



StarGAN Review

배경

- 모든 domain에 대해서 독립적인 모델들이 만들어져야함으로, 2가지 이상의 domain을 다루는데 제한된 **scalability(확장성)**와 **robustness(견고성)**가 있음
- 단 하나의 모델을 가지고 여러가지 domain에 대해 image-to-image translation을 수행 하여, 다른 domain을 가진 data들을 동시에 학습시키려 함. (mask vector method)

간단한 용어 소개

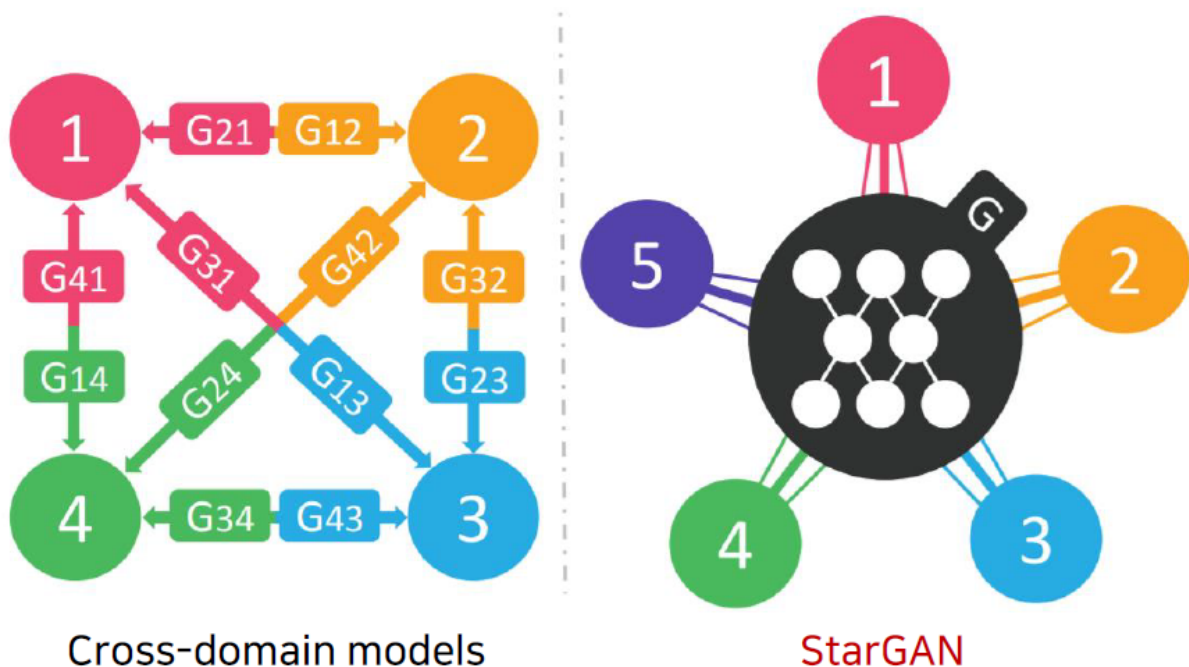
- **attribute**: image에 내제된 의미있는 feature ex) hair, color, gender, age
- **attribute value**: attribute의 특정한 값

attribute	attribute value
hair color	black, blond, brown
gender	male, female

- **domain**: 같은 attribute value를 가지는 이미지셋 ex) female인 이미지들은 하나의 domain

Introduction

- StarGAN 이전에 존재하는 모델들은 multi-domain image translation tasks에서 비효율적
 - K개의 domain들 사이의 모든 mapping을 배우기 위해서는 모든 domain을 서로 cross하여 $K(K-1)$ 개의 generator(발생기) 만들어야하기 때문!



- 왼쪽의 그림처럼, 4개의 domain간의 mapping을 모두 학습하려면 4*3개의 generator 필요
- 그러나 StarGAN은 오른쪽의 그림처럼, 1개의 generator로 모든 mapping들을 학습시킴

→ 하나의 뉴럴 네트워크를 이용해 다중 도메인(multi domain) 사이에서의 이미지 변환 가능!

Idea

고정된 translation만 학습하는 것이 아니라(흑발 → 금발 동작만 학습하는 것이 아니라)

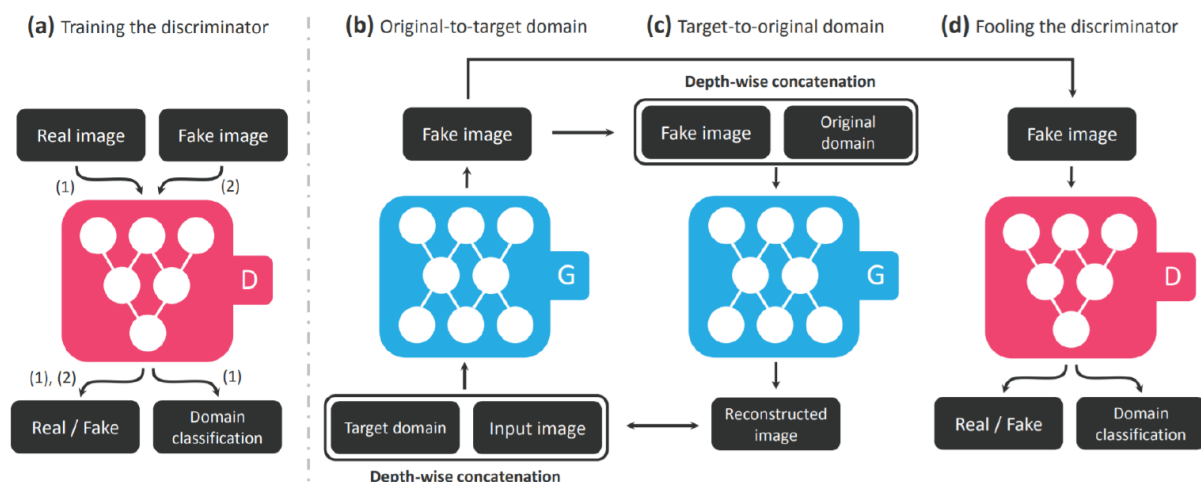
모델이 image와 domain 정보를 함께 input으로 받아서, input image에 대응되는 domain 으로

translation 할 수 있도록 학습시키는 것

- label: domain 정보를 담고 있는 label로, binary or one-hot vector 형태
- 학습과정에서 **target domain label**을 만들어 모델이 target domain으로 이미지를 변화시키도록 학습
- domain label에 **mask vector**를 추가해 다른 dataset의 domain 사이의 결합된 학습 가능

Multi-Domain Image-to-image Translation

1. Single generator G로 다수의 domain들 사이의 mapping을 학습시키기 위해서, G는 어떤 **target domain label c**를 이용하여 **input image x**를 **output image y**로 변환하는 것을 학습해야함
 - $G(x, c) \rightarrow y$
2. **discriminator D**는 source(Real의 image인지, G가 생성해낸 image인지)와 domain labels에 대한 **확률분포**를 만들어 내야함.
 - $D : x \rightarrow \{D_{src}(x), D_{cls}(x)\}$
 - discriminator: 판별자



(다른 GAN모델과 마찬가지로 2개의 모듈로 구성되어 있음 → **discriminator D** & **generator G**)

(a) D는 **real image** 와 **fake image**를 구별하는 것과 동시에,

real image일때 그것과 상응하는 domain을 분류해내는 것을 학습함

(b) G는 input으로 image와 동시에 target domain label을 받고 **fake image**를 생성함

(c) G는 original domain label로 **fake image**를 다시 original image로 reconstruction을 시도함

(d) G는 **real image**와 구분불가능하고 D에 의해 target domain이 분류가능한 이미지를 생성하려 함

→ 즉 **real image**처럼 보이려고 노력하는 것

Loss Function

들어가기에 앞서...

- G: x와 target domain label을 가지고 $G(x,c)$ 라는 이미지를 만들어냄
- D: src(=source)로 [Input data에서 온 **Real image**]와 [생성된 **Fake image**]를 구분하려고

노력한다는 의미의 Loss

StarGAN: ① **Adversarial loss** + ② **Domain classification loss** + ③ **Reconstruction loss**

$$\text{Adversarial } \mathcal{L}_{adv} = \mathbb{E}_x [\log D_{src}(x)] + \mathbb{E}_{x,c} [\log (1 - D_{src}(G(x, c)))]$$

$$\text{Domain classification} \left[\begin{array}{l} \mathcal{L}_{cls}^f = \mathbb{E}_{x,c} [-\log D_{cls}(c|G(x, c))] \\ \mathcal{L}_{cls}^r = \mathbb{E}_{x,c'} [-\log D_{cls}(c'|x)] \end{array} \right]$$

$$\text{Reconstruction } \mathcal{L}_{rec} = \mathbb{E}_{x,c,c'} [\|x - G(G(x, c), c')\|_1]$$

$$\text{최종 목적 함수} \left[\begin{array}{l} \mathcal{L}_D = -\mathcal{L}_{adv} + \lambda_{cls} \mathcal{L}_{cls}^r \\ \mathcal{L}_G = \mathcal{L}_{adv} + \lambda_{cls} \mathcal{L}_{cls}^f + \lambda_{rec} \mathcal{L}_{rec} \end{array} \right]$$

Adverarial Loss (적대적 손실)

$$\mathcal{L}_{adv} = \mathbb{E}_x [\log D_{src}(x)] + \mathbb{E}_{x,c} [\log (1 - D_{src}(G(x, c)))]$$

- D는 image가 **Real**로 판별된다면 1에 가깝게 출력을 낼 것
반대로, **Fake**로 판별된다면 0에 가깝게 출력을 낼 것
 - 0~1 사이에서 출력이 나오는 이유: D는 확률값을 가지기 때문!
- Loss는 작을 수록 좋으므로,
왼쪽의 $D_{src}(x)$ 는 1에 가깝게 판별하도록 학습하고, 오른쪽의 $1 - D_{src}(G(x, c))$ 는 전체에서
G가 Fake로 판별할 확률을 뺀 값이므로 1에 가깝게 판별하도록 학습한다면 Loss가 최소화될 것!
 - log 함수는 정의역이 1일 때 0, 0일때 $-\infty$ 로 발산함

Domain Classification Loss (도메인 분류 손실)

- 주어진 input image x 와 target domain label c에대해서,
x가 output image y로 변환되었을때, 그것이 target domain c로 분류되는 것이 목적.
그러기위해서 D와 G를 optimize(최적화)할 때 domain classification loss를 첨가한다.

1) Domain classification loss of real images used to optimize D

(D 최적화에 사용된 실제 이미지의 도메인 분류 손실)

$$\mathcal{L}_{cls}^r = \mathbb{E}_{x,c'} [-\log D_{cls}(c'|x)]$$

- $D_{cls}(c'|x)$ 는 real image x가 주어졌을때 D가 계산해낸 domain label c'일 확률분포
- Loss를 최소화함으로써, D는 real image x를 그것에 대응되는 original domain c'로

분류시키는 것을 학습함

2) Domain classification loss of fake image used to optimaize G

(G 최적화에 사용된 가짜 이미지의 도메인 분류 손실)

$$\mathcal{L}_{cls}^f = \mathbb{E}_{x,c}[-\log D_{cls}(c|G(x,c))]$$

- $D_{cls}(c|G(x,c))$ 는 x 와 target domain label을 가지고 만들어낸 이미지인 $G(x,c)$ 가 주어졌을때,
target domain label c 일 확률분포
- G 는 target domain c 로 분류되어질 수 있는 이미지를 생성하도록 Loss를 최소화 하려고 함

Reconstruction Loss (재건 손실)

- 위에서 소개한 Loss들 만으로는 input image의 target domain에 관련된 부분만을 변화시킬 때 input image의 본래 형태를 잘 보존 할 수 없기 때문에 **Reconstruction Loss** 등장

$$\mathcal{L}_{rec} = \mathbb{E}_{x,c,c'}[||x - G(G(x,c), c')||_1]$$

- G 는 변환된 image $G(x,c)$ 와 original doamin label c' 을 input으로 받고 original image x 를 reconstruction 하는데, 이때 original image x 와 reconstruction된 image의 손실을 최소화
- L1 norm

Objective Function

- 최종 G 와 D 에 대한 Objective Function

$$\mathcal{L}_D = -\mathcal{L}_{adv} + \lambda_{cls} \mathcal{L}_{cls}^r,$$

$$\mathcal{L}_G = \mathcal{L}_{adv} + \lambda_{cls} \mathcal{L}_{cls}^f + \lambda_{rec} \mathcal{L}_{rec}$$

- λ_{cls} 와 λ_{rec} 는 hyperparameter로,
domain classification loss와 reconstruction loss의 상대적인 중요도를 컨트롤함

Training with Multiple Datasets

- StarGAN의 중요한 이점은, 다른 domain을 가진 datasets들을 동시에 포함하는 것
그러나 다수의 dataset들을 학습시킬 때 label 정보가 각 dataset에 부분적으로만 있다는 문제

- ex) Celeb A & RaFD dataset

Celeb A: 머리색, 성별, 나이와 같은 facial attribute(얼굴 속성)과 관련된 40개의 label만을

가지고 있으므로, happy 와 angry 같이 facial expression(얼굴 표정)과 관련된

label은 가지지 않음

RaFD: happy 와 angry 같이 facial expression(얼굴 표정)과 관련된 8개의 label만을

가지고 있으므로, facial attribute(얼굴 속성)과 관련된 label은 가지지 않음

- 문제가 되는 이유

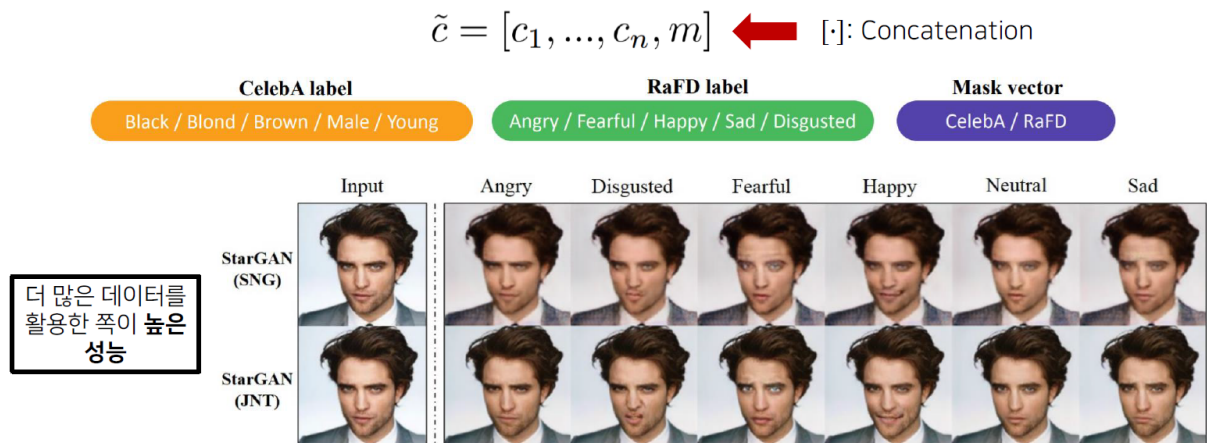
변환된 image인 $G(x,c)$ 로부터 input image x 를 reconstruction하려면

label vector c' 에 완전한 정보가 있어야하기 때문!

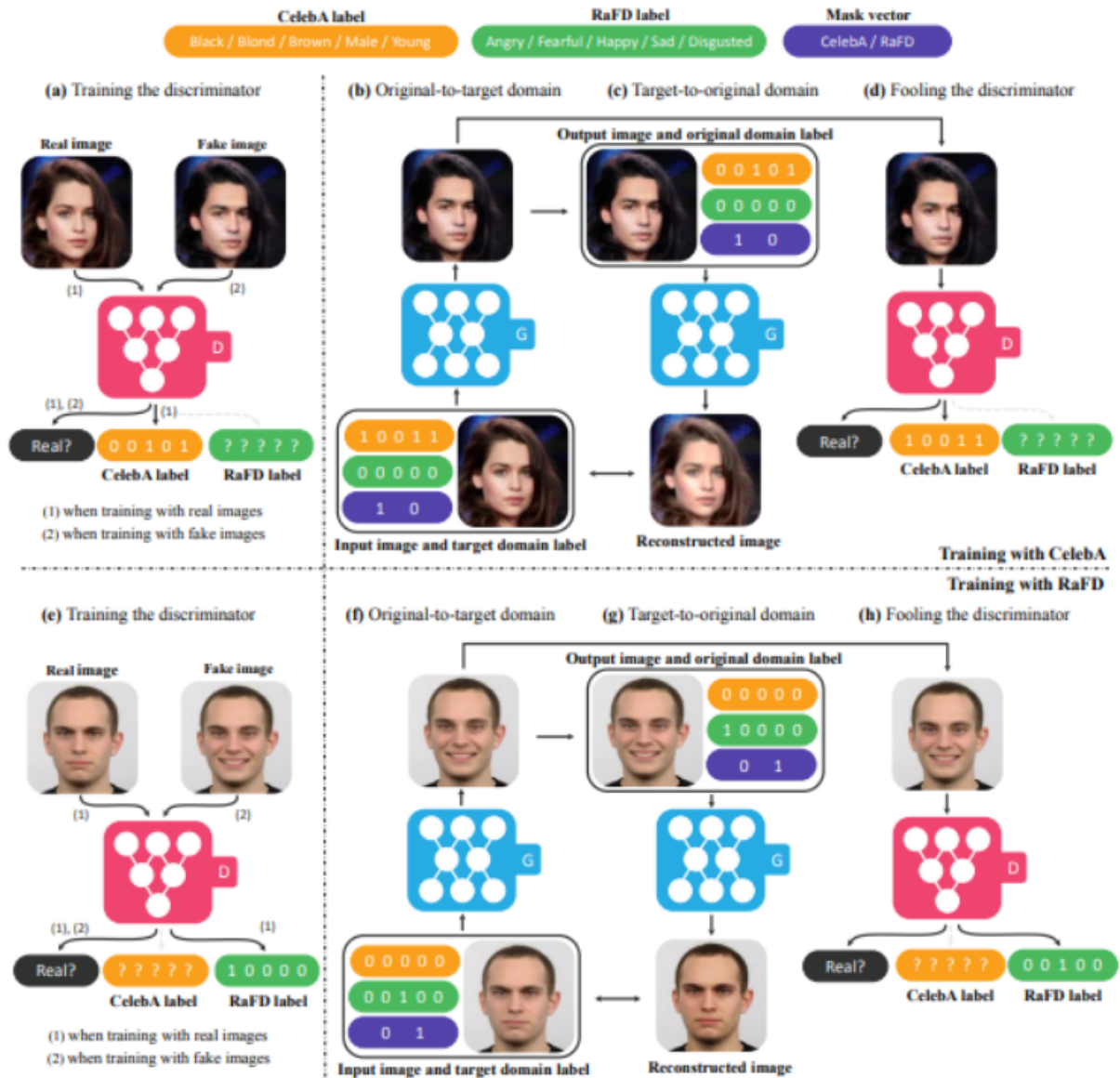
→ Mask Vector m 을 통해 해결!

Mask Vector

- StarGAN이 명시되지 않은 label에 대해서는 무시하고, 명시된 label에 대해서 집중하게 함
- n차원의 one-hot vector 사용! (n: dataset의 수)
 - ex) Celeb A와 RaFD 2개의 dataset을 사용하면, n=2



- C_i : i번째 dataset의 label들의 vector
 - binary attribute(성별) → binary vector
 - categorical attribute(머리색, 나이) → one-hot vector
- Mask vector에 어떤 dataset인지를 명시해줌으로써, 해당 dataset의 attribute label에 집중시킴
 - CelebA 학습을 위해 명시해 주었다면 RaFD에 관련된 facial expression들은 무시하고 학습



- StarGAN이 CelebA와 RaFD dataset을 가지고 학습할때의 모습을 보여주는 그림
- CelebA의 binary attribute(black,blond,brown,male,young)에 대한 label은 binary vector로 표현
반대로 RaFD의 categorical attribute(Angry, Fearful, Happy, Sad, and Disgusted)에 대한 label은 one-hot vector로 표현
- mask vector는 2차원 one-hot벡터로 CelebA와 RaFD중 valid 한 것을 가리킨다.
- CelebA와 RaFD를 교차시킴으로써 Discriminator D는 두 dataset에서 차이를 구분짓는
모든 feature들을 학습하게 되고, Generator G는 모든 label을 컨트롤하는 것을 학습

