FA-GAN: Face Augmentation GAN for Deformation-Invariant Face Recognition

Mandi Luo, Jie Cao, Xin Ma, Xiaoyu Zhang, and Ran He, *Senior Member, IEEE*

2018312824
Ryu Chaeeun

Introduction

FA-GAN: Network to **augment** the existing datasets by generating faces **with various deformations.**

<Expression>

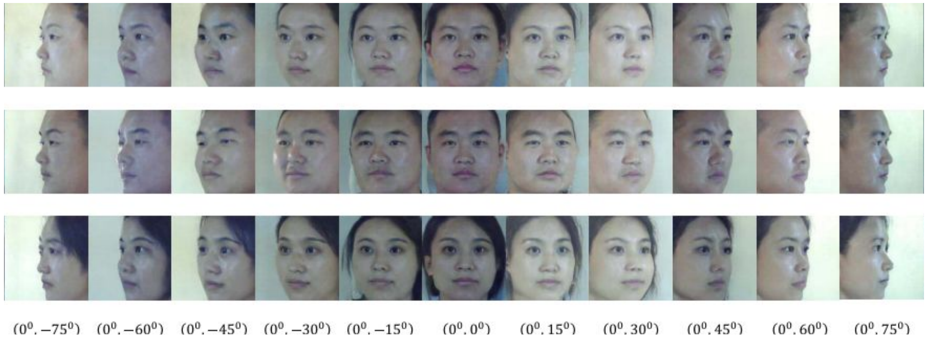
Deformation: Normal -> Happy

Input image (Normal)

Generated image (Happy)

Target image (Happy)

<Poses>



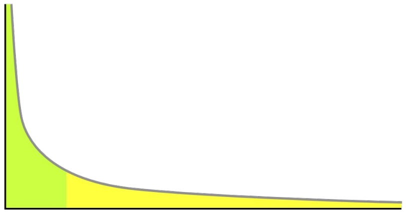
Background

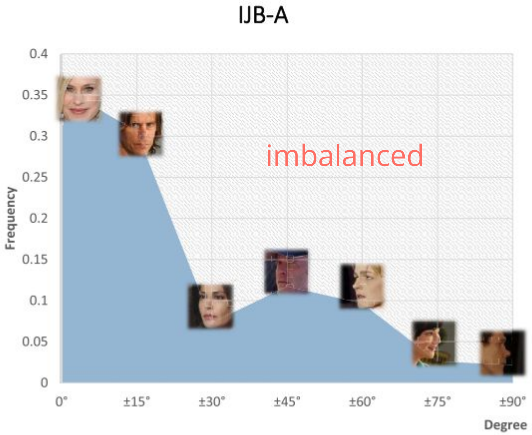
- **Long tail data distribution -> strong biases**

- **Limitations of precious methods when synthesizing face data: Geometry distortion in the synthesized faces.**

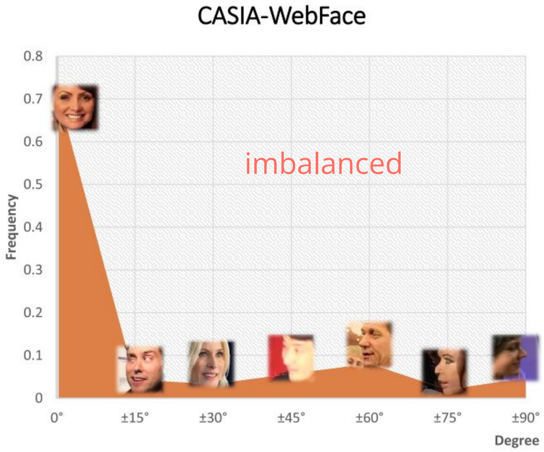
Background

- Long tail data distribution
  -> strong biases
  -> overfitting problem



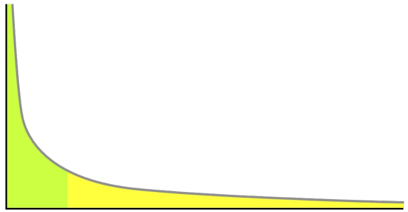


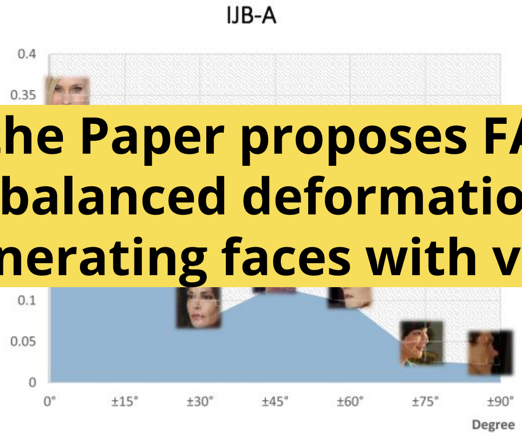
CASIA-WebFace



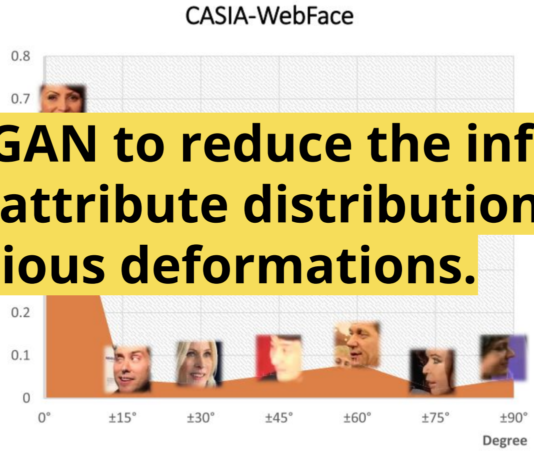
Background

- Long tail data distribution
  -> strong biases
  -> overfitting problem





CASIA-WebFace



⇛ the Paper proposes FA-GAN to reduce the influence of imbalanced deformation attribute distributions by generating faces with various deformations.

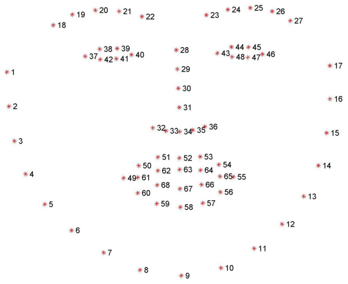
Background

- **Long tail data distribution -> strong biases**

- **Limitations of precious methods when synthesizing face data: Geometry distortion in the synthesized faces.**

<previous methods>

: Use mean facial landmarks



Neglect the fact that **landmarks** are **identity-dependent**.

Geometry distortion

Background

- **Long tail data distribution -> strong biases**

- **Limitations of precious methods when synthesizing face data: Geometry distortion in the synthesized faces.**

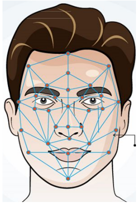
(a) (b)

Face parsing map

I_p

Contains:

- **Semantic information**
- **Spatial distribution of face regions**



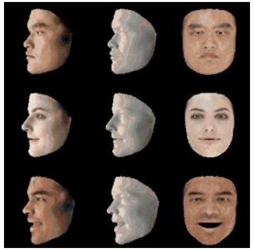
Better store **high-level** information for image generation

Goals & Approach

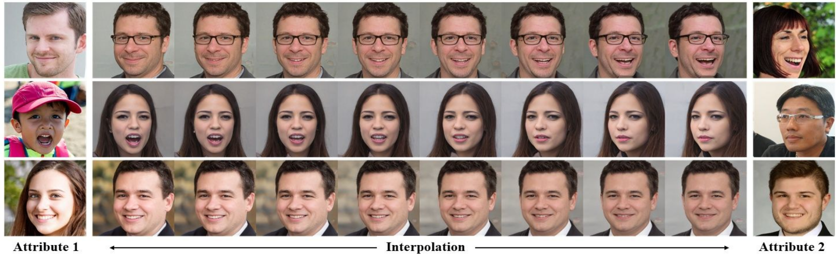
Goals of FA-GAN:
1. Learn efficient identity representations for face recognition
2. Augment face datasets with customized deformation demands

Trained in a supervised manner

1. **Geometry Preserving Module (GPM)**
   : learn identity-dependent geometry information by exploring spatial and semantic relations of different face regions.

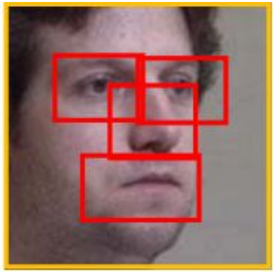


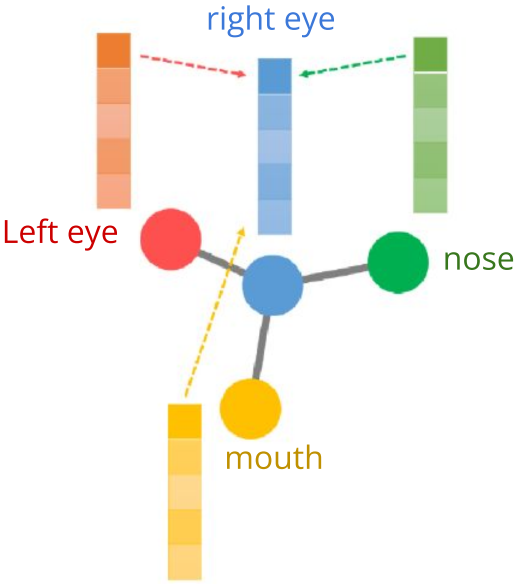
2. **Face Disentanglement Module (FDM)**
   : Disentangled representation learning.
   Disentangles deformation attributes representations from identity representations.



Network Architecture - GPM

GPM (Geometry Preserving Module)
Core: Use of Graph Convolutional Networks(GCNs)





G = {V,E}

Stack 2 layers of GC

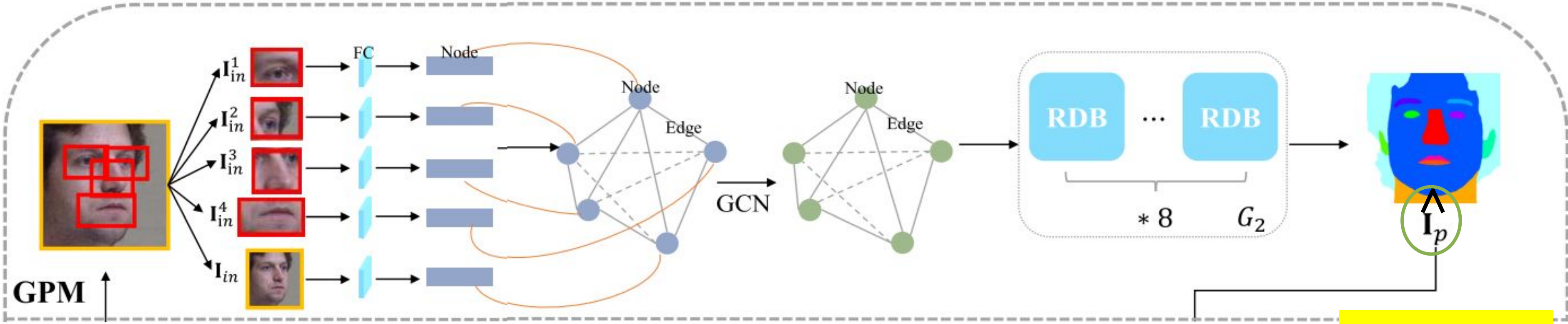
$$\mathbf{G = MXW} \longrightarrow \mathbf{G' = MXW_1W_2}$$

M: adjacency matrix (NxN)
X: input features (N x p)
W: weight matrix (pxq)

G = (Nxq)

Network Architecture - GPM

GPM (Geometry Preserving Module)



Crop strategy

Embed as nodes

GCN

Residual Dense Block

Normalized face parsing map

< supervised > $\mathcal{L}_p = ||\hat{\mathbf{I}}_p - \mathbf{I}_p||_1.$

Network Architecture - FDM

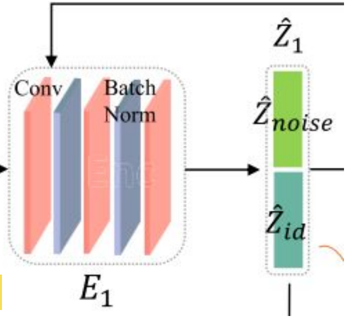
FDM (Face Disentanglement Module)
: Used to learn the representations of different identities and deformation attributes to further combine them freely. (These representations are tangled with each other in face images.)

-> Hierarchical Disentanglement

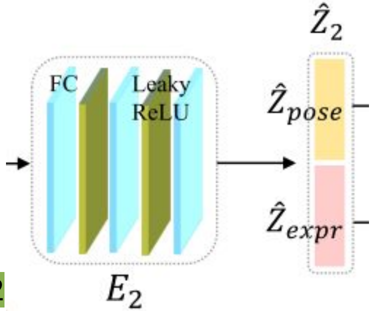
- **Level 1.** Learn identity feature embedding
  : Disentangle identities from “noise” (= nuisance factors ex. background)

- **Level 2.** Attribute embedding learning
  : Disentangles deformation attribute representations (pose and expression).

Level 1

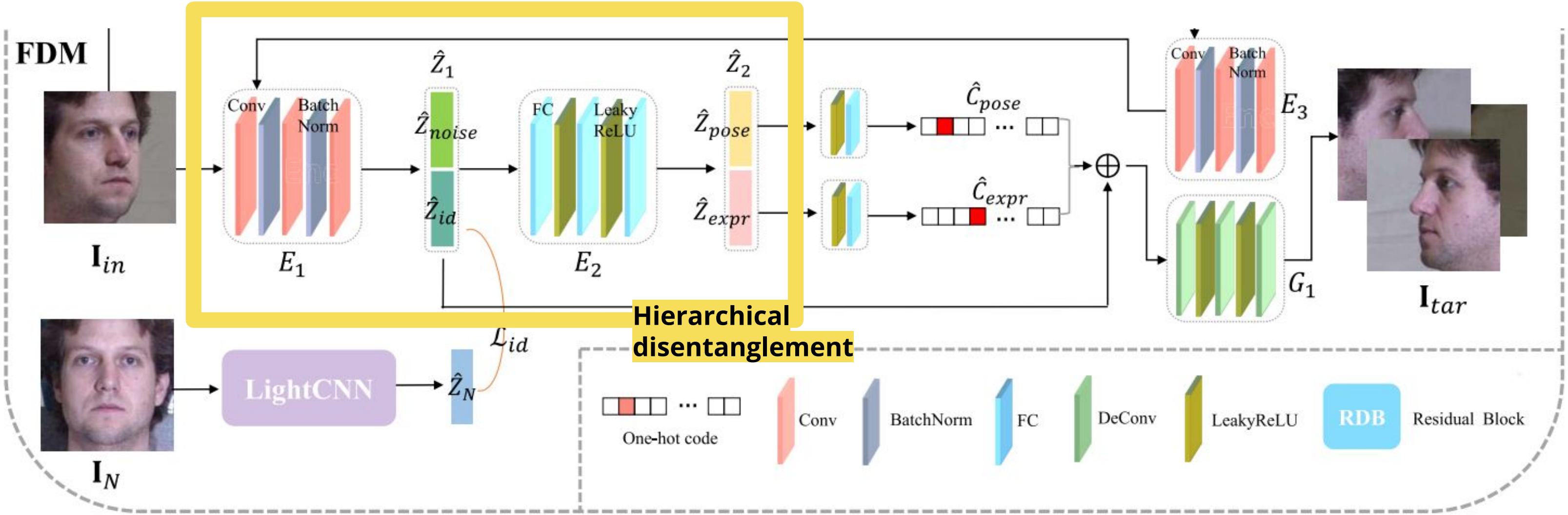


Level 2



Network Architecture - FDM

FDM (Face Disentanglement Module)

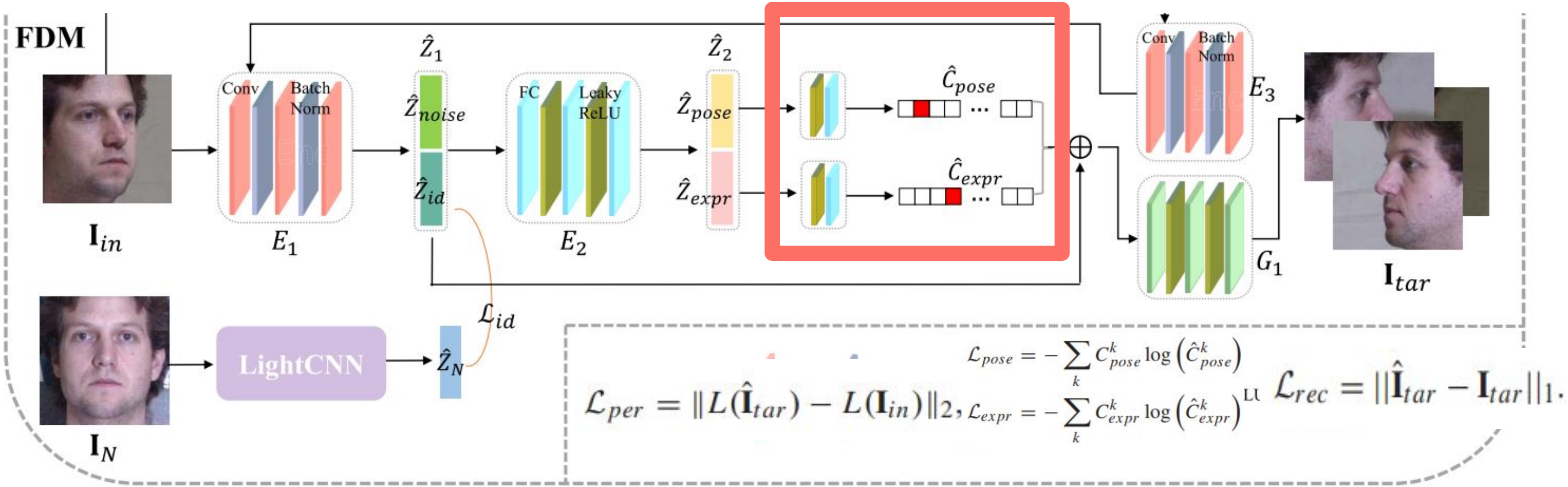


< supervised > $\mathcal{L}_{id} = ||\hat{Z}_{id} - L(\mathbf{I}_N)||_2$ (Same way for level 2)

Network Architecture - FDM

FDM (Face Disentanglement Module)

One-hot code



Adversarial loss

Discriminator

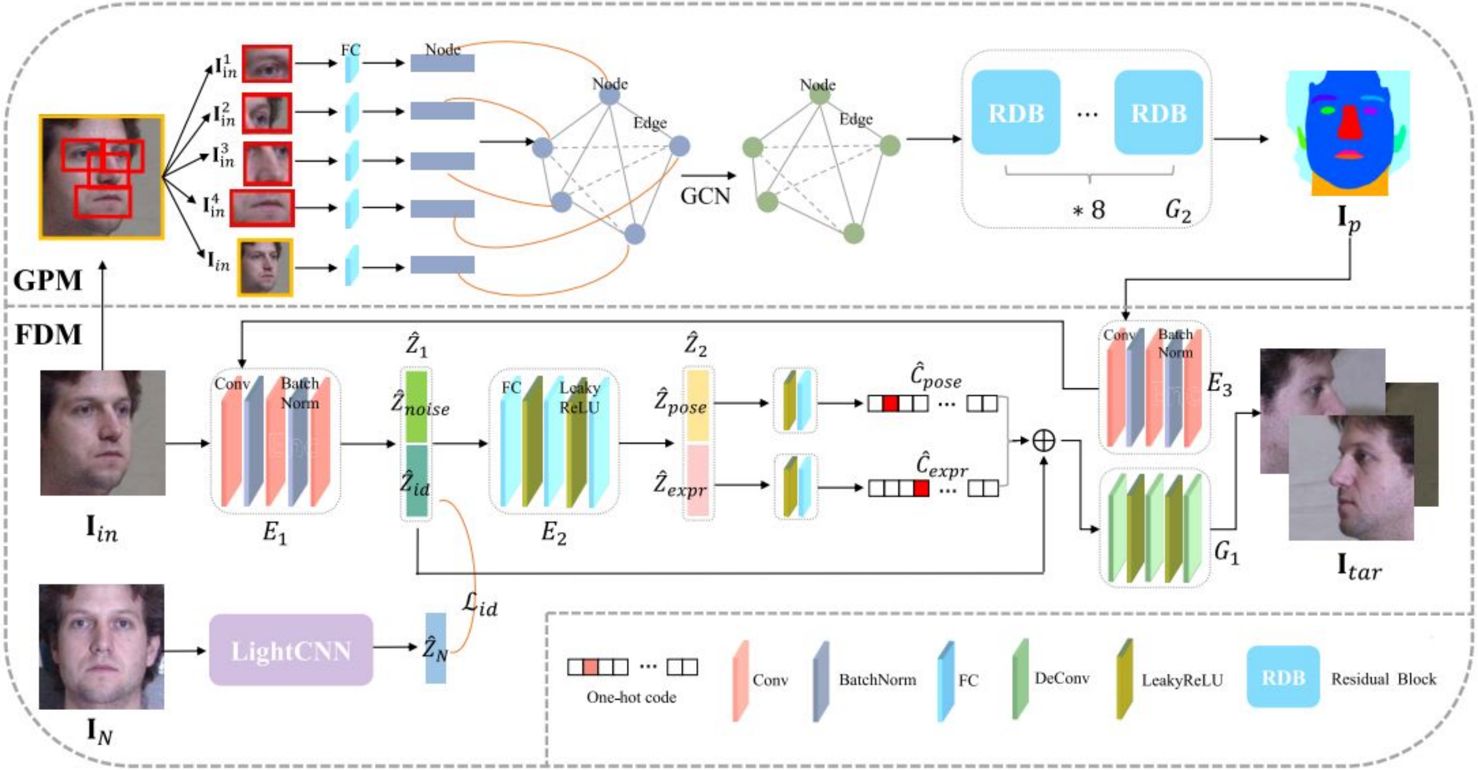
Generator

$$\mathcal{L}_{adv} = \mathbb{E}_{\mathbf{I}_{tar}\sim\mathbb{P}_{\text{train}}(\mathbf{I}_{tar})}\left[\log \boxed{D\left(\mathbf{I}_{tar}\right)}\right] + \mathbb{E}_{\mathbf{I}_{in}\sim\mathbb{P}_G(\mathbf{I}_{in})}\left[\log\left(1 - \boxed{D\left(G_{\theta_G}\left(\mathbf{I}_{in}\right)\right)}\right)\right]$$

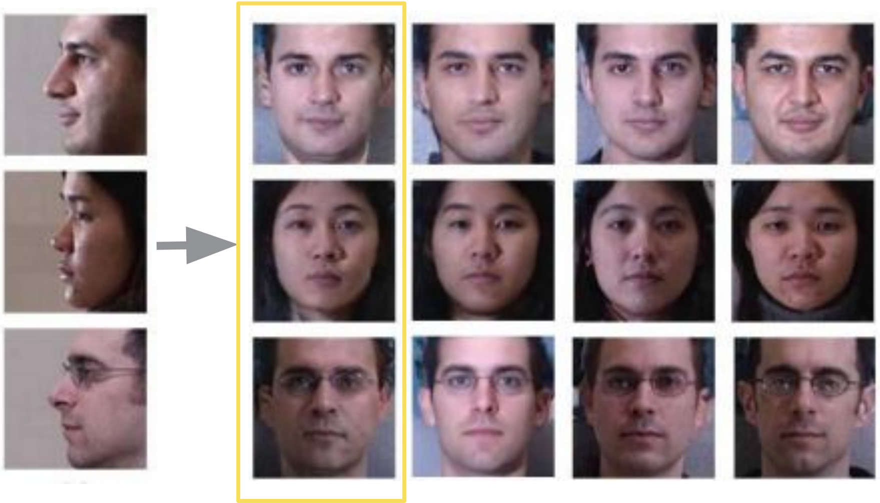
overall loss

$$\mathcal{L}_{total} = \lambda_{adv}\mathcal{L}_{adv} + \lambda_{rec}\mathcal{L}_{rec} + \lambda_{per}\mathcal{L}_{per} + \lambda_p\mathcal{L}_p + \lambda_{id}\mathcal{L}_{id} + \lambda_{pose}\mathcal{L}_{pose} + \lambda_{expr}\mathcal{L}_{expr},$$

Training entire network



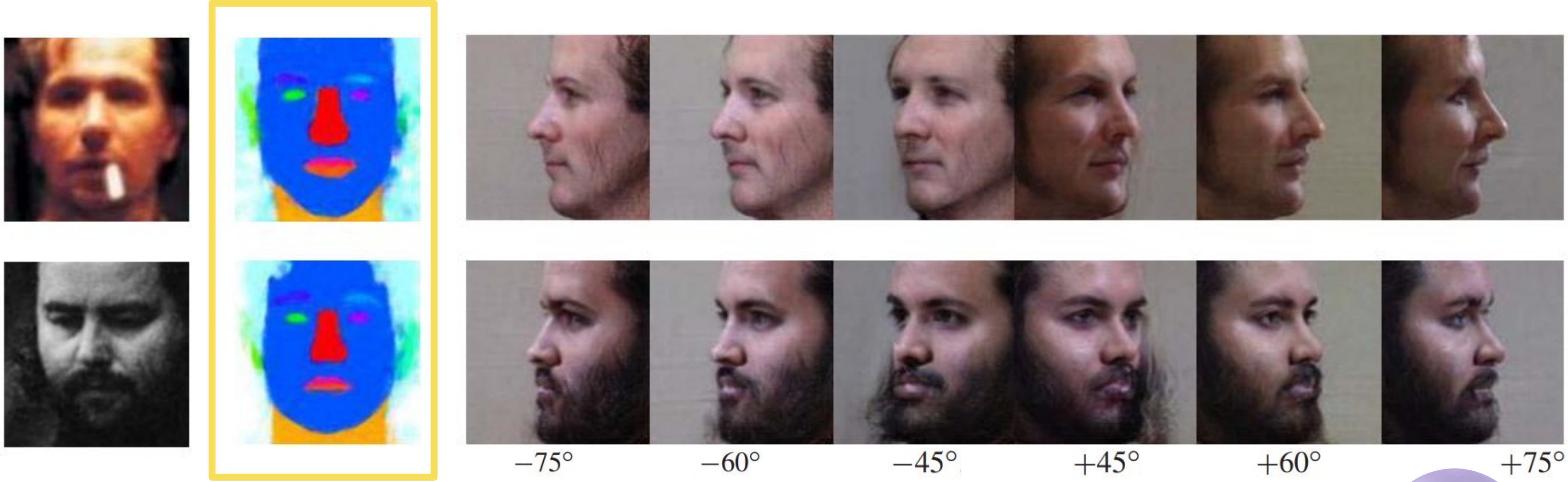
Performance



input

Performance

Same person



Very similar face parsing map