Everybody Dance Now

Paper summary by Akilesh B

September 5, 2018

1 Introduction

- This paper proposes a simple method for motion transfer.
- Given a source video of a person dancing, the performance is transferred to another target.
- It requires only few minutes of the target subject performing standard moves for training.

2 Network details

2.1 Key idea

- The pipeline is divided into three stages: a) pose detection, b) global pose normalization, c) mapping from normalized pose stick figures to the target subject as shown in figure 1
- **Pose detection** : Pretrained state-of-the-art pose detector is used to create pose stick figures from frames of the source video.
- **Global pose normalization** : This accounts for differences between the source and target body shapes such as difference in limb proportions or the fact that subjects can stand closer or farther to the camera than one another.
- **Mapping step**: This step is per-frame image-to-image translation using modified pix2pixHD with spatio-temporal smoothing.

2.2 Learning objective function

In the pix2pixHD framework [1], the generator network G is engaged in a min-imax game against multi-scale discriminators $D = (D_1, D_2, D_3)$

$$min_G\Bigg(\Big(max_{D_1,D_2,D_3}\sum_{k=1,2,3}L_{GAN}(G,D_k)\Big)+\lambda_{FM}\sum_{k=1,2,3}L_{FM}(G,D_k)+\lambda_{VGG}L_{VGG}(G(x),y)\Bigg) \tag{1}$$

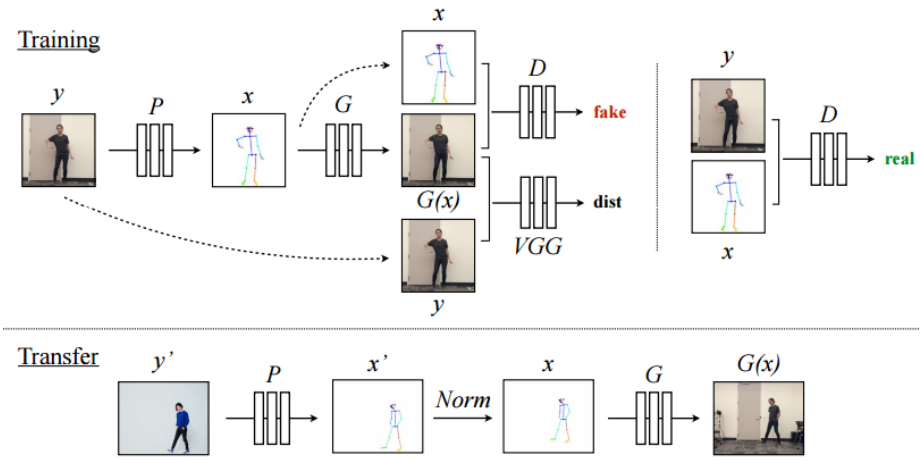

Figure 1: Training and Transfer steps

where $L_{GAN}(G, D)$ is the adversarial loss.

$$L_{GAN}(G, D) = E_{(x,y)}[logD(x, y)] + E_x[log(1 - D(x, G(x))] \tag{2}$$

$L_{FM}(G, D)$ is the discriminator feature-matching loss, $L_{VGG}(G(x), y)$ is the perceptual reconstruction loss.
They modify this to incorporate temporal smoothing, that is, $G(x_t)$ is conditioned on its corresponding pose stick figure x_t and the previous output $G(x_{t-1})$. Therefore, the discriminator is now tasked with distinguishing between the "fake" sequence $(x_{t-1}, x_t, G(x_{t-1}), G(x_t))$ and the "real" sequence $(x_{t-1}, x_t, y_{t-1}, y_t)$ as shown in figure 2. The modified GAN objective becomes:

$$L_{smooth}(G, D) = E_{(x,y)}[logD(x_{t-1}, x_t, y_{t-1}, y_t)] + E_x[log(1 - D(x_{t-1}, x_t, G(x_{t-1}), G(x_t))] \tag{3}$$

2.3 Face GAN

In order to better capture the face of the target and add more realism, they use a Face GAN wherein they feed the face region of the generated image$(G(x)_F)$ and that of the input stick figure(x_F) to another generator G_f and obtain a residual $r = G_f(x_F, G(x)_F)$. The final output is the addition of the residual with original face region $r + G(x)_F$ and this change is reflected in the relevant region of the full image as shown in figure 3

$$L_{face}(G_f, D_f) = E_{(x_F, y_F)}[logD_f(x_F, y_F)] + E_{x_F}[log(1 - D_f(x_F, r + G(x)_F))] \tag{4}$$

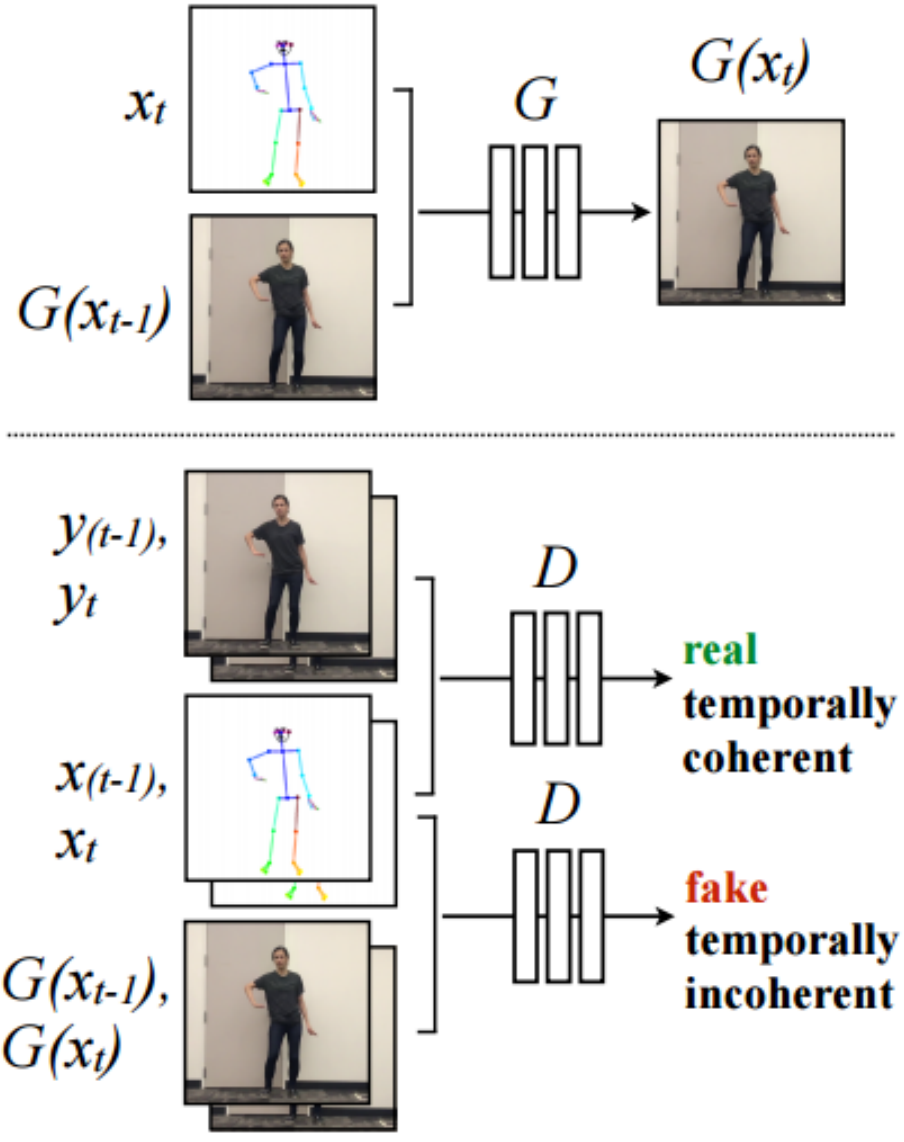

Figure 2: Temporal smoothing

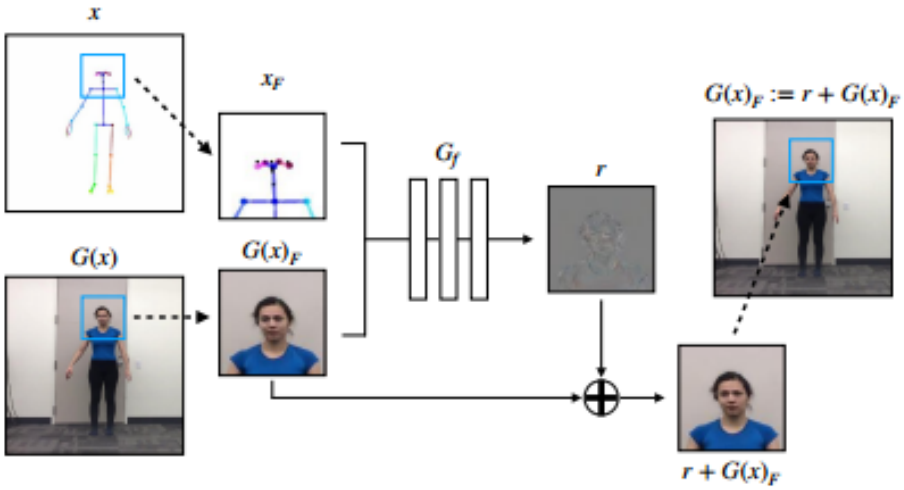

Figure 3: Face GAN

2.4 Training

First the full image GAN is optimized whose full objective is:

$$min_G\Bigg(\Big(max_{D_1,D_2,D_3}\sum_{k=1,2,3}L_{smooth}(G,D_k)\Big)+\lambda_{FM}\sum_{k=1,2,3}L_{FM}(G,D_k)+ \lambda_{VGG}\Big(L_{VGG}(G(x_{t-1}),y_{t-1})+L_{VGG}(G(x_t),y_t)\Big)\Bigg) \tag{5}$$

After this, the full image generator and discriminator weights are frozen and face GAN is optimized whose objective is:

$$min_{G_f}\Bigg(\Big(max_{D_f}L_{face}(G_f,D_f)\Big)+\lambda_{VGG}L_{VGG}(r+G(x)_F,y_F)\Bigg) \tag{6}$$

They show quantitative (using SSIM and LPIPS score) and qualitative assessment of the original pix2pixHD, pix2pixHD with temporal smoothing and pix2pixHD with both temporal smoothing and face GAN.

References

[1] Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J. and Catanzaro, B., 2018. High-resolution image synthesis and semantic manipulation with conditional gans. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Vol. 1, No. 3, p. 5).