Dialog Driven Face Construction using GANs

32nd International Conference on Tools with Artificial Intelligence

AUTHORS

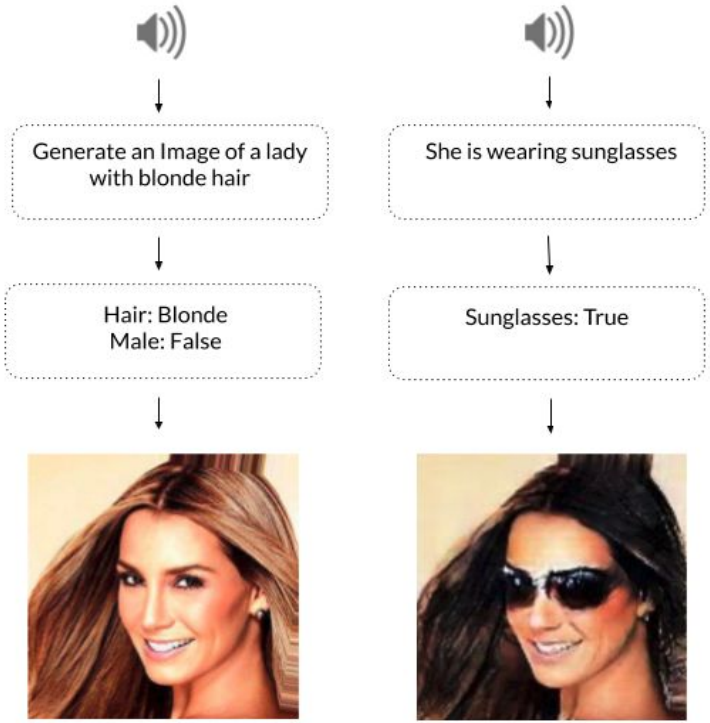
Malaika Vijay
Meghana
Nishant Aklecha
Dr. Ramamoorthy Srinath

PROBLEM STATEMENT

→ Given a speech based description of a face, generate an image that best approximates the description

→ Further improve the image by providing modifications to the base image through speech based dialog

→ Provide a conversational interface to the tool using speech based input



CONTRIBUTION AND USE CASES

→ An end-to-end pipeline for generation and manipulation of images of faces in multiple stages

→ An organic interface to the tool through speech

- Conversational dialog
- Facilitates ease of use without needing to understand how it works simulates real human interaction

→ A rule based parser to specifically designed to extract facial attribute descriptors from natural language text

→ Criminal Sketch Artists

- Automate the task of sketching suspect faces through conversation with a witness

→ Entertainment

- Creation of faces resembling a description

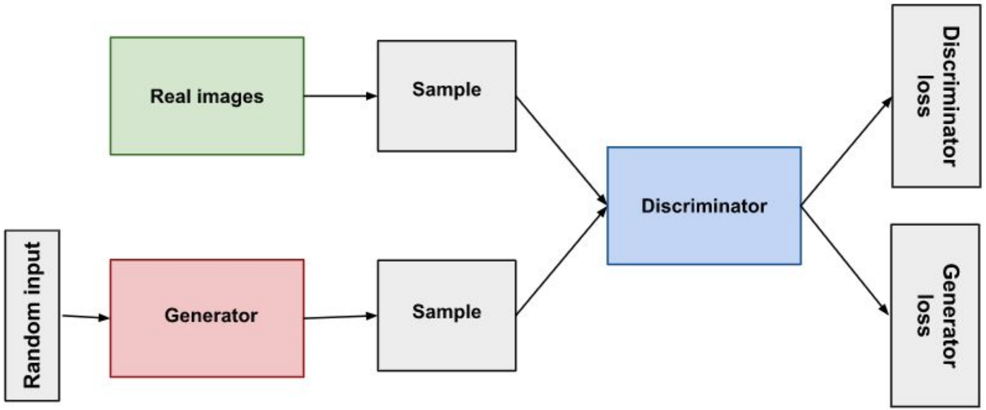
→ Advertisements

BACKGROUND - GENERATIVE ADVERSARIAL NETWORKS

GANs - Generative models that can create new instances resembling the training dataset

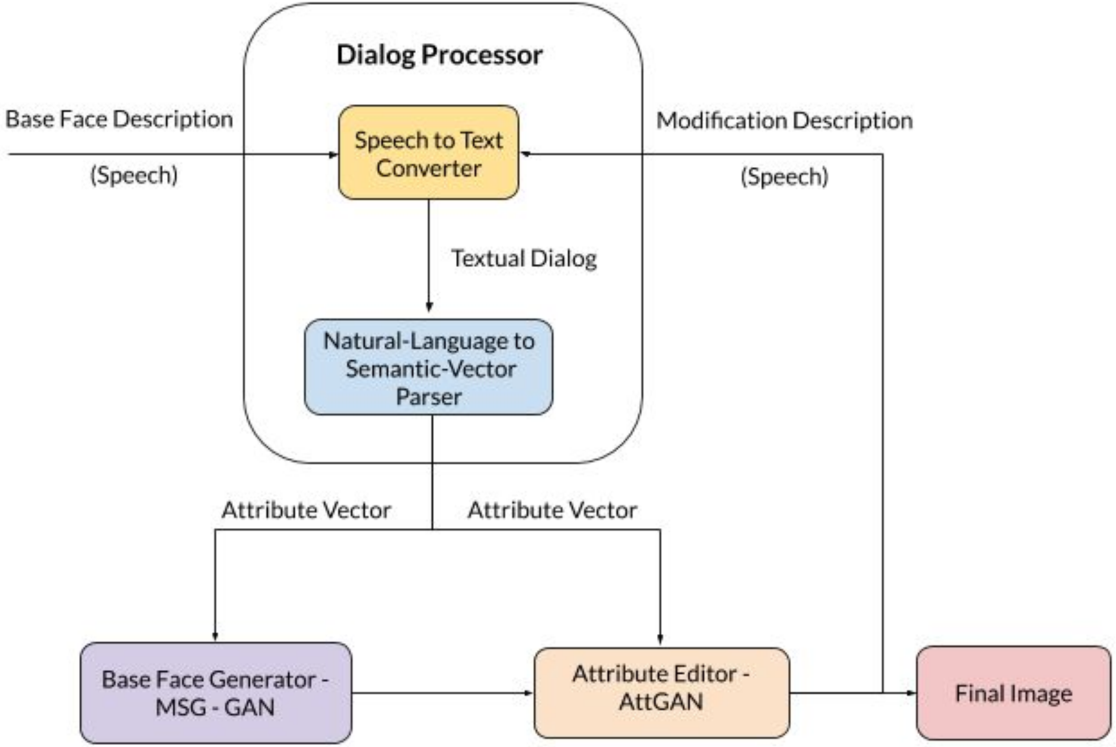
→ Generator - Generates new samples

→ Discriminator - Distinguishes between real and generated samples



Source: https://developers.google.com/machine-learning/gan/gan_structure

DESIGN DIAGRAM - PROPOSED APPROACH



DATASET DESCRIPTION

CelebFaces Attribute Dataset (Celeb-A)

→ 10,177 unique identities

  ○ Male:Female - 1:1.399

→ 202,599 images

→ 40 binary attribute annotations

Source: [CelebFaces Attribute Dataset]

NATURAL LANGUAGE PARSER

Natural Language Description

↓

Attribute-Value Pairs

↓

Attribute-Value Pairs contained in the dataset

“She has golden hair and is wearing glasses”

↓

{gender: female, hair: golden, glasses: true}

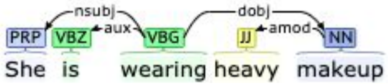
↓

{gender: female, hairColor: blonde, glasses: true}

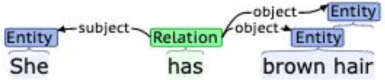
NATURAL LANGUAGE PARSER

→ Extract all attribute-value pairs from the raw text description

- ○ Dependency Annotations



- ○ Relation Extraction



amod dependencies → NN: attribute, JJ: value

"has" relations → object NN: attribute, object JJ: value

→ Map these pairs back to attributes and values enumerated in the dataset (e.g. cropped and short)

- ○ Synset extraction (e.g. spectacles and glasses)
- ○ Synset = (word, part of speech, sense)

Dictionary of recognised terms to their Synset

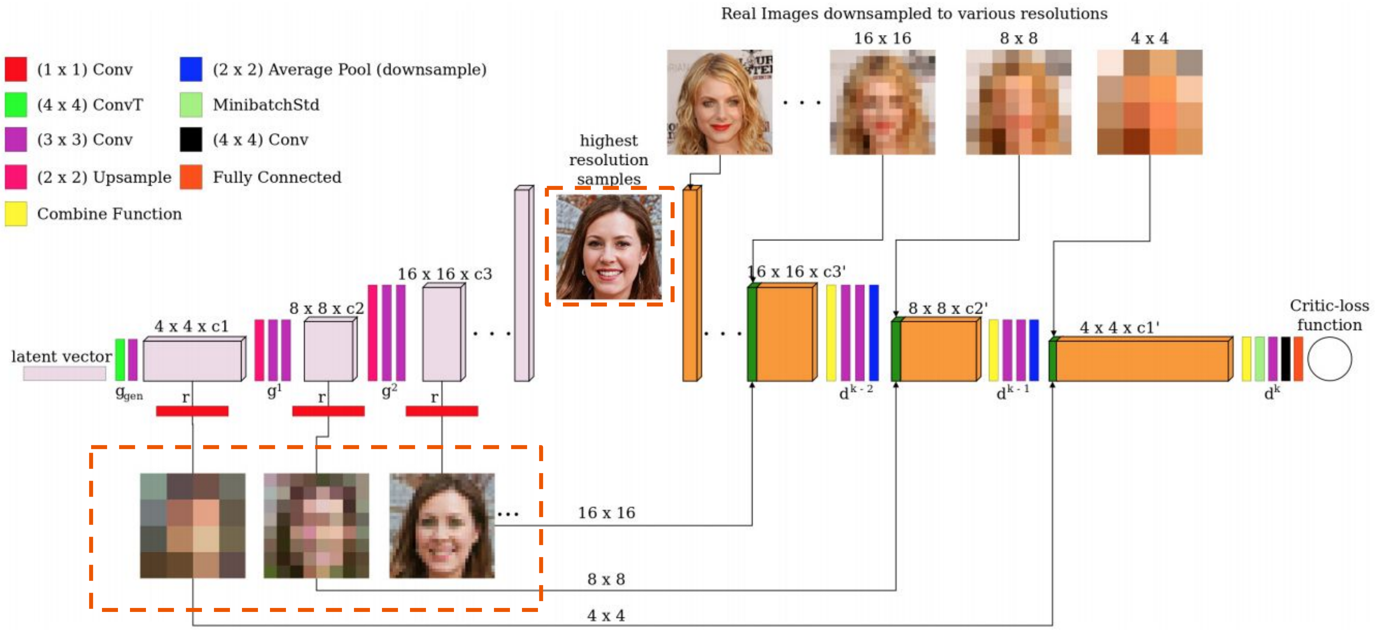
- Synset(Large) ∈ Synset(Big)

Dictionary of recognised terms to their Synonyms

- Cropped ∈ Synonyms(Short)

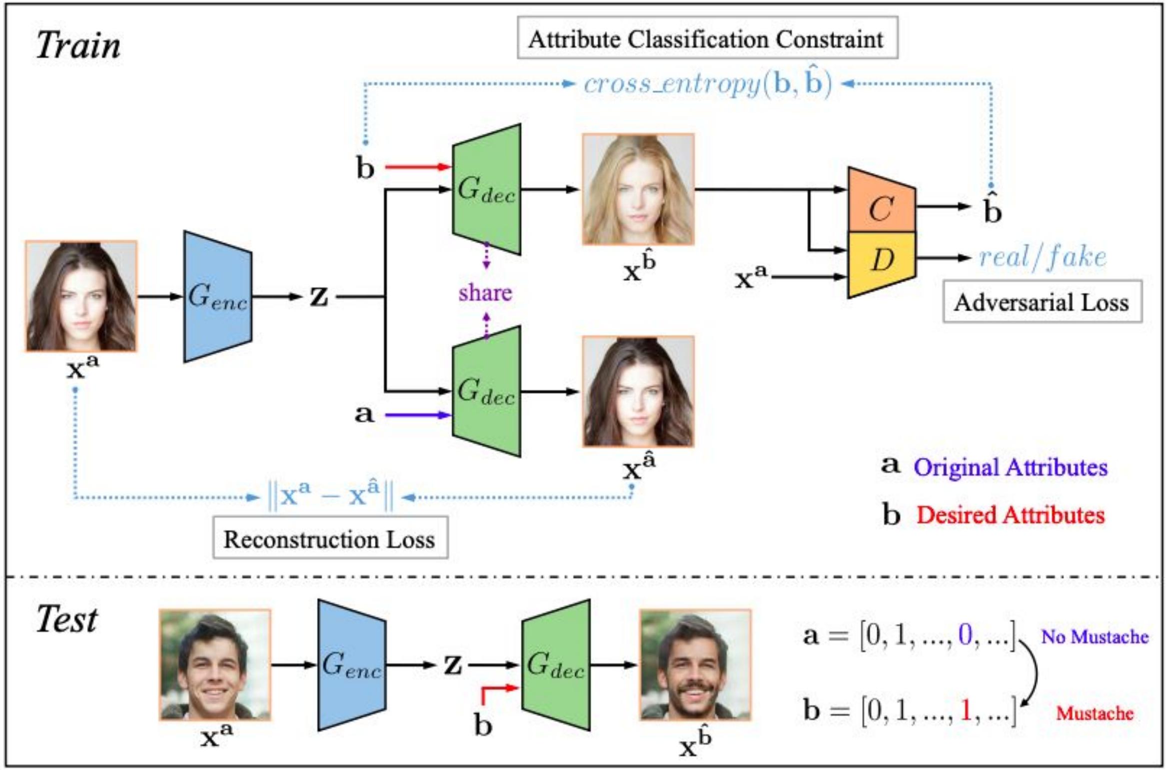
BASE FACE GENERATOR - MSG-StyleGAN

Multi-scale Gradient GAN (MSGGAN)



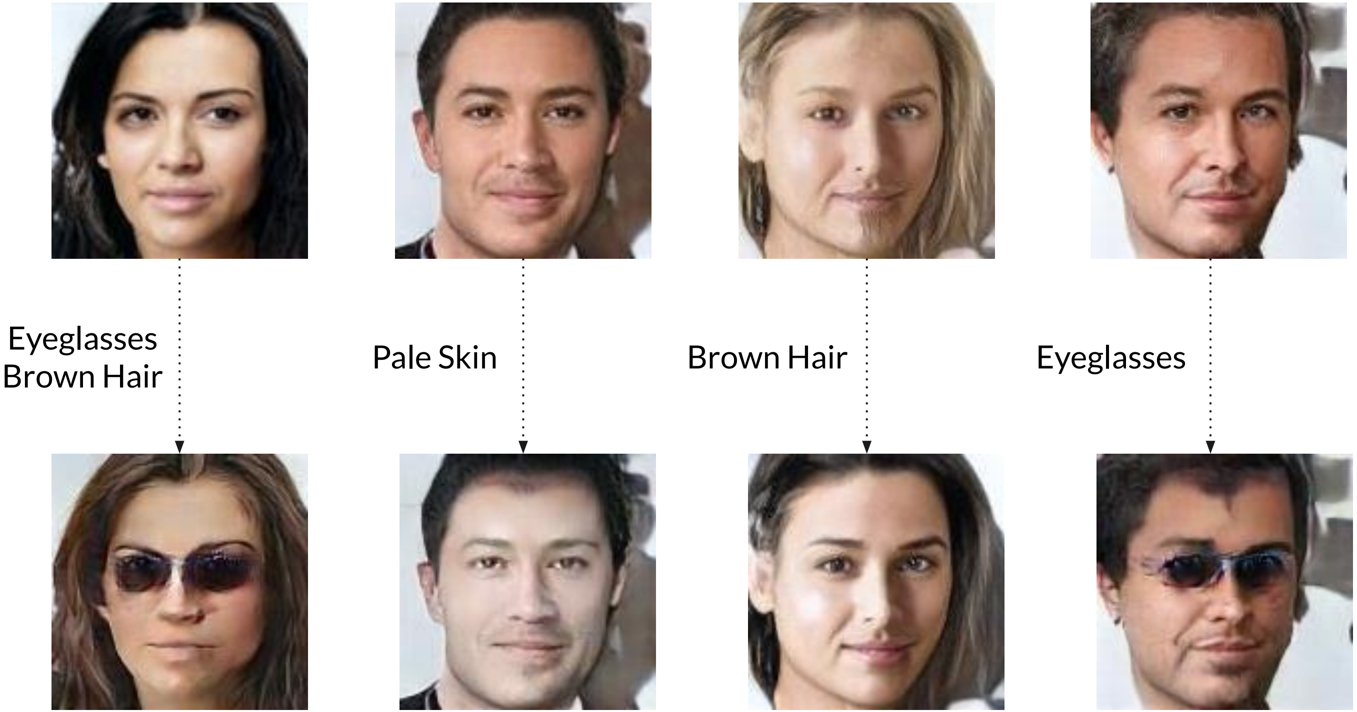
Source: [MSG-GAN: Multi-Scale Gradients for Generative Adversarial Networks]()

ATTRIBUTE EDITOR - AttGAN



Source: [AttGAN: Facial Attribute Editing by Only Changing What You Want]()

RESULTS



EVALUATION METRICS

→ Generate 100 images from 100 randomly chosen attribute vectors

→ Match them against images from a randomly chosen subset of 10,000 images

  - Cosine distance <= 0.4 is considered a match

→ Relevance - similarity between attribute vector of generated and retrieved image

EVALUATION METRICS

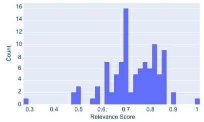
GENERATE 100 IMAGES FROM RANDOM ATTRIBUTE VECTORS

Match with CelebA

MATCH GENERATED FACES WITH 10,000 FACES FROM CELEBA
(Cosine distance between VGG-Face feature vector <0.4)

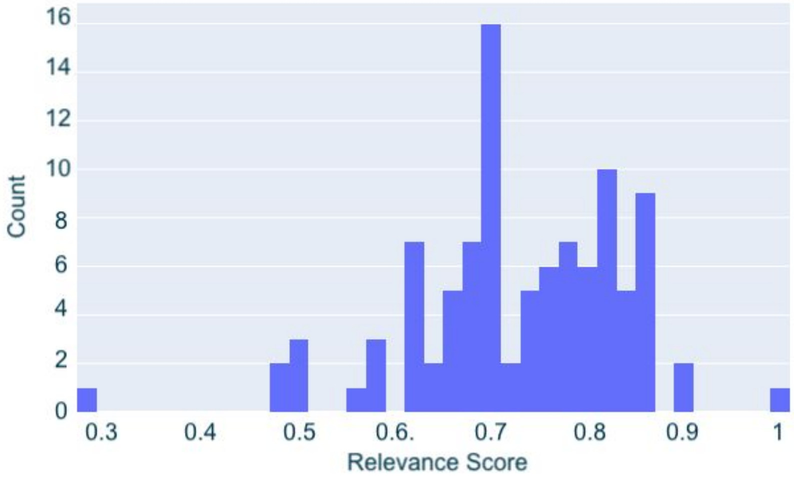
Matched CelebA attribute vectors

SIMILARITY COMPUTED BETWEEN GENERATED AND RETRIEVED ATTRIBUTE VECTORS



EVALUATION METRICS

- Average Maximum Relevance Score - 0.73
- 70% of the images have a Maximum Relevance Score > 0.7



- Top: Generated Image
- Bottom: Retrieved Image with maximum relevance score

Q/A

APPENDIX

GAN TRAINING

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

  for k steps **do**

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
  - Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

  end for

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Update the generator by descending its stochastic gradient:

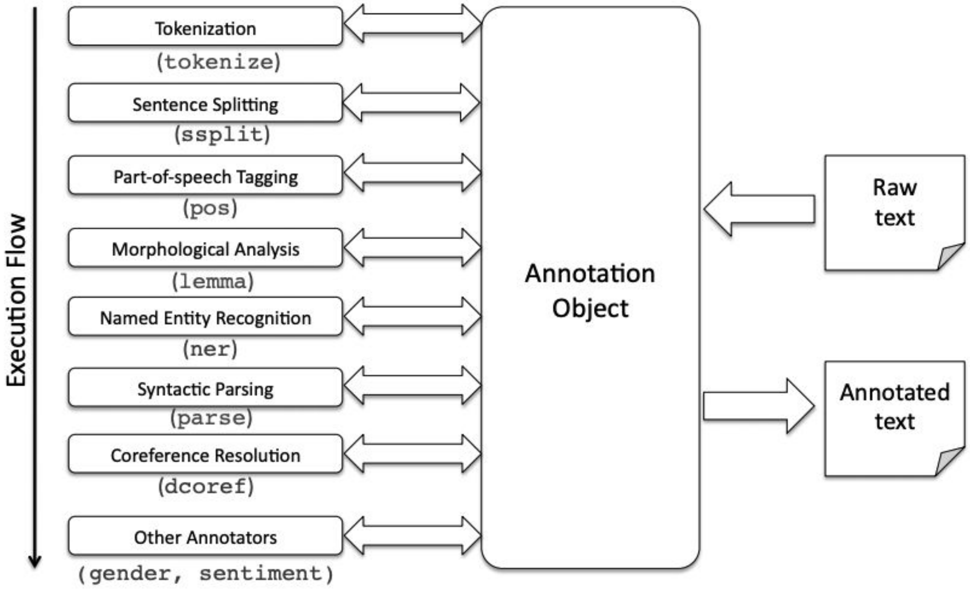
$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

Source: [Generative Adversarial Networks]

NATURAL LANGUAGE PARSER



MSG-STYLE GAN

StyleGAN

- Style transfer - using style vectors
- Takes in style vector(s) in addition to the random vector
- Aid in producing realistic images with seamless blending of features

MSG-STYLE GAN

- Number of GPUs used: 1
- Minibatch Size: 2
- Generator learning rate: 0.003
- Discriminator learning rate: 0.003
- Generator Loss Function: Non-Saturated GAN Loss
- Discriminator Loss Function: Simple Logistic Loss

The implementation of the MSG-StyleGAN was done with Tensorflow 1.13. The MSG-StyleGAN was trained for ~ 40 hours on a Tesla K80 GPU.

ATTRIBUTE GAN - ADVERSARIAL LOSS

Discriminator Loss

$$\min_{\|D\|_L \leq 1} \mathcal{L}_{adv_d} = -\mathbb{E}_{\mathbf{x^a} \sim p_{data}} D(\mathbf{x^a}) + \mathbb{E}_{\mathbf{x^a} \sim p_{data}, \mathbf{b} \sim p_{attr}} D(\mathbf{x^{\hat{b}}})$$

Generator Loss

$$\min_{G_{enc}, G_{dec}} \mathcal{L}_{adv_g} = -\mathbb{E}_{\mathbf{x^a} \sim p_{data}, \mathbf{b} \sim p_{attr}} [D(\mathbf{x^{\hat{b}}})]$$