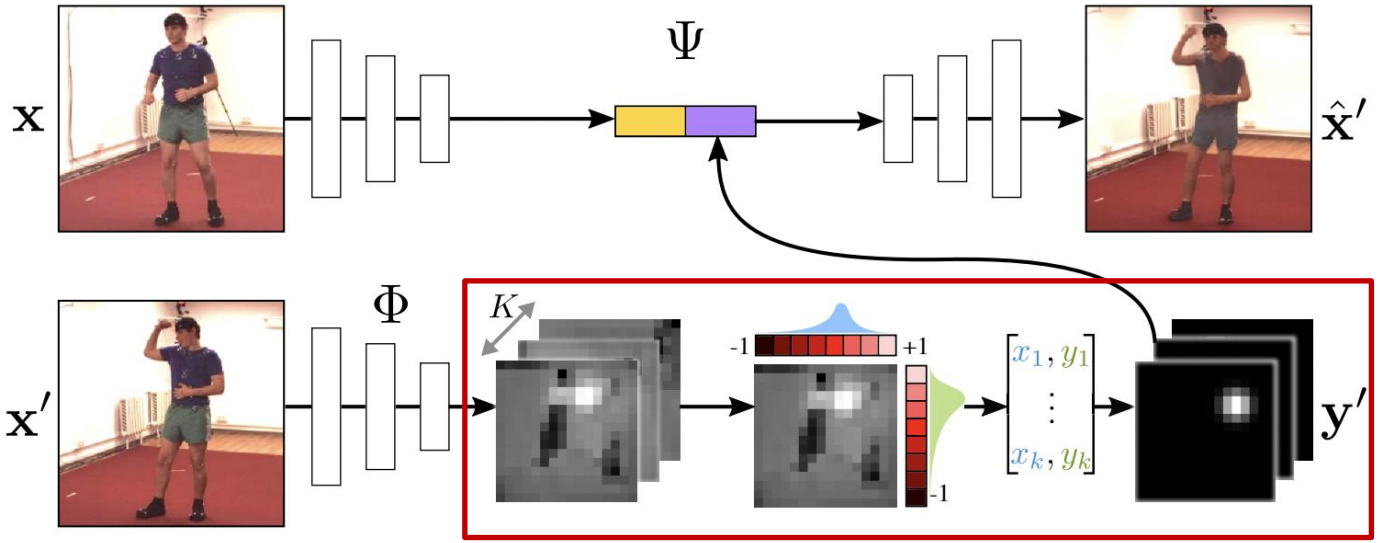


Aliaksandr et al., "Animating Arbitrary Objects via Deep Motion Transfer", CVPR 2019

Unsupervised Keypoint Detector

Keypoint를 추출하기 위해 중간 과정에서 Heatmap을 예측한 후에,
x축과 y축에 대해 각각 (x_k, y_k) 를 계산함.

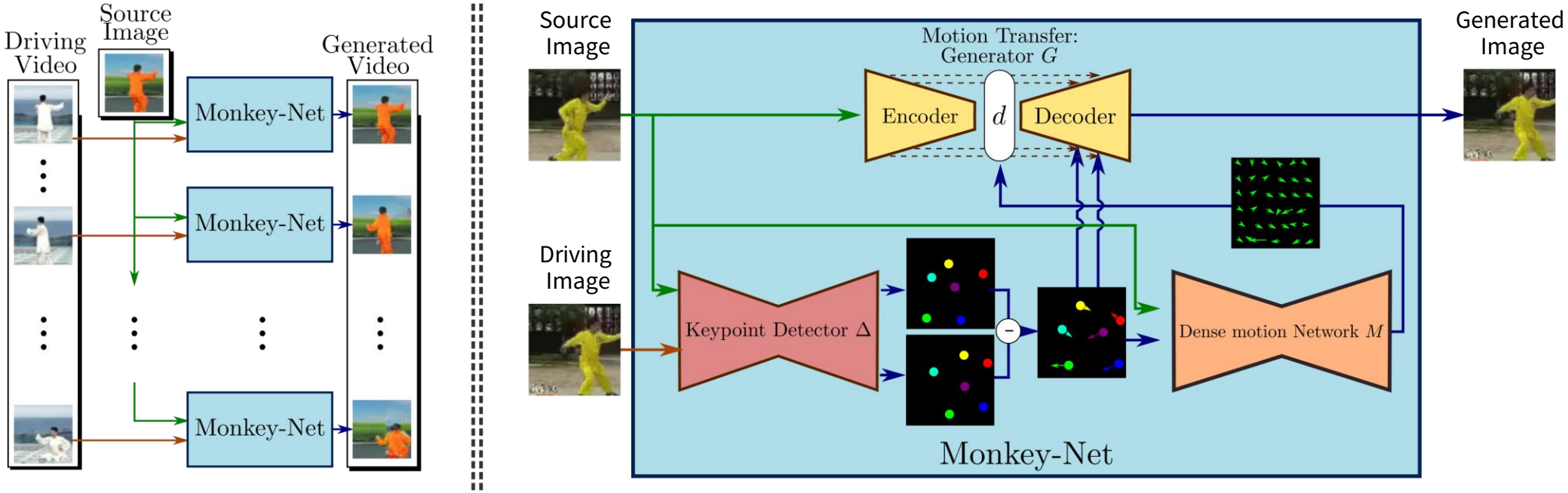
(1) 각 축에 대해 Softmax를 계산한 후에, 위치에 따라 -1~1 값을 곱한 후에 더해서 좌표 (x_k, y_k) 를 계산함.
(2) 계산한 좌표 (x_k, y_k) 으로 Gaussian Heatmap을 생성함.



Jakab et al., "Unsupervised Learning of Object Landmarks through Conditional Image Generation", NeurIPS 2018

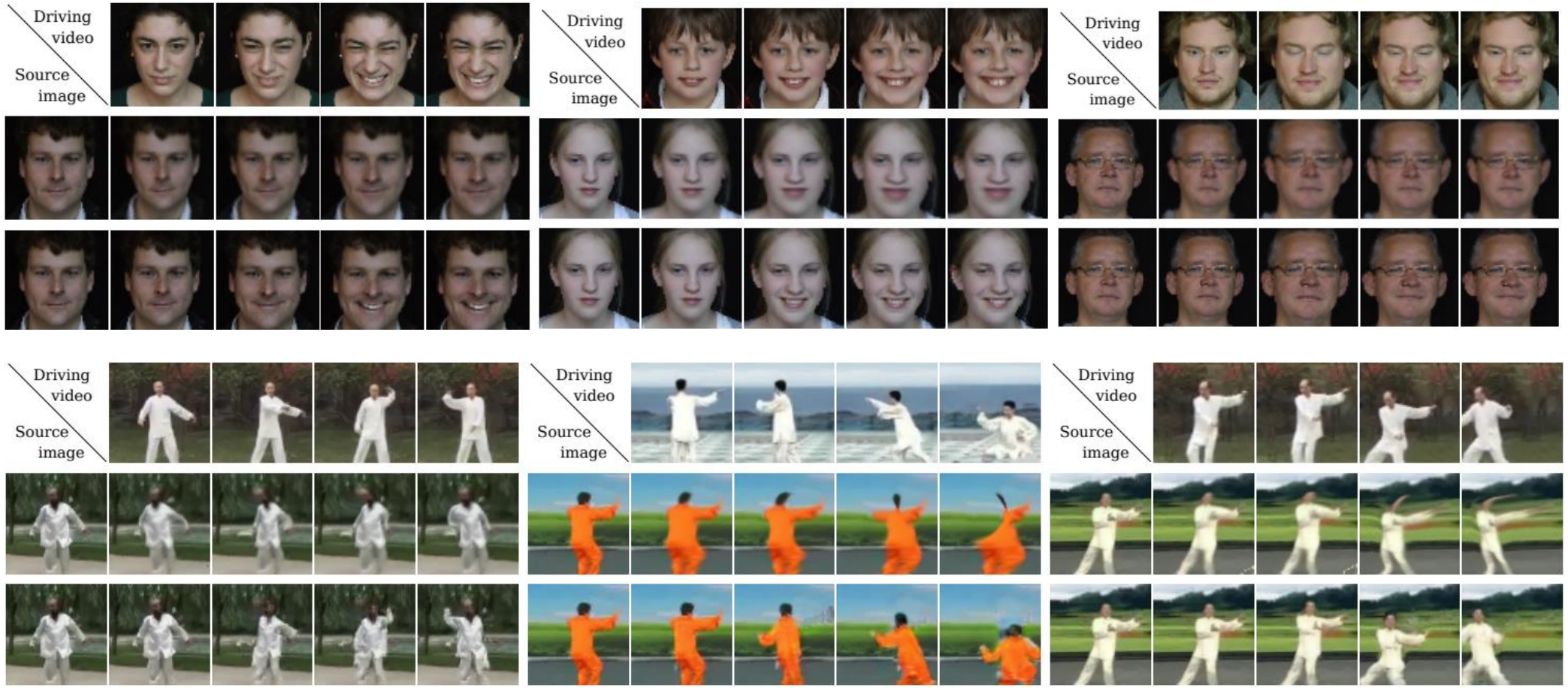
MonkeyNet의 Loss Function

MonkeyNet은 최종적으로 다음과 같은 모델 구조를 가짐.
모델을 학습시키기 위해 Adversarial Loss와 Feature-Matching Loss를 사용함.



Aliaksandr et al., "Animating Arbitrary Objects via Deep Motion Transfer", CVPR 2019

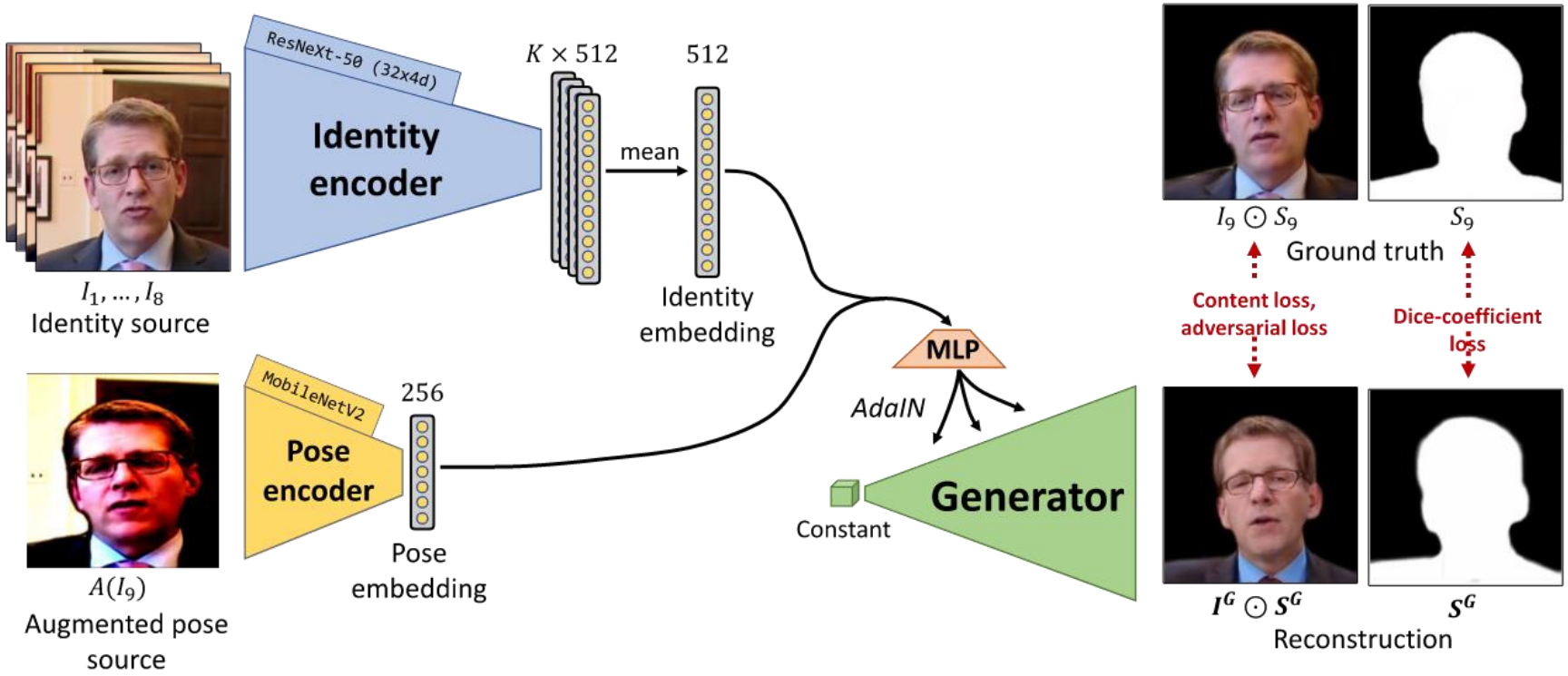
MonkeyNet의 결과



Aliaksandr et al., "Animating Arbitrary Objects via Deep Motion Transfer", CVPR 2019

Latent Pose Descriptor의 Overview

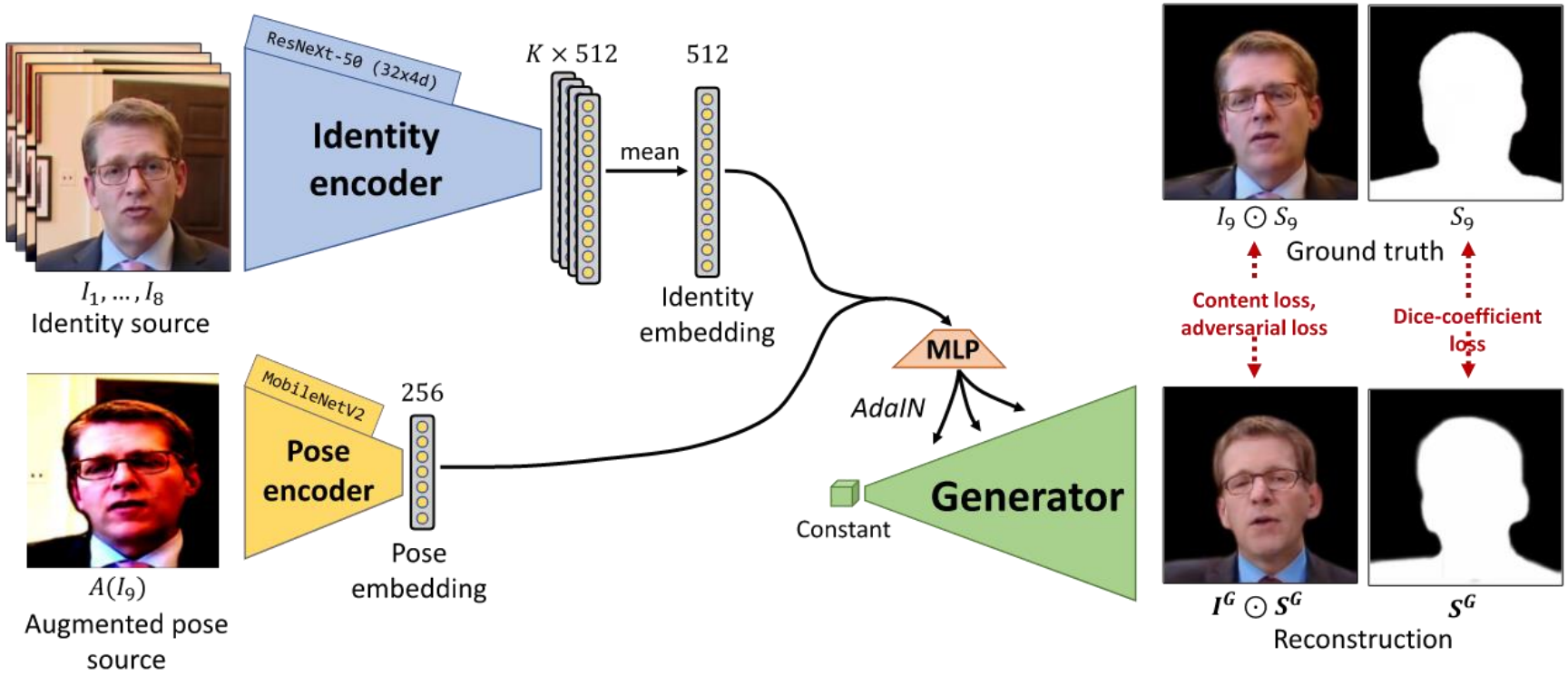
Latent Pose Descriptor를 이용하여 Neural Talking Head를 수행하는 모델임.



Egor et al., "Neural Head Reenactment with Latent Pose Descriptors", CVPR 2020

Latent Pose Descriptor의 Overview

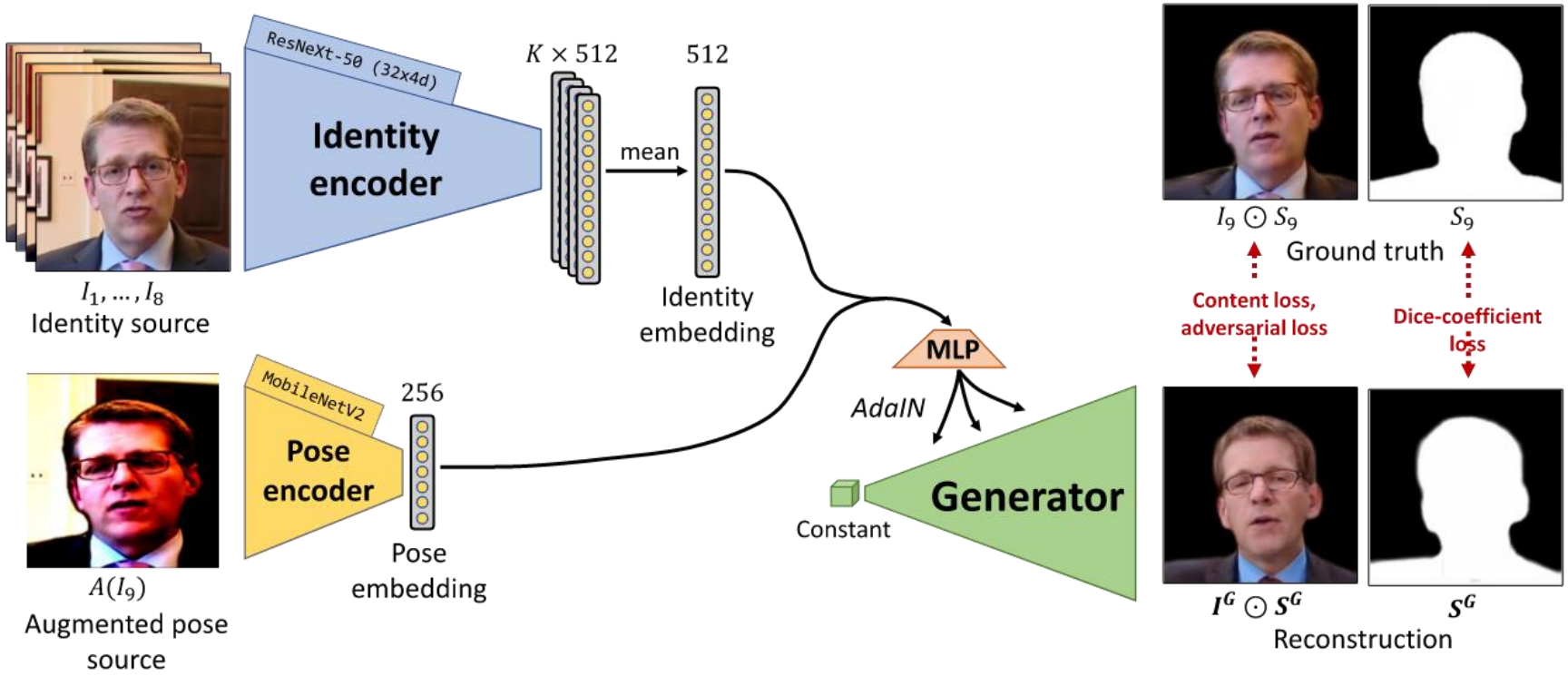
Foreground Segmentation을 예측하며,
명시적인 Keypoint 추출과 같은 추가적인 장치 없이
Pose와 Identity의 Disentanglement가 가능함을 보여줌.



Egor et al., "Neural Head Reenactment with Latent Pose Descriptors", CVPR 2020

Latent Pose Descriptor의 Loss Function

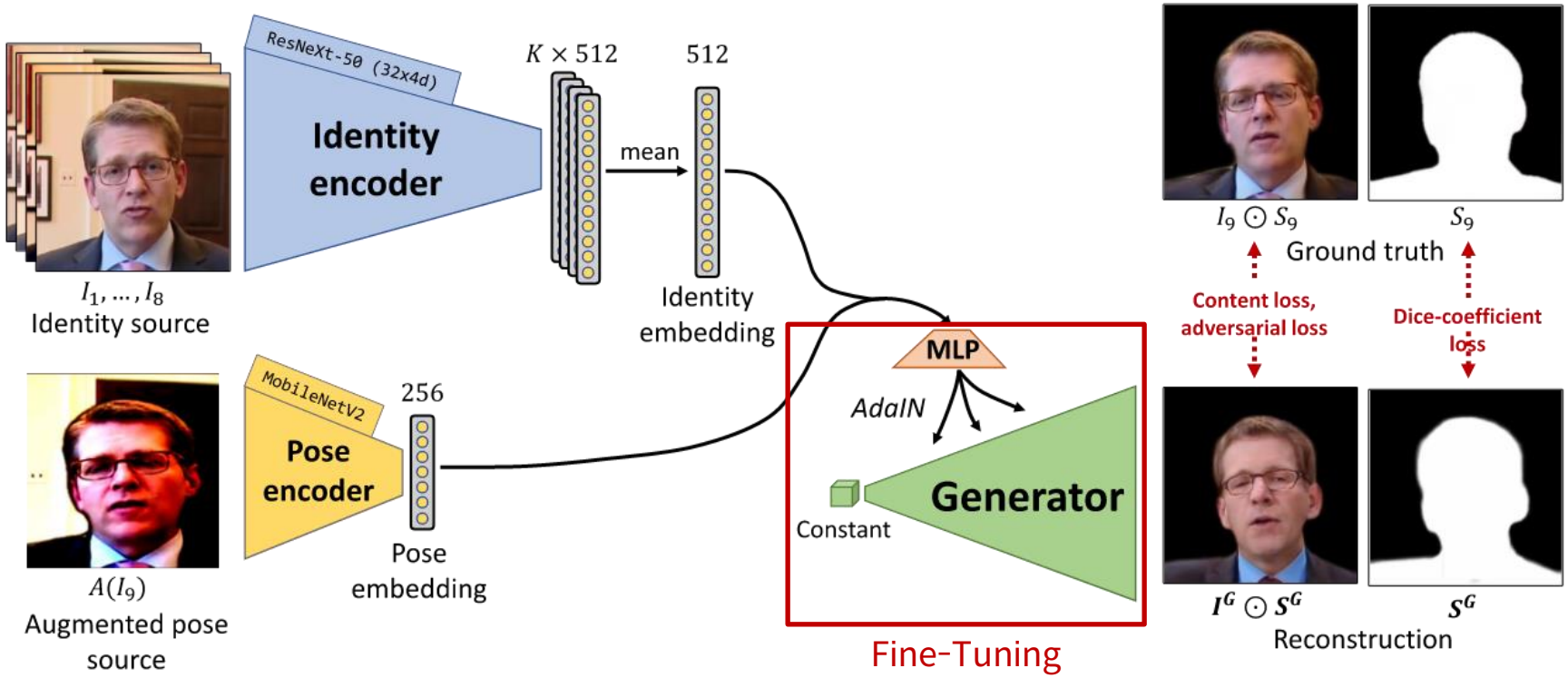
모델을 학습시키기 위해 Adversarial Loss와 Content Loss (VGG19을 활용한 Perceptual Loss)
그리고 Foreground Segmentation Map에 Dice-coefficient Loss를 사용함.



Egor et al., "Neural Head Reenactment with Latent Pose Descriptors", CVPR 2020

Latent Pose Descriptor의 Fine-Tuning 방법

Test phase에서 Identity Images를 이용하여,
MLP, Generator 그리고 Discriminator 부분을 Fine-Tuning함으로써 Identity Gap을 줄일 수 있음.



Egor et al., "Neural Head Reenactment with Latent Pose Descriptors", CVPR 2020

Latent Pose Descriptor의 결과

이러한 Fine-Tuning을 통해 사람 뿐만 아니라 모나리자와 같은 이미지에도 적용 가능함.

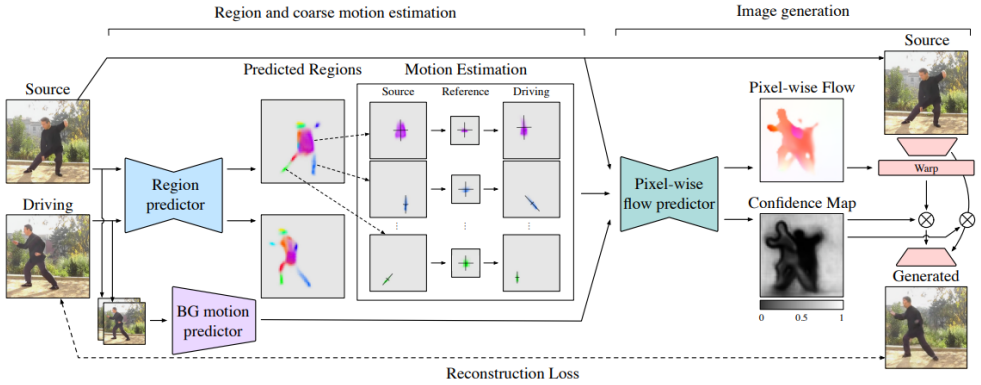
Egor et al., "Neural Head Reenactment with Latent Pose Descriptors", CVPR 2020

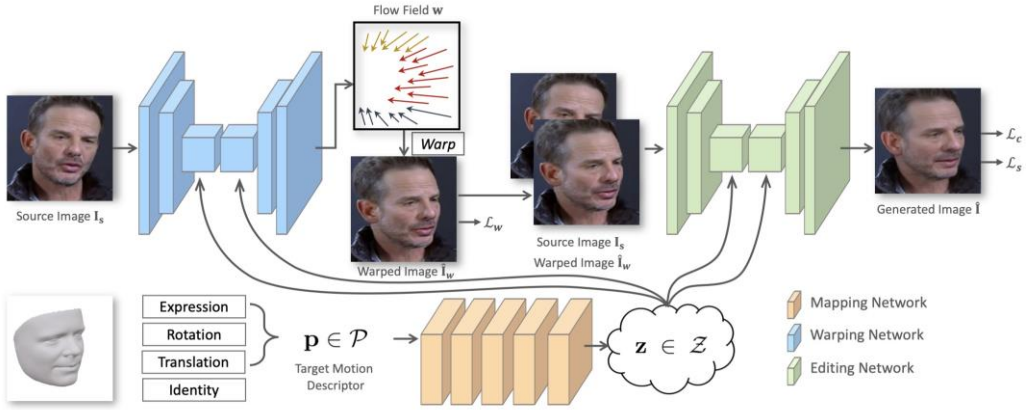
Research Trends

Neural Talking Head 모델에서는 Identity에 관련된 정보(예:생김새, 눈의 크기 등)를 제외한
Pose를 어떤 방법으로 encoding하는지가 중요함.

최근에는 MonkeyNet과 같이 Keypoint를 사용하거나,
Latent Pose Descriptor과 같이 Latent space를 사용하는 방법 이외에도 다양한 연구가 진행되고 있음.

최근에는 Region 혹은 3D Morphable Model 등을 활용하는 방법이 제안됨.





Aliaksandr et al., "Motion Representations for Articulated Animation", CVPR 2021
Yurui et al., "PIRenderer: Controllable Portrait Image Generation via Semantic Neural Rendering", ICCV 2021

Ch 4. Advanced Image-to-Image Translation Applications

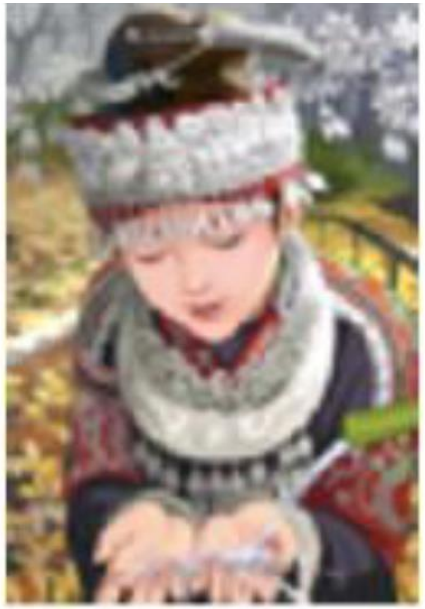
Clip 2. Image Super-Resolution

Image Super-Resolution

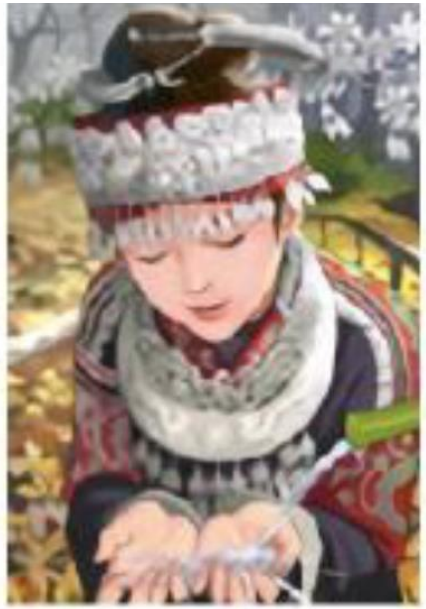
Image Super-Resolution은 고해상도를 복원하는 태스크임.

original

Bicubic
(21.59dB/0.6423)



SRRcsNet
(23.44dB/0.7777)

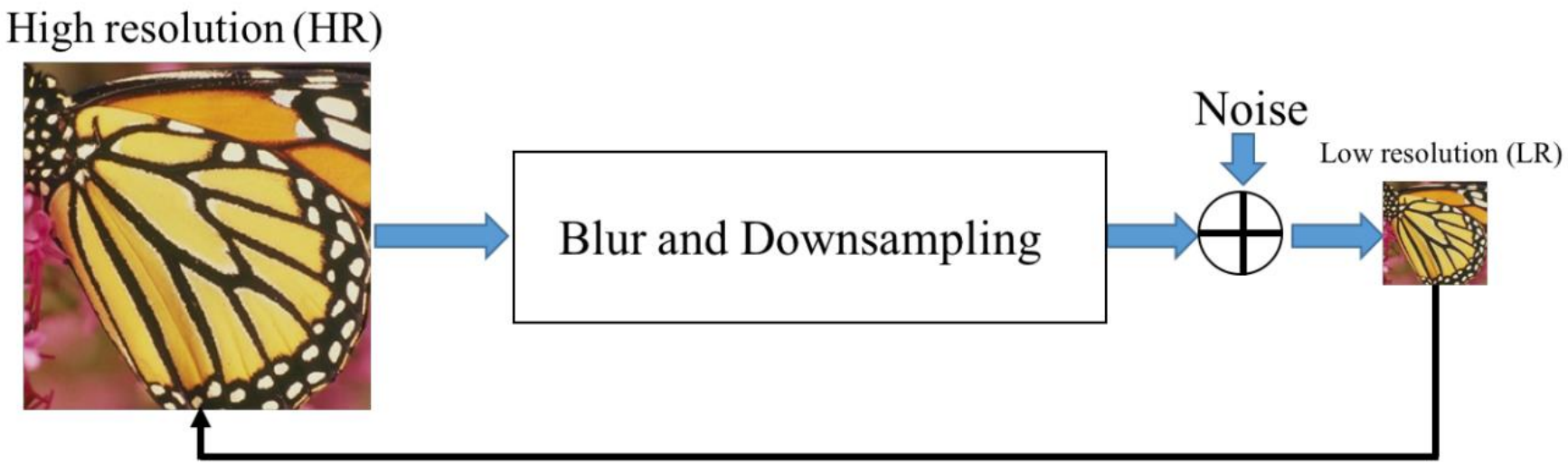


SRGAN
(20.34dB/0.6562)

Ledig et al, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", CVPR 2017

Image Super-Resolution

일반적으로 Image Super-Resolution 모델을 학습할 때,
고화질의 이미지를 저화질의 이미지로 해상도를 낮추고,
저화질의 이미지를 Input Image로 / 고화질의 이미지를 Ground Truth Image로 사용함.

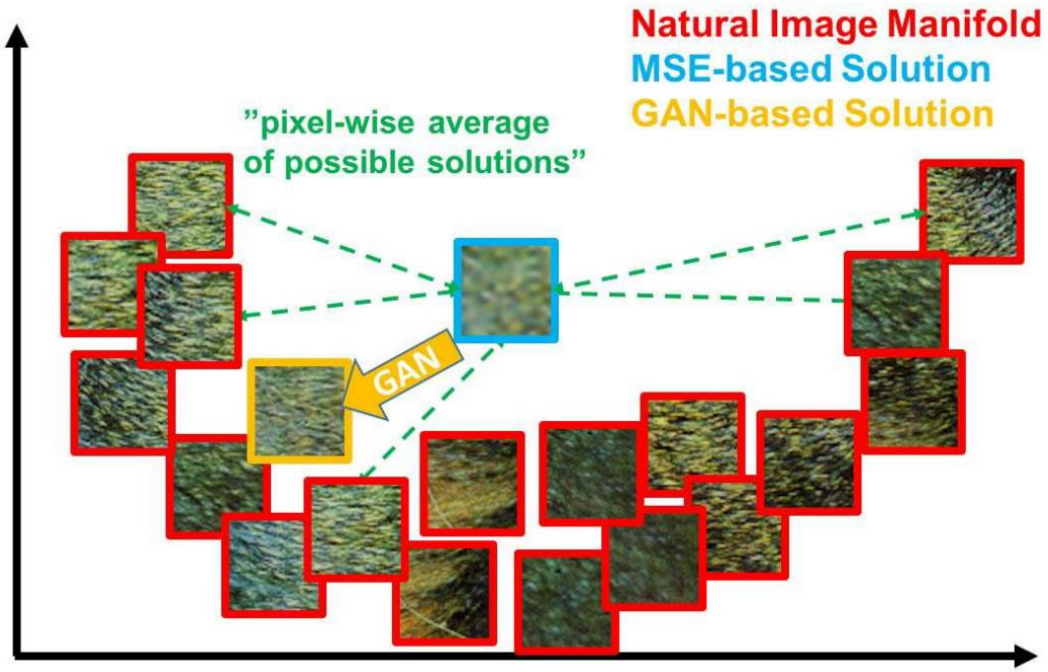


SISR: Try to recover HR from its LR counterpart

Ledig et al, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", CVPR 2017

MSE Loss vs. GAN Loss

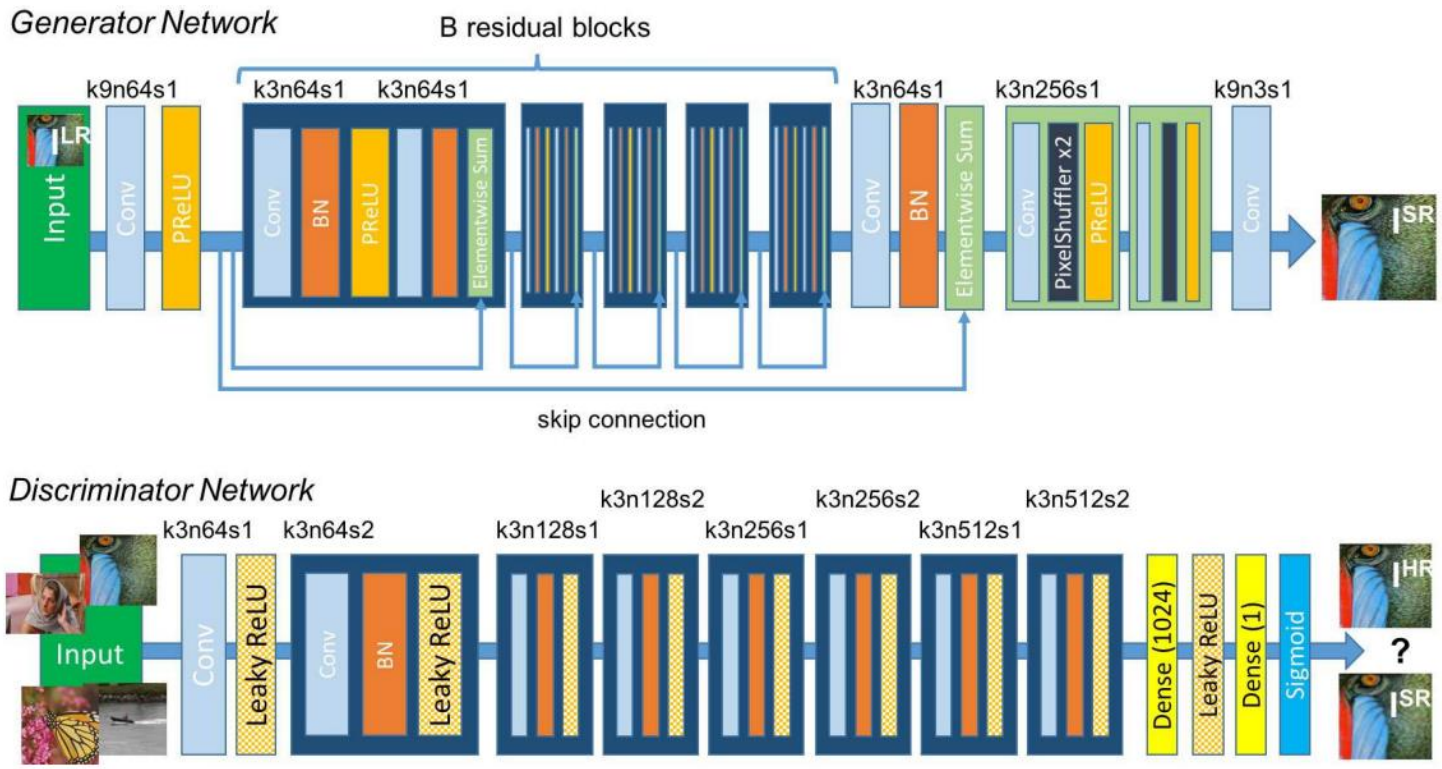
기존의 MSE Loss만 사용하는 Super-Resolution 모델들은 최종 결과가 blurry하다는 단점이 있음.
이를 해결하고자 GAN을 응용한 SRGAN이 제안됨.



Ledig et al, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", CVPR 2017

SRGAN의 Overview

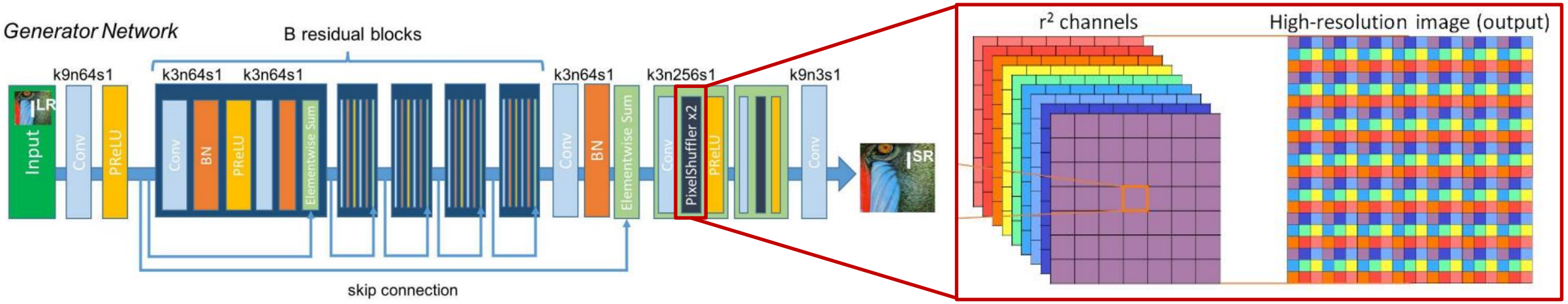
SRGAN은 Generator와 Discriminator로 구성되어 있으며,
모델 구조는 다음과 같음.



Ledig et al, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", CVPR 2017

PixelShuffle

SRGAN에서는 Resolution을 늘리기 위해 Generator 후반부에서
PixelShuffle을 사용함.

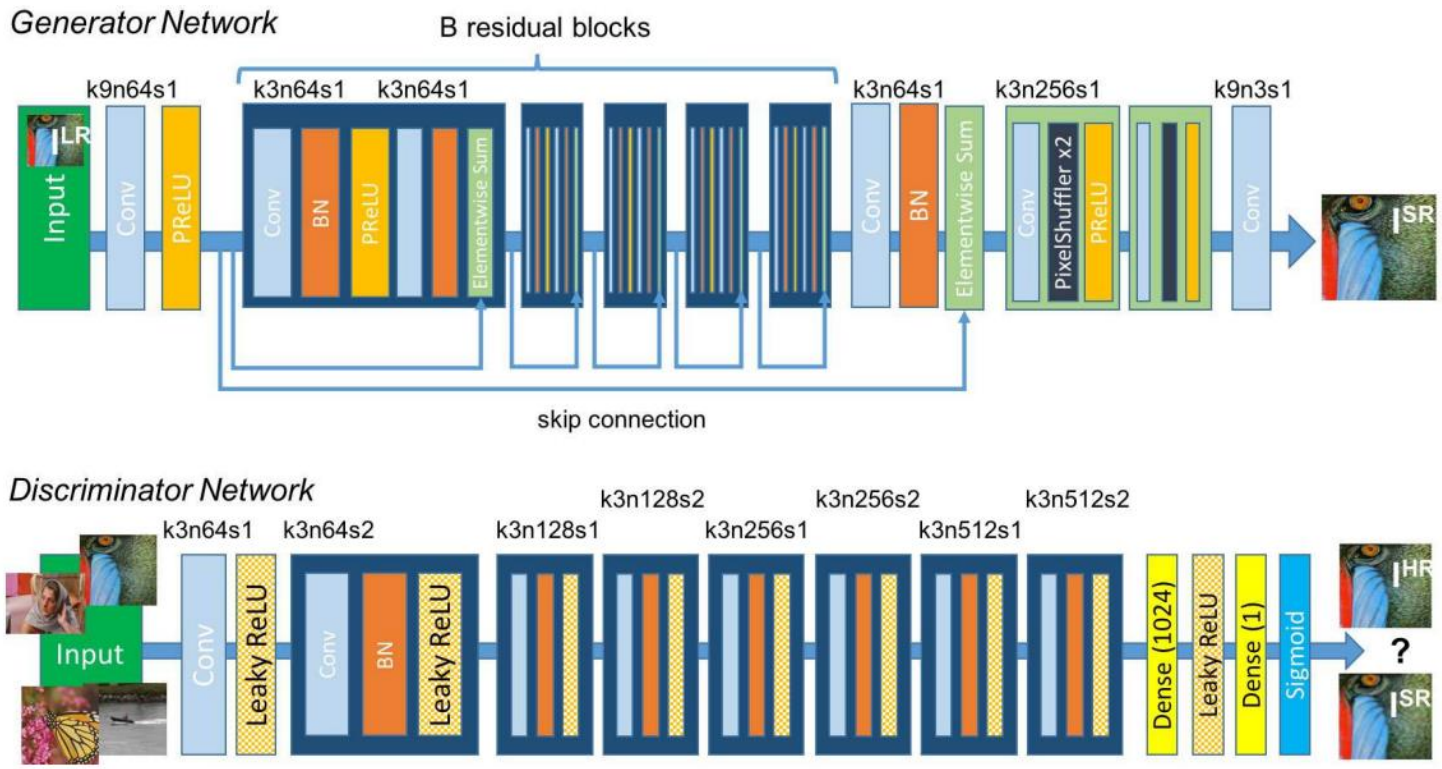


Pixel Shuffle

Ledig et al, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", CVPR 2017

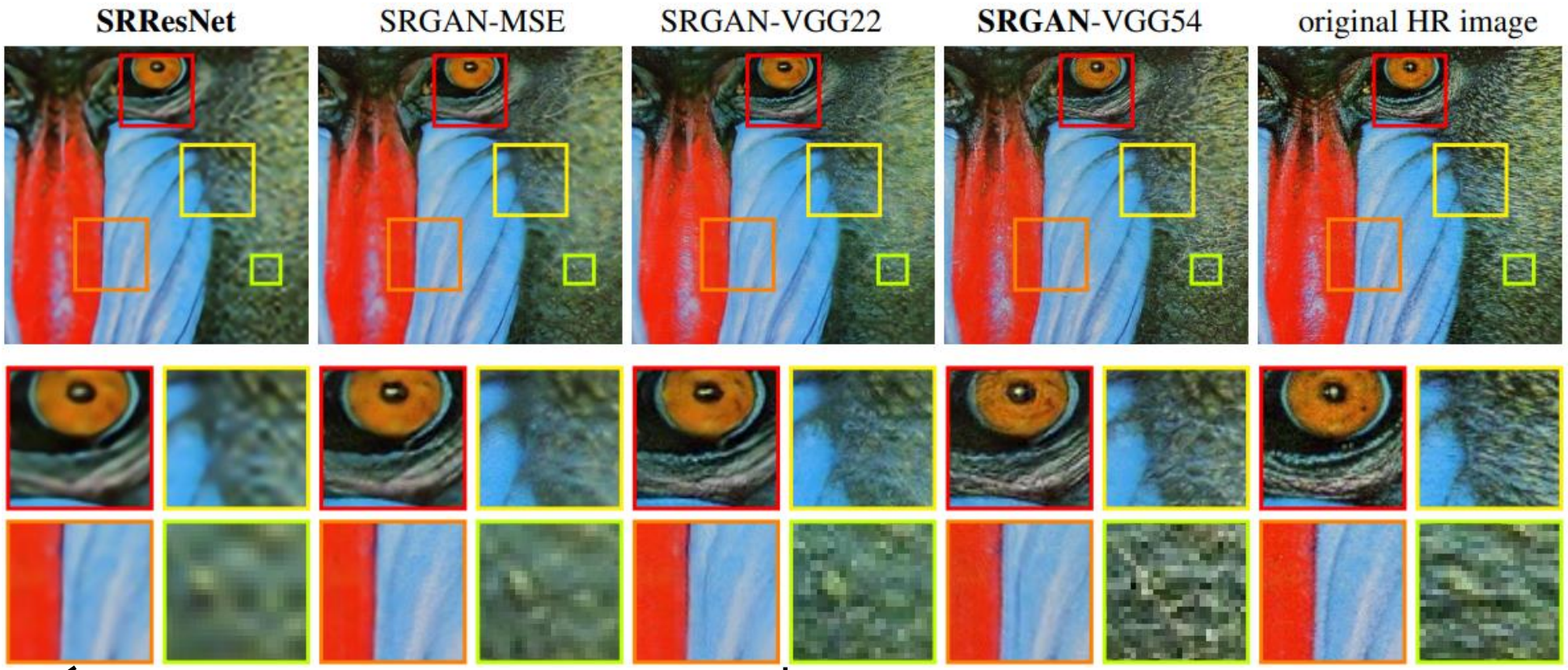
SRGAN의 Loss Function

SRGAN을 학습시키기 위해 Adversarial Loss, MSE Loss 그리고 Perceptual Loss를 사용함.



Ledig et al, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", CVPR 2017

SRGAN의 결과



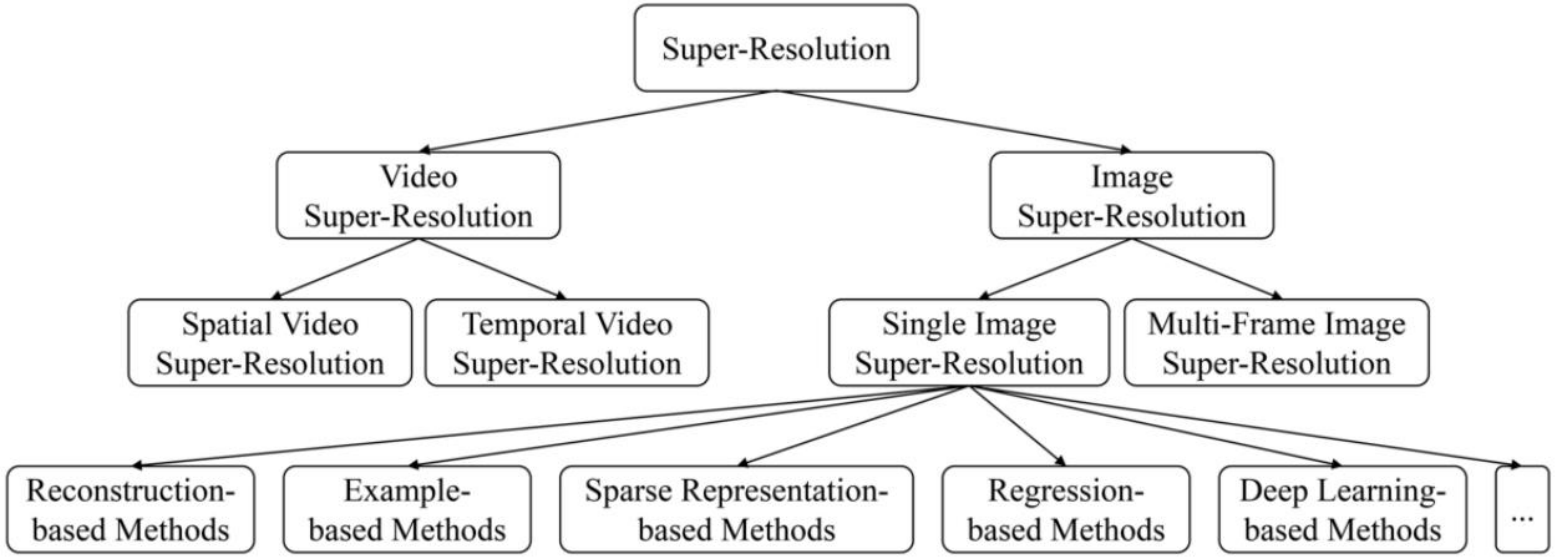
MSE Loss만 사용하면, Blur함.

Perceptual Loss 없이 학습하는 것보다는
Deeper Network로 Perceptual Loss를 사용하는 것이 성능에 도움이 됨.

Ledig et al, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", CVPR 2017

Research Trends

강의에서는 Single Image Super-Resolution 방법 중 GAN을 활용하는 SRGAN을 다룸.
하지만 Super-Resolution에는 비디오의 화질을 높이는 Video Super-Resolution 등이 있으며,
Image Super-Resolution에서도 다양한 해결 방법들이 존재함.



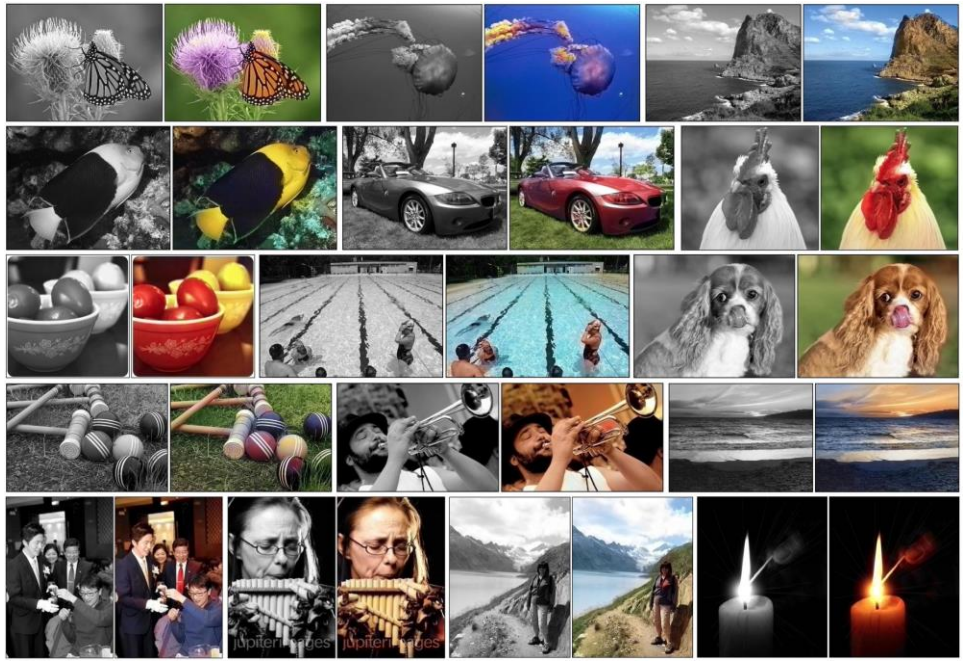
Chen et al., "Real-World Single Image Super-Resolution: A Brief Review", arXiv 2021

Ch 4. Advanced Image-to-Image Translation Applications

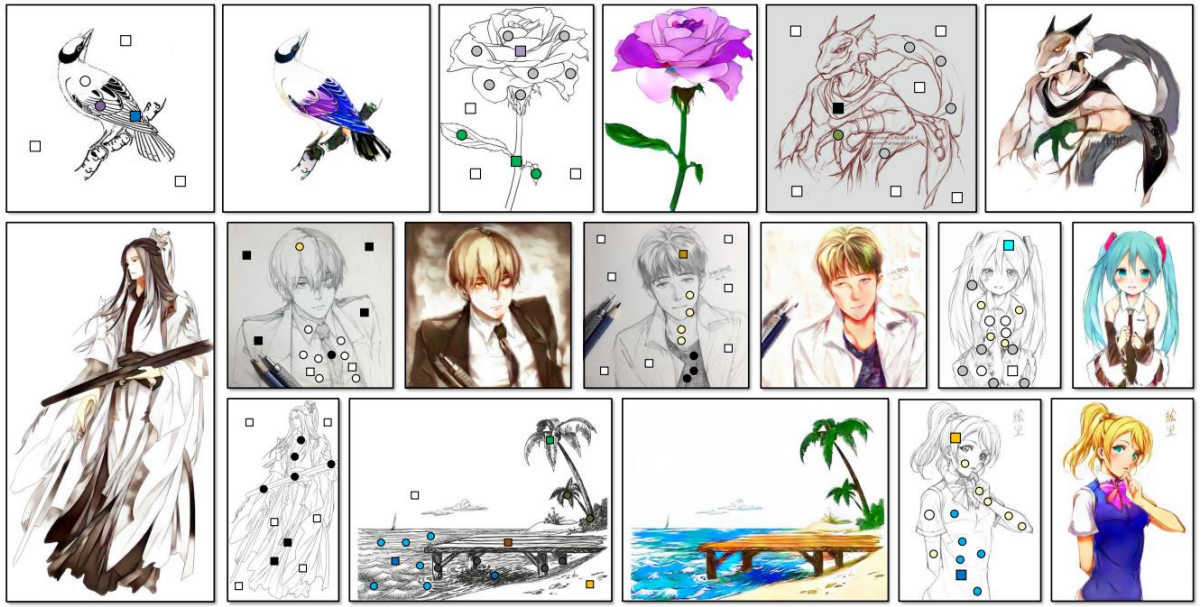
Clip 3. Automatic Colorization

Automatic Colorization

Colorization에는 크게 2가지 태스크가 있음.
Grayscale 이미지 / Sketch 이미지를 색칠하는 방법이 있음.



Grayscale Colorization

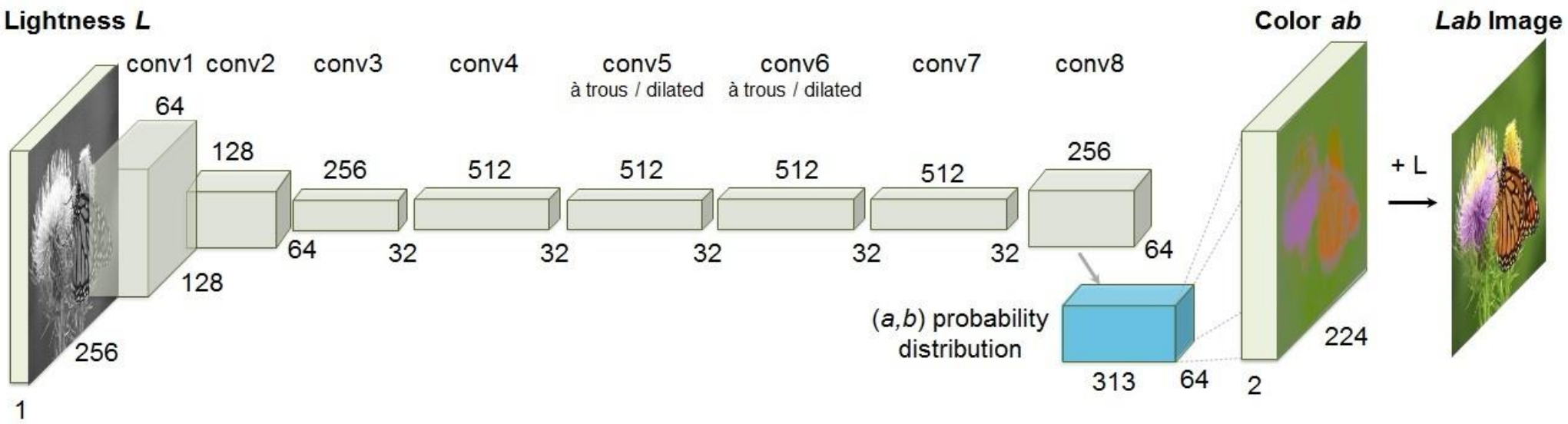


Sketch Colorization

Zhang et al., "Colorful Image Colorization", ECCV 2016
Zhang et al., "Two-stage Sketch Colorization", SIGGRAPH Asia 2018

Colorful Image Colorization의 Overview

Colorful Image Colorization 논문에서는 처음으로 Grayscale 이미지의 색을
Convolutional Neural Network (CNN)으로 채우는 방법을 제안함.

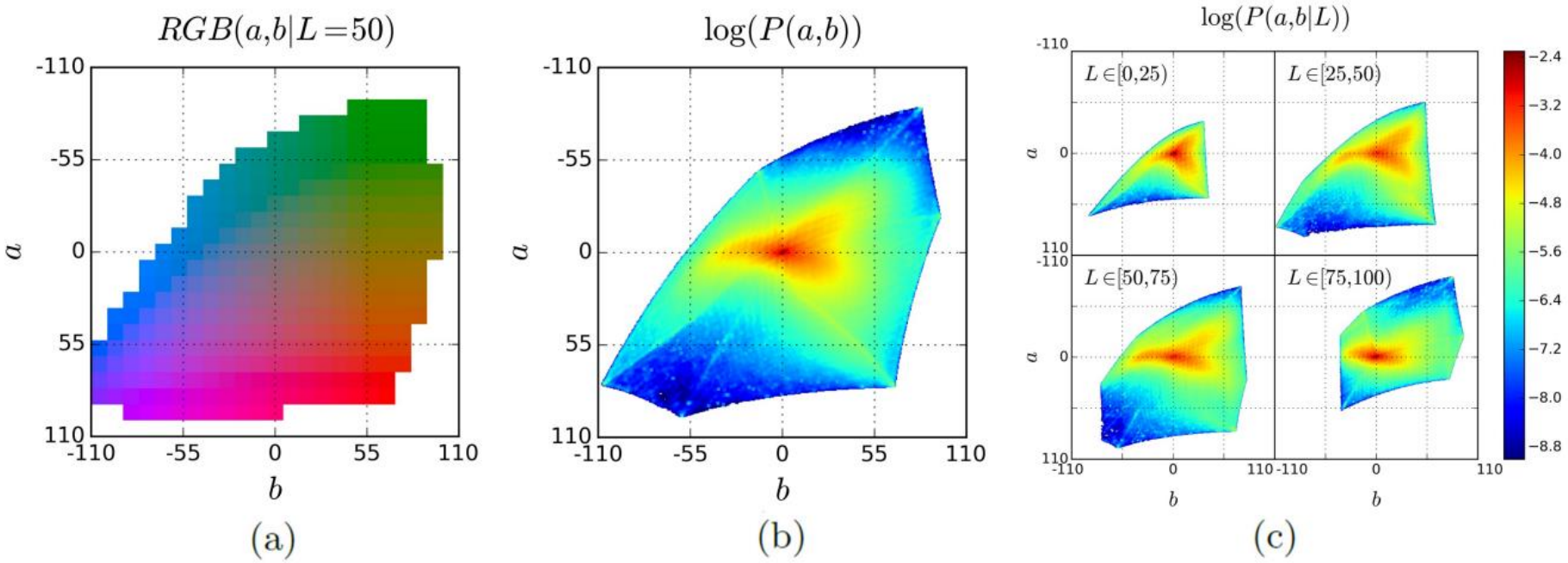


Zhang et al., "Colorful Image Colorization", ECCV 2016

Lab Channel

Color를 표현하는 방법에는 RGB, CIE Lab 등이 존재하는데,
여기서는 Lightness channel인 L과 a와 b의 color channel로 색을 표현하는 CIE Lab color space를 사용함.

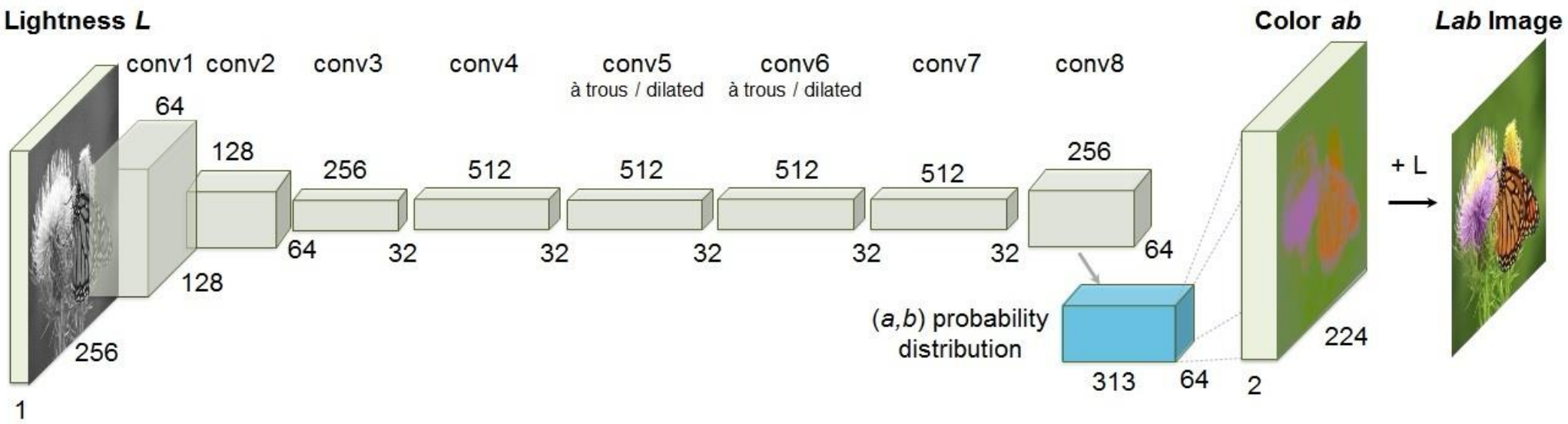
아래 그림에서는 Lightness channel L에 따라 표현할 수 있는 color space를 시각화한 결과임



Zhang et al., "Colorful Image Colorization", ECCV 2016

Colorful Image Colorization의 Overview

Grayscale 이미지를 Lightness channel로 사용하고,
CNN으로 color space ab 를 예측함.
최종적으로 Input으로 사용했던 L channel과 ab channel을 합쳐서
최종적인 Output 이미지를 생성함.



Zhang et al., "Colorful Image Colorization", ECCV 2016

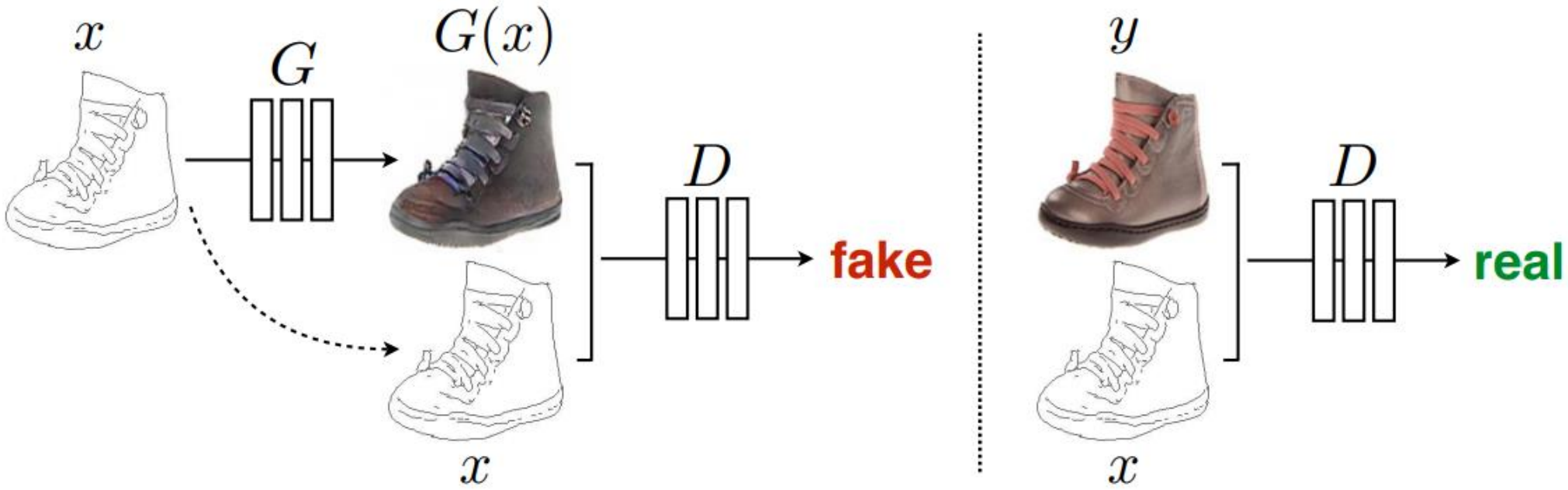
Pix2Pix

Pix2Pix 모델은 Sketch 이미지를 색칠할 수 있는 대표적인 모델임.

Isola et al, "Image-to-Image Translation with Conditional Adversarial Nets", CVPR 2017

Pix2Pix

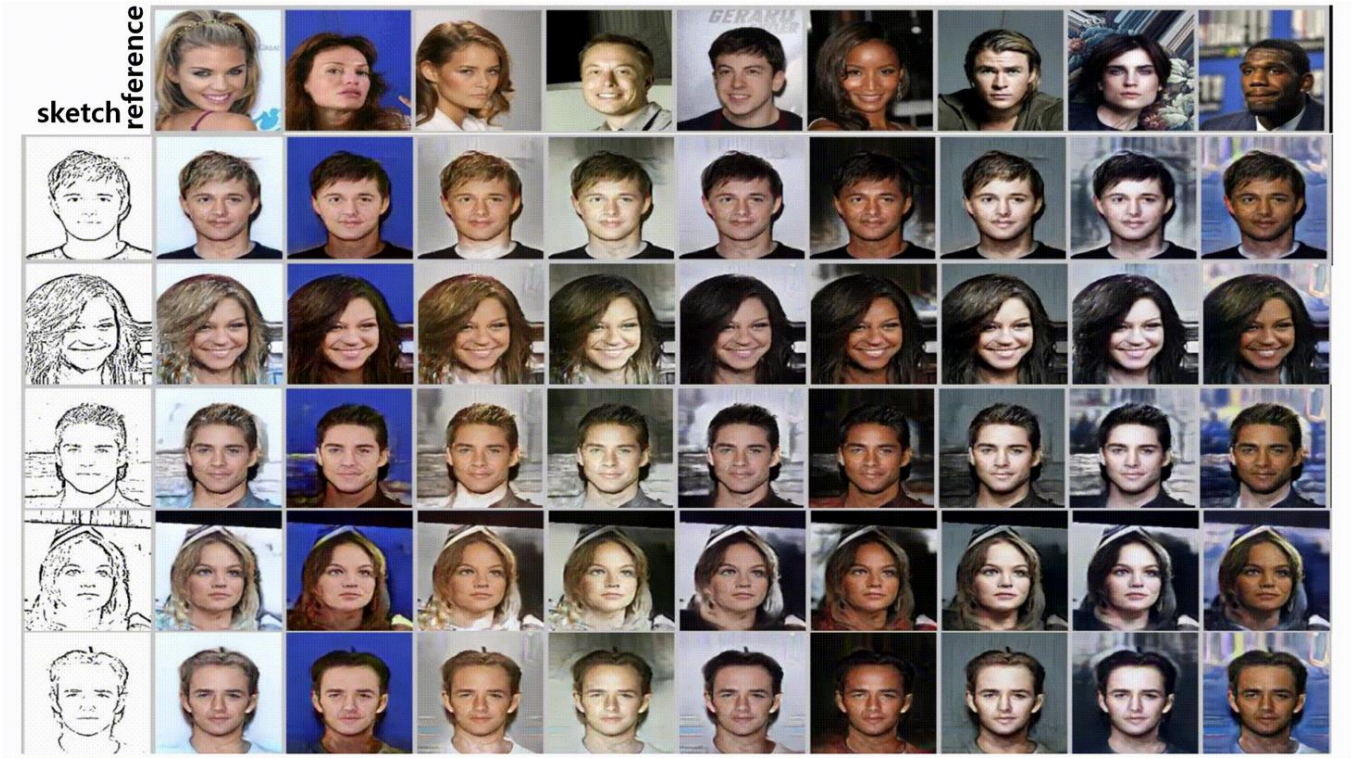
Pix2Pix 모델은 Sketch 이미지를 색칠할 수 있는 대표적인 모델임.
아래와 같이 Discriminator에 Sketch를 Condition으로 같이 Input으로 넣음.



Isola et al, "Image-to-Image Translation with Conditional Adversarial Nets", CVPR 2017

Reference-based Colorization

Reference 이미지를 사용하여, Sketch 이미지의 색을 입히는 태스크를
Reference-based Colorization이라고 정의함.



Lee et al, "Reference-Based Sketch Image Colorization using Augmented-Self Reference and Dense Semantic Correspondence", CVPR 2020

Reference-based Colorization

Reference 이미지를 사용하여, Sketch 이미지의 색을 입히는 태스크를
Reference-based Colorization이라고 정의함.

Lee et al, "Reference-Based Sketch Image Colorization using Augmented-Self Reference and Dense Semantic Correspondence", CVPR 2020

Reference-based Colorization

Reference 이미지를 사용하여, Sketch 이미지의 색을 입히는 태스크를
Reference-based Colorization이라고 정의함.

Lee et al, "Reference-Based Sketch Image Colorization using Augmented-Self Reference and Dense Semantic Correspondence", CVPR 2020

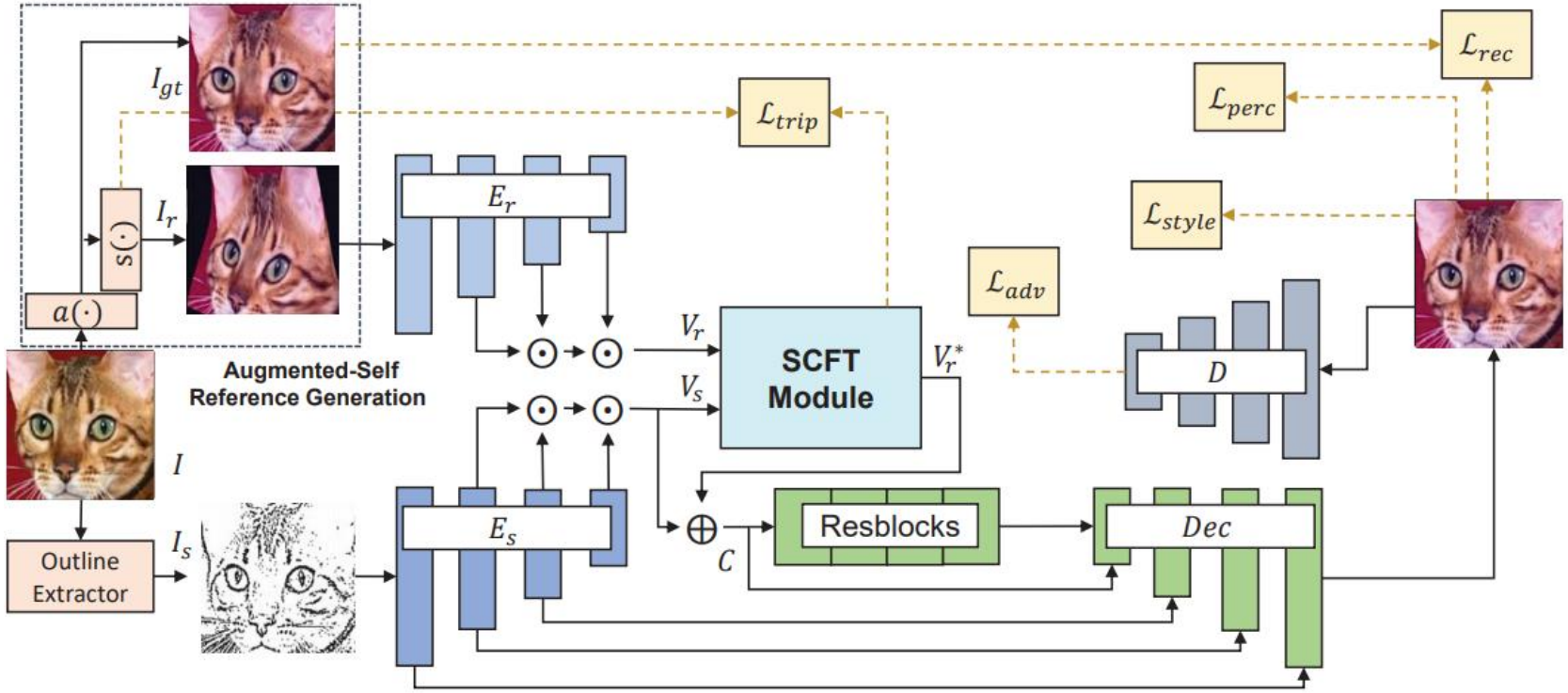
Reference-based Colorization

Reference 이미지를 사용하여, Sketch 이미지의 색을 입히는 태스크를
Reference-based Colorization이라고 정의함.

Lee et al, "Reference-Based Sketch Image Colorization using Augmented-Self Reference and Dense Semantic Correspondence", CVPR 2020

Reference-based Colorization의 Overview

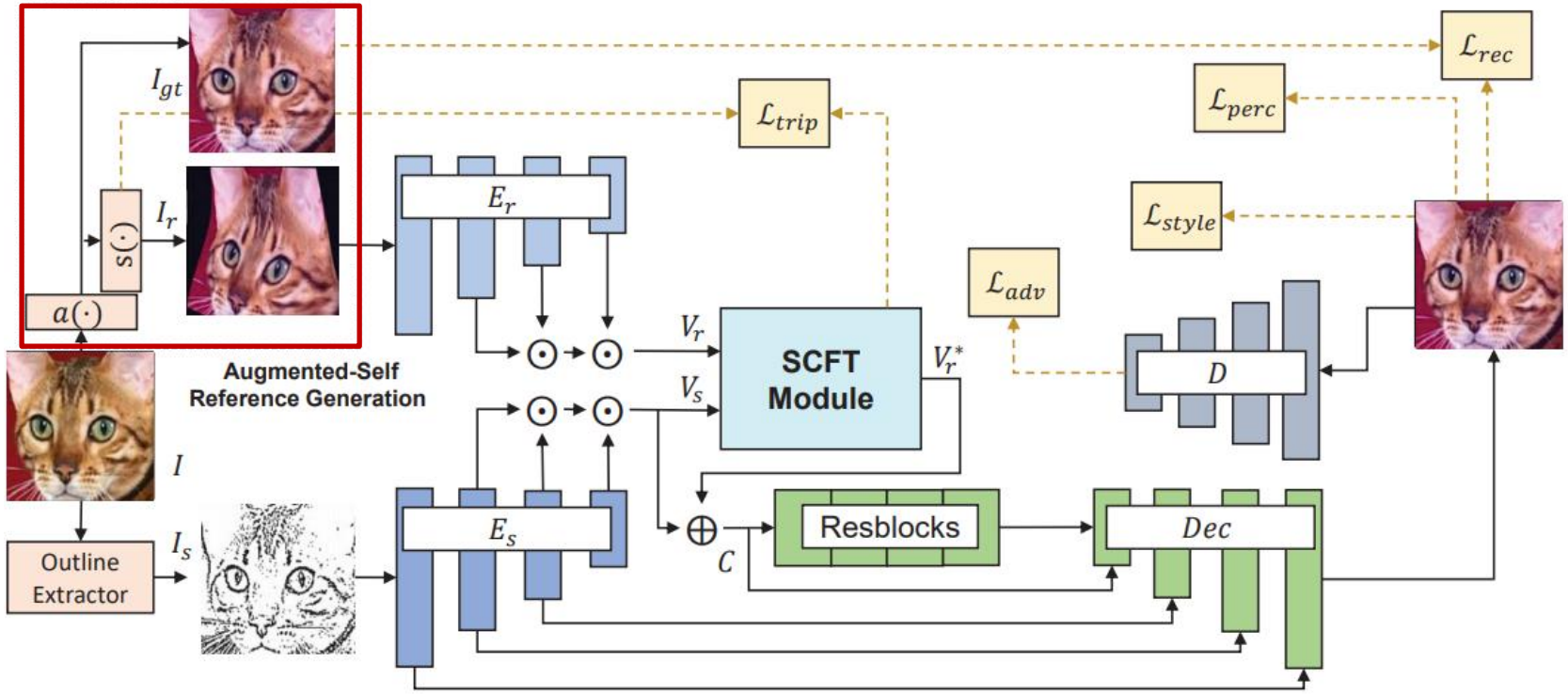
Reference-based Colorization 모델 구조는 다음과 같음.



Lee et al, "Reference-Based Sketch Image Colorization using Augmented-Self Reference and Dense Semantic Correspondence", CVPR 2020

Reference-based Colorization의 Overview

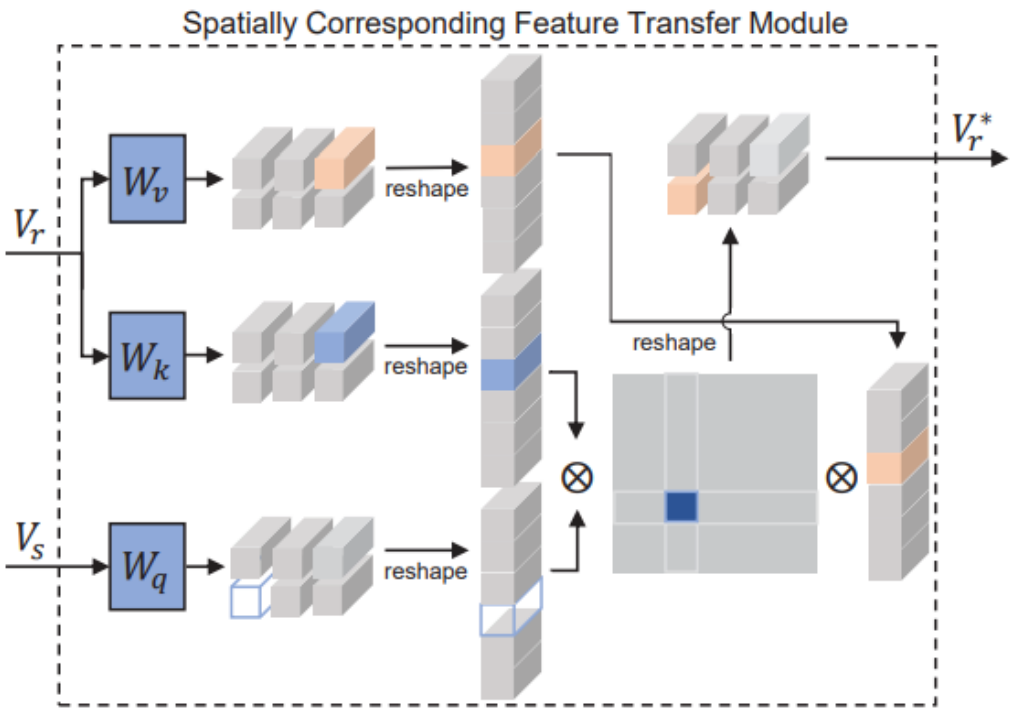
학습 과정에서 Color Jittering과 Spatial Transformation을 적용한 이미지를 Reference 이미지로 활용하고,
Color Jittering만 된 이미지를 GT 이미지로 사용함.



Lee et al, "Reference-Based Sketch Image Colorization using Augmented-Self Reference and Dense Semantic Correspondence", CVPR 2020

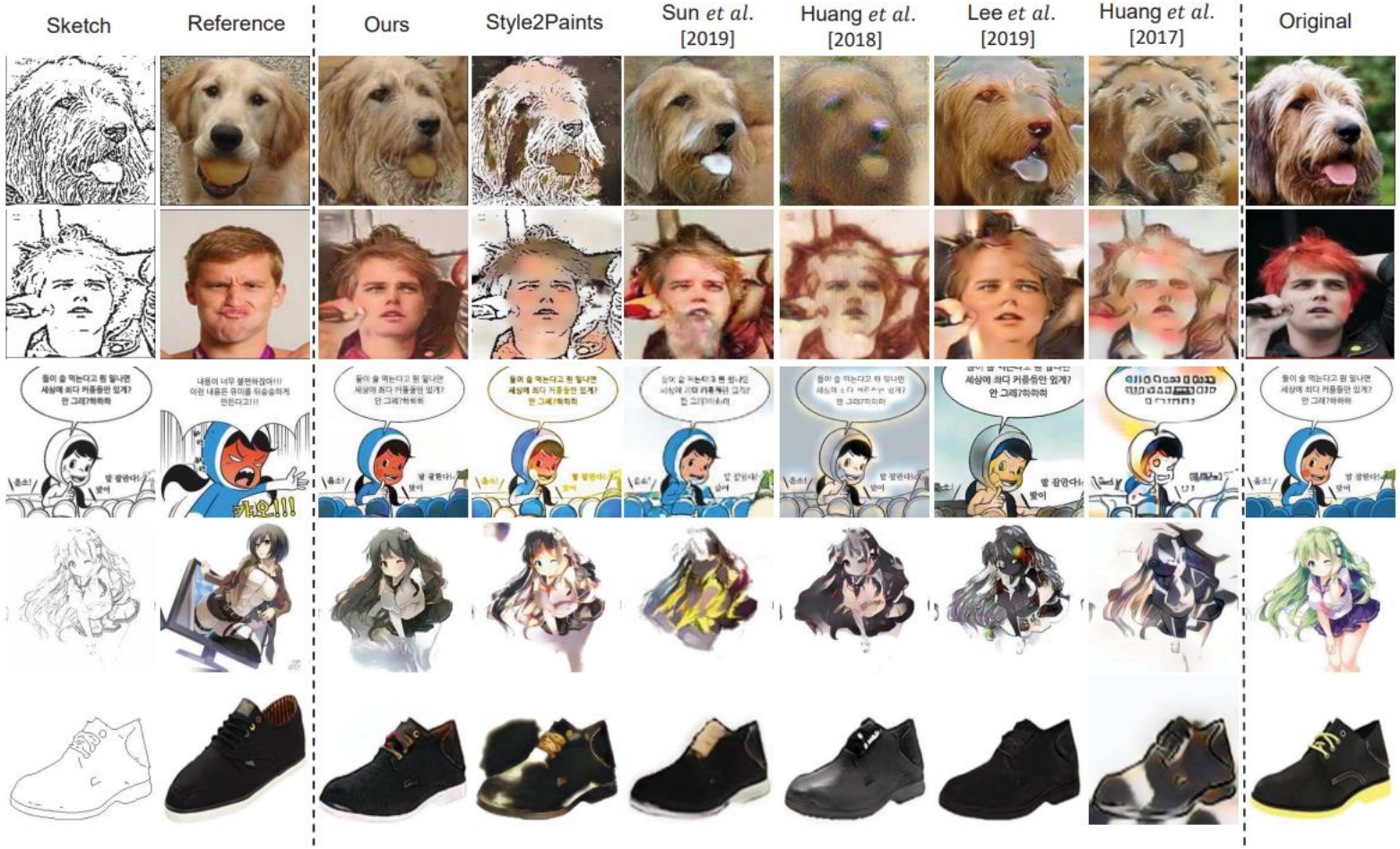
Spatially Corresponding Feature Transfer Module

Self-attention을 응용한 Spatially Corresponding Feature Transfer (SCFT) module을 제안함.



Lee et al, "Reference-Based Sketch Image Colorization using Augmented-Self Reference and Dense Semantic Correspondence", CVPR 2020

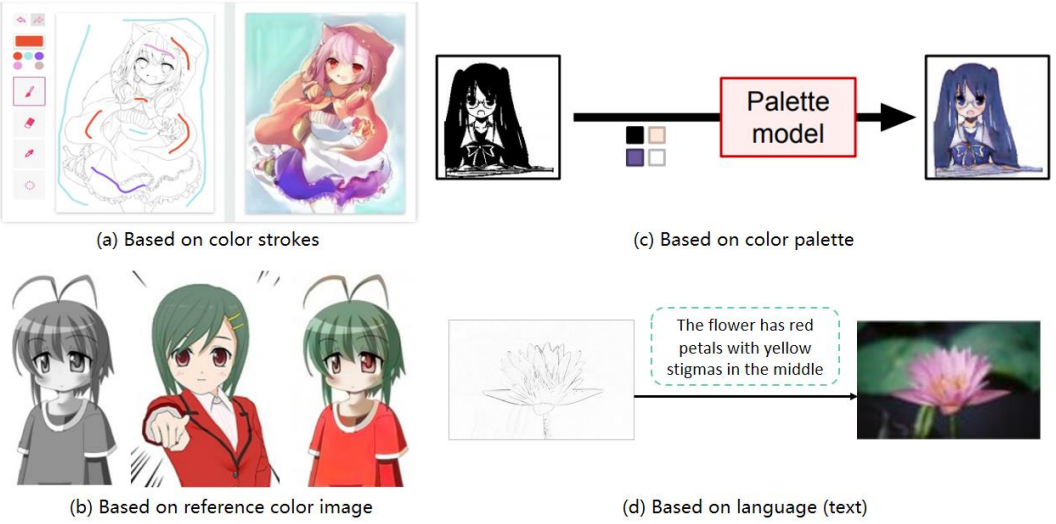
Reference-based Colorization의 결과



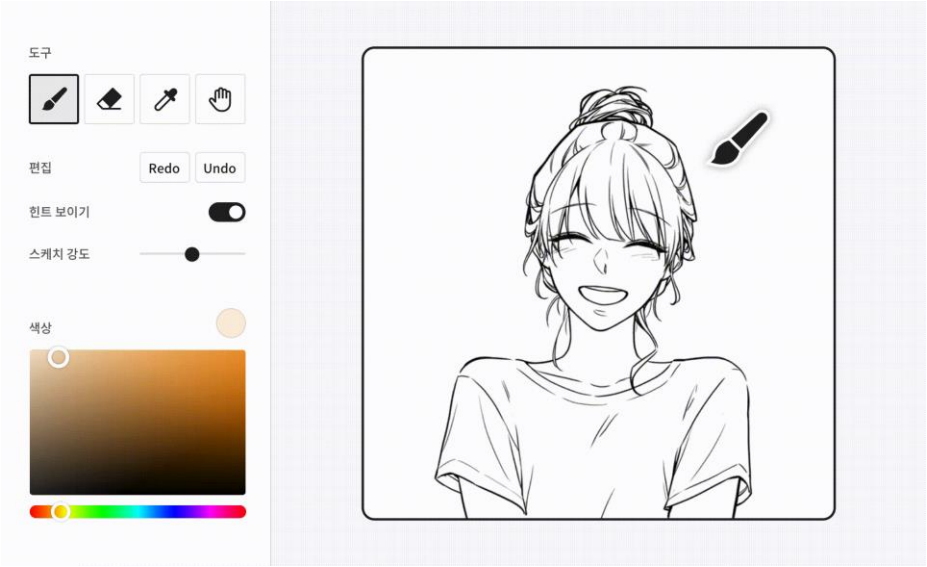
Lee et al, "Reference-Based Sketch Image Colorization using Augmented-Self Reference and Dense Semantic Correspondence", CVPR 2020

Research Trends

Colorization 분야에서는 사용자가 원하는 색을 입히기 위해 다양한 방식으로 User Interaction을 모델에게 전달함.
Stroke나 Language 등을 활용하여 Colorization을 하는 연구가 활발히 진행 중임.
User의 Stroke 혹은 Point를 활용하는 대표적인 Colorization으로는 네이버웹툰의 AI Painter 서비스가 있음.



(a) Based on color strokes
(c) Based on color palette
(b) Based on reference color image
(d) Based on language (text)



Naver Webtoon AI Painter

https://github.com/MarkMoHR/Awesome-Image-Colorization
https://ai.webtoons.com/painter/howto

Ch 4. Advanced Image-to-Image Translation Applications

Clip 4. Virtual Try-On

Virtual Try-On

Virtual Try-On은 원하는 옷을 입은 사람 이미지를 생성하는 태스크임.
Image-based / 3D-based Virtual Try-On 2가지가 있으나,
여기서는 Image-based Virtual Try-On만을 다룸.

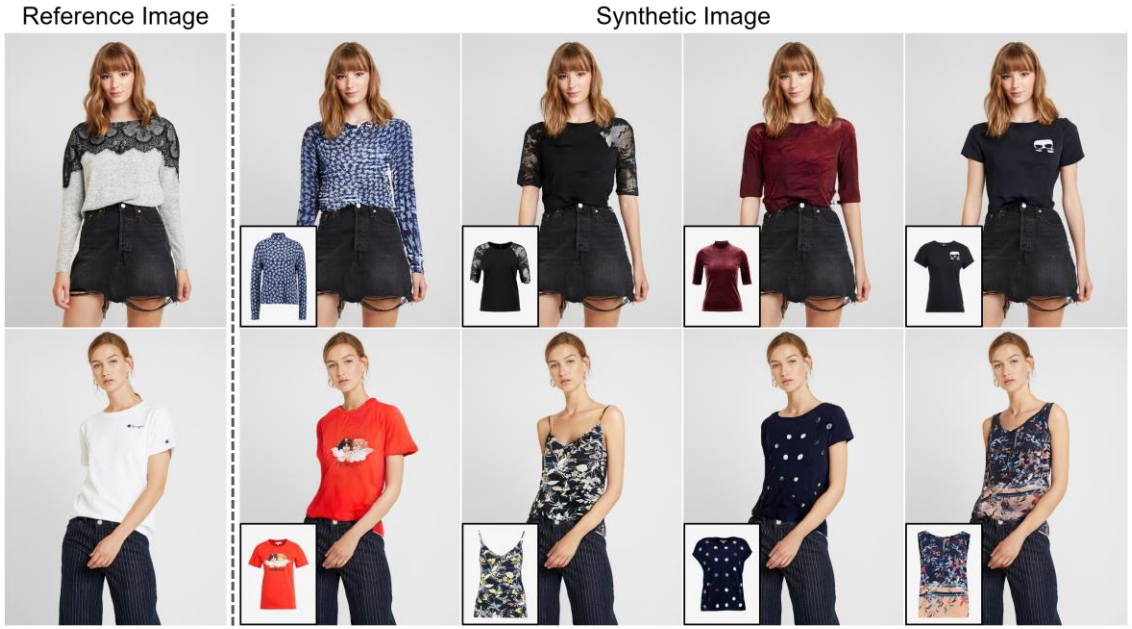


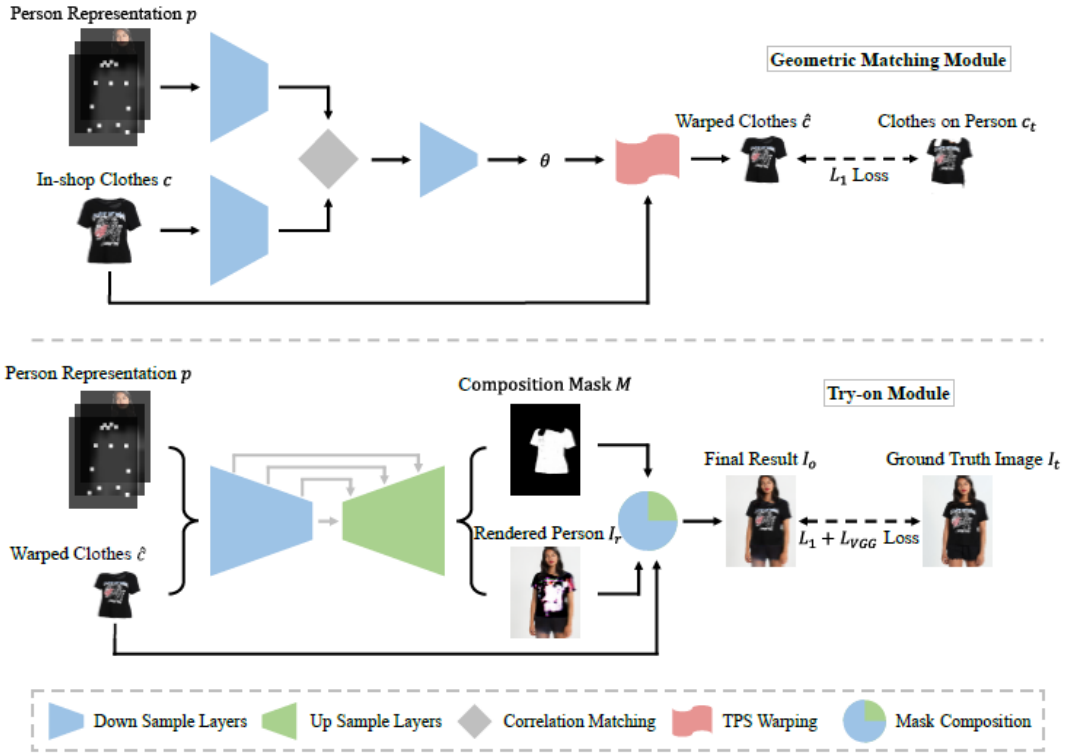
Image-based Virtual Try-On

3D-based Virtual Try-On

Choi et al, "VITON-HD: High-Resolution Virtual Try-On via Misalignment-Aware Normalization", CVPR 2021
Zhao et al., "M3D-VTON: A Monocular-to-3D Virtual Try-On Network", ICCV 2021

CP-VTON의 Overview

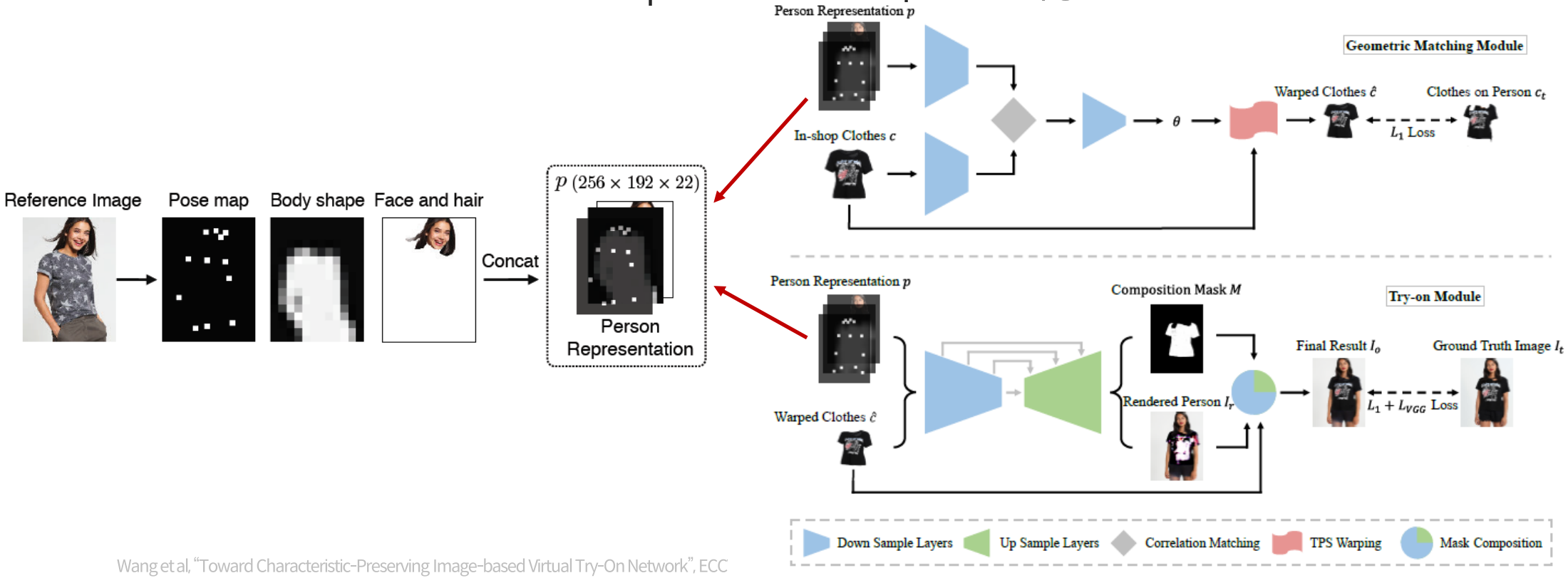
CP-VTON은 Virtual Try-On을 수행하기 위한 대표적인 모델이며, 크게 2가지 단계로 이루어짐.
(1) 옷을 사람의 몸에 맞춰서 변형함. (Geometric Matching Module (GMM))
(2) Warped Cloth를 이용하여 최종적인 이미지를 생성함. (Try-On Module)



Wang et al, "Toward Characteristic-Preserving Image-based Virtual Try-On Network", ECCV 2018

Person Representation

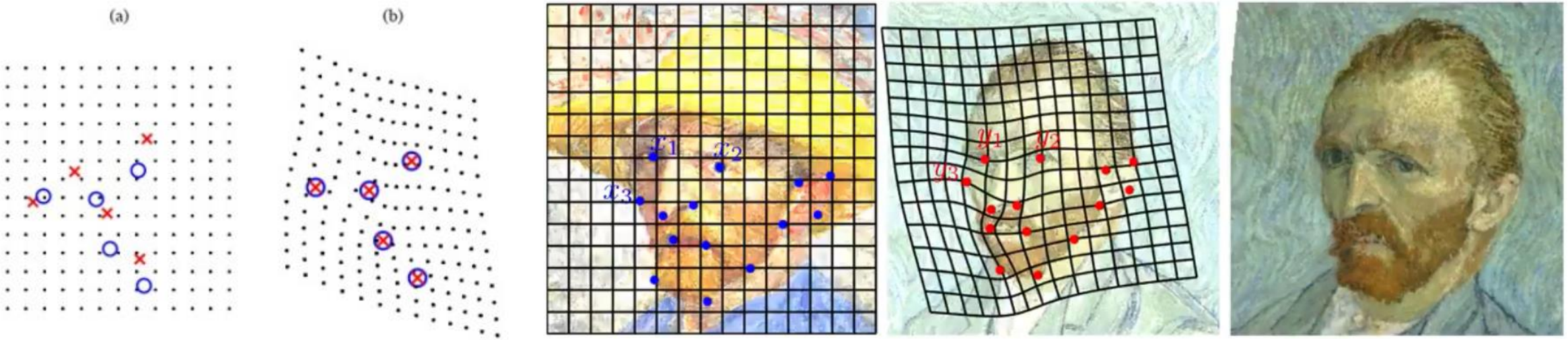
CP-VTON은 사람이 기존에 입고 있던 옷에 대한 정보를 지우기 위해
사람 이미지 대신 Pose map, Body shape, Face and hair로 구성된
Person Representation를 Input으로 사용함.



Wang et al, "Toward Characteristic-Preserving Image-based Virtual Try-On Network", ECC

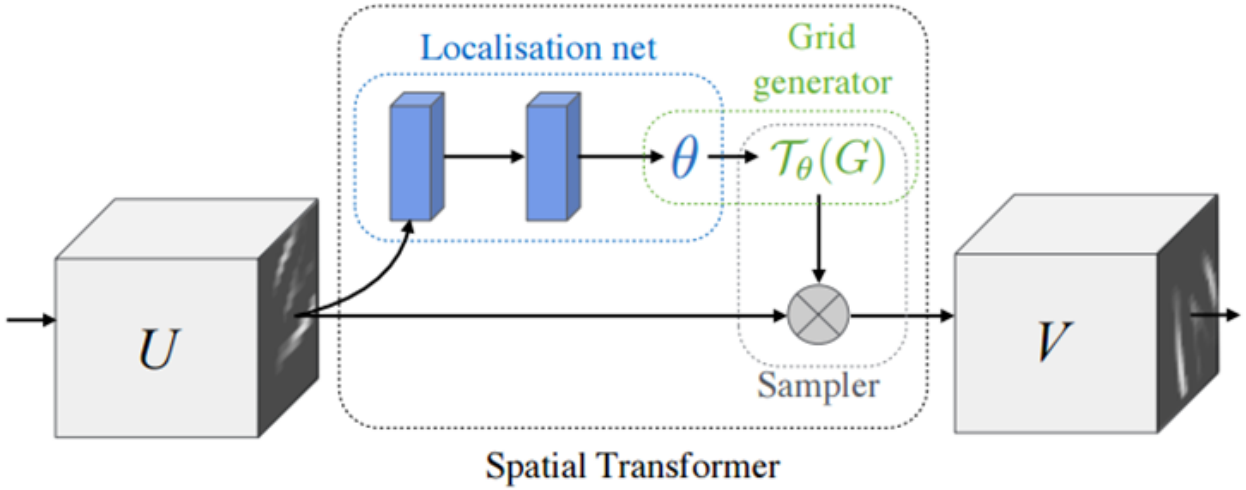
TPS Transformation

Thin Plate Spline (TPS) Transformation은
몇 개의 점 (Control Point) 의 위치를 이동시키고, 이를 이용하여 곡면 워핑 (warping) 변환을 수행하는 연산임.



Spatial Transformer Network

VITON과 다르게 CP-VTON에서는 옷을 변형하기 위해 Neural Network를 활용함.
이를 위해 Spatial Transformer Network를 응용함.
CP-VTON에서는 TPS Transformation의 파라미터를 Neural Network로 예측한 후,
이를 이용하여 옷을 Warping함.



A Spatial Tranfomer module

Jaderberg et al, "Spatial Transformer Networks", NeurIPS 2015

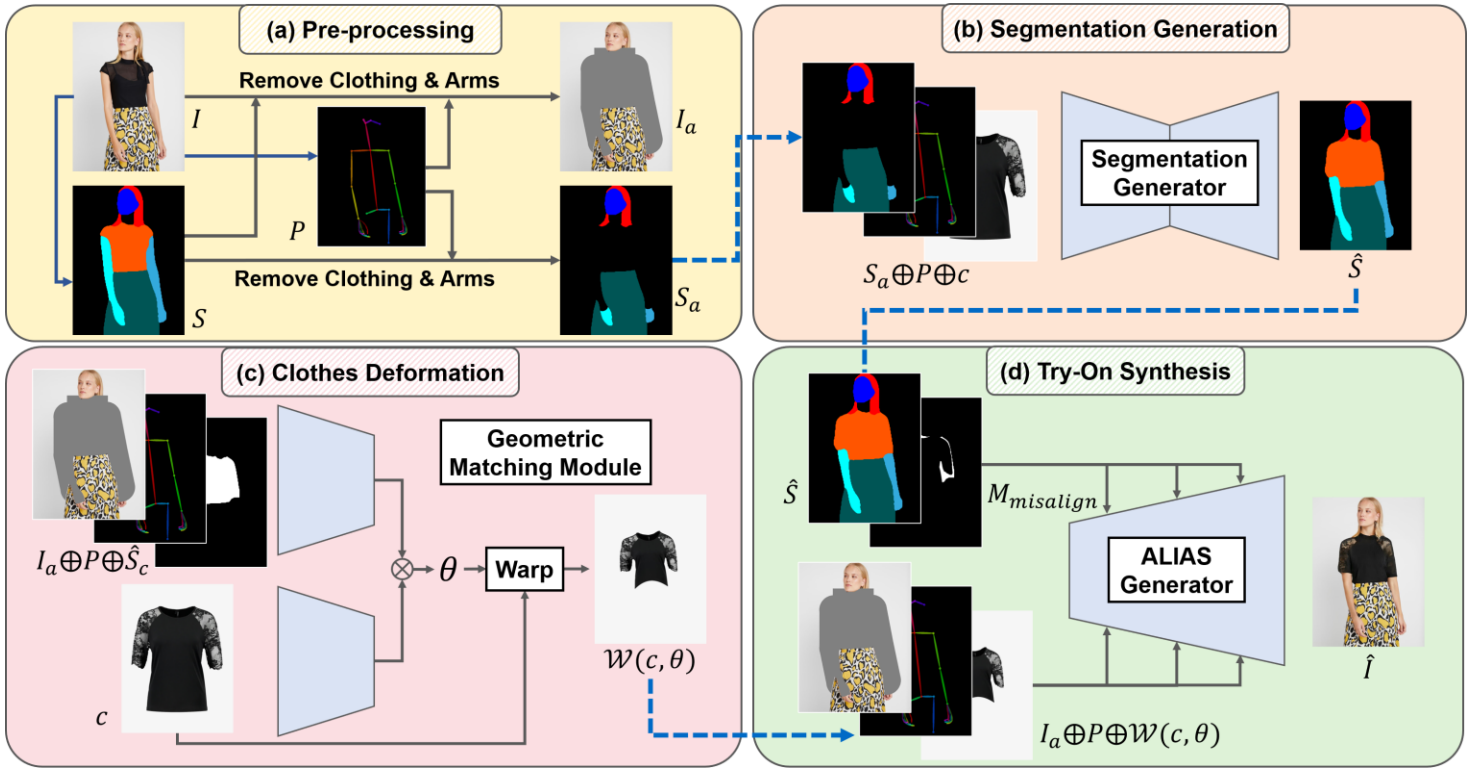
CP-VTON의 결과

CP-VTON은 GMM을 통해 더 정확하게 옷을 변형하고,
Try-On Module의 Mask Composition 연산으로 옷의 디테일을 유지하는데 성공함.

Wang et al, "Toward Characteristic-Preserving Image-based Virtual Try-On Network", ECCV 2018

VITON-HD의 Overview

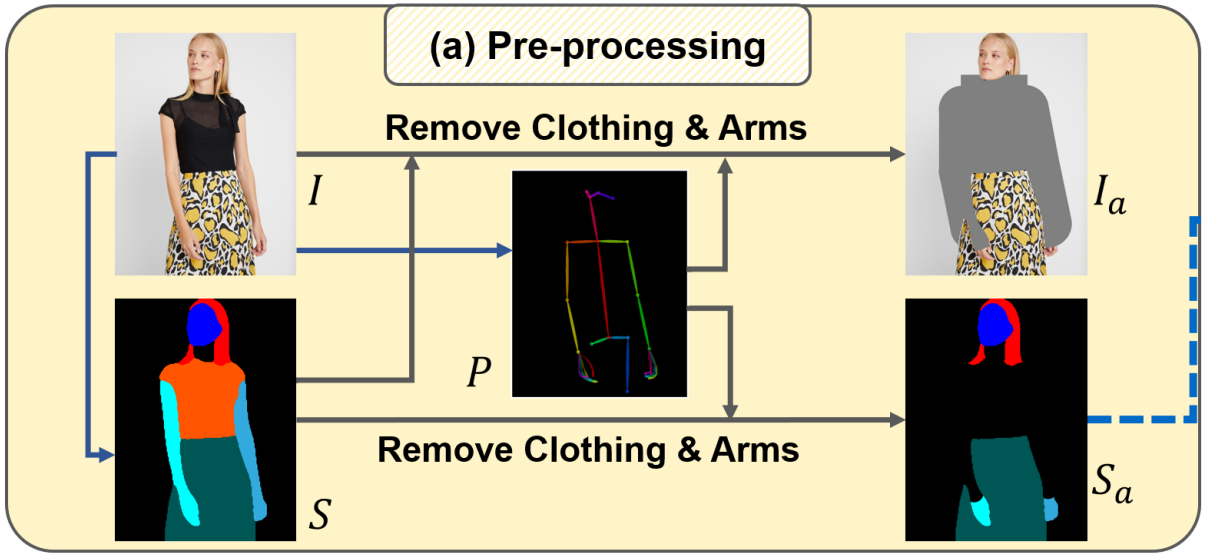
고화질 이미지를 생성해낼 수 있는 VITON-HD는 크게 4가지 과정을 수행함.



Choi et al, "VITON-HD: High-Resolution Virtual Try-On via Misalignment-Aware Normalization", CVPR 2021

VITON-HD의 Pre-processing

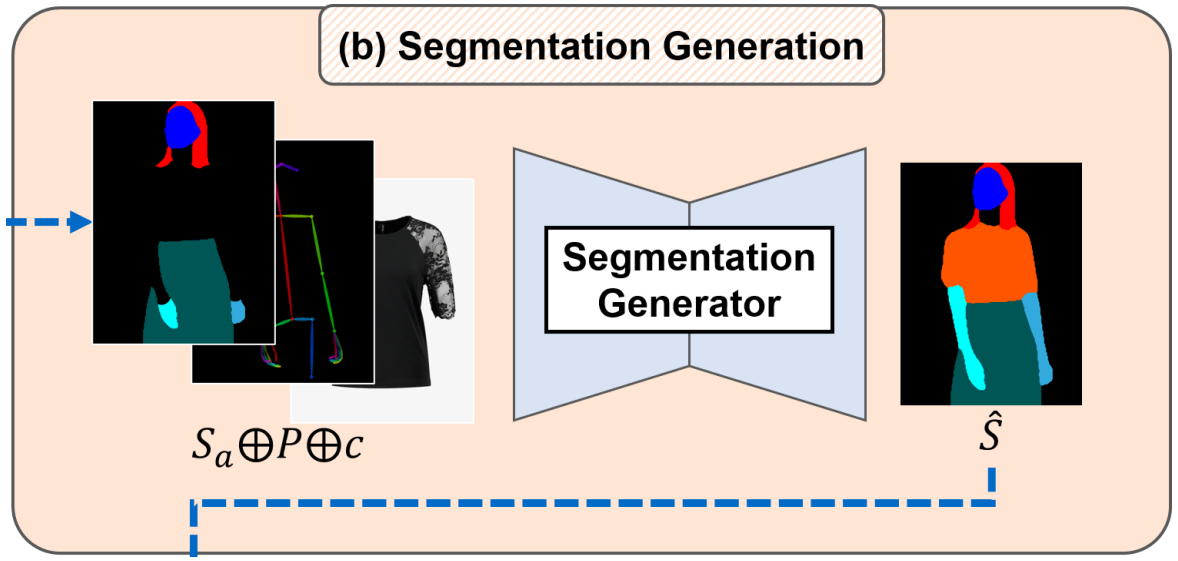
옷을 바꾸고자 하는 사람 이미지에서 Pre-trained된 모델로
Pose와 Segmentation Map을 추출함.
그 다음 변형하고자 하는 옷 부분을 지우는 작업을 수행함.



Choi et al, "VITON-HD: High-Resolution Virtual Try-On via Misalignment-Aware Normalization", CVPR 2021

VITON-HD의 Segmentation Generation

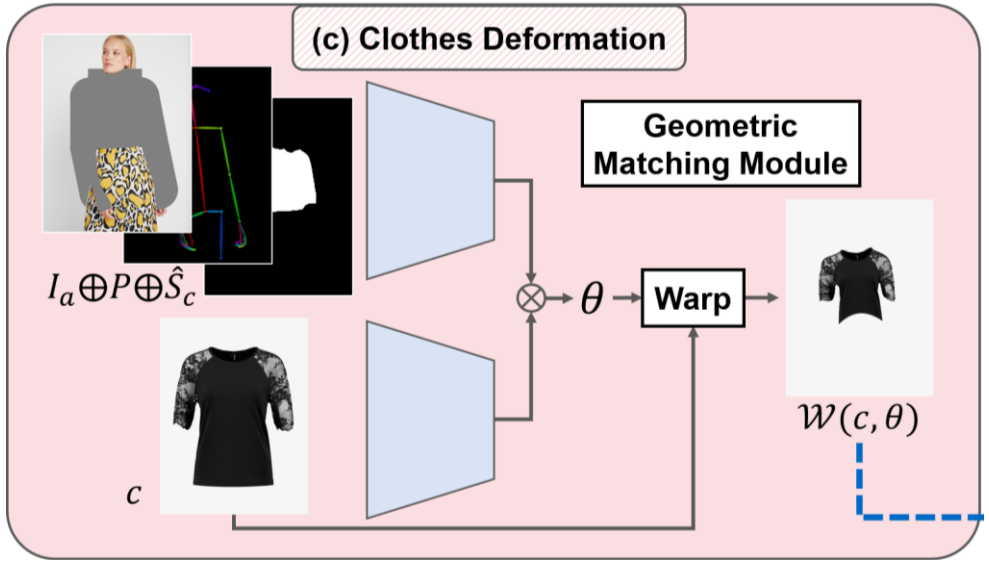
입히고자 하는 옷을 입은 사람의 Segmentation을 먼저 생성함.
Segmentation Map을 먼저 생성하는 것은 생성하고자 하는 이미지의 윤곽을 잡는데 도움이 됨.



Choi et al, "VITON-HD: High-Resolution Virtual Try-On via Misalignment-Aware Normalization", CVPR 2021

VITON-HD의 Clothes Deformation

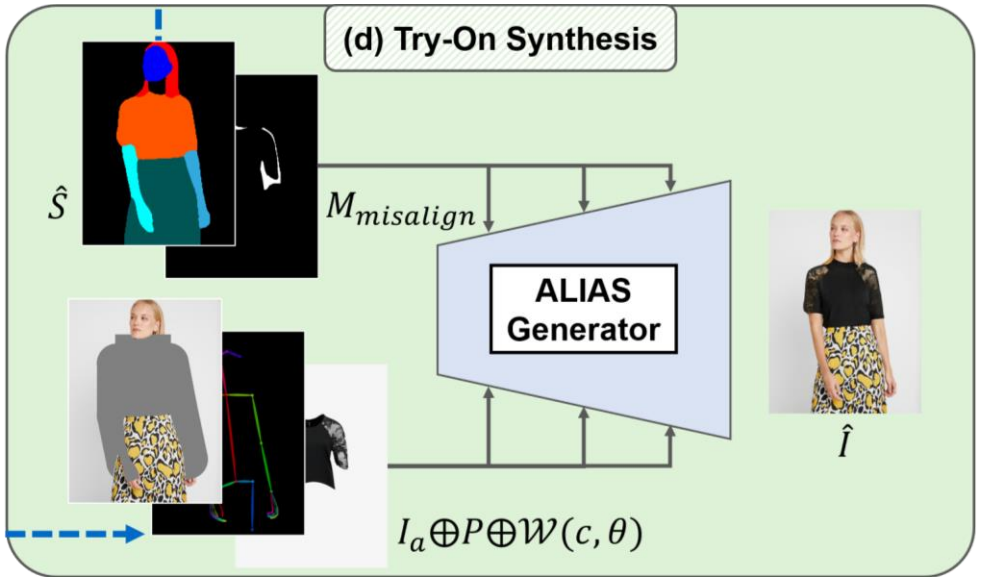
CP-VTON과 같이 옷을 사람의 몸에 맞추기 위해 입히고자 하는 옷을 변형하는 과정을 수행함.
Geometric Matching Module (GMM) 은 CP-VTON과 유사함.



Choi et al, "VITON-HD: High-Resolution Virtual Try-On via Misalignment-Aware Normalization", CVPR 2021

VITON-HD의 Try-On Synthesis

최종적으로 이전 과정에서 생성한 Segmentation Map과 Warped Cloth 등을 이용하여, 최종 이미지를 생성함.
이 때, Warped Cloth와 사람의 몸 사이에서 생긴
Misalignment를 해결할 수 있는 ALIAS Generator를 활용함.



(a) Warped Clothes on Reference Image | (b) Warped Clothes on Segmentation | (c) Misaligned Regions

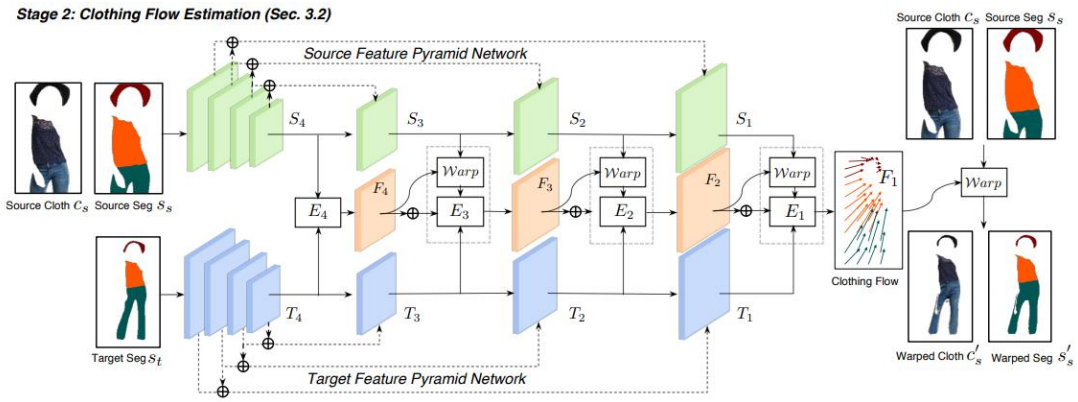
Choi et al, "VITON-HD: High-Resolution Virtual Try-On via Misalignment-Aware Normalization", CVPR 2021

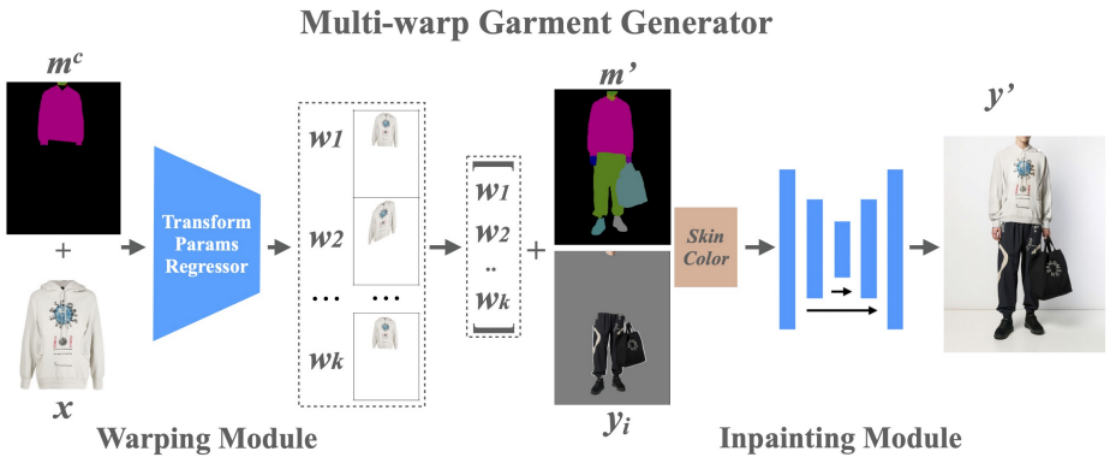
VITON-HD의 결과

Choi et al, "VITON-HD: High-Resolution Virtual Try-On via Misalignment-Aware Normalization", CVPR 2021

Research Trends

Virtual Try-On 분야에서는 옷을 입히는 과정에서 좀더 정확한 warping 연산을 위한 방법들이 새로 제안되고 있음.
최근에는 Optical Flow를 사용하거나, 여러 번 warping 연산을 적용하는 등의 방법들이 제안됨.
이외에도 3D Virtual Try-On 분야의 연구도 활발히 진행되고 있음.





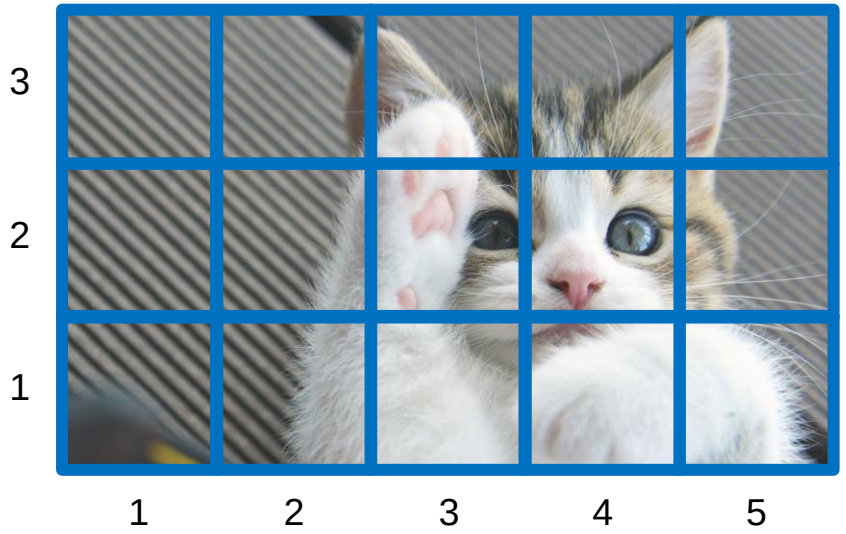
Han et al., "ClothFlow: A Flow-Based Model for Clothed Person Generation", ICCV 2019
Li et al., "Toward Accurate and Realistic Outfits Visualization with Attention to Details", CVPR 2021

Ch 4. Advanced Image-to-Image Translation Applications

Clip 5. 3D-Aware Multi-View Synthesis

Implicit Representation

Implicit representation은 2가지 장점이 있음.:
(1) Lighter than voxel representation. (2) Continuous.



$f(3,2) =$

$f(4,3) =$

$f(1,1) =$

...

Example of implicit representation for 2D image

Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

Neural Radiance Field

Neural Radiance Field는 neural network로
5D coordinate (x, y, z, θ, ϕ)를 volume density (σ)와 RGB color (r, g, b)로 mapping시킴.

$$f(x, y, z, \theta, \phi) = (r, g, b, \sigma)$$

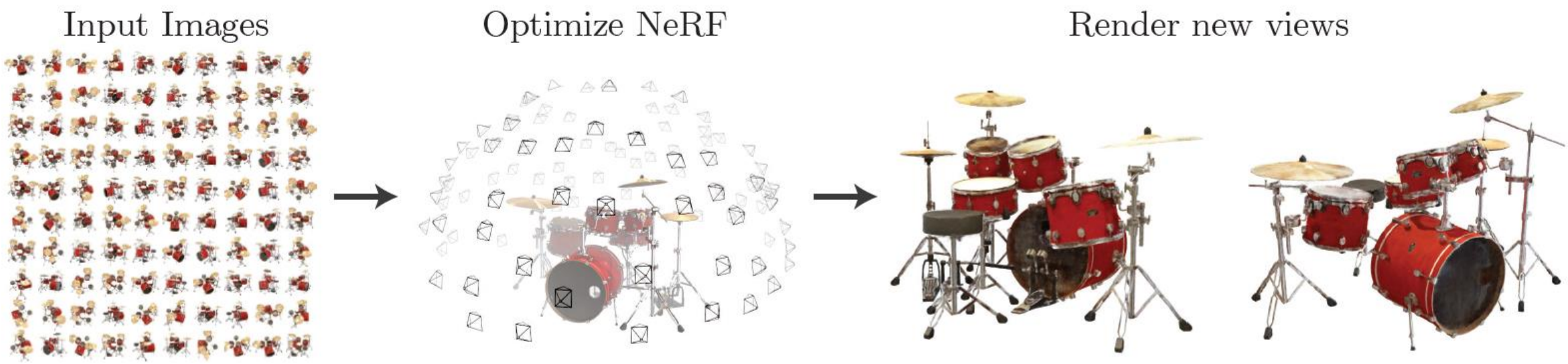
왜 viewing angle과 volume density가 필요할까? (예: 거울, 유리)

Viewing angle이 바뀌면,
거울 속의 이미지도 바뀜.

유리 뒤의 물체도 볼 수 있음.

Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

Novel View Synthesis



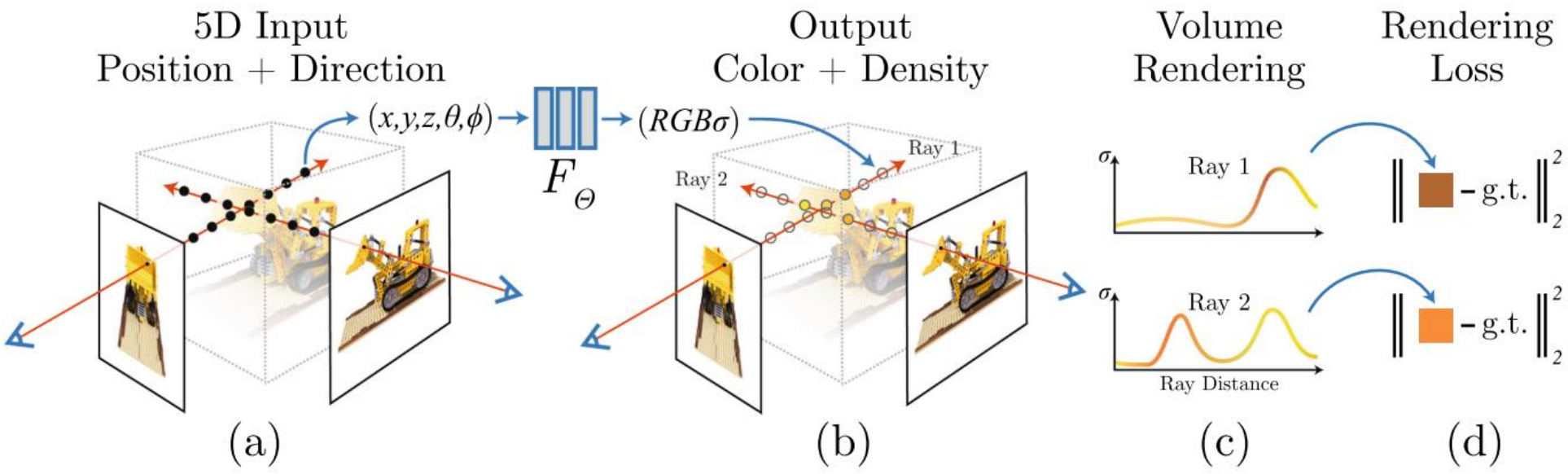
Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

Novel View Synthesis

Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

NeRF

NeRF는 sparse한 input view들을 통해 novel view를 생성하는 방법임.
이 때, volume rendering 기술을 활용함.
Input : 3D location $\boldsymbol{X} = (x, y, z)$ 과 2D Viewing direction (θ, ϕ)
Output : Emitted color $\boldsymbol{C} = (r, g, b)$ 와 volume density σ

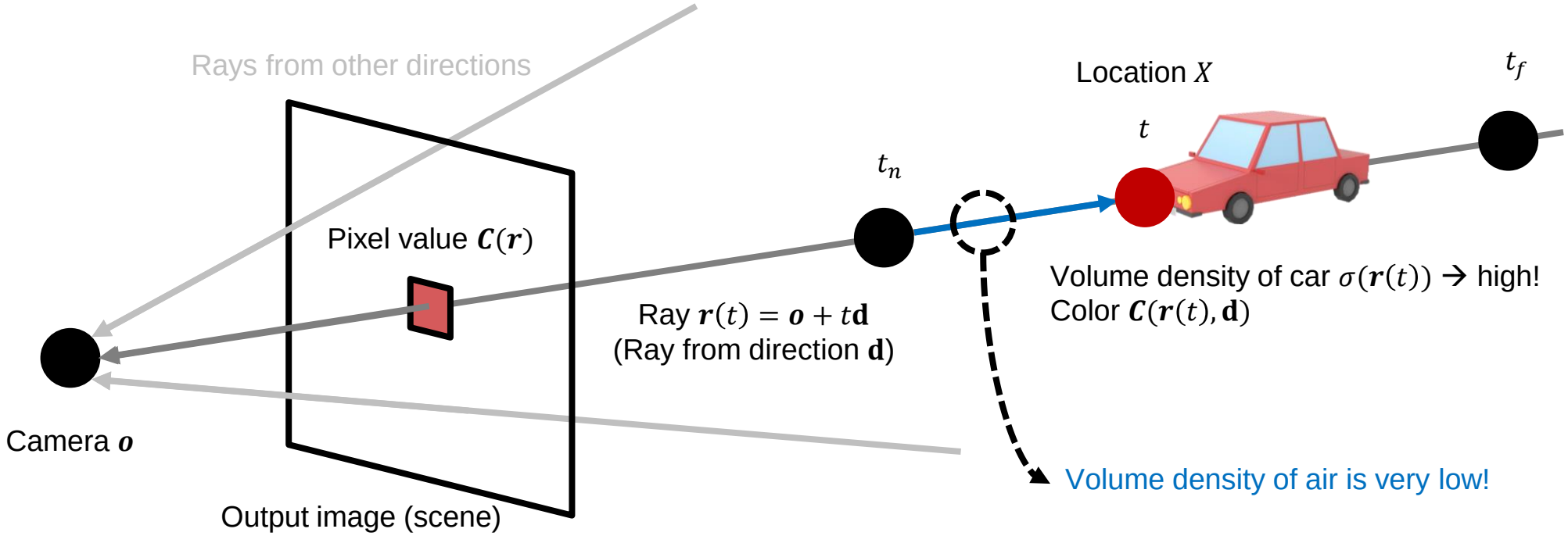


Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

Volume Rendering with Radiance Fields

The expected color $C(\boldsymbol{r})$ of camera ray $\boldsymbol{r}(t) = \boldsymbol{o} + t\mathbf{d}$ with near and far bound t_n and t_f is:

$$C(\boldsymbol{r}) = \int_{t_n}^{t_f} T(t)\sigma\big(\boldsymbol{r}(t)\big)\boldsymbol{c}(\boldsymbol{r}(t), \mathbf{d})dt, \qquad \text{where } T(t) = \exp\left(-\int_{t_n}^{t} \sigma\big(\boldsymbol{r}(s)\big)ds\right)$$



Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

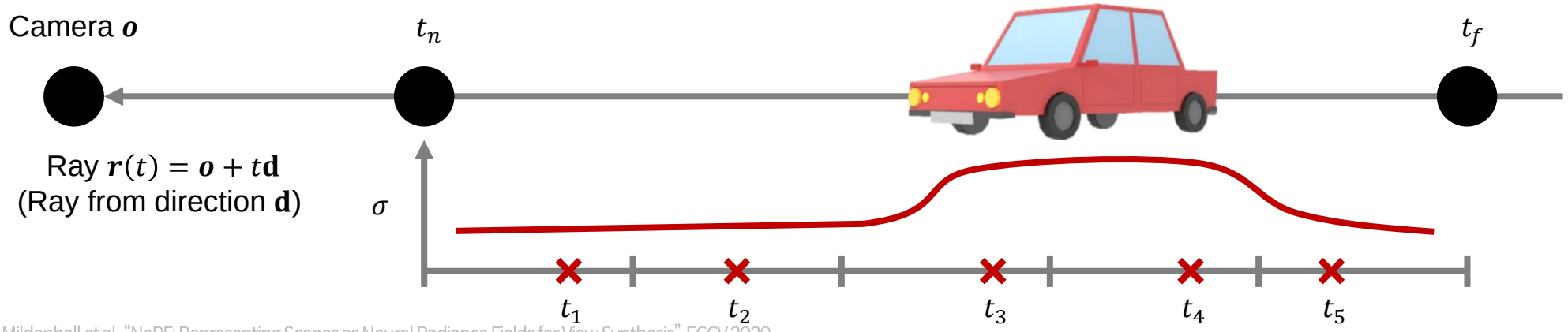
Volume Rendering with Radiance Fields

어떻게 적분을 계산할까? → Sampling!

$$\hat{C}(\boldsymbol{r}) = \sum_{i=1}^{N} T_i(1 - \exp(-\sigma_i \delta_i))\boldsymbol{c}_i\,, \qquad \text{where } T_i = \exp(-\sum_{j=1}^{i-1} \sigma_j \delta_j)\,, \qquad \delta_i = t_{i+1} - t_i$$

구간 $[t_n, t_f]$을 일정한 간격으로 N개로 나눈 후, Sampling을 진행:

$$t_i \sim \text{Uniform}\left[t_n + \frac{i-1}{N}(t_f - t_n), t_n + \frac{i}{N}(t_f - t_n)\right].$$



Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

Volume Rendering with Radiance Fields

효과적인 camera ray의 sampling을 위해 Hierarchical volume sampling을 제안함.

$$\hat{C}_c(\boldsymbol{r}) = \sum_{i=1}^{N_c} w_i C_i \,, \qquad w_i = T_i(1 - \exp(-\sigma_i \delta_i))$$

1) Coarse network $\hat{C}_c(\boldsymbol{r})$ 가 먼저 N_c locations에 대해 계산함.
2) 다음, w_i에 normalization을 적용하고, camera ray에 따른 piecewise-constant PDF를 계산함.
3) PDF를 이용하여, N_f locations를 sampling하고, fine network $\hat{C}_f(\boldsymbol{r})$를 활용함.

$\hat{C}_f(\boldsymbol{r})$는 최종적인 rendering에 $N_c + N_f$ sample을 사용함.
Coarse network와 Fine network를 학습시키기 위한 loss function은 다음과 같음.

$$L_{total} = \sum_{\boldsymbol{r} \in R} \left[\left\| \hat{C}_c(\boldsymbol{r}) - C(\boldsymbol{r}) \right\|_2^2 + \left\| \hat{C}_f(\boldsymbol{r}) - C(\boldsymbol{r}) \right\|_2^2 \right]$$

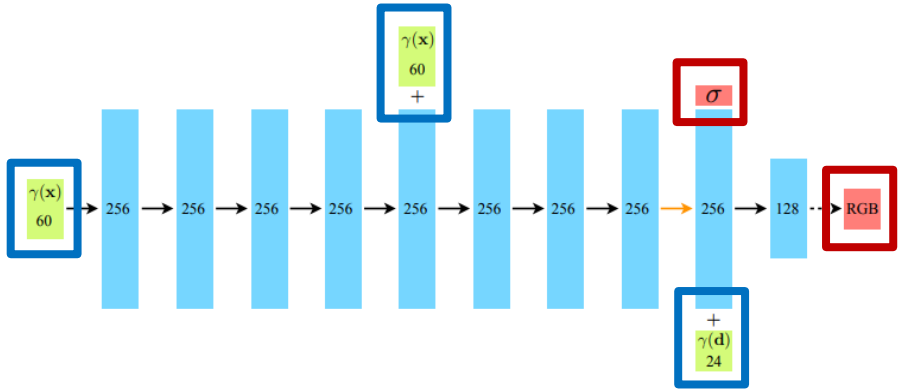
Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

Positional Encoding

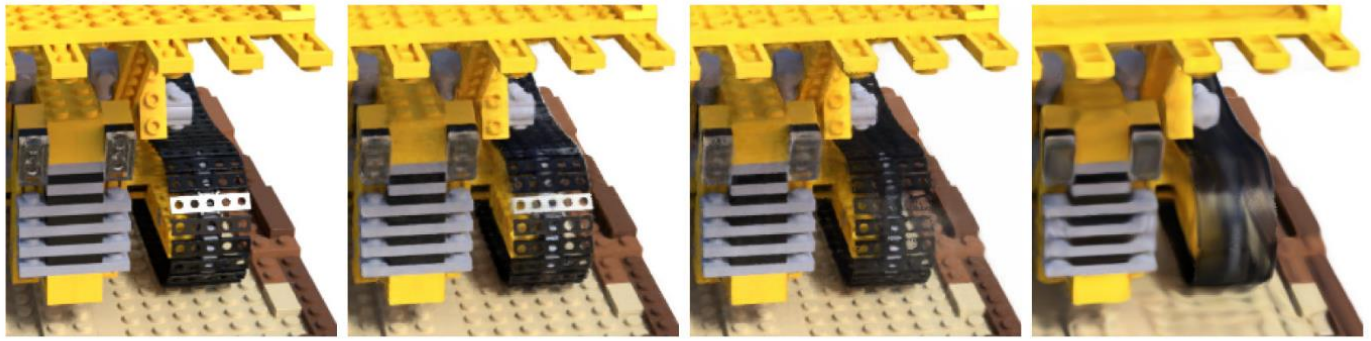
NeRF에서는 input 값들을 고차원의 space로 mapping한 후에 모델에 넣음.
이러한 trick을 positional encoding이라고 정의하며, 방법은 다음과 같음.

$$\gamma(p) = (\sin(2^0\pi p), \cos(2^0\pi p), \dots, \sin(2^{L-1}\pi p), \cos(2^{L-1}\pi p))$$

→ Single scalar value가 2L-dimensional vector로 encoding됨.
(NeRF에서는 3D location는 $L = 10$으로, 2D viewing direction은 $L = 6$로 설정하여 사용함.)



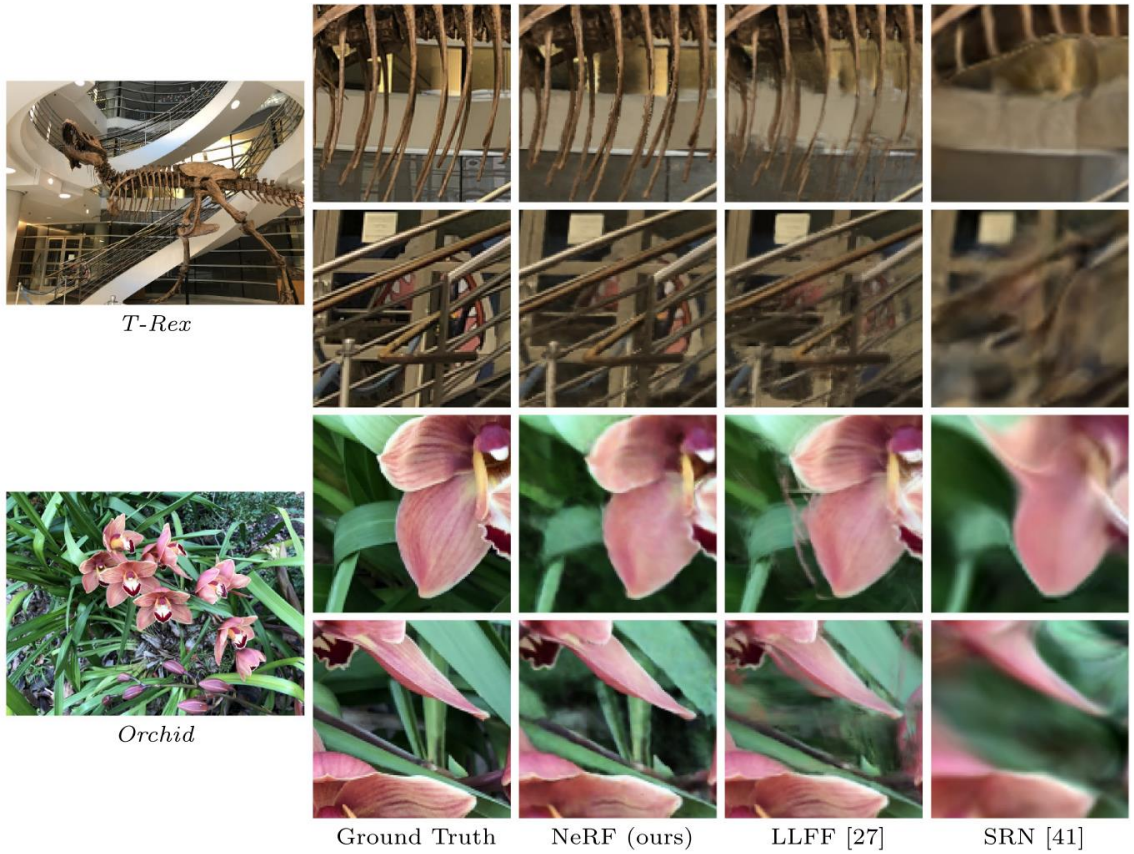
Model architecture (See assumption)

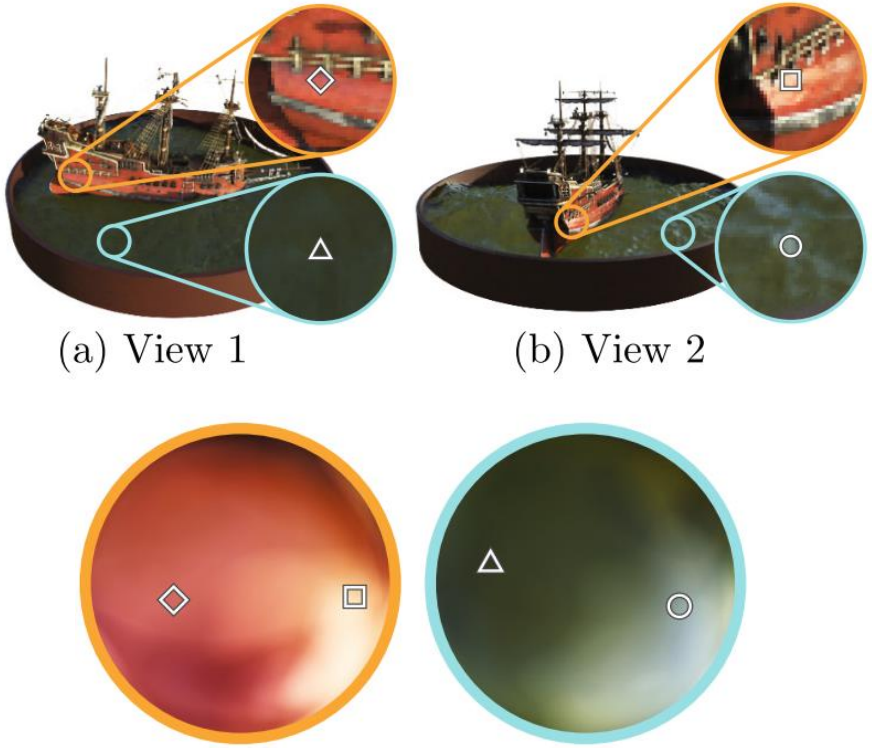


Ground Truth | Complete Model | No View Dependence | No Positional Encoding

Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

NeRF의 결과



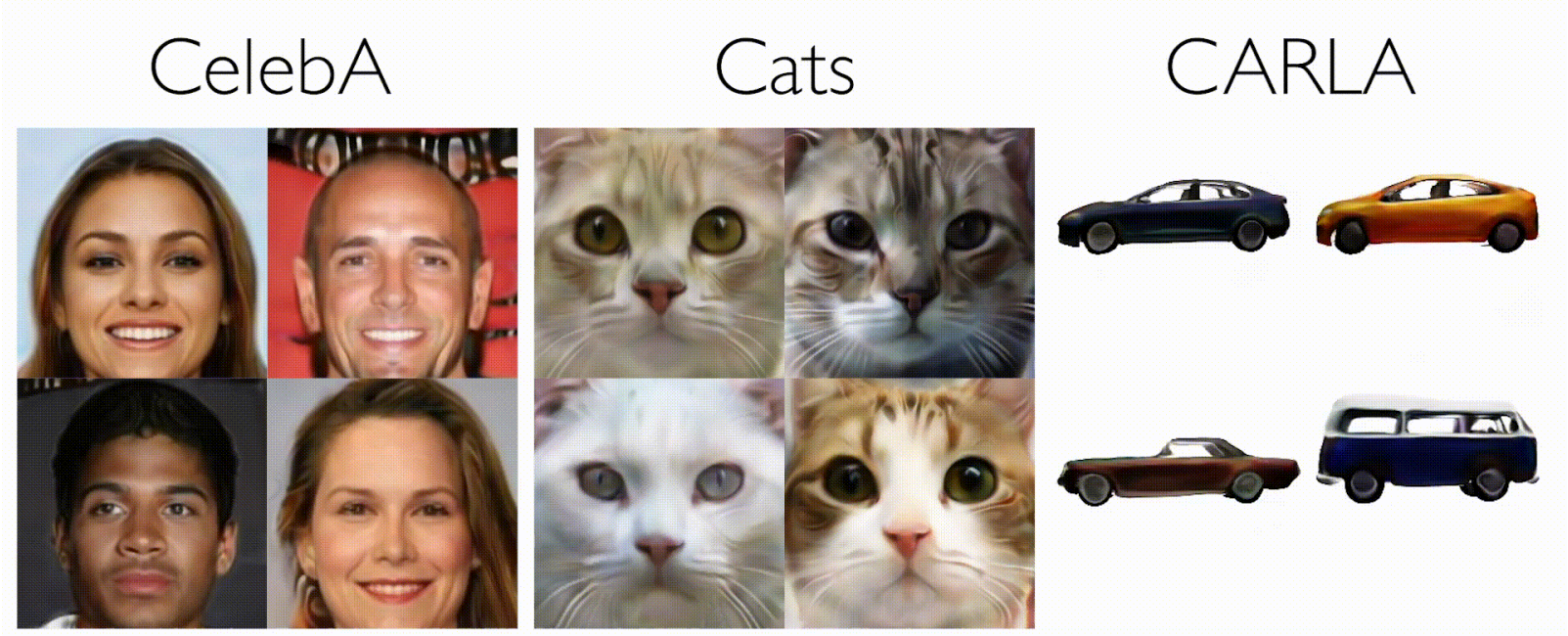


Hemisphere of viewing direction (θ, ϕ)

Mildenhall et al., "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis", ECCV 2020

Research Trends

NeRF 논문이 나온 이후, 수많은 연구들이 활발하게 진행되고 있음.
기존의 생성 모델과 NeRF를 결합하여 다양한 3D 각도에서 본 이미지 합성할 수 있는 모델도 많이 연구되고 있음.



Chan et al., "pi-GAN: Periodic Implicit Generative Adversarial Networks for 3D-Aware Image Synthesis", CVPR 2021

Useful Resource

- Paper with Code : https://paperswithcode.com/newsletter
- Tensorflow-KR : https://ko-kr.facebook.com/groups/TensorFlowKR/
- arXiv Sanity : http://www.arXiv-sanity.com/
- CVPR Buzz : https://mattdeitke.com/cvpr-buzz/

- Neural Talking Head Paper List : https://github.com/DaddyJin/awesome-faceReenactment
- Super-Resolution Paper List : https://github.com/ChaofWang/Awesome-Super-Resolution
- Colorization Paper List : https://github.com/MarkMoHR/Awesome-Image-Colorization
- Virtual Try-On Paper List : https://github.com/minar09/awesome-virtual-try-on
- NeRF Paper List : https://github.com/yenchenlin/awesome-NeRF

Thank you.