A Novel Hairstyle Transfer Method using Contextual Transformer Blocks

Dongjin, Huang

Shanghai Film Academy, Shanghai University, djhuang@shu.edu.cn

Zhenyan, Li

Shanghai Film Academy, Shanghai University, lizhenyanny@163.com

Jinhua, Liu

Shanghai Film Academy, Shanghai University, jinhua1427@hotmail.com

Abstract. Hair editing is a research highlight of computer vision. Due to its complex structure, synthesizing and editing realistic fine-grained hair is a very challenging task. In order to improve the visual quality and realism of generated images, we propose a hairstyle transfer method based on generative adversarial networks by using the Contextual Transformer Blocks. Firstly, we design a Hair Segmentation Module (HSM) that embeds Contextual Transformer Blocks in convolutional operations to reconstruct the segmentation architecture, further enhancing visual representation and making images produce more refined hairstyle structures and hair color quality. Secondly, we propose a Hair Mapper Module (HMM) that maps hair color information to higher-level semantic information by changing the direction of the mapper, to achieve color consistency with the reference image. Finally, we introduce new hair color loss function to manipulate the hair in a decoupled manner. Numerous experiments on the CelebA-HQ dataset have shown that our method can generate images with higher consistency in hairstyle and hair color. Compared with state-of-the-art methods, our method has best comprehensive performance.

CCS CONCEPTS • Artificial intelligence • Computer vision

Additional Keywords and Phrases: Hairstyle Transfer, Generative Adversarial Networks, Hair Segmentation Module, Hair Mapper Module

1. Introduction

The hair, as an important component of the face and a prominent feature of human image, has attracted many research works dedicated to hair modeling and synthesis. Due to the rapid development of generative adversarial networks (GANs)[1], which can generate realistic and vivid images, hair editing based on GAN has become possible.

Many internet experts consider using generative adversarial networks to achieve hair editing in the image or text domain as input, in order to meet people's visual needs of automatically changing hair. In the field of hairstyle transfer, several noteworthy related works include LOHO[2], MichiGAN[3], and HairCLIP[4]. Due to the unique geometric shape and material characteristics of hair, it is more difficult and challenging to characterize. Firstly, the complex boundaries of the hair region make it difficult to segment the hair shape from the background. Secondly, the hair in the source image and reference image is different in position and shape, making it hard to achieve perfect hair alignment. After hairstyle transfer, there may be misalignment situations; Thirdly, when a large number of areas in the source image are obstructed by the original hair, while converting long hair to short hair, the

shoulders and background obstructed by long hair need to be repaired; Finally, the hair texture is not refined enough after the hairstyle transfer, and local structural details are lost to some extent.

To solve these problems, we proposed a novel hairstyle transfer method based on generative adversarial networks by using the Contextual Transformer Blocks. Initially, we proposed a Hair Segmentation Module (HSM) to generate more refined hair segmentation results. Then, we designed a Hair Mapper Module (HMM) to maintain color consistency with the reference. Finally, we introduced customized loss functions for hairstyle transfer.

2. Related work

2.1. Generative Adversarial Networks

Due to the powerful modeling ability of deep neural networks, various generation models for learning unconditional image synthesis have been proposed one after another. Due to GAN's ability to model fine-grained details, it has been proven to be more popular. GAN[1] can convert latent code into images by learning the underlying distribution of training data, and generate fake images with the same distribution as natural images in the target domain. GAN has achieved success in computer vision applications such as image to image translation and video generation.

Several state-of-the-art generative adversarial networks have shown significant improvements in visual quality and diversity of samples. For example, ProgressiveGAN[5] first utilizes a progressive generation network structure. StyleGAN[6] further merges latent and noise information into shallow and deep layers, and StyleGAN2[7] modifies the shortcomings of the StyleGAN algorithm to produce better results. These network architectures can generate high-quality images that cannot be distinguished from real images, especially in the field of facial recognition.

2.2. Hair editing and hairstyle transfer

For the algorithms where conditional input from the image domain, hair editing is essentially hair conversion. Given the hairstyle reference image and hair color reference image, the purpose of hair conversion is to convert their corresponding attributes into the source image while preserving other features of the source image. MichiGAN's[3] the first conditional image generation method only focuses on hair editing. LOHO[2] proposes a normalization-based hairstyle latent optimization method. However, these algorithms perform hairstyle transfer by directly copying in the spatial domain, when the input image and reference image pose are not consistent, they will result in poor visual quality.

With the flourishing development of cross modal vision and natural language processing, text guided hair editing has also become possible. TediGAN[8] has pioneered text driven image manipulation. StyleCLIP[9] explores the power of using contrastive language image pretraining model CLIP[10] to achieve text-based image manipulation. HairCLIP[4] has been improved on the basis of StyleCLIP. It designs a unified hair editing framework to achieve the goal of hairstyle transfer under different input conditions. However, the segmentation algorithm is not precise enough and cannot achieve the transformation of the real hair structure in the reference image.

3. Method

3.1. Overview

We improve on the HairCLIP algorithm[4] and propose a novel hairstyle transfer method based on generative adversarial network by using Contextual Transformer Blocks[12]. The algorithm improvements include two modules: the Hair Segmentation Module (HSM) and the Hair Mapper Module (HMM). The entire network architecture is shown in Figure 1. Utilizing the powerful synthesis ability of pretrained StyleGAN[6], learn an additional mapper network to achieve hair editing functionality. Considering real image editing, for the input image we use the StyleGAN inversion method "e4e"[11] to obtain its latent code w in $W+$ space; For the reference images of

hairstyle and hair color, we encode them using a CLIP image encoder[10] to obtain the hairstyle condition input $\boldsymbol{e_s}$ and hair color condition input $\boldsymbol{e_c}$ of a 512-dimensional vector. In the Hair Mapper Module (HMM), we change the direction of the mapper by pointing the hairstyle condition information $\boldsymbol{e_s}$ to the hair mappers $\boldsymbol{M_c}$ and $\boldsymbol{M_f}$, and the hair color condition information $\boldsymbol{e_c}$ to the mapper $\boldsymbol{M_m}$. Afterwards, a mapping network consisting of three hair mappers predicts latent code changes $\boldsymbol{\Delta w}$ based on latent code w and editing condition inputs. Finally, the modified latent code $\boldsymbol{w' = w + \Delta w}$ is fed back to the pretrained StyleGAN to obtain the target editing results.

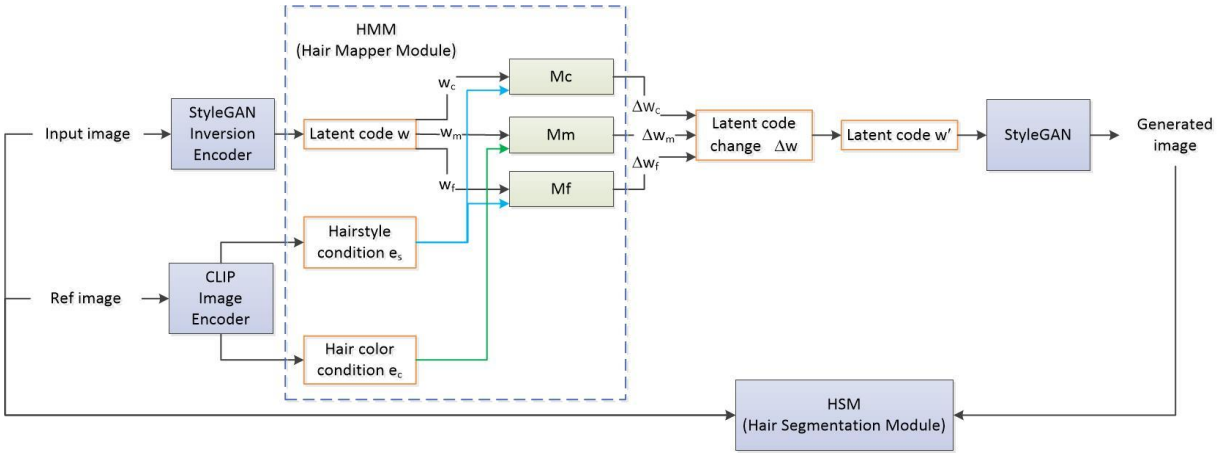

Figure 1. The overview of our network architecture

3.2. Hair Segmentation Module (HSM)

In the Hair Segmentation Module (HSM), we adopt more advanced segmentation algorithms and add Contextual Transformer Blocks to the convolutional modules that repeatedly appear in the new segmentation algorithm. Therefore, our Hair Segmentation Module has stronger visual representation learning ability, which can achieve an improvement in hair segmentation accuracy in experiments, especially in the visualization effect of hairstyle and hair color transfer. Our HSM can not only segment facial hair from reference images, but also assist in monitoring by calculating the hairstyle loss and hair color loss when only converting hairstyles between the generated and reference images.

3.2.1. Hair Segmentation. Semantic segmentation is an important branch of image processing and machine vision. For the hairstyle transfer of facial images, hair segmentation is also an extremely important module, which is related to the shape, structure, color, etc. of the converted hairstyle. Using the segmentation algorithm in HairCLIP always loses some edge information of hair, so we use a more advanced segmentation algorithm UNet++[13] and add convolutional operations with embedded Contextual Transformer Blocks which can enhance visual representation learning.

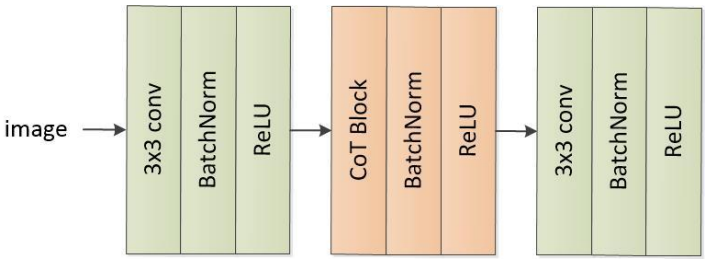

Figure 2. Embedding the Contextual Transformer Block (CoT Block) in the segmentation algorithm.

UNet++ has receptive fields of different sizes, which can capture features at different levels. The features with large receptive fields can easily recognize large objects, while the features with small receptive fields can recognize the edge information of large objects and the small objects themselves. Thereby it can recover the information lost by repeated down-sampling and up-sampling of deep networks, achieving an improvement in accuracy. From the UNet++ network, it can be seen that regardless of the down-sampling process or up-sampling process, each layer undergoes two consecutive convolution operations, which are repeated many times in the network structure. Because the design of the Contextual Transformer Block is a unified self-focused building block, we add a new convolution operation in two consecutive convolution operations in UNet++, including 3*3 convolution replaced by the Contextual Transformer Block, BatchNorm, and ReLU to further enhancing the visual representation. Our design is shown in **Figure 2.**

3.2.2. Contextual Transformer Block. As a new attention structure, the Contextual Transformer Block[12] aggregates the mining of contextual information and the learning of self-attention into a structure, fully utilizing the static contextual information inside the key to guide the learning of self-attention, further enhancing the visual representation ability. So embedding it into segmentation algorithms can help improve segmentation accuracy. The structure is shown in **Figure 3**. Firstly, a 3*3 convolution is used to model the local static contextual information K_1, fully utilizing the contextual information of the key; Then concatenate the query Q and local static contextual information modeled key K_1, and use two consecutive 1*1 convolution operations W_θ and W_δ to self-attention to obtain matrix A:

$$A = [K_1, Q]W_\theta W_\delta \tag{1}$$

Multiply Attention Map A and Value Map V to generate global dynamic contextual information K_2:

$$K_2 = V \circledast A \tag{2}$$

Finally, an SE attention module is used to fuse local static contextual information K_1 and global dynamic contextual information K_2 as output:

$$output = SE([K_1, K_2]) \tag{3}$$

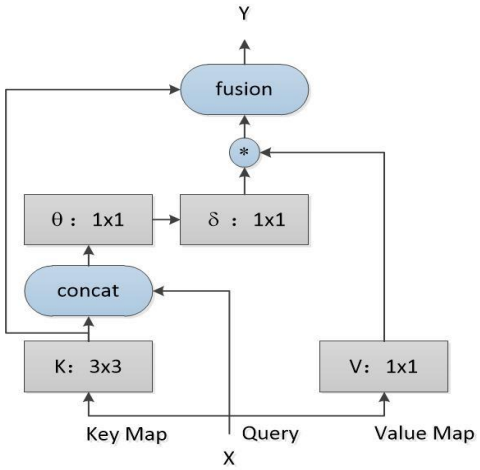

Figure 3. The structure of Contextual Transformer Block

3.3. Hair Mapper Module (HMM)

We explicitly separate hairstyle and hair color information and feed them to different hair mappers based on their semantic levels. After using HSM for hair segmentation, the shape and structure of the hair have been well transformed, but the visual conversion effect of hair color is not better. Due to the limited semantic information of low-level features in the image and the rich semantic information of

high-level features, we have pointed the hair color towards the higher-level layer to further improve color consistency.

The different layers of StyleGAN[6] correspond to different levels of semantic information in the generated image, with the more preceding layers corresponding to higher levels of semantic information. We use three hair mappers M_c, M_m, and M_f with the same network structure in HairCLIP[4] to predict latent code changes in different parts. These three mappers point to the high semantic layer, medium semantic layer, and low semantic layer, respectively. So we embed hairstyle information e_s from CLIP as the conditional input for M_c and M_f, and embed hair color information e_c from CLIP as the conditional input for M_m. The specific information of HMM is in **Figure 1**.

3.4. Loss Functions

For hairstyle transfer, hairstyle loss and hair color loss are two important operational losses. We introduce below loss functions to achieve various operations in a decoupled manner.

Hairstyle loss. We utilize the powerful potential of CLIP[10] to describe the similarity between hairstyles before and after conversion:

$$\mathcal{L}_s = 1 - \cos(E_i(x_M * P_h(x_M)), E_i(x * P_h(x))) \tag{4}$$

where the image being manipulated $x_M = G(w + M(w, e_s, e_c))$, G denotes the StyleGAN generator, w represents the latent code generated by the e4e inversion encoder[11], M means the hair mapper, $e_s = E_i(x * P_h(x))$, P denotes a hair segmentation network using the HSM module, $P_h(x_M)$ represents the mask of the x_M hair area, x means the given reference image, E_i represents encoding using CLIP's image encoder.

Hair color loss. We refer to the SmoothL1Loss function to calculate the average color difference of the hair region between the reference image and the processed image as the hair color loss:

$$\mathcal{L}_c = \frac{1}{n}\sum_{i=1}^{n} \begin{cases} .5 * \left(avg\left(x_M * P_h(x_M)\right) - avg\left(x * P_h(x)\right)\right)^2, if \left|avg\left(x_M * P_h(x_M)\right) - avg\left(x * P_h(x)\right)\right| < 1 \\ \left|avg\left(x_M * P_h(x_M)\right) - avg\left(x * P_h(x)\right)\right| - 0.5, otherwise \end{cases} \tag{5}$$

In summary, image operation loss is defined as:

$$\mathcal{L}_i = \lambda_s \mathcal{L}_s + \lambda_c \mathcal{L}_c \tag{6}$$

where λ_s, λ_c are set to 5, 0.02 respectively by default.

4. Experiments

4.1. Implementation details

We use StyleGAN2[7] which was pretrained on the FFHQ dataset[6] as the generator. And we adopt "e4e" as inversion encoder[11]. Our hair mapper is trained and evaluated on the CelebA-HQ dataset[5], which is also used to provide reference images for hairstyles and hair colors. Our training dataset includes 24176 images, and the test dataset includes 2000 images. During training, the hair mapper is randomly assigned tasks to edit only the hairstyle or only hair color, or both.

For the training environment, our model is trained on an NVIDIA GeForce RTX 3080 GPU with PyTorch 1.7 and Cuda 11.0. For the training strategy, our basic learning rate is 0.0005, batch size is 1, and the number of training iterations is 500000. We use the Adam optimizer, β_1 and β_2 set to 0.9 and 0.999, respectively.

For all comparison methods, we use official training codes or pretrained models. We compare our method with the SOTA algorithms in the current hairstyle transfer field, such as LOHO[2], MichiGAN[3], and HairCLIP[4]. When comparing with the LOHO algorithm, we use Graphonomy[14] to obtain the hair mask, and use the GatedConv[15] to repair the background of the image. In comparison, all hyperparameters and configuration options remain at default values.

4.2. Quantitative comparison

In order to quantitatively evaluate the effect of hairstyle transfer and the visual quality of generated images, we use three evaluation metrics: Frechet Inception Distance (FID), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity (SSIM). We compare our method with LOHO, MichiGAN, and HairCLIP. The quantitative evaluation results are shown in **Table 1.** We can see that our proposed method has lower FID and higher PSNR, which is superior to other methods in terms of FID and PSNR evaluation metrics. Overall, our method can generate images with higher visual quality and it has the best comprehensive performance.

Table 1. Quantitative evaluation results of different algorithms. ↓ indicates that lower is better, ↑ indicates that higher is better.

Methods	FID↓	PSNR↑	SSIM↑
LOHO	71.04	15.45	0.62
MichiGAN	36.70	16.14	**0.71**
HairCLIP	28.52	13.84	0.58
Ours	**26.81**	**16.15**	0.65

4.3. Qualitative comparison

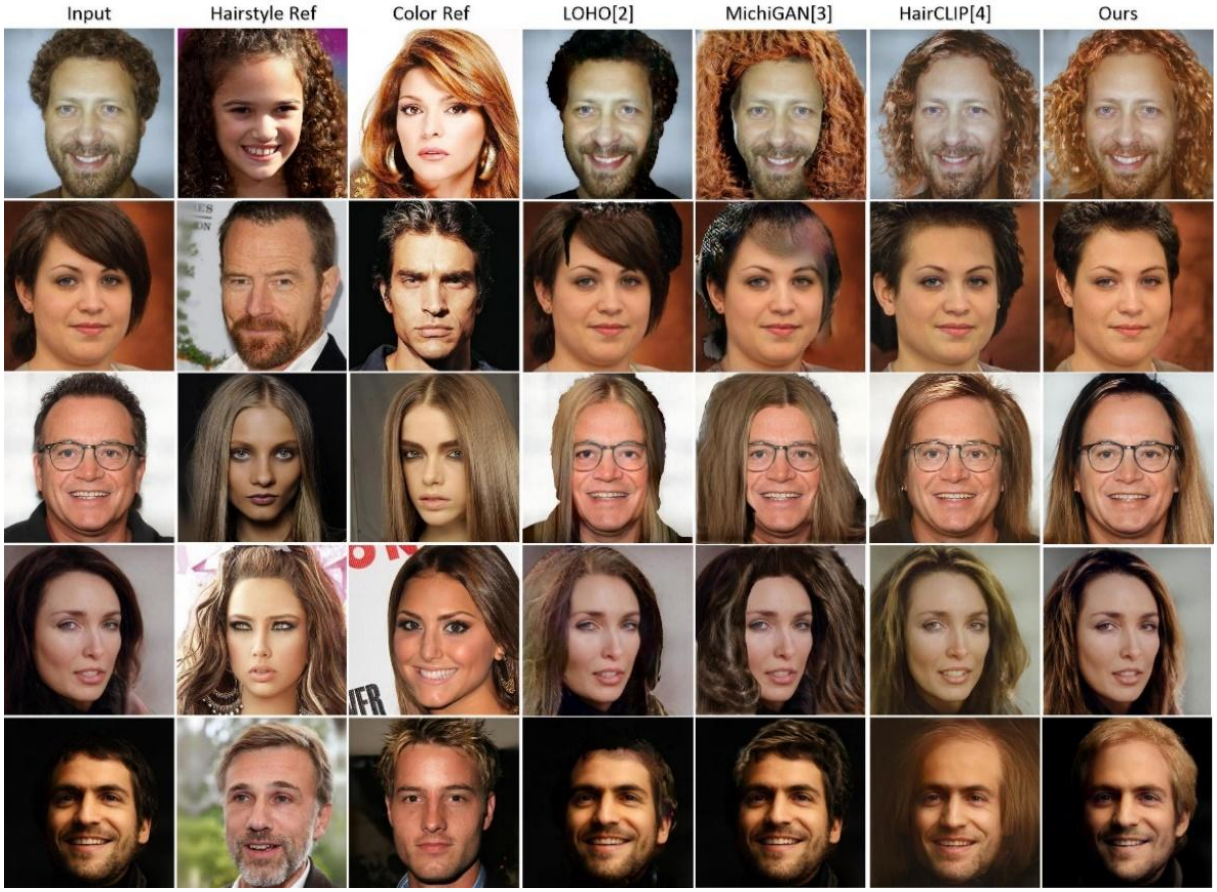

Figure 4. Visualization results of hairstyle transfer using different algorithms

The visualization results of hairstyle transfer are shown in **Figure 4**. The results indicate that compared to other methods our method has smoother visual effect regardless of the differences of gender, hair length and hair color. LOHO and MichiGAN perform hairstyle transfer by directly replicating in the spatial domain. From five sets of data, we can see that when the reference image

hairstyle is not well aligned with the input image hairstyle, hairstyle transfer can result in misalignment. Our method inherits the idea of HairCLIP, converts the similarity metric space into the latent space of CLIP, and uses the embedding of the reference image hair region in CLIP as the conditional input, so it can solve the problem of posture misalignment. Compared to HairCLIP, we have made further improvements in hair segmentation, and we can see that both hairstyle and hair color maintain structural and color consistency with the reference image.

4.4. Ablation study

To verify the effectiveness of our proposed Hair Segmentation Module (HSM) and Hair Mapper Module (HMM), we alternately remove one of the key components to retrain the variants of our method, keeping all components except for the key components unchanged. **Table 2** shows the quantitative results of the ablation study. It can be seen that removing any module will result in worse evaluation metrics, indicating the necessity of our improvement.

Table 2. Quantitative evaluation results of the ablation experiments. ↓ indicates that lower is better, ↑ indicates that higher is better.

Methods	FID↓	PSNR↑	SSIM↑
Ours	**26.81**	**16.15**	**0.65**
w/o HSM	33.14	15.29	0.60
w/o HMM	33.66	13.95	0.58

5. Conclusions

In this paper, we propose a novel hairstyle transfer method based on generative adversarial networks by using the Contextual Transformer Blocks. Using our proposed Hair Segmentation Module and Hair Mapper Module, we can generate results with the best visual quality for hairstyle transfer. Experiments have verified the effectiveness of our method.

References

[1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y. (2014) Generative adversarial nets. Advances in neural information processing systems., 27.

[2] Saha, R., Duke, B., Shkurti, F., Taylor, G.W., Aarabi, P. (2021) Loho: Latent optimization of hairstyles via orthogonalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1984-1993.

[3] Tan, Z., Chai, M., Chen, D., Liao, J., Chu, Q., Yuan, L., Tulyakov, S., Yu, N. (2020) Michigan: multi-input-conditioned hair image generation for portrait editing. arXiv preprint arXiv:2010.16417.

[4] Wei, T., Chen, D., Zhou, W., Liao, J., Tan, Z., Yuan, L., Zhang, W., Yu, N. (2022) Hairclip: Design your hair by text and reference image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18072-18081.

[5] Karras, T., Aila, T., Laine, S., Lehtinen, J. (2017) Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv: 1710. 10196.

[6] Karras, T., Laine, S., Aila, T. (2019) A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401-4410.

[7] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T. (2020) Analyzing and improving the image quality of stylegan. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8110-8119.

[8] Xia, W., Yang, Y., Xue, J.-H., Wu, B. (2021) Tedigan: Text-guided diverse face image generation and manipulation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2256-2265.

[9] Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., Lischinski, D. (2021) Styleclip: Text-driven manipulation of stylegan imagery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2085-2094.
[10] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J. (2021) Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748-8763.
[11] Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., Cohen-Or, D. (2021) Designing an encoder for stylegan image manipulation. ACM Transactions on Graphics (TOG)., 40: 1-14.
[12] Li, Y., Yao, T., Pan, Y., Mei, T. (2022) Contextual transformer networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence., 45: 1489-1500.
[13] Zhou, Z., Rahman Siddiquee, M.M., Tajbakhsh, N., Liang, J. (2018) Unet++: A nested u-net architecture for medical image segmentation. arXiv preprint arXiv:1807.10165.
[14] Gong, K., Gao, Y., Liang, X., Shen, X., Wang, M., Lin, L. (2019) Graphonomy: Universal human parsing via graph transfer learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7450-7459.
[15] Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S. (2019) Free-form image inpainting with gated convolution. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4471-4480.