

CVPR 2019 | STGAN: 人脸高精度属性编辑模型

标签（空格分隔）： 陈扬

github:<https://github.com/bluestyle97/STGAN-pytorch>

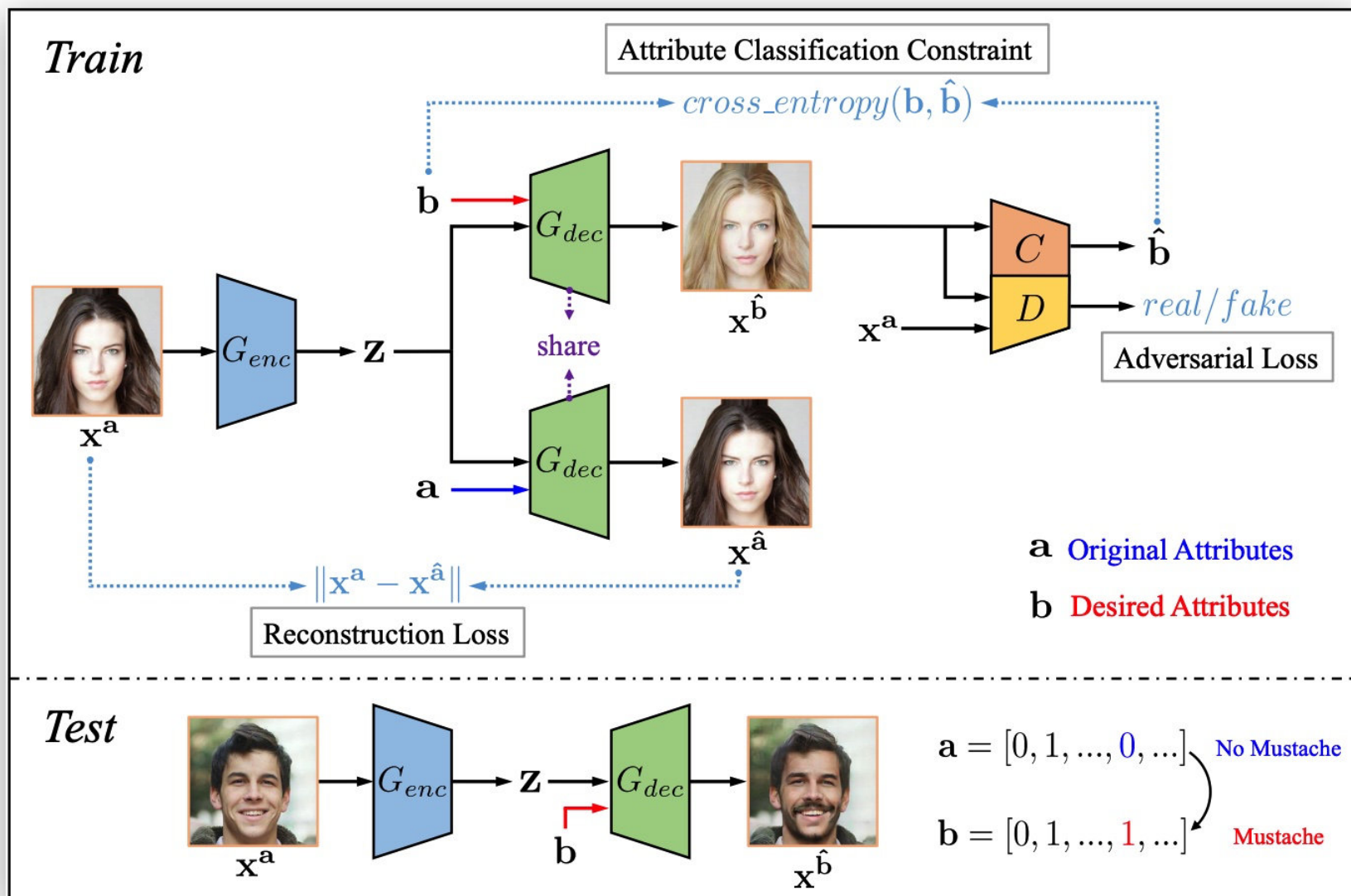
arxiv:<https://arxiv.org/abs/1904.09709>

Abstract

AttGAN和**StarGAN**在人脸属性编辑上取得了很大的成功，但当人脸属性之间相互交集或者目标人脸属性比较复杂时，这两种方式对于控制属性的标签上的精细化就显示了些许的不足。

STGAN是一个建立在**AttGAN**基础上的人脸属性编辑模型，通过差分属性标签下选择性传输单元的跳跃连接实现了人脸高精度属性的编辑。

前言--ATTGAN



属性分类限制

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

$$\ell_g(\mathbf{x}^a, \mathbf{b}) = \sum_{i=1}^n -b_i \log C_i(\mathbf{x}^{\hat{\mathbf{b}}}) - (1-b_i) \log(1-C_i(\mathbf{x}^{\hat{\mathbf{b}}})),$$

classification也要训练的，和auto-encoder一起训练，介样练：

$$\min_C \mathcal{L}_{cls_c} = \mathbb{E}_{\mathbf{x}^a \sim p_{data}} [\ell_r(\mathbf{x}^a, \mathbf{a})],$$

$$\ell_r(\mathbf{x}^a, \mathbf{a}) = \sum_{i=1}^n -a_i \log C_i(\mathbf{x}^a) - (1-a_i) \log(1-C_i(\mathbf{x}^a)).$$

重建误差

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

对抗误差

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

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

总误差（分为两步）

$$\min_{G_{dec}, G_{enc}} \mathcal{L}_{enc_{dec}} = \lambda_1 \mathcal{L}_{rec} + \lambda_2 \mathcal{L}_{cls_g} + \mathcal{L}_{adv_g},$$

$$\min_{D, C} \mathcal{L}_{dis_{cls}} = \lambda_3 \mathcal{L}_{cls_c} + \mathcal{L}_{adv_d},$$

承上启下

ATTGAN 模型都是架构在编码器 - 解码器上，同时将源图像和目标属性向量作为输入，AttGAN不是对潜在表示施加约束，而是对生成的图像应用属性分类约束，以保证所需属性的正确变化，同时引入重建学习以保留属性排除细节。StarGAN只用一个generator网络，处理多个domain之间互相generate图像的问题，这是比AttGAN更深一步的人脸属性迁移。

不过,我们今天要介绍的是 STAGN:

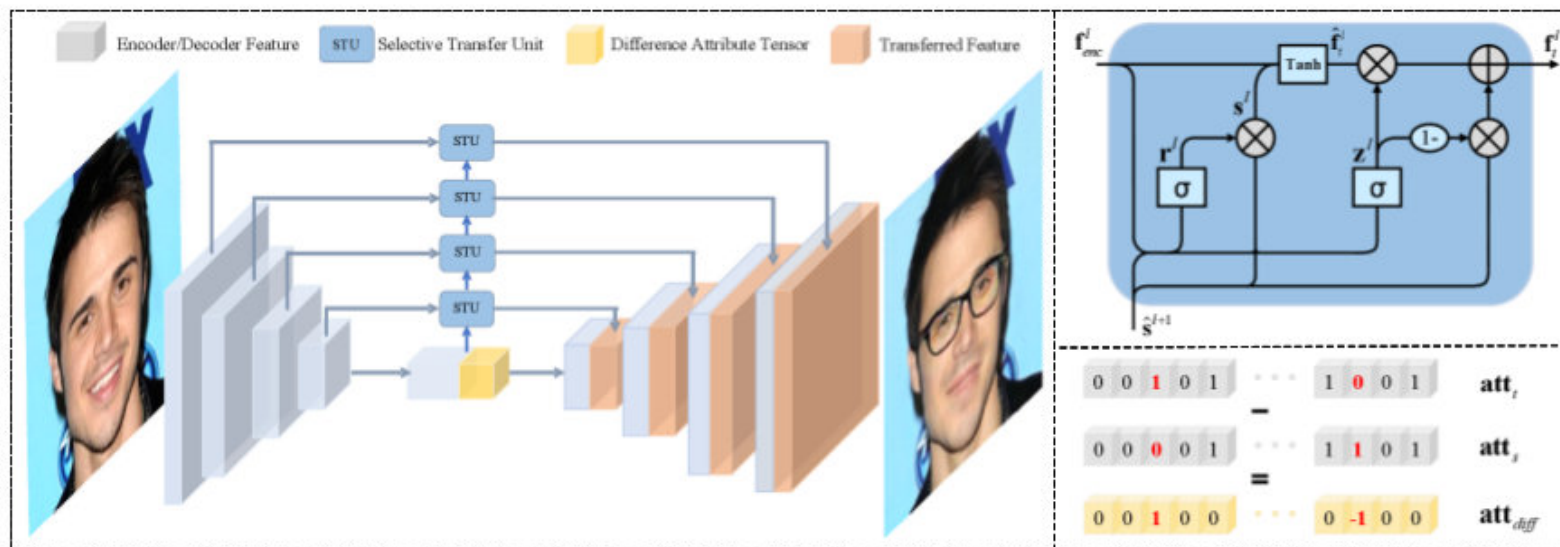


Figure 5: The overall structure of STGAN. On the left is the generator. The top-right figure shows detailed STU structure, and all variables marked in this figure share same dimension (e.g., 64×64). The difference attribute vector of adding *Eyeglasses* and removing *Mouth Open* attributes is shown on the bottom-right.

STGAN 用的的是一个类 UNET 的网络结构做生成器,在 skip-connection 的时候使用了一个叫 STU(提出了选择性传输单元)的操作,来自适应地选择和修改编码器特征,其进一步与解码器特征连接以增强图像质量和属性操纵能力,相当于给生成器加上了硬解耦的作用.

选择性传输单元

在介绍选择性传输单元之前，我们先把文章对目标属性和源属性的标签处理交代一下。StarGAN 和AttGAN 都将目标属性向量 Att_t 和源图像 x 作为输入到生成器。实际上，使用完整目标属性向量是多余的，可能对编辑结果有害。如果目标属性向量 Att_t 与源 Att_s 完全相同，此时，理论上输入只需要对图像进行重构即可，但StarGAN和AttGAN 可能会错误地操作一些未更改的属性，比如把本来就是金色头发变得更加的金色。

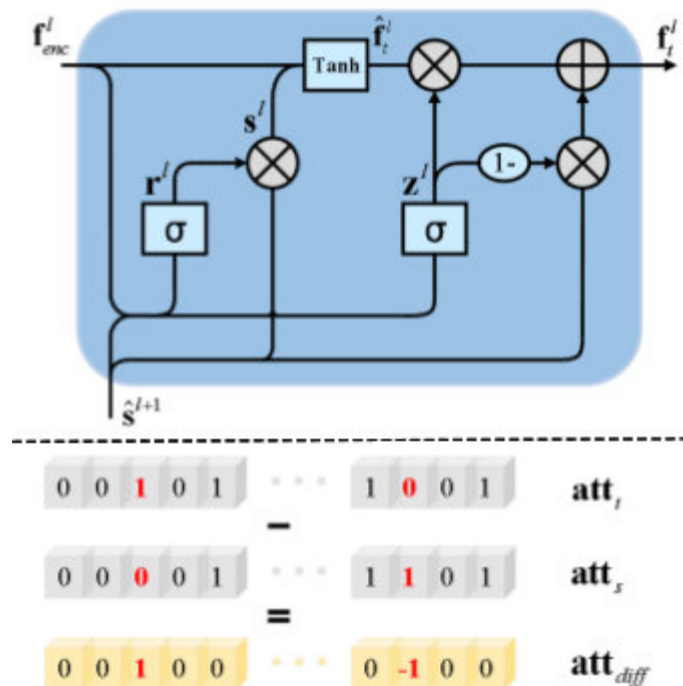
对于任意图像属性编辑，而不是完整目标属性向量，只应考虑要更改的属性以保留源图像的更多信息。因此，将差异属性向量定义为目标和源属性向量之间的差异是合适的：

$$Att_{diff} = Att_t - Att_s$$

比如男性有胡子戴眼镜的源图编辑到男性无胡子戴眼镜秃头目标图，这里面仅仅是添加了秃头这一属性，减少了胡子这一属性，其它的可以保持不变。 Att_{diff} 可以为指导图像属性编辑提供更有价值的信息，包括是否需要编辑属性，以及属性应该改变的方向。然后可以利用该信息来设计合适的模型，以将编码器特征与解码器特征进行变换和连接，并且在不牺牲属性操纵精度的情况下提高图像重建质量。

选择性传输单元（STU）来选择性地转换编码器特征，使其与解码器特征兼容并互补，而不是通过skip connection直接将编码器与解码器特征连接起来。这个变换需要适应变化的属性，并且在不同的编码器层之间保持一致，作者修改GRU[3]的结构以构建用于将信息从内层传递到外层的STU。

我们来看一下STU的结构：



f_{enc}^l 为编码器第 l 层输出， s^{l+1}

为数据编码在 $l + 1$ 层的隐藏状态，隐藏状态 \hat{s}^{l+1} 则是结合了 Att^{diff} 得到的：

$$s^{l+1} = W_{\{t * T\}}[s^{l+1}, Att_{diff}]$$

其中 $[\cdot, \cdot]$ 表示为concatenation操作， $\ast T$ 为转置卷积，然后，STU采用GRU的数学模型来更新隐藏状态 s 和转换后的编码器特征 f_t^l :

$$r^l = \sigma(W_r \ast [f_{enc}^l, s^{l+1}]), z^l = \sigma(W_z \ast [f_{enc}^l, s^{l+1}]), s^l = r^l \circ s^{l+1}$$

$$\hat{f}_t^l = \tanh(W_h \ast [f_{enc}^l, s^l]), f_t^l = (1 - z^l) \circ \hat{f}_t^l + z^l \circ f_t^l$$

其中 \ast 表示卷积运算， \circ 表示逐项乘积， $\sigma(\cdot)$ 表示sigmoid函数。复位门 r^l 和更新门 z^l 的引入允许以选择性方式控制隐藏状态，差异属性向量和编码器特征。输出 f_t^l 提供了一种自适应的编码器特征传输方法及其与隐藏状态的组合。

选择性传输单元（STU）说白了就是在GRU的结构上实现的，差分标签控制下的编码特征的选择。

```
if use_stu:
    self.stu = nn.ModuleList()
    for i in reversed(range(\
        self.n_layers - 1 - self.shortcut_layers,\
        self.n_layers - 1)):
        self.stu.append(C\
            onvGRUCell(self.n_attrs, conv_dim * 2 ** i, \
                conv_dim * 2 ** i, stu_kernel_size))
```

模型结构

有了上述的分析，我们再看模型的结构则是比较容易理解了：

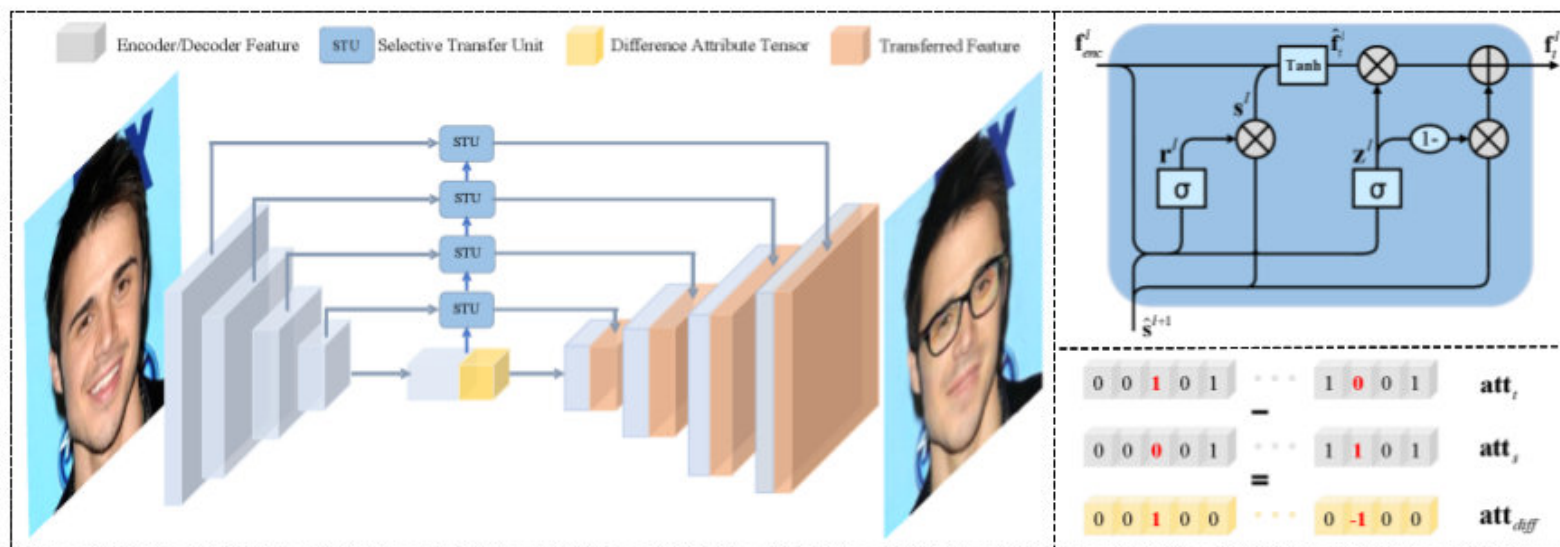


Figure 5: The overall structure of STGAN. On the left is the generator. The top-right figure shows detailed STU structure, and all variables marked in this figure share same dimension (e.g., 64×64). The difference attribute vector of adding *Eyeglasses* and removing *Mouth Open* attributes is shown on the bottom-right.

整个模型比较简单，在编码器和解码器过程中，加入 STU 选择单元，从而获得人脸属性编辑后的输出。编码器的输入端包括源图 x 和差分属性标签 Att_{diff} 。对于判别器，也是判别生成器输出真假和对应的属性标签。

对抗损失采用WGAN-GP来实现生成优化，对应着 $L_{D_{adv}}, L_{G_{adv}}$ 。对于属性标签和生成器的属性优化通过源真实样本和标签优化判别器，再通过判别器去判别目标生成的属性结果来优化生成器：

$att_s^{(i)}, att_t^{(i)}$ 为源域和目标域属性标签，文章还加入了一个不使用标签的重构误差，也就是差分标签置为0：

$$\mathcal{L}_{rec} = \|x - G(x, 0)\|_1$$

最终得到的损失为：

$$\min_D \mathcal{L}_D = -\mathcal{L}_{D_{adv}} + \lambda_1 \mathcal{L}_{D_{att}}$$

$$\min_G \mathcal{L}_G = -\mathcal{L}_{G_{adv}} + \lambda_2 \mathcal{L}_{G_{att}} + \lambda_3 \mathcal{L}_{rec}$$

$att_s^{(i)}, att_t^{(i)}$ 为源域和目标域属性标签，文章还加入了一个不使用标签的重构误差，也就是差分标签置为0：

$$\mathcal{L}_{rec} = \|x - G(x, 0)\|_1$$

最终得到的损失为：

$$\min_D \mathcal{L}_D = -\mathcal{L}_{Dadv} + \lambda_1 \mathcal{L}_{Datt}$$

$$\min_G \mathcal{L}_G = -\mathcal{L}_{Gadv} + \lambda_2 \mathcal{L}_{Gatt} + \lambda_3 \mathcal{L}_{rec}$$

实验

CelebA数据集包含裁剪到 178×218 的202,599个对齐的面部图像，每个图像有40个带/不带属性标签。图像分为训练集，验证集和测试集，文章从验证集中获取1,000张图像以评估训练过程，使用验证集的其余部分和训练集来训练STGAN模型，并利用测试集进行性能评估。实验考虑13种属性，包括秃头，爆炸，黑发，金发，棕色头发，浓密眉毛，眼镜，男性，嘴微微开口，小胡子，无胡子，苍白皮肤和年轻，实验中，每个图像的中心 170×170 区域被裁剪并通过双三次插值调整为 128×128 。

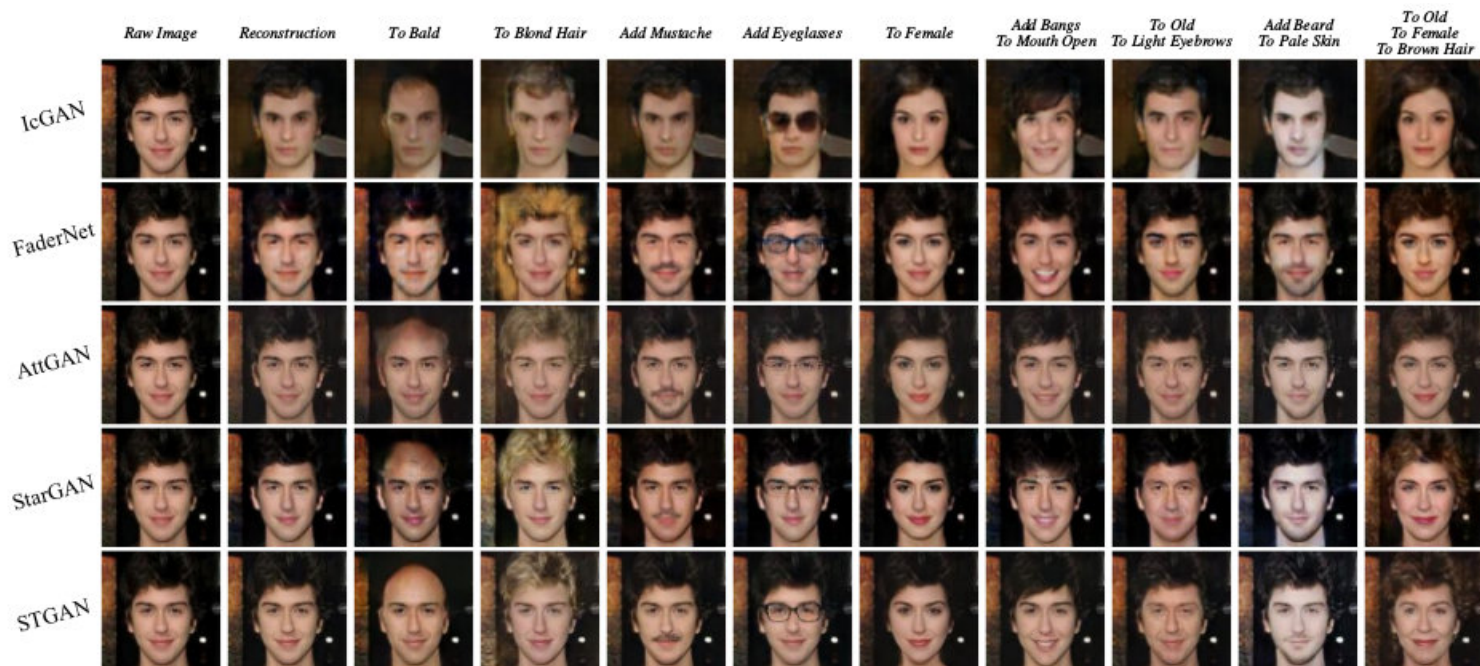


Figure 6: Facial attribute editing results on the CelebA dataset. The rows from top to down are results of IcGAN [26], FaderNet [17], AttGAN [11], StarGAN [7] and STGAN.

定量评估上，文章从两个方面评估属性编辑的性能，即图像质量和属性生成准确性。图像质量上，保持目标属性向量与源图像属性相同，得到了PSNR / SSIM结果：

Method	IcGAN	FaderNet	AttGAN	StarGAN	STGAN
PSNR/SSIM	15.28/0.430	30.62/0.908	24.07/0.841	22.80/0.819	31.67/0.948

Table 2: Reconstruction quality of the comparison methods on facial attribute editing task.

吐槽:PSNR 和 SSIM 在这里当做评价指标简直是瞎扯淡.....

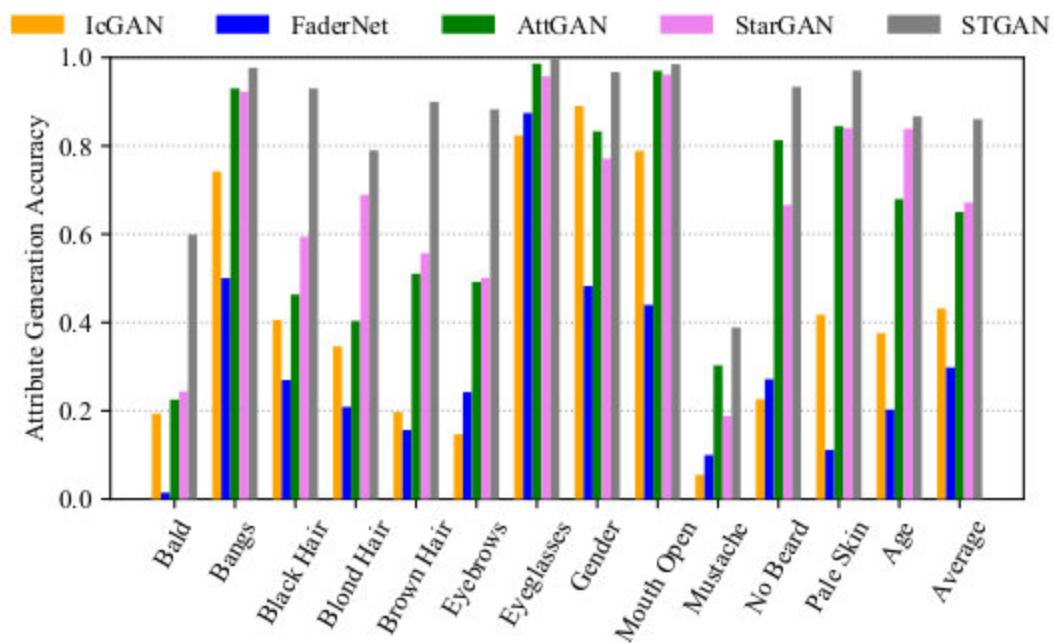


Figure 7: Attribute generation accuracy of IcGAN [26], FaderNet [17], AttGAN [11], StarGAN [7] and STGAN.

这个指标还行.....



Figure 8: Results of season translation, the top two rows are *summer*→*winter*, and the bottom two rows are *winter*→*summer*.

实验在用户的选择测试上也取得了最佳效果，Ablation Study实验上也证实了模型的每一部分的优势和必要。最后放一张STGAN在图像季节转换的实验效果：

总结

文章研究了选择性传输视角下任意图像属性编辑的问题，并通过在编码器 - 解码器网络中结合差分属性向量和选择性传输单元 (STU) 来提出STGAN模型。通过将差异属性向量而不是目标属性向量作为模型输入，STGAN可以专注于编辑要改变的属性，这极大地提高了图像重建质量，增强了属性的灵活转换。

