GAN 및 키포인트와 로컬 아핀 변환을 이용한 스타일 변환 동적인 이미지 애니메이션 네트워크 구축

장준보 [1]
[1] 한국외국어대학교 산업경영공학과

datu0615@gmail.com

Construction of Dynamic Image Animation Network for Style Transformation Using GAN, Keypoint and Local Affine

Jun-Bo Jang[1]
[1]Dept. of Industrial Management Engineering, Hankuk University of Foreign Studies

Abstract

High-quality images and videos are being generated as technologies for deep learning-based image style translation and conversion of static images into dynamic images have developed. However, it takes a lot of time and resources to manually transform images, as well as professional knowledge due to the difficulty of natural image transformation. Therefore, in this paper, we study natural style mixing through a style conversion network using GAN and natural dynamic image generation using the First Order Motion Model network (FOMM).

1. Introduction

With the development of technology for deep learning-based image style conversion and conversion of static images into dynamic images, high-quality images and videos are being created to the extent that it is difficult to distinguish authenticity. The growing demand for these technologies also highlights the importance of natural image style transformation and dynamic images. However, relying on manual work and converting images requires a lot of time and resources, as well as difficulties in natural image conversion and professional knowledge. To solve this problem, we use a technology that converts it into an image similar to an image desired by a user through a GAN network called DualStyleGAN [1]. It also uses a technology that converts a converted image into a dynamic image through a network called the First Order Motion Model (FOMM) [2]. The system is first learned via a deep learning style conversion network using the FFHQ dataset (Flickr-Faces-HQ Dataset) [3] and the cartoon dataset from Toonify [4]. A video sequence is then generated for the transformed image to be animated according to the motion of the video using the VoxCeleb dataset [5].

2. Related Work

1. DualStyleGAN

DualStyleGAN [1] is a model that uses a pair of style networks by adding a new network in StyleGAN2 [6], and performs well with fewer data compared to other models. Although traditional fine-tuning sometimes translates space as a whole, losing style diversity, this model can use an external style network to map existing domains to various style domains at the same time as not destroying them. DualStyleGAN aims to form fixed face-portrait pairs by training and supervising realistic faces from artistic portraits through Facial Destination. Since the difference between the recovered face and the portrait may be large, we propose a multi-stage de-style method that gradually improves the realism of the portrait to maintain a balance between the two. The first step is later initialization. Latent initialization maps a portrait to a stylegan latent space using a PSP Encoder [7]. The PSP encoder learns about a particular stylegan2 network. When an image S is input, the Z+ spatial vector of the network closest to the input image is output. The second step is Latent Optimization [8]. Latent optimization uses a fine-tuned generator g' to reconstruct the face image, which finds g' and the acceptance loss, identity loss, and zeˆ+, which minimizes standard errors of the input image. After finding zeˆ+, we design a regularization term that pulls zeˆ+ into a well-defined Z space to avoid overfitting. The third step is image embedding. Image embedding finally eliminates additional zi+ (=E(g(zeˆ+))) unrealistic face information by passing zeˆ+ through the original model and encoder. zi+ has a more rational facial structure and proceeds with training in pairs with zeˆ+ to provide effective supervision on how to transform and abstract facial structures to mimic input images. The existing internal style network (mapping network) on the left side of Figure 1 and the parameters of the generator remain fixed and receive either a stylized style code zi+ or a real face style code z+ or a unit Gaussian noise z as input. The external style network simply receives a ze+ or noise z encoding the portrait. The overall training is carried out by G(E(I), E(S), and w), receiving a face image I and a portrait image S, and a style transfer is performed by G(E(I), E(S), and w, where w is a weight vector for regulating the style combination of each layer. Color control is fused in fine-resolution layers (8–18) with outer style paths going through mapping networks and affine transformation blocks in the same way as inner styles and reflecting the value of weight w in AdaIN. Block Tc is used to characterize color values for each domain. Structural control proposes Ts, characterizing domain-specific styles to adjust structural styles in coarse-resolution layers (1–7), and modulation residual blocks (ModRes) containing ResBlock and AdaIN.

2. First Order Motion Model for Image Animations

This paper is an advanced model from Monkey-Net (MOviNg KEYpoint Network) [9], the first object-agnostic deep model for image animation. Monkey-Net encodes motion information via keypoints learned in a self-supervised fashion. The major weakness of Monkey-Net is that it poorly models object appearance transformations in the keypoint neighborhoods, assuming a zeroth order model. The model developed to solve this problem is the First Order Motion Model. This model models complex behavior using a set of self-learning keypoints with local affine transformations. We then introduce an occlusion-aware generator, which adopts an occlusion mask automatically estimated to indicate object parts that are not visible in the source image and that should be inferred from the context. To improve the estimation of local affine transformations, we extend the equivariance loss commonly required for keypoint detector training. Look at Figure 2, The image of the object to which the animation is to be applied proceeds in the direction of learning the video sequencing of a similar object with movement. The Motion Module sends these two data and outputs a local affine transformation and an occlusion mask. The Keypoint Detector within the Motion Module predicts key points individually for both data. And since Local Affine Transformation not only moves keypoints, but also identifies motion near each keypoint, complex transformations can be performed through this affine transformation.

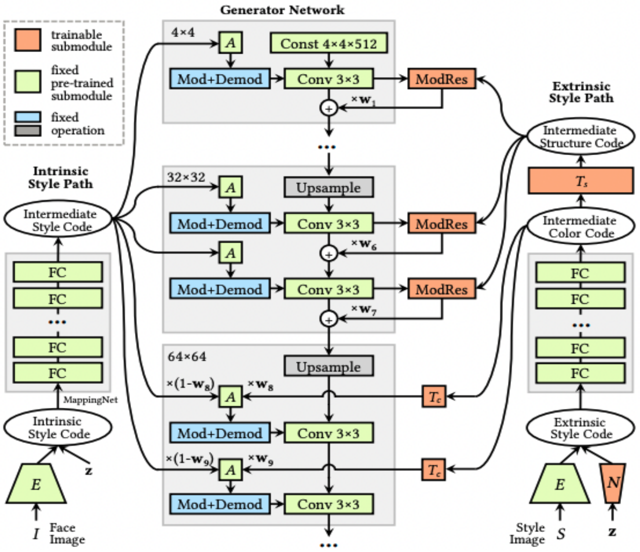

Figure 1. Details of the DualStyleGAN Network

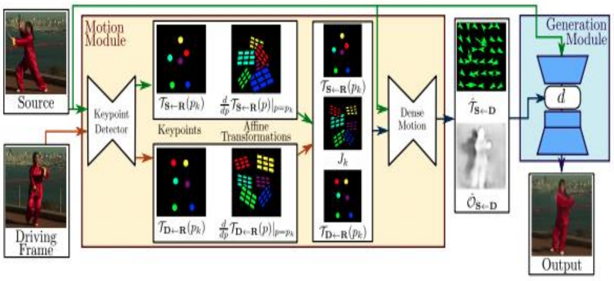

Figure 2. Details of the motion learning model

3. Style Trasformation Dynamic Image Generation System

1. System Overview

The overall structure of the proposed system is shown in Figure 3. First, an RGB camera is opened to receive an image, and if the camera is not used, an image is selected and input. The input image is input to a deep learning model learned in advance, and the generated image is saved after style conversion for the image is performed. After that, the saved image is converted into a dynamic image by inputting a video to be converted into another pre-trained deep learning model.

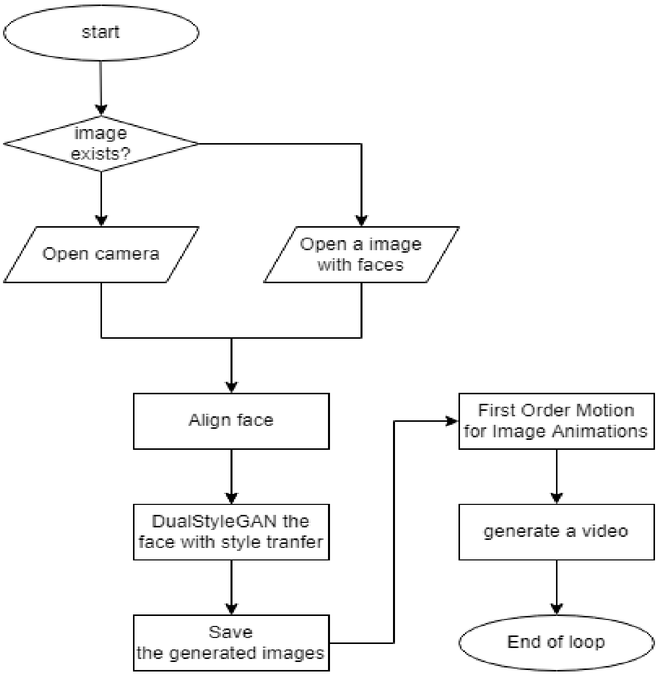
start
image exists?
Open camera
Open a image with faces
Align face
DualStyleGAN the face with style tranfer
Save the generated images
First Order Motion for Image Animations
generate a video
End of loop

Figure 3. Proposed Style Mixing Dynamic Image Genenration System Overall Flowchart

2. Style Transfer Network Learning

2.1 Dataset

In this paper, we use the FFHQ dataset and the cartoon face image dataset. The FFHQ dataset is a high-quality human face image dataset produced by NDVIA as a benchmark for generative adversarial networks (GAN). The dataset consists of 70,000 high-quality PNG images (1024x1024 resolution) with varying ethnic, age and image backgrounds. The Cartoonface image dataset is a dataset used by a total of 317 Toonify. Toonify is a model that uses the existing StyleGAN2 to extract the picture's feature later vector and insert it into the Blended model to make a human face into a cartoon face.

2.2 Implementation details

Progressive fine-tuning uses Google collaborative pro + environment, NVIDIA T4, P100 GPU, and 32 batch sizes. In the source domain, structural transmission is performed at λadv = 0.1, λperc = 0.5, and trains on l = 7, 6, 5 for 300, 300, 3000 iterations. In the target domain, style transfer sets s λadv = 1, λperc = 1, λCX = 0.25, λFM = 0.25. sets (λID, λreg) to (1,0.015) and sets trains that 1400 iterations on the Cartoon.

3. Motion Learning Network

3.1 Dataset

It uses the VoxCeleb dataset and the image dataset converted to DualStyleGAN. The VoxCeleb dataset is a face dataset of 22,496 videos extracted from YouTube. The bounding box is extracted from the first video frame for preprocessing. Then trace the face away from its initial position and cut the video frame to the smallest size containing all bounding boxes. The video is then adjusted to 256x256 size, maintaining the aspect ratio.

3.2 Implementation details

Progressive fine-tuning uses Google collaborative pro + environment, NVIDIA T4, P100 GPU, and 40 batch sizes. The generator, discriminator, and keypoint detector learning rates were all set to 2.0e-4, and sigma_affine=0.05, sigma_tps=0.005, and points_tps=5. sets trains that 100 iterations on the VoxCeleb Dataset.

4. Results

As a result of performing style transfer and dynamic image generation for learning data, it can be seen that style transfer for side images is superior to other style transfer models such as Toonify or StyleGAN2. In addition, we could see that image animation results based on key-point and local affine transformations also move naturally. On the other hand, there are cases in which non-face textures such as hats are poorly detected and characteristics not present in the external style dataset are not expressed, or certain body parts are unnatural. Also, for motion learning, there is an unnatural part of the video where a person's face turns to the side, because only the style-transfered images appear to be in a specific location.

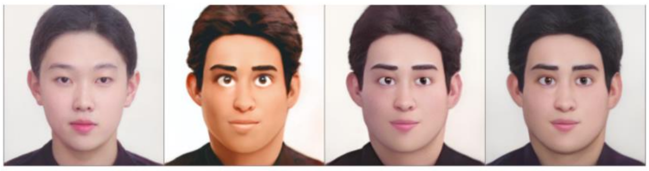

Figure 4. Result using DualStyleGAN
From left to right, 1. PSP reconstructed content image 2. Style transfer result : both color and structure styles are transferred 3. Structure transfer result : preserve the color of the content image by replacing the extrinsic color codes with intrinsic color codes 4. Structure transfer result : preserve the color of the content image by deactivating color-related layers

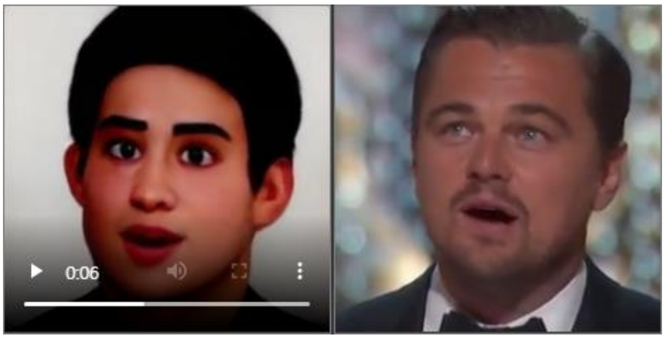

Figure 5. Results using the First Order Motion Model

The result of using the First Order Motion Model, which created the converted image using DualStyleGAN and followed the movement of the right video

5. Conclusions

Future research will improve the performance of style transfer through models trained on large amounts of data, such as BERT and GPT-3, to express characteristics that are not present in non-face textures or external style datasets and will naturally use 360° images of humans for learning in the process of converting static images. In addition, we are planning to create our own emoticons by converting the style-converted images into GIF format rather than video.

References

[1] Yang, Shuai, et al. "Pastiche Master: Exemplar-Based High-Resolution Portrait Style Transfer." arXiv preprint arXiv:2203.13248 (2022).

[2] Siarohin, Aliaksandr, et al. "First order motion model for image animation." Advances in Neural Information Processing Systems 32 (2019).

[3] Karras, Tero, Samuli Laine, and Timo Aila. "A style-based generator architecture for generative adversarial networks." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

[4] Pinkney, Justin NM, and Doron Adler. "Resolution dependent gan interpolation for controllable image synthesis between domains." arXiv preprint arXiv:2010.05334 (2020).

[5] Nagrani, Arsha, Joon Son Chung, and Andrew Zisserman. "Voxceleb: a large-scale speaker identification dataset." arXiv preprint arXiv:1706.08612 (2017).

[6] Karras, Tero, et al. "Analyzing and improving the image quality of stylegan." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.

[7] Richardson, Elad, et al. "Encoding in style: a stylegan encoder for image-to-image translation." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[8] Pinkney, Justin NM, and Doron Adler. "Resolution dependent gan interpolation for controllable image synthesis between domains." arXiv preprint arXiv:2010.05334 (2020).

[9] Siarohin, Aliaksandr, et al. "Animating arbitrary objects via deep motion transfer." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.