Preprint of the paper presented at IEEE Face and Gestures, 2021.

Hairstyle Transfer between Face Images

Adéla Šubrtová, Jan Čech and Vojtěch Franc
Faculty of Electrical Engineering, Czech Technical University in Prague

***Abstract*— We propose a neural network which takes two inputs, a hair image and a face image, and produces an output image having the hair of the hair image seamlessly merged with the inner face of the face image. Our architecture consists of neural networks mapping the input images into a latent code of a pretrained StyleGAN2 which generates the output high-definition image. We propose an algorithm for training parameters of the architecture solely from synthetic images generated by the StyleGAN2 itself without the need of any annotations or external dataset of hairstyle images. We empirically demonstrate the effectiveness of our method in applications including hair-style transfer, hair generation for 3D morphable models, and hair-style interpolation. Fidelity of the generated images is verified by a user study and by a novel hairstyle metric proposed in the paper.**

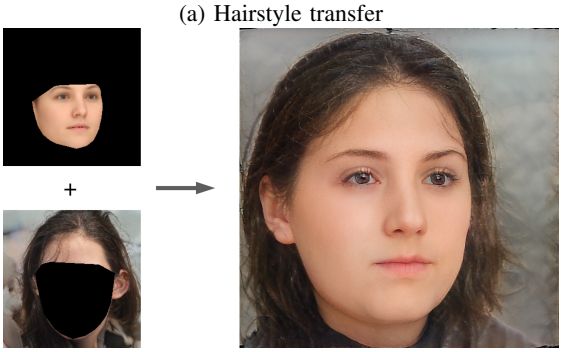
I. Introduction

Hair, or a lack of thereof, plays an important part in one's visual identity, and different hairstyles can vastly change the way one looks. It can be interesting to envision how one would look with a different hairstyle. It can be desirable to generate an image of oneself with a specific haircut before letting a hairdresser make a dramatic change. Alternatively, one can wish to edit their hair in a photo to share a visually pleasing picture on social media.

However, human hair is known for its complex intrinsic structure and variability. Until recently, synthesizing a realistic hair image was a challenging problem in computer graphics. As early as 1999, Blanz and Vetter introduced a 3D morphable model in their seminal paper [4] which had become the main tool for face synthesis. Fitting parameters of the 3D morphable models requires a point-to-point correspondence of the training faces, which is impossible to establish for hair strands, hence they are used only to model the inner part of the face. Recently, deep neural networks have been successfully applied as the underlying model for face synthesis. The main workhorses in this area are the Generative Adversarial Networks (GAN)s [12] and the DCGAN architecture [29]. The current state-of-the-art in face synthesis is then the StyleGAN2 achitecture [18], mapping a low-dimensional latent space into high-fidelity portrait images. However, the task of hair manipulation and generation is still not sufficiently addressed.

This paper aims to produce an end-to-end solution for hairstyle manipulation with applications such as hair generation for 3D morphable models and hairstyle transfer between images, see Fig. 1. We propose a neural network which takes two inputs, a hair image and a face image, and produces an output image having the hair of the hair image seamlessly merged with the inner face of the face image. Our architecture consists of two encoder networks, one for the inner face and one for the hair, that map the inputs into two embeddings. The embeddings are then merged and translated by a simple network to the latent code of StyleGAN2 generating the output image. The two encoders and the merging net are trained from synthetic images generated by the StyleGAN2 itself.

(a) Hairstyle transfer



(b) 3D morphable model

Fig. 1: Applications of the proposed method: Realistic hairstyle transfer between *unagligned* face images (a), and 3D morphable model hair generation (b). The results were achieved by running our neural network, without any post-processing or user interaction.

978-1-6654-3176-7/21/$31.00 ©2021 IEEE

There are three main contributions distinguishing our method from existing works. First, our method does not require any annotations or any external dataset of hairstyle images which are difficult to collect in large numbers. Second, we bypass the intricate GAN training by exploiting the pretrained StyleGAN2. Thus the generated images have a high resolution of 1024×1024. Third, we implement a hairstyle metric to objectively evaluate the fidelity of the generated images.

The rest of the paper is structured as follows. Related existing works are surveyed in Section II. In Section III, we present the novel architecture and the method for its training from synthetic images. A thorough empirical evaluation of the proposed method is given in Section IV. Lastly, the summary of the entire work is the subject of Section V.

II. Related Work

A significant number of methods have been proposed for the acquisition of models capturing hair geometry and its appearance from 2D images [28], [25], [9], [8], [35], [13], [14], [7]. The models have been used for various purposes like, for example, hairstyle transfer [9], [19], animation [8], or morphing [35]. A survey of modeling techniques for hairstyling, hair simulation, and hair rendering can be found in [34]. The geometric models often capture only a coarse hair geometry and its properties that do not allow for the generation of photorealistic images.

With the advent of deep learning, there has been a paradigm shift to using neural networks as the underlying model for face synthesis. An efficient algorithm for learning deep generative models was introduced in the seminal work [12] which proposed the Generative Adversarial Networks (GANs). The GANs allow to generate a high-dimensional distribution by transforming a fixed low-dimensional distribution via a neural network. The parameters of the network are learned such that the distribution of the high-dimensional outputs matches the training data. A surge of papers applying GANs to image modeling was initiated by a fully convolutional architecture DCGAN [29]. Since then, the quality of images synthetized by GANs has been steadily growing [15], [5], [16]. The current state-of-the-art is represented by the StyleGAN2 achitecture [18] which produces high-resolution photorealistic faces that are for humans hard to distinguish from real photos.

In our work, we use the StyleGAN2 as the underlying generator of the facial images. A similar idea was previously considered by others in e.g., [1], [2], [30]. In [1], the authors propose an embedding algorithm transforming a given photograph into the latent space of StyleGAN. The algorithm starts from a randomly generated latent code, which is then optimized by a gradient method until the output of StyleGAN matches the input image in terms of perceptual and L2 loss. The authors investigate how algebraic operations on the latent codes correspond to semantic processing operations on the images. The work [2] addresses the problem of interactive image editing via manipulation of latent codes. As the backbone generator, they use pretrained progessive GAN [15]. The latent codes are computed by the same iterative algorithm as in [1]. An encoder network directly mapping the input images into latent vectors which are fed to StyleGAN was proposed in [30]. The encoder is trained by minimizing a combination of L2-loss, perceptual loss, and identity capturing loss that are used to evaluate a similarity between the input and the generated images. The authors show that the direct encoding works on par or better than the methods optimizing the latent vector iteratively [1], [2]. In our work, we exploit the same idea of directly mapping the image to the latent vectors and we also use a similar loss function, however, we do not require real images for training.

Deep generative networks specifically designed for hairstyle modeling were proposed in [36], [26], [33]. The work [36] introduces *Hairstyle30k* database which is composed of 30k face images, each labeled by one out of 64 different hairstyles annotated by a semiautomated process. The database is exploited to train H-GAN model which is a variant of the Conditional Variational Auto-Encoder. Using the conditioning allows the H-GAN to explicitly control the hairstyle of the generated facial images. The authors show how to use their model for hairstyle classification and hair editing in portrait images. In our work, we use a different architecture and we do not require annotated faces for training. RSGAN [26] is a generative neural network designed for face swapping and edditing of facial attributes. Similarly to our approach, they train two encoding networks, one for the inner face and one for the hair region. The latent representations of the inner face and the hair region input the generator network which outputs the synthesized face image. In contrast, our approach uses a pretrained StyleGAN2 generator, while they train all components of the architecture from scratch using annotated face images. MichiGAN [33] is an interactive hair editing system for portrait images. The system is based on a complicated architecture which implements a sequential conditioning mechanism composed of three condition modules. The MichiGAN architecture allows for an orthogonal control over four hair attributes including shape, structure, appearance, and background.

To summarize, the main distinguishing feature of our approach compared to existing works is that our learning algorithm does not require any external set of training examples, instead it *learns solely from synthetic images* generated by the pretrained StyleGAN2. In contrast, the existing hairstyle modeling GANs, i.e. H-GAN[36], RSGAN [26] and MichiGAN [33], all require a large database of real faces, which in the case of H-GAN and RSGAN has to be annotated with the hairstyle. Collecting such training databases is difficult because it has to contain faces with a large variation of hairstyles, annotation of which is a tedious task with often inconsistent outcomes. Apart from this, our method generates photorealistic 1024×1024 images in contrast to existing architectures producing lower-resolution outputs, namely, 128×128 in case of H-GAN and RSGAN, and 512×512 in the case of MichiGAN.

A related problem to hairstyle transfer is face swapping,

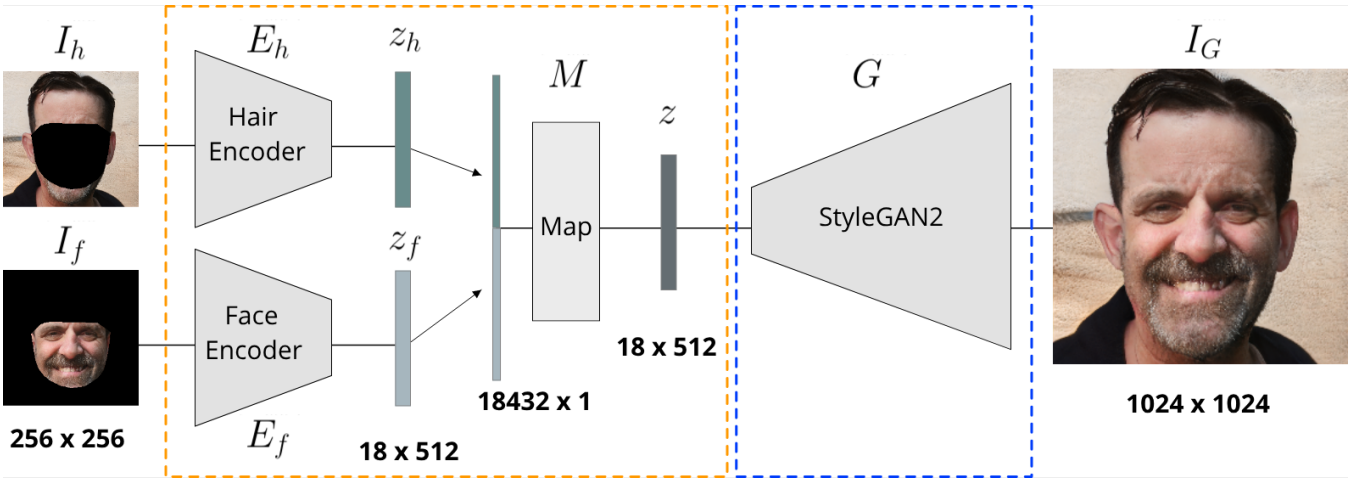


Fig. 2: Architecture of the proposed hairstyle transfer method. The networks in the orange rectangle are updated during training, the model in the blue rectangle is fixed.

e.g. [3], [27], which aims at transferring a face from a source photo to a target photo. However, the face swapping is achieved by a geometrical transform of the segmented inner part of the face without changing the hair area and hence an explicit model of the hairs is not required.

III. Method

We propose a neural network which takes two inputs, a hair image and a face image, and produces an output realistic portrait image having the hair of the hair image and the inner face of the face image.

A. Architecture

The architecture, depicted in Fig. 2, consists of two encoders E_h, E_f for each of the hair and the face inputs respectively, a mapping network M and a generator G.

The input images are of size 256×256 pixels. The hair image has the inner face masked, while the face image is masked complementarily. The mask is found as a convex hull of the 2D facial landmark [6].

The encoders map the input images I_h, I_f into embedding z_h, z_f of hair and face of dimension 18×512 each. The architecture of the encoders is ResNet-IR SE 50 adopted from the repository accompanying paper [30]. However, the paper [30] employs a single encoder of a different architecture based on feature pyramid network [23]. Then, a subtle mapping network M combines the flattened and concatenated embeddings z_h, z_f (dimension: $2*18*512 \times 1$) into latent code z of dimension 18×512. The mapping consists of a single fully connected layer with leaky ReLU activation of negative slope 0.2. Finally, the *fixed* StyleGAN2 [17] generator G produces the output image of resolution 1024×1024 pixels from the style space code z. Note that we use the extended 18×512 StyleGAN2 input $\mathcal{W}+$, following a recent practice for StyleGAN inversion, e.g., [1].

B. Training

During training, the weights of the generator G stay fixed, and only the encoders E_f, E_h, and the mapping network M are updated. The training is fully self-supervised and uses only synthetic data randomly generated from the StyleGAN2.

The network is trained in auto-encoder fashion. An input randomly generated image I is subsampled to 256×256 pixels and split into hair I_h and face I_f images using facial landmarks. Then the network is trained to generate an output image I_G that matches the input image $I = I_h + I_f$.

Formally,

$$I_G = G\Big(M\Big(E_h(I_h) \oplus E_f(I_f)\Big)\Big) \downarrow 256,$$

where $\oplus$ represents concatenation and $\downarrow 256$ subsampling to 256×256. The loss function consists of three components: Perceptual, Pixelwise, and Identity. Perceptual loss $\mathcal{L}_{LPIPS}$, that correlates well with human perception of similarity as reported in [37], is computed as

$$\mathcal{L}_{LPIPS} = \|F(I) - F(I_G)\|_2,$$

where F denotes features extracted from AlexNet. Pixel-wise $\mathcal{L}_{L_2}$ loss is

$$\mathcal{L}_{L_2} = \|I - I_G\|_2.$$

Identity loss $\mathcal{L}_{ID}$, to preserve person's identity between the generated and the target image, is computed as a cosine similarity between descriptors provided by ArcFace net [10]

$$\mathcal{L}_{ID} = 1 - \langle A(I), A(I_G) \rangle,$$

where A represents unit-length-normalized descriptors produce by ArcFace net, computed on cropped-out faces of the output and target images (subsampled to 112 x 112). The complete objective is

$$\mathcal{L} = \lambda_{LPIPS}\mathcal{L}_{LPIPS} + \lambda_{L_2}\mathcal{L}_{L_2} + \lambda_{ID}\mathcal{L}_{ID}.$$

The weights are set as: λ_{LPIPS} = 0.8, λ_{L_2} = 1.0, λ_{ID} = 0.1. The same setting was used in [30] for StyleGAN inversion. We also used the same optimizer, the Ranger optimizer, to train the network. Batch size was 2 only, due to the GPU memory limitation.

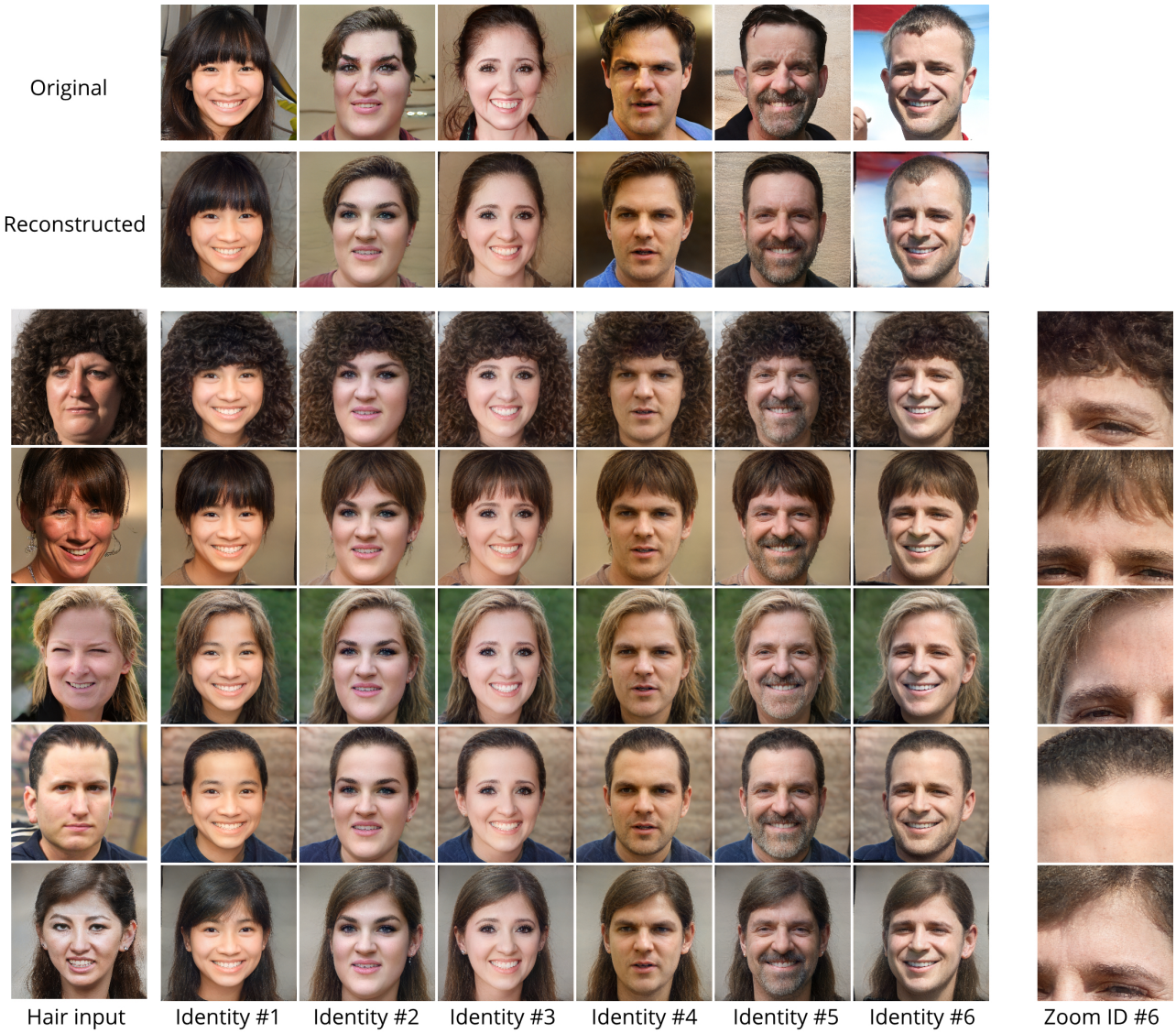


Fig. 3: Example of hairstyle transfer. The top two rows display the original identities and their reconstructions generated by our network. The remaining rows depict hairstyle transfer with various hair inputs. Identities are preserved column-wise. The rightmost column shows zoomed-in views of forehead regions of identity 6.

C. Dataset and Data Augmentation

We randomly generated 70,000 images using StyleGAN2 network. Each image was then manipulated in the latent space 18×512 to generate a 5-step-long sequence of yaw angles in range $[-25°,25°]$ with the same identity.

To manipulate, we followed a standard approach, e.g., [32], where a yaw angle of the face was estimated from facial landmarks [6], and a linear classifier to discriminate between positive and negative yaw angles was found. The normal of the discriminative hyperplane is the direction that changes the face yaw.

During training, the images were sampled as follows. For a particular identity, we randomly drew two yaw angles from the sequence, one for the hair image I_h and the other for the face image I_f. Thus, the input images have nonmatching hair and face poses. The ground-truth image I has the yaw of the face image I_f.

To achieve better results for input pairs with misaligned hair and face inputs, we used the following data augmentation. We first trained the network for 500,000 iterations with a random affine transformation of hair images I_h. Namely, translation $\pm$13px, scaling 0.85–1.15, and in-plane rotation $\pm 20°$ was uniformly sampled.

After the training, we observed that the network was sensitive to a dislocation of the input face. Input face images having a slightly different position, scale, or in-plane rotation resulted in certain artifacts. Therefore, we finetuned the trained network with further 60,000 iterations with face images I_f transformed with the same affine transformation as the hair images, except for a smaller $\pm 10°$ range of rotations. The same transformation was applied to the target images to preserve consistency.

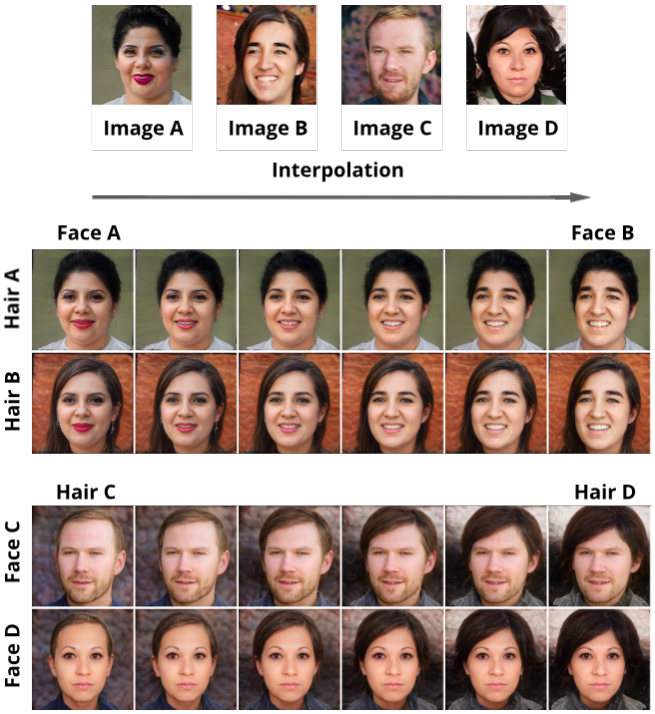


Fig. 4: Interpolation in the face and hair domains. Input images (top). Interpolation between identities A and B that keeps the original hair intact (middle), and interpolation between hairs C and D that keeps the original identity intact (bottom).

Interestingly, the network did not converge well when the inner face transformation was used from the beginning, and the resulting images suffered from artifacts. Typically, the contour of the hair leaked into the background. Two stage training solved the problem. The reason is probably that the network trained to capture a simpler transformation is able to guide the network to capture a more complex transformation.

IV. Experiments

We present several experiments to evaluate the consistency and fidelity of the generated hairstyles. The qualitative analysis includes hairstyle transfer, hair synthesis for 3D morphable model [4], interpolation, and comparison with other methods for face swapping. The quantitative evaluation was performed by using an independently learned hair similarity metric. To quantitatively assess the visual quality of our results, we conducted a user study. All experiments were done on held-out test set which was not used in the training.

A. Qualitative Analysis

1) Hairstyle Transfer: This experiment demonstrates the ability of the network to transfer hairstyles between different identities. The hairstyle transfer was executed on several image pairs by simply swapping the inner face inputs I_f.

In Fig. 3, we present several examples of hairstyle transfer between a variety of identities and haircuts. The pairs are challenging, since the poses of the original images and of the hair inputs are different, as well as the illumination. The input identities differ in head size and shape, in gender and ethnicity. Some of the input hair cuts are very complicated.

The hairstyles are faithfully transferred while still preserving the input identity and facial expression. The output images provide seamless hairstyle transfer despite the challenges. Note that, e.g., the skin color is well preserved and that even female hairstyles to male identities are transferred without apparent artifacts.

The reasons that the network can handle mismatching pairs of hair and inner face input images and produces appealing results are twofold. The robustness to geometrical misalignment is probably due to the complex data augmentation used in the training. The illumination consistency is probably due to the state-of-the-art StyleGAN2 generator embedded in our architecture. StyleGAN2 pretrained on a large amount of data produces photorealistic results with consistent illumination.

2) 3D Morphable Model: Since the introduction, the 3D morphable model (3DMM) has come a long way with increasingly detailed features. Nevertheless, the model is still missing hair. Our framework provides uncomplicated hair generation for faces rendered by 3DMM.

In Fig. 5, we show two examples using the proposed hairstyle transfer method and the 3DMM. First, we fitted a 3D morphable model to a test image and rendered a sequence of poses with increasing yaw angle. Each rendered image was masked to obtain the inner face image I_f. All images from the sequence were paired with the original image containing hair I_h. Those pairs were fed into the network to generate the output images. The other results show the same process with a randomly generated 3DMM. The rendered models and hair input image from the previous experiment were again fed in our network to provide a synthetic 3DMM model rendering with natural hair.

We can see that the hair rotates consistently with the input face. The identity is well preserved for all orientations. Only for extreme yaw angles, the output rotations seem slightly smaller than the inner inputs. The reason is probably a scarcity of extreme orientations in the training set. We only have images up to $25°$ in our datasets, since it was problematic to generate larger orientations without artifacts.

3) Interpolation: To uncover the behavior of the latent spaces (of hair and faces), we experiment with interpolation. A pair of images was taken, and the latent-space embeddings z_h, z_f of the respective hair and face encoders were found for both images. Then, we generated a sequence of several steps by linearly interpolating between either the latent codes of hair or face. We reconstructed the output images feeding the network with the interpolated codes of hairs, while keeping the face code the same or vice versa.

Results for both versions are presented in Fig. 4. Face interpolation shows a gradual change in identity without modifying the hairstyle. The hair is naturally adjusted to fit a face size. The hair interpolation shows progressive hair lengthening and darkening, while preserving the identity.

Fig. 5: 3D morphable model with hairs. The 3DMM was fit to a real image (a), or generated randomly (b). A sequence of yaw angles in range $[-30°, 30°]$ was rendered. Hair was added from the real image.

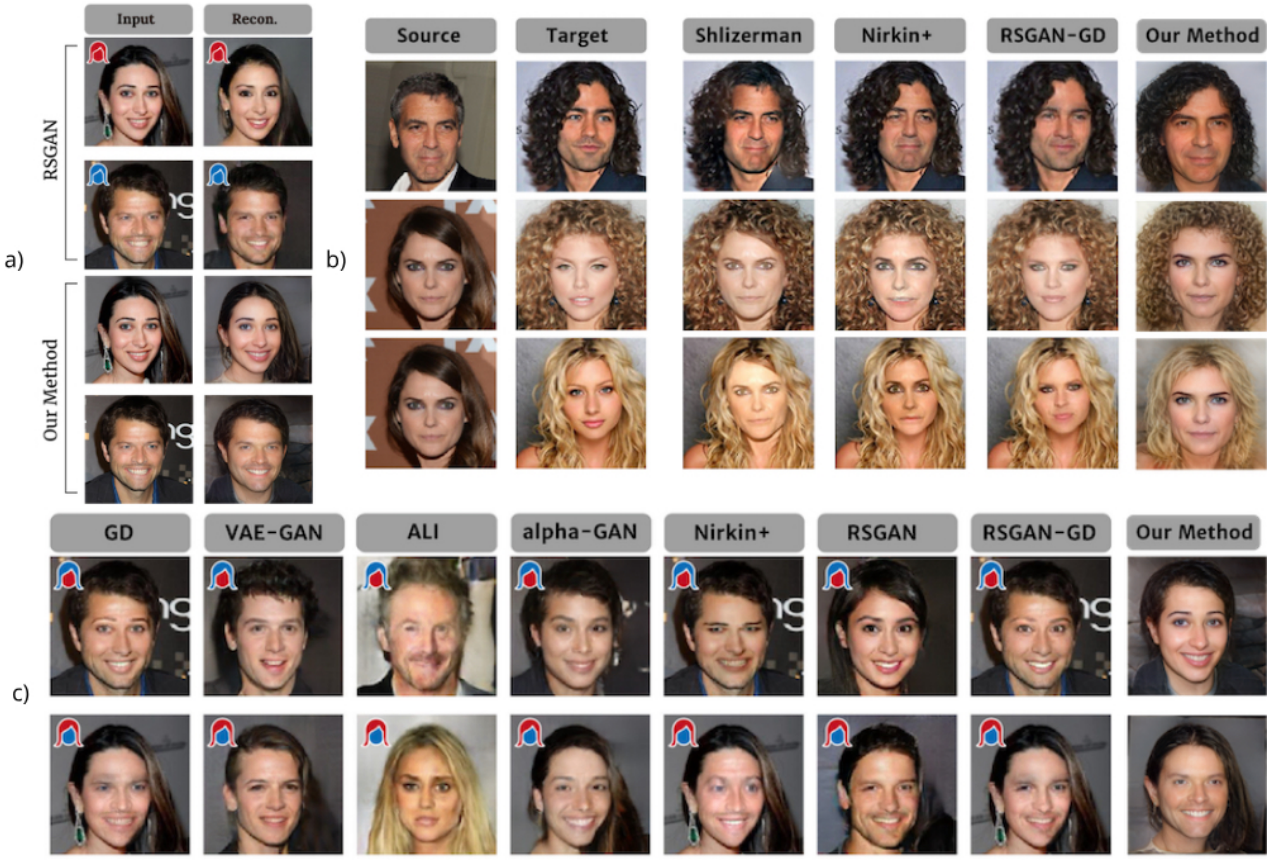


Fig. 6: Comparison with other methods. Reconstruction (a), and hairstyle transfer/face swapping (b), (c). All images except for our results were adopted from [26]. Note that in (b) our method provides hair transfer results onto face image of the source image unlike the other methods. Compared methods are from Shlizerman [19], Nirkin [27], GD [22], VAE-GAN [21], ALI [11], α-GAN [31].

4) Comparison with other methods: The closest works to ours are RSGAN [26] and MichiGAN [33]. The latter requires a user interaction, therefore cannot be easily compared. Comparison with RSGAN is shown in Fig. 6. The images are taken from the electronic version of the paper and then enriched by our results. This is a challenging scenario, since we do not have the choice of the images in our hands. The RSGAN presents two methods for face swapping/hair transfer; RSGAN and RSGAN-GD. RSGAN is the output of their network, whereas RSGAN-GD reconstructs only the facial part and blends it with the input background and hair using gradient-domain image stitching [22]. We compare

auto-reconstruction in Fig. 6 (a) and hairstyle transfer in (b) and (c). Note that our results in (b) have a different pose, since the pose is given by the source image and not by the target image. We simply fit the hairstyle to the input face and not the face to the hairstyle as other methods. This is given by our training process.

Our model achieves better results in reconstruction (a) and in preserving identity than RSGAN-GD, and Nirkin+ (b). Hairstyle fidelity in (b) is comparable among all images, nevertheless residuals of the original hair are visible in Shlizerman. In (c), we can see that our method outperforms other results, e.g. RSGAN fails to transfer long hair to male identity and vice versa.

B. Quantitative Evaluation

To evaluate the proposed method, we performed three experiments. First two experiments show a quantitative evaluation of the hair transfer fidelity and identity preservation of the generated image. The third experiment is a user study to assess the visual quality of the generated images of our method against random images from StyleGAN2 generator.

1) Hair Similarity: To the best of our knowledge, there is no hairstyle similarity metric available. Therefore, we trained a hair similarity metric on an annotated dataset of hair styles. Our method allows to train from databases with incompatible hairstyle annotations which was exploited in our case.

a) Hairstyle Dataset: We combined two annotated datasets; CelebAMask-HQ and a part of Hairstyle 30k - "Sixkind". In summary, about 20k images. For CelebAMask-HQ we only took images with positive annotation in at least one hair attribute (colors, straight, wavy, bald). The images with hats were discarded. The images with the same attribute vector were considered as the same class. The total number of classes was 19. For Sixkind, we choose only a subset of the dataset containing 6 classes, since some classes of the whole dataset do not contain many examples. In each image, we detected the face with dlib detector [20]. We increased the size of each bounding box by a factor of 0.75 on all sides and shifted the larger box up by a factor of 0.5 (of the original bbox size) to capture the whole haircut.

b) Training: We employed the architecture ResNet 50 without its last layer. The network was trained in Siamese setup using triplet loss with a margin equal to 1 and L_2 distance. Since the classes were not compatible across the datasets, each randomly drawn triplet contained only images from either dataset. We augmented each batch with "mirrored" triplets. These triplets were constructed from an anchor and a negative image; the positive example was created by horizontally flipping the anchor image. When feeding into the network, the images were resized to 224 $\times$ 224. We used the optimizer AdamW [24] with learning rate 0.0001 and weight decay 0.0001 for 300 epochs. Batch size was 32 with 8 more randomly chosen "mirror" triplets.

c) ROC curves: The learned hair similarity metric was used to evaluate the hair transfer fidelity. In Fig. 7, we show ROC curves for a binary classifier which uses the similarity metric to distinguish pairs of images with the same or different hairstyle. We compare the results for image pairs generated from the real dataset with manually labeled hairstyles and synthetic pairs generated by our hairstyle transfer method. It is seen that the ROC curves are indistinguishable. The metric learned from manually created labels simulates the answer of a human to the question: "Are the two hairstyles similar?". Hence, we see that a human would agree with our method as frequently as he/she agrees with another human on what pair of images shows the same hairstyle. This evaluation technique replaces an expensive user study which would require to manually judge all image pairs generated by our method.

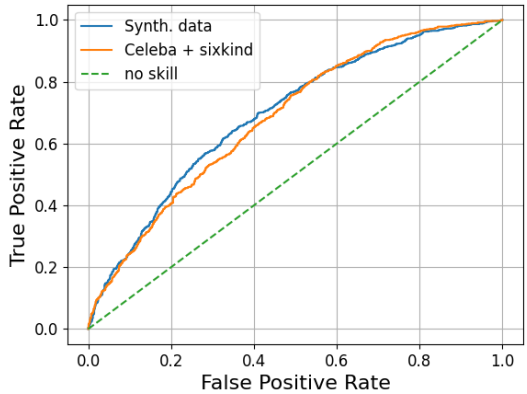


Fig. 7: Hair similarity metric ROC curves computed on an annotated dataset (orange) and an unannotated synthetic dataset generated by our hairstyle transfer method (blue).

The positive and negative image pairs were generated as follows. Pairs for the manually labeled dataset were constructed using the class labels. ROC curve is computed from 2500 random pairs from the held-out test sets. A synthetic set of images generated by our method was prepared. Let us have a pair of images A and B, randomly generated from StyleGAN2. Then we denote $\mathcal{G}(A_h, B_f)$, the output of the proposed hairstyle transfer network taking the hair input from A and the face input from B. For every pair (A, B) we create two positive $\{\mathcal{G}(A_h, A_f), \mathcal{G}(A_h, B_f)\}$, $\{\mathcal{G}(B_h, B_f), \mathcal{G}(B_h, A_f)\}$ and two negative pairs $\{\mathcal{G}(A_h, A_f), \mathcal{G}(B_h, A_f)\}$, $\{\mathcal{G}(B_h, B_f), \mathcal{G}(A_h, B_f)\}$. We generated 2500 test pairs as in case of the real dataset.

2) Identity Preservation: To assess the method does not alter the identity of the subject, we conducted the following retrieval experiment. We randomly generated 100k images from StyleGAN2. For a random subset of 1k images, we randomly switched hair. For all images, ArcFace net [10] recognition descriptor was calculated. The hair-changed images were queried to the entire dataset using cosine similarity. The results show that the first retrieved image was correct in 98.3% of the queries. The average rank was 1.143, which corroborates the identity is well preserved. The experiment is challenging, due to the large size of the dataset and since the face detector was likely not trained to be perfectly robust to such drastic hair changes.

3) User Study: To quantitatively analyze the visual quality of the hair-transferred results, we conducted a human-evaluated survey. The goal of the study is to compare our results with raw images generated by StyleGAN2.

We set up four questionnaires, all containing 25 pairs. Each pair comprised a randomly generated image by StyleGAN2 and an image generated by our method with the same identity but a different hairstyle, randomly chosen from the image set. The question was: "Which of the two images is more realistic?". All images captured only women to avoid unrealistic results due to unusual hairstyles from male to female hair transfers.

Each respondent was assigned a score, representing the number of votes for the image generated by our method. 109 respondents participated in the study. The mean score is 46%, standard deviation 15.3%. The best theoretical score is 50%, since we use the StyleGAN2 as an output generator. The result suggests that our method is not easily distinguishable from the StyleGAN2. It means that the image quality perceived by study participants is comparable as that of the state-of-the-art StyleGAN2 and the hair transfer does not introduce additional artifacts.

V. Conclusions

We presented a method that was trained from synthetic data without manual annotation, providing a high-definition hairstyle transfer between face images. The method was thoroughly evaluated using both qualitative and quantitative analysis including a user study. Comparison with other state-of-the-art methods was demonstrated. The results of our method show high fidelity of hair transfer, preserving the identity and facial expressions of the subject, even in challenging cases of nonmatching pairs, neither aligned geometrically nor with consistent illumination.

Unlike other methods, we do not use any postprocessing of the results, e.g., for blending the original background to output [26]. The pipeline is simple and fully automatic, unlike [33], which requires user interaction. The results are provided by a single pass of the proposed neural network, without iterative optimization.

Acknowledgements

The research was supported by the OP VVV funded project CZ.02.1.01/0.0/0.0/16_019/0000765 "Research Center for Informatics", by the Czech Science Foundation project GACR GA19-21198S, and by the CTU Student Grant SGS20/171/OHK3/3T/13.

References

[1] R. Abdal, Y. Qin, and P. Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proc of ICCV*, 2019.
[2] D. Bau, H. Strobelt, W. Peebles, J. Wulff, B. Zhou, J.-Y. Zhu, and A. Torralba. Semantic photo manipulation with a generative image prior. *ACM TOG*, 38(4), 2019.
[3] D. Bitouk, N. Kumar, S. Dhillon, P. Belhumer, and S. Nayar. Face swapping: automatically replacing faces in photographs. *ACM TOG*, 2008.
[4] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *Proc. SIGGRAPH*, 1999.
[5] A. Brock, J. Donahue, and K. Simonyan. Large scale gan training for high fidelity natural image synthesis. In *Proc. of ICLR*, 2019.
[6] A. Bulat and G. Tzimiropoulos. How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3D facial landmarks). In *ICCV*, 2017.
[7] M. Chai, T. Shao, H. Wu, Y. Weng, and K. Zhou. Autohair: Fully automatic hair modeling from a single image. *ACM TOG*, 35(4), 2016.
[8] M. Chai, L. Wang, Y. Weng, X. Jin, and K. Zhou. Dynamic hair manipulation in images and videos. *ACM TOG*, 32(4), 2013.
[9] M. Chai, L. Wang, Y. Weng, Y. Yu, B. Guo, and K. Zhou. Single-view hair modeling for portrait manipulation. *ACM TOG*, 31(4), 2012.
[10] J. Deng, J. Guo, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. *CVPR*, 2019.
[11] V. Dumoulin, I. Belghazi, B. Poole, A. Lamb, M. Arjovsky, O. Mastropietro, and A. C. Courville. Adversarially learned inference. *ArXiv*, abs/1606.00704, 2017.
[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Adv. Neural Inf. Process. Syst.*, 2014.
[13] L. Hu, C. Ma, L. Luo, and H. Li. Robust hair capture using simulated examples. *ACM TOG*, 33(4), 2014.
[14] L. Hu, C. Ma, L. Luo, and H. Li. Single-view hair modeling using a hairstyle database. *ACM TOG*, 34(4), 2015.
[15] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *ICLR*, 2018.
[16] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proc. CVPR*, 2019.
[17] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila. Analyzing and improving the image quality of stylegan. *CoRR*, abs/1912.04958, 2019.
[18] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila. Analyzing and improving the image quality of stylegan. In *Proc. CVPR*, 2020.
[19] I. Kemelmacher-Shlizerman. Transfiguring portraits. *ACM TOG*, 35(4), 2016.
[20] D. E. King. Dlib-ml: A machine learning toolkit. *JMLR*, 10(60), 2009.
[21] A. B. L. Larsen, S. K. Sønderby, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. *CoRR*, abs/1512.09300, 2015.
[22] A. Levin, A. Zomet, S. Peleg, and Y. Weiss. Seamless image stitching in the gradient domain. In *ECCV*, 2004.
[23] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.
[24] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
[25] L. Luo, H. Li, S. Paris, T. Weise, M. Pauly, and S. Rusinkiewicz. Multi-view hair capture using orientation fields. In *Proc. of CVPR*, 2012.
[26] R. Natsume, T. Yatagawa, and S. Morishima. Rsgan: Face swapping and editing using face and hair representation in latent spaces. In *ACM SIGGRAPH 2018 Posters*, SIGGRAPH '18, 2018.
[27] Y. Nirkin, I. Masi, A. T. Tran, T. Hassner, and G. G. Medioni. On face segmentation, face swapping, and face perception. *CoRR*, abs/1704.06729, 2017.
[28] S. Paris, W. Chang, O. I. Kozhushnyan, W. Jarosz, W. Matusik, M. Zwicker, and F. Durand. Hair photobooth: Geometric and photometric acquisition of real hairstyles. *ACM TOG*, 27(3), 2008.
[29] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proc. of ICLR*, 2016.
[30] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, and D. Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. *ArXiv*, abs/2008.00951, 2020.
[31] M. Rosca, B. Lakshminarayanan, D. Warde-Farley, and S. Mohamed. Variational approaches for auto-encoding generative adversarial networks. *ArXiv*, abs/1706.04987, 2017.
[32] Y. Shen, J. Gu, X. Tang, and B. Zhou. Interpreting the latent space of gans for semantic face editing. *IEEE/CVF CVPR*, 2020.
[33] Z. Tan, M. Chai, D. Chen, J. Liao, Q. Chu, L. Yuan, S. Tulyakov, and N. Yu. Michigan: multi-input-conditioned hair image generation for portrait editing. *ACM TOG*, 39, 2020.
[34] K. Ward, F. Bertails, T. Kim, S. R. Marschner, M. Cani, and M. C. Lin. A survey on hair modeling: Styling, simulation, and rendering. *IEEE TVCG*, 13(2), 2007.
[35] Y. Weng, L. Wang, X. Li, M. Chai, and K. Zhou. Hair Interpolation for Portrait Morphing. *Comput. Graph. Forum*, 2013.
[36] W. Yin, Y. Fu, Y. Ma, Y.-G. Jiang, T. Xiang, and X. Xue. Learning to generate and edit hairstyles. In *Proc. of ACM MM*, 2017.
[37] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. *IEEE/CVF CVPR*, 2018.