Improving quality of pixel-wise transfer using Abortion method

Ohyul Kweon[1,2], Seoyeon Bang[1,2], Jehyun Yang[1,2], Wonseok Oh[1,2]
[1]Korea University, [2]KUCC

Abstract

This paper proposes a more efficient E2Style model using the abortion method. In the training process, we exclude inefficient iterations by prompting an abortion upon meeting certain conditions. This novel approach has proven effective in addressing the issue of overfitting and improves upon other aspects by comparing previous models.

1 Introduction and Related works

The emergence of deep-learning-based image processing technology has given rise to countless new studies utilizing GAN(Generative Adversarial Networks) which generates new images from input images. GAN structure consists of the generator and the discriminator in an adversarial training process. GAN model learns the latent space, the distribution of the latent vectors of the input image. The encoder converts image to feature vector, while the decoder reconstructs these features - *latent vectors* to new image. GAN inversion is the process of finding the latent vector that allows us to derive an output image most similar to the input. StyleGAN [1] emerged with its novel approach of incorporating a style-based generator. With the use of its new generator based on style transition, StyleGAN made possible the control over not only overall characteristics but also elaborate details - *skin color, age, gender etc*. E2Style [2] aims to improve the efficiency and effectiveness of StyleGAN inversion. E2Style's process is as follows: First, its encoder network takes into account the hierarchical structure to expect the latent vectors. These are extracted from the encoder's various spatial levels, and with the various details of the pre-trained StyleGAN generator, the learning difficulty is lowered. Second, E2Style accepts shared efficient prediction heads for each level. This includes global average pooling layers of varying size and a full-connected layer, resulting in a more lightweight and efficient network.

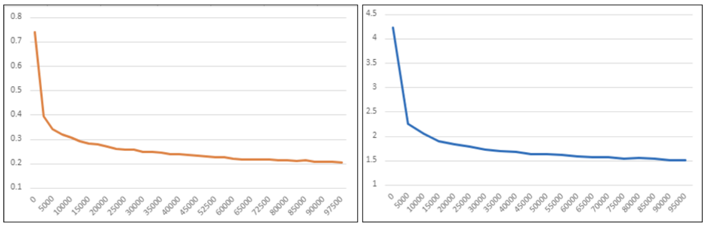

Figure 1: Change of loss values in pSp(*left*) and E2Style(*right*)

Fig.1 shows a gradual but continuous decline in loss function values. We identify points where the change in the loss value has slowed down and become trivial. These suggest the potential occurrence of overfitting in the training set. To address this issue while maintaining the quality of generated images, we propose a novel approach. We highlight the iteration process in E2Style's hierarchical structure, which proposed methods of improving the efficiency and effectiveness of StyleGAN inversion. We introduce the proposal of reducing the fixed number of iterations from the basic E2Style method. This is expected to ease the burden of training calculation as well as lessen the chances of overfitting in training datasets.

2 Loss functions and Abortion method

2.1 Loss function

We adopt the same functions introduced by E2Style. E2Style's loss function can be largely divided into two parts.

Common losses. Common losses consist of $\mathcal{L}_2$ loss and $\mathcal{L}_{LPIPS}$ [3] loss . $\mathcal{L}_2$ refers to the difference between the input image and the output image, which has been reconstructed through the encoder and the StyleGAN generator, acts as decoder, on a pixel level.

$$\mathcal{L}_2 = \|\mathbf{x} - D(E(\mathbf{x}))\|_2 \tag{1}$$

Using $\mathcal{L}_2$ solely is insufficient to discern the features of the reconstruction result. Therefore, we derive the feature-level loss through the additional use of $\mathcal{L}_{LPIPS}$ loss.

$$\mathcal{L}_{LPIPS} = \|F(\mathbf{x}) - F(D(E(\mathbf{x})))\|_2 \tag{2}$$

Multi-Layer Loss. Multi-Layer Loss consists of Identity Loss and Parsing Loss between multi layer outputs. In GAN inversion, it is crucial to conserve the identity information of the original image's attributes. Multi-Layer Identity Loss refers to maintained consistency between the input image and the inverted output image

$$\mathcal{L}_{ID} = \sum_{k=1}^{5}(1 - cos(N_f(\mathbf{x}), (N_f(D(E(\mathbf{x}))))) \tag{3}$$

cos refers to cosine similarity. N_f(x) is the feature that corresponds to semantic level k in the facial recognition network N [4] for image **x**. The Parsing Loss function in the Multi-Layer works in the opposite way from Identification Loss by separating different features.

$$\mathcal{L}_{PAR} = \sum_{k=1}^{5}(1 - cos(P_k(\mathbf{x}), (P_k(D(E(\mathbf{x}))))) \tag{4}$$

Likewise, P_k(x) refers to the feature that corresponds to the kth semantic level in the pre-trained facial parsing network P [5] for input image **x**. Parsing loss operates in tandem with Identity loss in a complementary way, enabling the two loss functions to operate as each feature in the multi layer forms clusters.

Summary. To sum up, the value of the total loss function can be expressed as below:

$$\mathcal{L}_{total} = \lambda_1\mathcal{L}_2 + \lambda_2\mathcal{L}_{LPIPS} + \lambda_3\mathcal{L}_{ID} + \lambda_4\mathcal{L}_{PAR} \tag{5}$$

2.2 Abortion method

In this paper, we proposes the abortion technique in the E2Style training process. Previous E2Style does not allow for flexible adjustment of the number of iterations. So it is expected a vast load of data in order to extract the images, and also lies the risk of an overfitting in the training set. Our approach proposes two type of abortion methods to counter such drawbacks, through which we expect more efficient results.

The first method is **relative method**. We determine a relative call condition for the abortion function. In this method we begin by designating previous loss = ∞. After the 1st iteration, we calculate the value of loss and subtract it from the previous loss. Here, if the result is positive, we reset the value of *abort_count* to 0, because it has yielded a result image of higher quality. Otherwise, we continuously add 1 to the value of *abort_count*. The previous loss, which we denoted as ∞ prior to the 1st iteration will now be updated to the newly derived loss. Once count reaches 10 after a series of reiterations, the abortion function is called and *abort_count* is reset. The abortion function then indefinitely reduces the number of repetitions the input images will undergo.

The second is the **absolute method**, in which we set a random parameter a. This parameter acts as an absolute criterion that determines whether to reset *abort_count* to 0 or proceed to add increments of 1. Then, the value of loss is derived by continuously adding one iteration until abortion occurs. If the value of (a − loss) is positive, this, as mentioned above results in a higher quality image and therefore resets *abort_count* to 0. Otherwise, *abort_count* will increase by increments of 1. Likewise, *abort_count* exceeding 10 will call the abortion function and result in *abort_count* being reset. This process is as shown in Fig. 2.

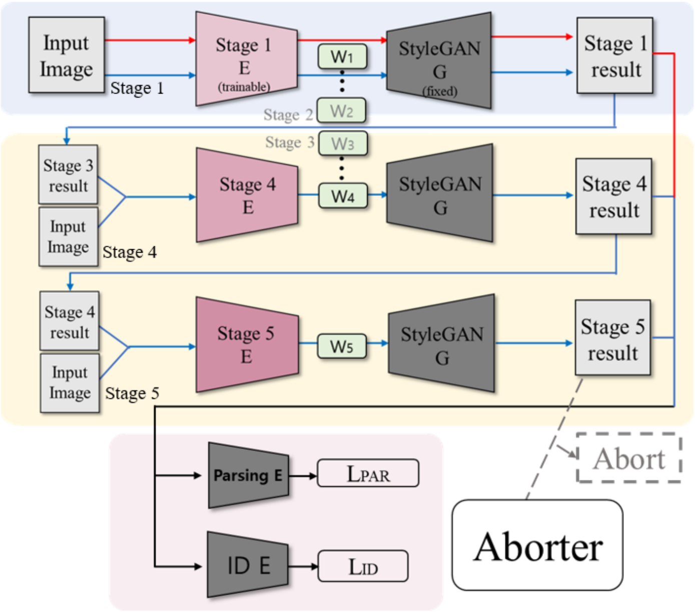

Figure 2: E2Style with Abortion. Abortor prevents overffitng by truncating the last stage under certain conditions.

By applying the new methods proposed in this paper, we can reduce the amount of calculations by excluding inefficient iterations, and expect positive results by avoiding overfitting issues with our training dataset.

In terms of implementation, the method of reducing iteration should be approached thoughtfully. In particular, when using absolute abortion method, setting a high threshold may result in inappropriate abortion which occurs before the sufficient training. It leads to sub-optimal performance due to an insufficiently trained encoder. Therefore, careful consideration and optimization of the threshold value are necessary to ensure the training process which achieves the desired performance while avoiding potential negative impacts.

3 Experiment and Result

To demonstrate the benefits of abortion in the training, we compared our module with the existing modules using multiple images.

Implementation detail. For absolute abortion method to be used, prior knowledge of the loss function in E2Style is necessary. Therefore, in our experiments, we used the relative abortion method instead. The initial weights for the weighting factor of each loss function were set as 1, 0.8, 0.5, and 1, respectively. We trained our model using a dataset of 25,000 images from

CelebAMask-HQ [6] and evaluated it on a randomly selected set of 5,000 images that were not used in the training set.

Results. Examples of the training results for our model and other models are shown below.

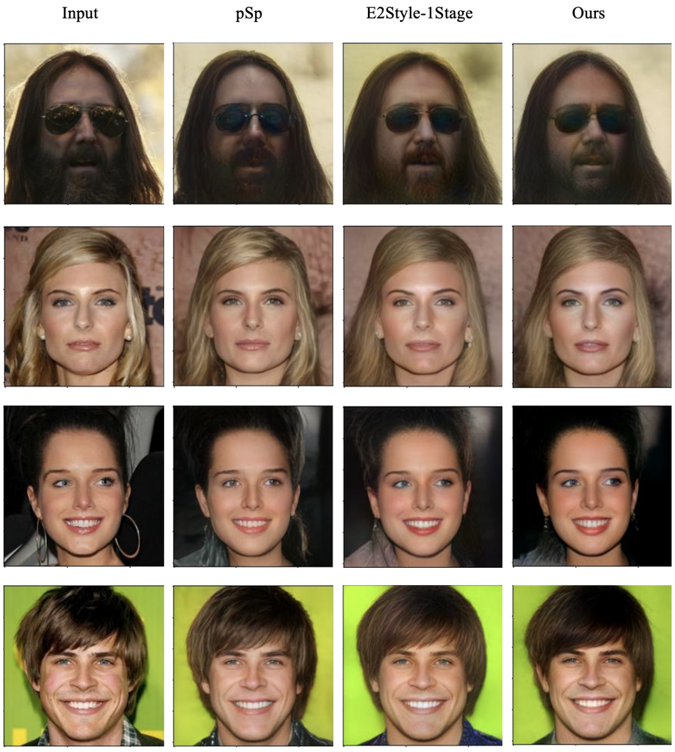

Figure 3: Visual comparison of the GAN inversion models

Fig.3 shows that our model demonstrates excellent detail reproduction of the original images. Our model excels at implementing small details such as visible teeth in an open mouth and directionally oriented pupils, which set our model apart from others. This result can be thought to have occurred by preventing overfitting to the training set through our abortion method.

In terms of the amount of computation for training, pSp model has only 1 fixed stage, and the E2Style model has N fixed stages. If solely one stage is passed to be trained, the performance will be lower than a model that passes through N-stages. However, a model with N fixed stages carries a greater risk of overfitting than a model that completes only one stage. Moreover, N-stages require N times more operations on all training sets, resulting in a slower learning speed. Our model is implemented in a direction that overcomes these two drawbacks, starting from the N-stages and gradually reducing the number of stages, maintaining performance level similar to that of a model that completes the N-stages while dealing with overfitting and reducing the computational load.

4 Conclusion

We suggest a novel approach to improve the E2Style model by introducing the abortion technique to allow model to process dynamic stages to prevent overfitting. Our model demonstrates three advantages over the previous model, namely, **i**. reducing overfitting, **ii**. decreasing computational complexity compared to the fixed N-stage E2Style model, and **iii**. preserving fine-grained details of the original images.

References

[1] Tero Karras, Samuli Laine, Timo Aila. (2019). A Style-Based Generator Architecture for Generative Adversarial Networks. eprint arXiv:1812.04948, ()

[2] Tianyi Wei, Dongdong Chen, Wenbo Zhou, Jing Liao, Weiming Zhang, Lu Yuan, Gang Hua, Fellow, IEEE, NEnghai Yu. (2022). E2Style: Improve the Efficiency and Effectiveness of StyleGAN Inversion. Wei-2022, 3267-3280

[3] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 586–595, 2018.

[4] J. Deng, J. Guo, and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4685–4694, 2019.

[5] Z. Liu, https://github.com/switchablenorms/CelebAMask-HQ/tree/ master/face parsing, accessed: Mar. 2021. [Online].

[6] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of gans for improved quality, stability, and variation,” ArXiv, vol. abs/1710.10196, 2018.