Region-Based Semantic Factorization in GANs

Jiapeng Zhu, Yujun Shen, Yinghao Xu, Deli Zhao, Qifeng Chen

ICML 2022

Method

Preliminary—manipulation model

The formation of image editing with pre-trained GANs:

$$\text{Original image:} \quad x = G(z)$$

$$\text{Editing:} \quad x' = G(z + \alpha n)$$

Method

Preliminary—change in the edited image

By the Taylor series, we have

$$G(z + \alpha n) = G(z) + \alpha J_z n + o(\alpha)$$

Hence, we can further write

$$\|G(z + \alpha n) - G(z)\|_2^2 \approx \alpha^2 n^T J_z^T J_z n$$

The eigenvalue of the Jacobian is the variance of the image pixels after editing with Jacobian principal subspace n.

Method

The change in the foreground:

$$\left\|x_f^{edit} - x_f\right\|_2^2 \approx \alpha^2 n^T J_f^T J_f n$$

The change in the background:

$$\left\|x_b^{edit} - x_b\right\|_2^2 \approx \alpha^2 n^T J_b^T J_b n$$

Method

Key: We argue that a good attribute vector n for local control should take sufficient influence on the pixels within foreground, while maintaining the pixel values within the background. Hence, we have the following optimization:

$$
\begin{cases}
\text{argmax}_n n^T J_f^T J_f n \\
\text{argmin}_n n^T J_b^T J_b n
\end{cases}
$$

Method

Reformulation: The optimization problem can be solved through convex optimization. Alternatively, we replace the dual-objective optimization problem with a single-objective problem as

$$\begin{cases} \text{argmax}_n \ \dfrac{n^T J_f^T J_f n}{n^T J_b^T J_b n} \end{cases}$$

where the new object is the generalized Rayleigh quotient[1]. And this equation can be solved efficiently through a generalized eigenvalue problem
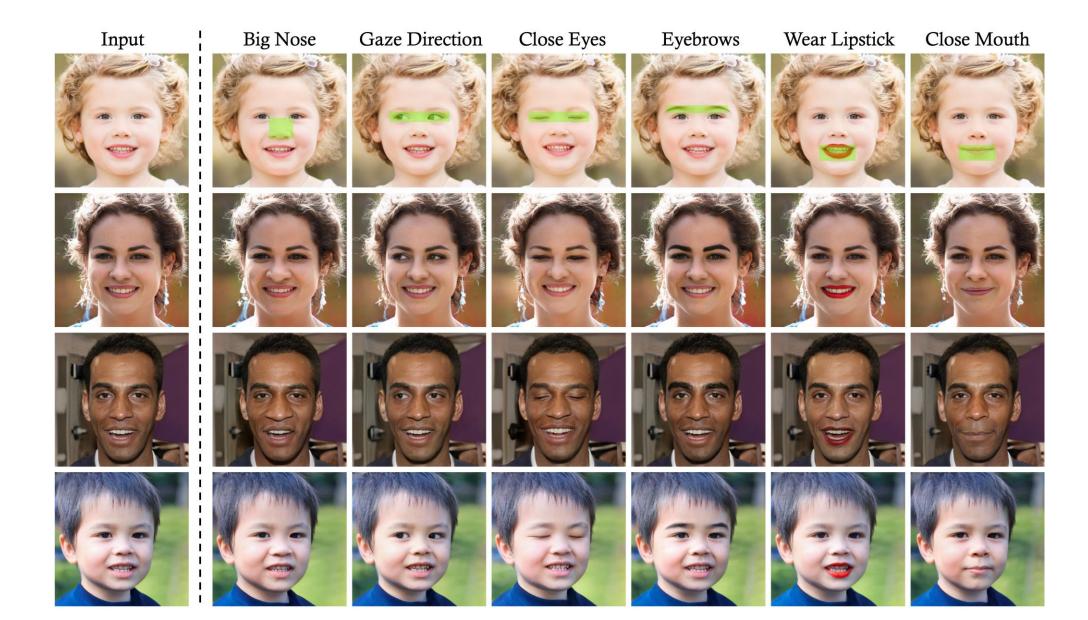
$$J_f^T J_f n = \lambda J_b^T J_b n \quad \text{or} \quad \left(J_b^T J_b\right)^{-1} J_f^T J_f n = \lambda n$$

1.Eigenvalue and Generalized Eigenvalue Problem: Tutotial . https://arxiv.org/abs/1903.11240

Method

Pseudo code

```python
import numpy as np
def compute_direction(J_fore, J_back):
    """Compute directions using the given Jacobians.
    Args:
        J_fore: Jacobian on the foreground.
        J_back: Jacobian on the background.
    Returns:
        Editing directions.
    """
    temp = np.linalg.inv(J_back).dot(J_fore)
    eig_val, eig_vec = np.linalg.eig(temp)
    return eig_vec.T[np.argsort(-eig_val)]
```

Experiment Results (StyleGAN2-FFHQ)

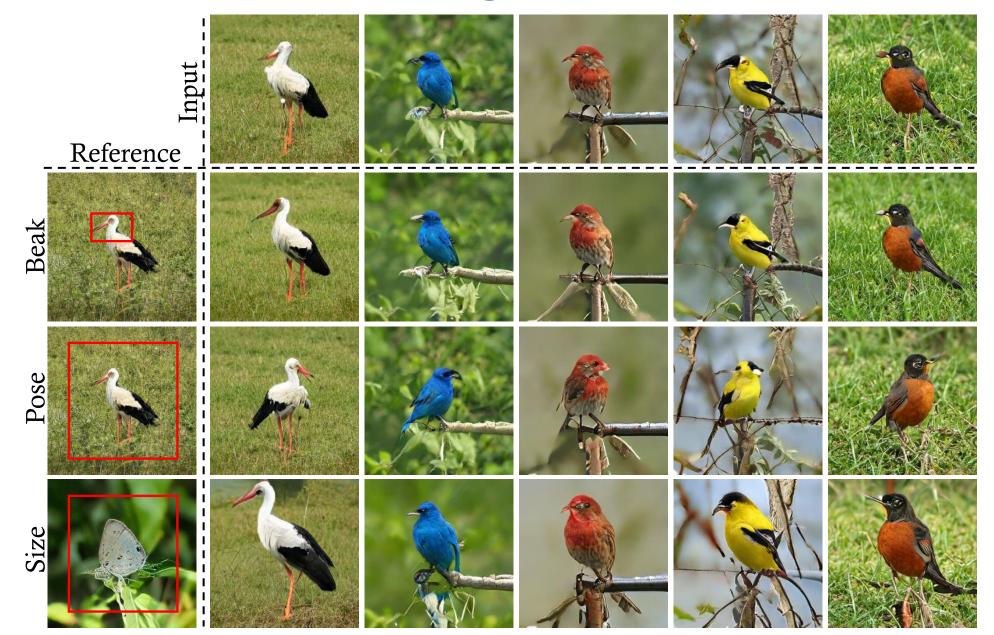| Input | Big Nose | Gaze Direction | Close Eyes | Eyebrows | Wear Lipstick | Close Mouth |

Experiment Results (StyleGAN2-Car)

Experiment Results (StyleGAN3-AFHQ)

Experiment Results (BigGAN)

Thanks!