TEAM PPAC

ANALYZING GANS FOR FEATURE TRANSLATION

- Anshul Verma (210050015)
- Chaitanya Garg(210050039)
- Priyanshu Yadav(210050125)
- Pulkit Goyal(210050126)

INTRODUCTION

- Aim of our project is to analyze and compare state-of-the-art models for feature translation in human faces.
- Feature translation falls in the category of image to image translation, and the most widely used models for such tasks are Generative Adversarial Networks.

GENERATIVE ADVERSARIAL NETWORKS

- Generative modeling is a unsupervised learning task in machine learning that involves automatically discovering and learning the regularities or patterns in input data in such a way that the model can be used to generate or output new examples that plausibly could have been drawn from the original dataset.
- GANs are a clever way of training a generative model by framing the problem as a supervised learning problem.
- It contains two sub-models: the generator model that we train to generate new examples, and the discriminator model that tries to classify examples as either real (from the domain) or fake (generated). They act as adversaries. Generator tries to create an image that looks realistic whereas Discriminator tries to find fake images. They are used to train each other, both getting better with time.

TYPES OF GANS USED

- StarGAN
- Attention Gan
- AttGAN

StarGAN

- StarGAN can be used to translate different features in an image by training a single model
- It has 3 types of losses in the overall loss function for Generator and Discriminator.
- A**dversarial loss** : Used in both Generator loss and Discriminator loss. It is useful in generating realistic images.
  - Lossadv = Ex [log Dsrc (x)] + Ex,c [log (1-Dsrc (G(x,c)))]
  - Dsrc gives a probability distribution over sources given by D.
  - The generator G tries to minimize the objective, while the discriminator D tries to maximize it.

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

Domain Classification Loss : Used for improving translation of required features. L_cls for fake images is used in Generator whereas L_cls for real images is used in Discriminator.

This loss is mean of the cross-entropy loss over the batch of images. **Fake image classification loss** is used to improve performance of generator whereas **real image classification** loss is for improving discriminator.

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

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

Reconstruction Loss : Used in Generator to keep other features unchanged, so that generated image looks similar to real image.

The generator first generates a fake image from real image x conditioned on target domain c, G(x,c). Then this fake image is again passed through generator, this time conditioned on original domain c‘. This should bring back the original image (or very close to it).

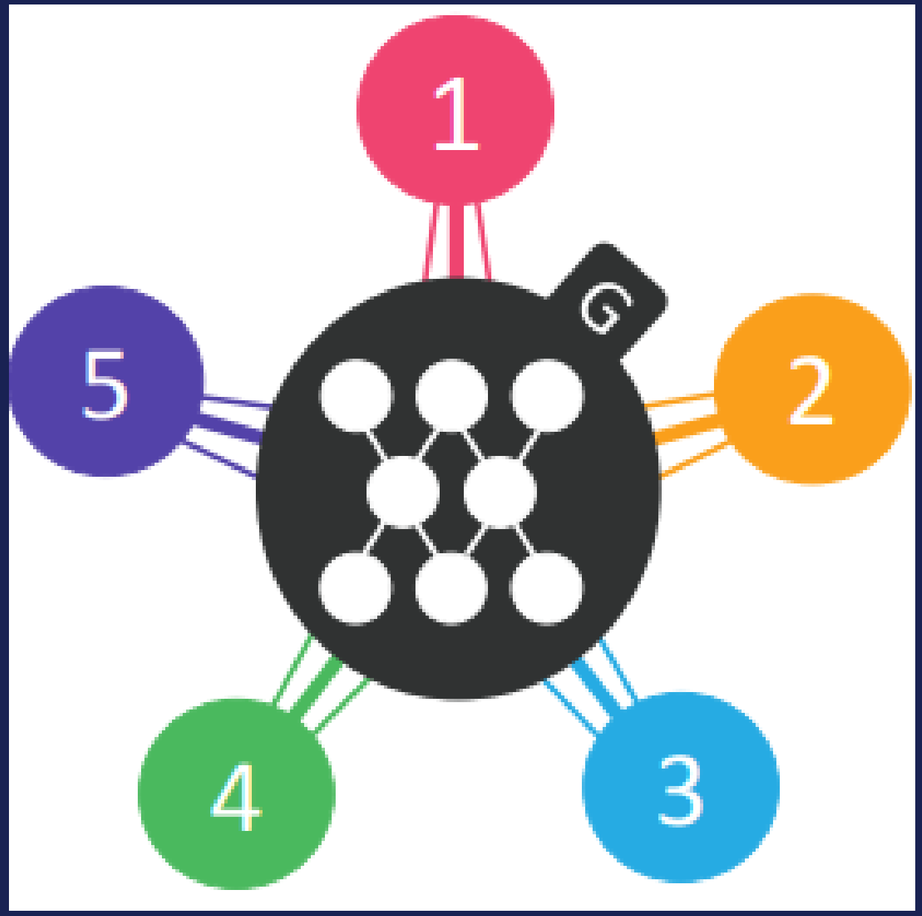
$$\mathcal{L}_{rec} = \mathbb{E}_{x,c,c'}[||x - G(G(x,c),c')||_1],$$

Total loss for Generator and Discriminator for StarGAN :

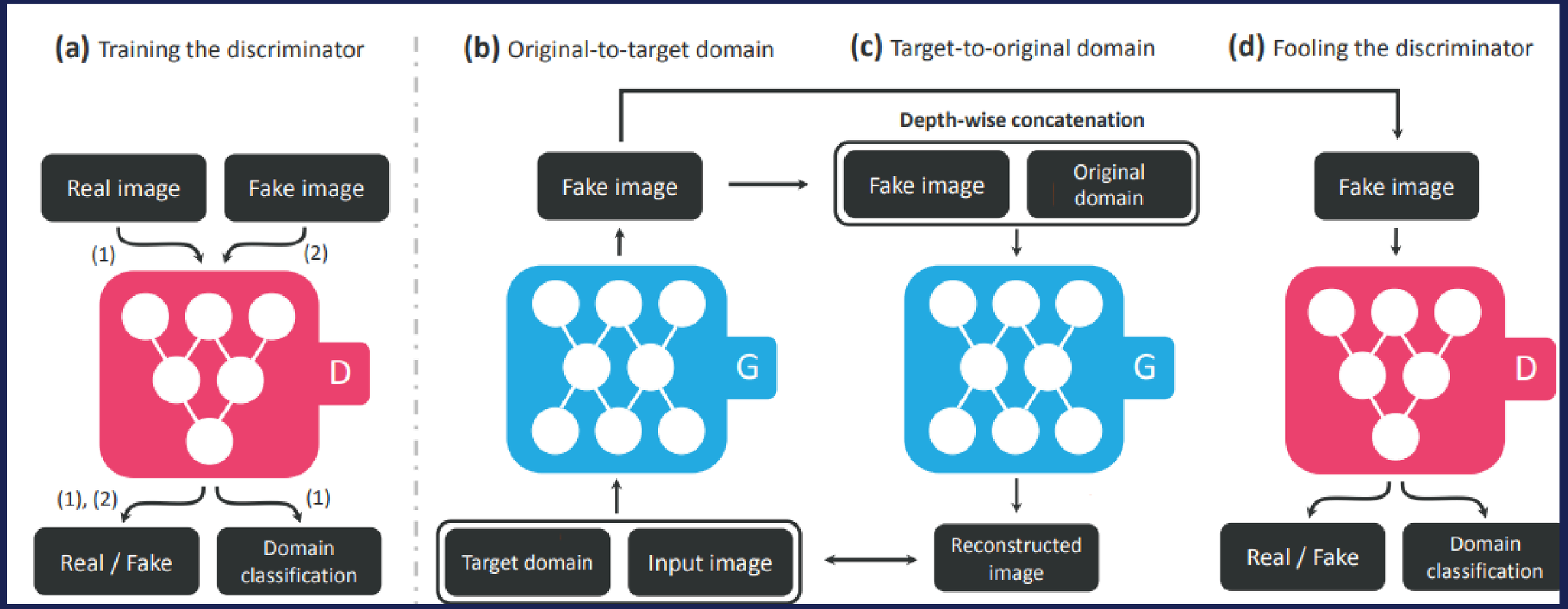
$$\mathcal{L}_D = -\mathcal{L}_{adv} + \lambda_{cls}\,\mathcal{L}^r_{cls},$$

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

StarGAN overview



StarGAN overview



AttentionGAN

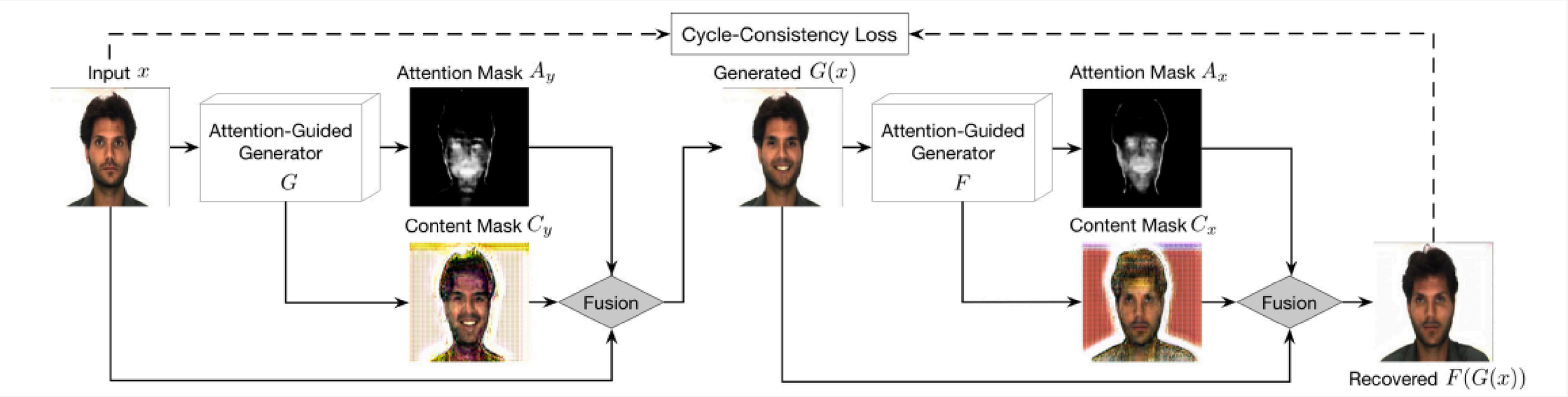
Attention GANs are another type of GANs for translating feature in real images to generate fake images. The defining characteristic of AttentionGANs is that its generator outputs attention and content masks which in combination with real image are used to generate fake image whereas in other GANs, the generator directly outputs the fake image.

For a given image x, generator G attention masks A_y and content mask C_y which are then used to create fake image as follows:

$$G(x) = C_y * A_y + x * (1 - A_y),$$

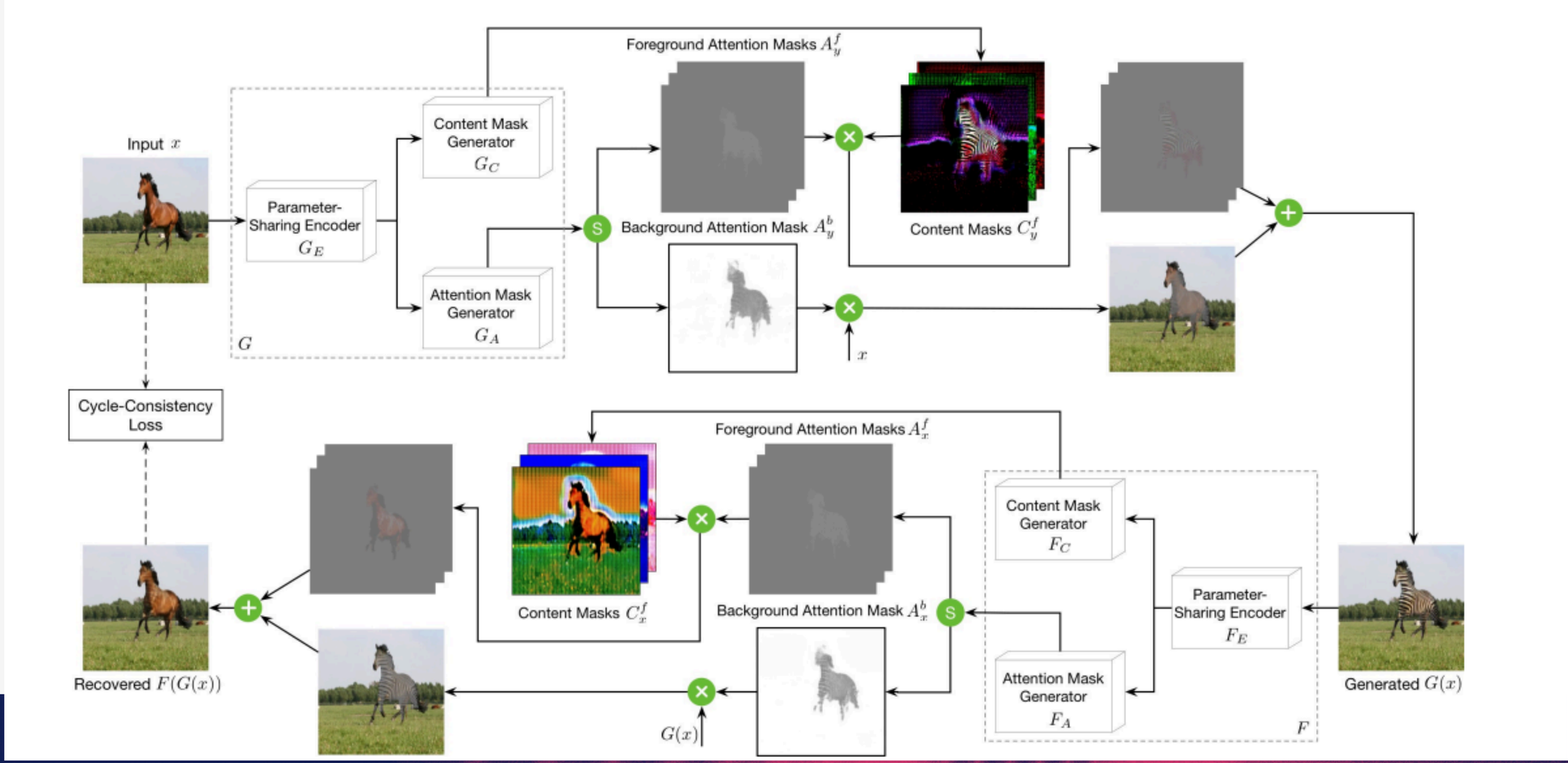
Type 1 Attention GAN : basic approach

$$G(x) = C_y * A_y + x * (1 - A_y),$$



Type 2 Attention GAN : advanced approach

$$G(x) = \sum_{f=1}^{n-1} (C_y^f * A_y^f) + x * A_y^b.$$



AttributeGAN

AttributeGAN is different in that it used an encoder-decoder structure for its Generator model.

For a given image x_a (with origin domain a), it is passed through the encoder to create a latent image z. This latent image z preserves other features and removes feature a from the image. Then z is fed in the decoder along with the target domain b to generate x_b', the fake image with feature b in place of a.

$$\mathbf{z} = G_{enc}(\mathbf{x^a}).$$

$$\hat{\mathbf{x^b}} = G_{dec}(\mathbf{z}, \mathbf{b}),$$

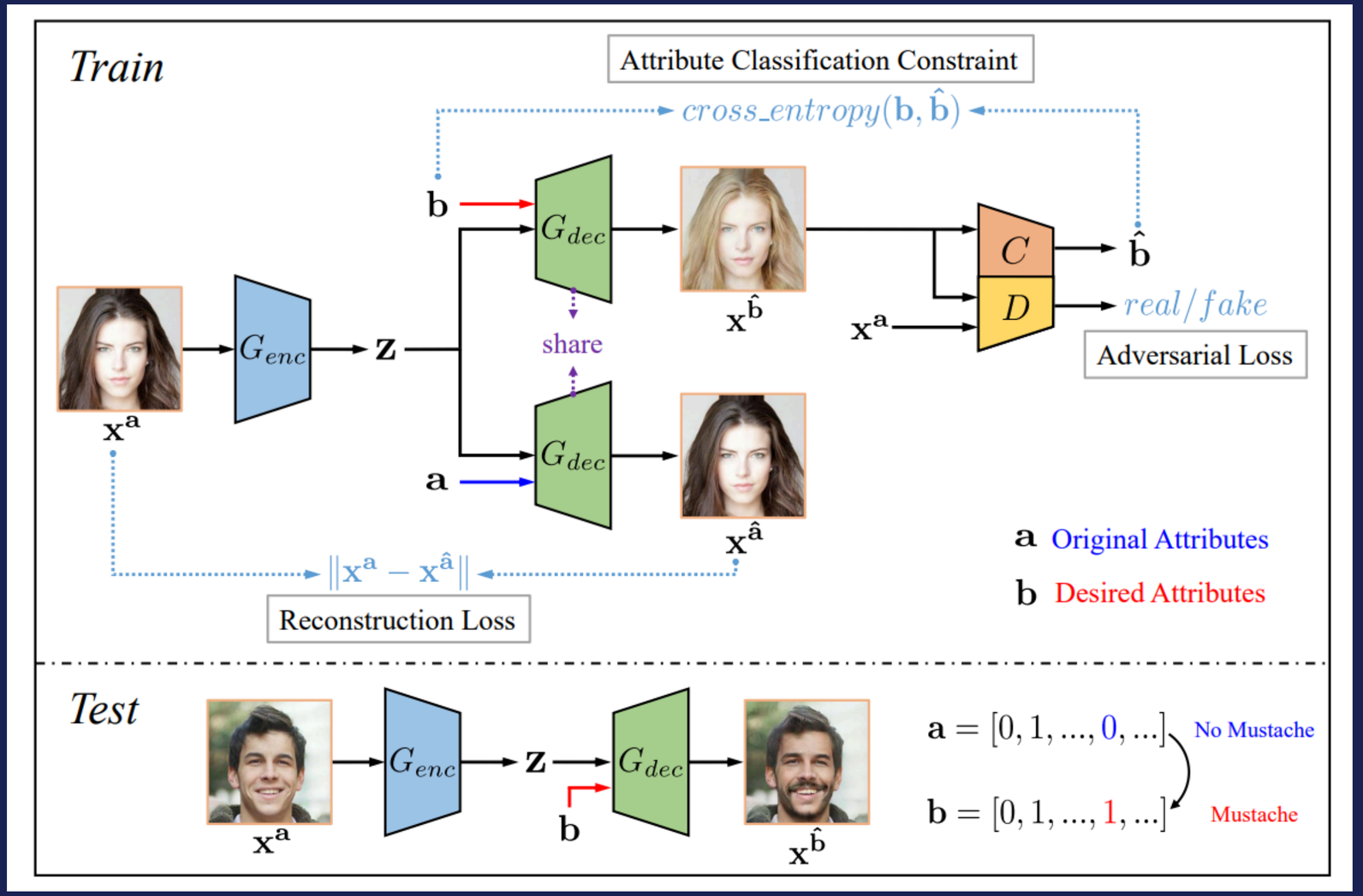
$$\hat{\mathbf{x^b}} = G_{dec}(G_{enc}(\mathbf{x^a}), \mathbf{b}).$$

Reconstruction loss in attGAN is calculated differently compared to other GANs. For this loss, we first pass the image x_a through the decoder, G_enc to get z and then pass z along with a through G_dec to obtain the reconstructed version of original image.

Note that in other GANs, we passed the *FAKE image in Generator*, instead of the latent image in the decoder.

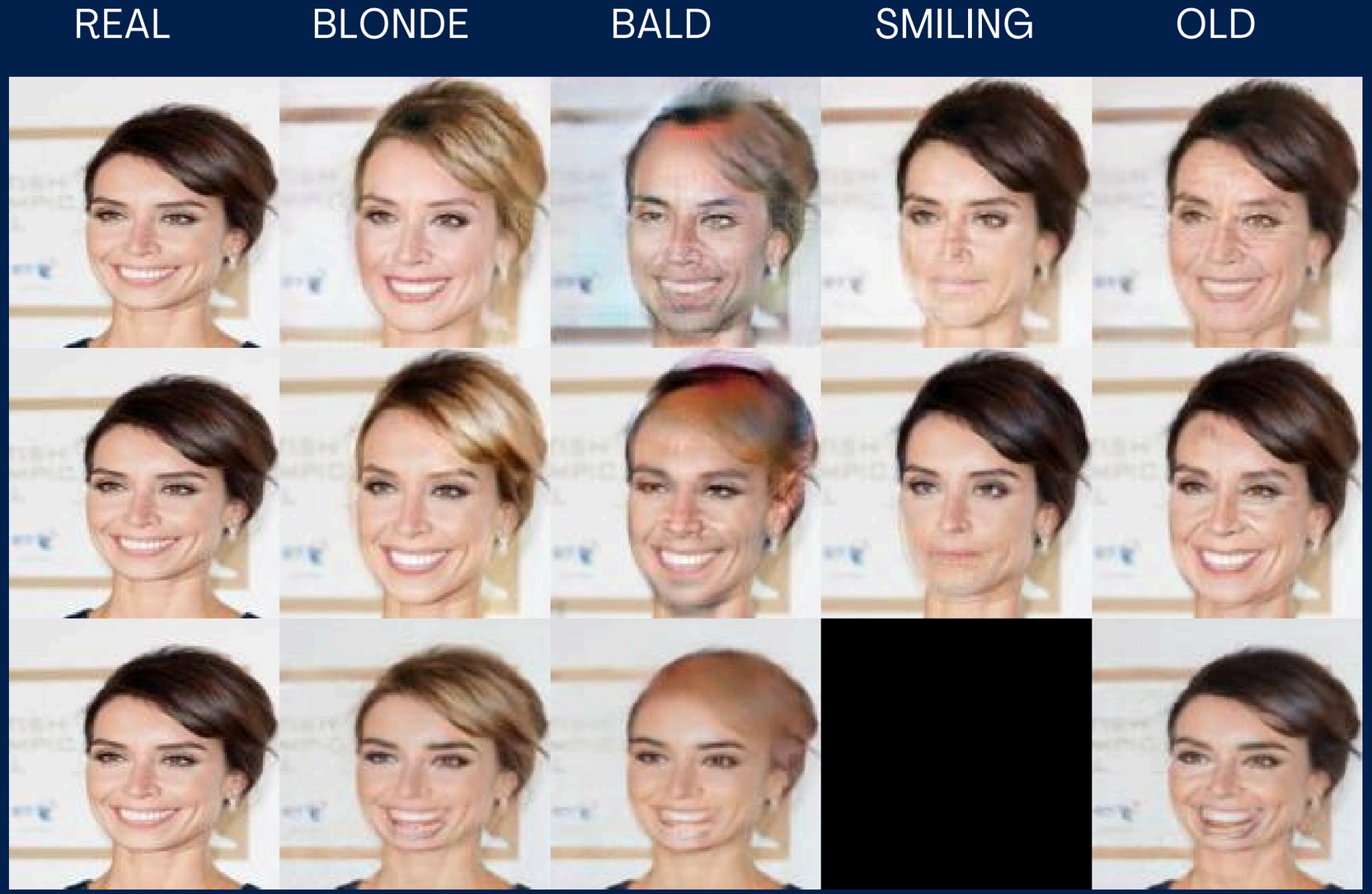
$$\min_{G_{enc},G_{dec}} \mathcal{L}_{rec} = \mathbb{E}_{\mathbf{x^a}\sim p_{data}}[\|\mathbf{x^a} - \mathbf{x^{\hat{a}}}\|_1],$$

AttGAN overview



RESULTS

Visual Comparison on CelebA data



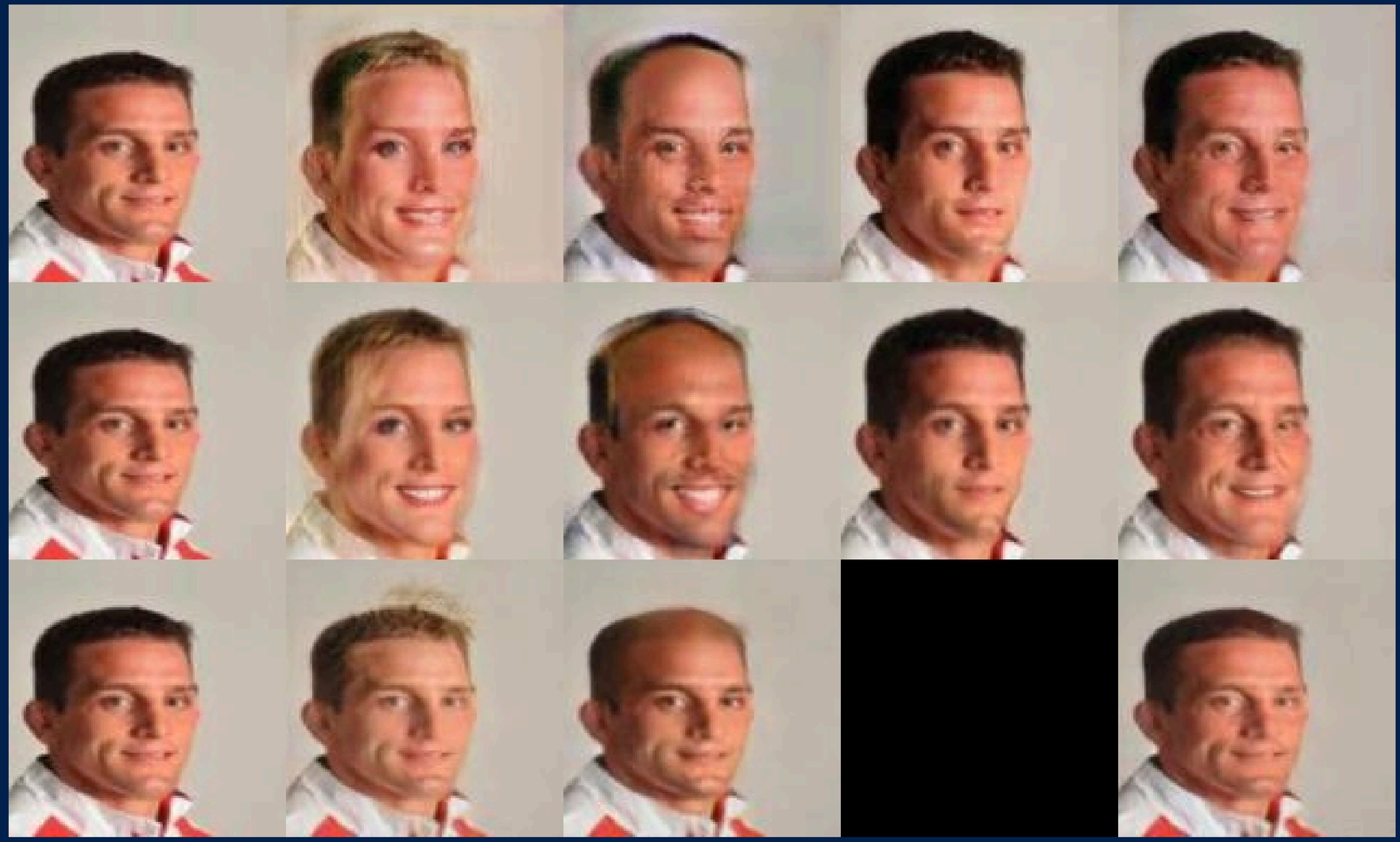
Visual Comparison on CelebA data

REAL BLONDE BALD SMILING OLD

STAR-GAN

ATTENTION GAN

ATT-GAN



Quantitative comparison on CelebA data

Generator Loss (for fixed Discriminator)

	BLONDE	BALD	SMILING	YOUNG
StarGAN	39.872	42.233	42.051	42.119
AttGAN	40.895	41.639		42.086
AttentionGAN	40.435	42.181	42.445	42.493

Quantitative comparison on CelebA data

FID(Frechet Inception Distance)

	BLONDE	BALD	YOUNG
StarGAN	566	624	419
AttGAN	417	482	323
AttentionGAN	512	554	351

Performance of models on out-of-domain data

StarGAN

AttGAN

AttentionGAN

THANK YOU