SelfieArt: Interactive Multi-Style Transfer for Selfies and Videos with Soft Transitions

Lucas N. Alegre
Instituto de Informática - UFRGS
Porto Alegre, RS - Brazil
lnalegre@inf.ufrgs.br

Manuel M. Oliveira
Instituto de Informática - UFRGS
Porto Alegre, RS - Brazil
oliveira@inf.ufrgs.br

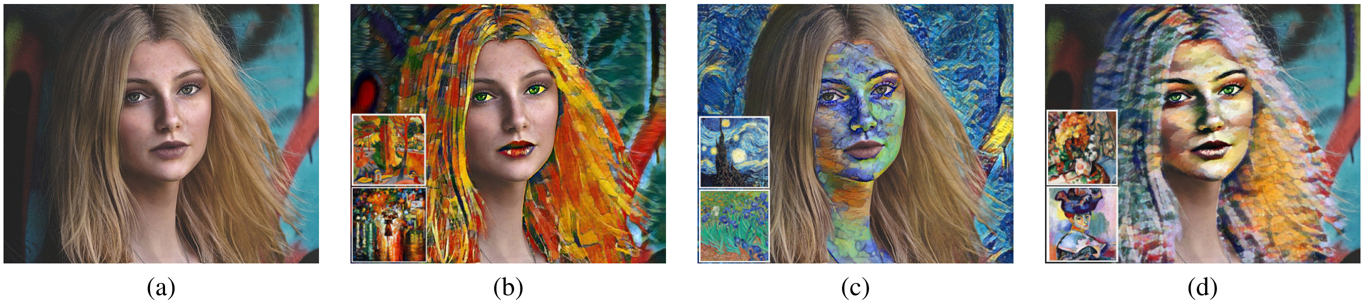


Fig. 1: Examples of images created with *SelfieArt*. (a) Input photograph. (b)-(d) Results obtained applying sytles from paintings (shown as insets at the bottom left of each image) to different parts of the photograph. (b) Style from the top painting applied to the hair; style from the bottom painting applied to the eyes, lips, and background. (c) Top style applied to the background and bottom style applied to the face and neck. (d) Top style applied to the face and bottom style applied to the hair.

***Abstract*—We introduce SelfieArt, an interactive technique for performing multi-style transfer for portraits and videos. Our method provides a simple and intuitive way of producing exquisite artistic results that combine multiple styles in a harmonious fashion. It uses face parsing and a multi-style transfer model to apply different styles to the various semantic segments. This is achieved using parameterized soft masks, allowing users to adjust the smoothness of the transitions between stylized regions in real-time. We demonstrate the effectiveness of our solution on a large set of images and videos. Given its flexibility, speed, and quality of results, our solution can be a valuable tool for creative exploration, allowing anyone to transform photographs and drawings in world-class artistic results.**

I. Introduction

Image and video stylization are important applications in computational photography, computer graphics, and image and video processing. In this context, stylization consists of transferring the style (*i.e.*, line stokes, textures, colors, etc.) of one image to other images or videos. In case multiple styles are simultaneously transferred to one or more images or videos, the process is called *multi-style transfer*.

Style transfer can be seen as the process of synthesizing textures while respecting the semantic content of the target image or video frames [1], [2]. In computer graphics, several texture-synthesis algorithms have been developed based on image resampling [3]–[6], or performing multi-scale analysis and synthesis [7], [8]. The Image Analogies technique [9] uses a pair of unfiltered and filtered images as training data to learn a filter that can be subsequently applied to other images. More recently, Gatys et al. [10] have shown that a convolutional neural network (CNN) can successfully separate the content from the style of natural images. Moreover, by recombining the content of a given image A with the style of a different image B, they demonstrated high-quality style transfer results, and sparkled a revival in the interest for style transfer methods. Since then, faster techniques have been developed [2]. Automated style transfer software can be used to generate derivative works of a particular artist or painting, and style transfer based on deep neural networks is behind popular apps like *Prisma* [11], which convert real-world photos into different artistic styles.

Motivated by several computer vision applications, CNNs have also been successfully used to perform semantic segmentation [12]. Recently, researchers have combined style transfer with semantic segmentation in order to stylize only certain elements of an image (*e.g.*, cars and buildings) in a traffic scenario) [13], [14].

Our technique also combines style transfer with semantic segmentation. But unlike previous approaches, we exploit the huge popularity of selfies and "video selfies" among young people. Selfies and photos with some applied filters are prevalent in social media such as Instagram, Snapchat, and Twitter. Other social networks, such as Tik Tok, use short videos. Thus, our technique uses face parsing [15] to obtain pixel-wise label maps for different semantic components of one's face (*e.g.*, hair, mouth, eyes). We use this information to perform interactive multi-style transfer for selfies and videos in a simple and intuitive way. Our technique produces artistically pleasing results by smoothly combining multiple styles.

Figure 1 illustrates some results generated by our technique. It shows a photograph on the left, and three results obtained applying the styles from pairs of paintings (shown as insets at the bottom left of each image) to different portions of the photograph. Note the variety of achieved effects.

The **contributions** of this paper include:

- A technique to perform interactive multi-style transfer of portraits and videos (Section III). Our method uses semantic segmentation of facial elements, which can then receive different styles.
- A method for blending the multiple styles with adjustable soft masks (Section III-C). Our solution uses the probabilities provided by the segmentation network as blending factors, which can be adjusted by the user in real-time.

II. Related Work

Recently, researchers have combined style transfer and semantic segmentation to stylize only certain regions of an image. However, to the best of our knowledge, no previous work has combined style transfer and face parsing to stylize different facial semantic segments.

A few works have applied style transfer methods to face portraits. Seleim et al. [16] extended the method of Gatys et al. [10] introducing spatial constraints in order to maintain the integrity of facial structures in the stylized images. Zhao et al. [17], [18] use segmentation methods to generated masks for both the content and style images. These masks are used in the loss function enforcing that pixels in the same segment are stylized similarly. Although these methods improve the quality of the style transfer, like in Gatys et al. [10] and Seleim et al. [16], a single style is applied to the entire image.

Castillo et al. [13] use a semantic segmentation algorithm to allow the user to select the region onto which the transfer will occur. A Markov random field (MRF) based model is used to merge the extracted stylized object with the non-stylized background. To blend the images using an MRF, a narrow band of ambiguous pixels around the target object must be specified by hand. The method produces a binary labeling, which results in sharp transitions at the boundaries of the stylized regions.

Kurzman et al. [14] combine a fast style-transfer method [19] with binary segmentation to stylize user-defined regions in frames of a video in real-time. The use of binary masks to merge the stylized and original elements introduces sharp transitions at the boundaries of the stylized objects.

In contrast to these methods, ours uses semantic segmentation of facial elements to support interactive multi-style transfer with smooth transitions between styles. Our method works for both images and videos, and provides interactive control over the smoothness of the transitions between boundaries.

III. Proposed Method

Our multi-style transfer method works as follows: given an input portrait/video frame c, a face parsing method (Section III-A) generates a set of (logits) matrices, one for each predefined segment (*e.g.*, skin, eyes, lips, neck, hair, background). These matrices indicate the segment to which each input pixel belongs. Given a set of user-selected style images, their styles are transferred to segments returned by the face parsing method according to user selection (Section III-B). The user can control the smoothness of the transitions involving stylized regions. Figure 2 shows the pipeline of our method.

A. Face Parsing

The goal of face parsing is to assign a pixel-wise label for each semantic components (*e.g.*, hair, eyes, nose, and mouth) in an input face image. Recently, many works have achieved impressive results using deep CNN's for image segmentation [12]. Following this line, we employ a CNN for face segmentation. Given an input (color or monochromatic) image/frame $c \in \mathbb{R}^{3\times H\times W}$ (with three channels), the face parsing model outputs a logits matrix $z \in \mathbb{R}^{H\times W\times K}$, where H and W are the height and width of the image c, respectively, and K is the number of semantic segments. These logits matrices are then transformed into a blending mask used to composite the multiple stylized segments and c.

In this work, we used the Bilateral Segmentation Network (BiSeNet) [20], a real-time semantic segmentation method. BiSeNet works by combining a Spatial Path and a Context Path. The Spatial Path preserves spatial information and generates high-resolution features. The Context Path with a fast downsampling strategy is used to obtain a sufficiently large receptive field. Our current prototype uses an implementation of BiSeNet [21] which was pre-trained with CelebAMask-HQ [22], a large-scale high-resolution face dataset with fine-grained mask annotations. For this dataset, there are $K = 19$ possible semantic segments in the images, including background, skin, hair, eyes, nose, and cloth.

B. Multi-Style Transfer

Neural style transfer (NST) was introduced by Gatys et al. [10]. It builds on the key idea that it is possible to separate the style and content representations of an image using a CNN. Given an input (content) image c and a style image s, NST generates an image g by minimizing the loss function:

$$\mathcal{L}_{total} = \alpha\mathcal{L}_{content} + \beta\mathcal{L}_{style}, \tag{1}$$

where α and β are weighting factors for content and style reconstruction, respectively.

The content loss is defined as the mean squared error of the activations in the feature maps from a pre-selected convolutional layer of a feature extraction network. Given these activations for the content and generated image, the content loss is then computed as:

$$\mathcal{L}_{content} = ||A_l(g) - A_l(c)||^2, \tag{2}$$

where $A_l(x)$ is a matrix containing the activations for all N_l feature maps of size M_l in layer l for the image x.

The style loss, in turn, is computed as:

$$\mathcal{L}_{style} = \sum_{l\in L} ||G_l(g) - G_l(s)||^2, \tag{3}$$

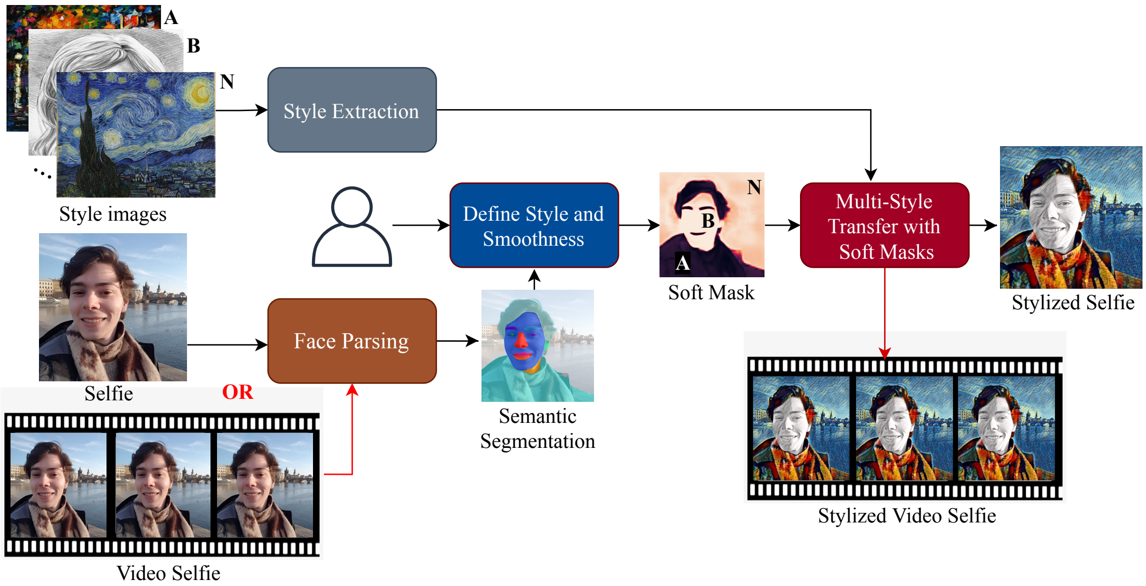


Fig. 2: The *SelfieArt* pipeline. Given a selfie or a video as input, face parsing performs semantic segmentation. The user then interactively associates styles to the segmented elements and defines a desired smoothness level (τ) for the transitions involving stylized regions. A soft mask generated based on the selected segments and on τ guides the multi-style transfer process.

where $G_l(x)$ denotes the Gram's matrix obtained using the feature maps of layer l, computed as:

$$G_l(x) = \frac{1}{N_l M_l} A_l(x) A_l(x)^T. \tag{4}$$

The Gram's matrix measures the covariance between features in space, *i.e.*, which feature maps tend to activate together. This way, it captures color and texture information, not affecting spatial structure (which is preserved due to the content loss). The set of layers L from which the feature maps are computed is obtained from a pre-trained VGG CNN [23].

Zhang et al. [2] introduced Multi-style Generative Network (MSG-Net), a feed-forward neural network that runs in real-time. Differently from NST, which directly optimizes over the pixels of image g by minimizing the loss in Eq. 1, MSG-Net is trained to generate g with a single forward pass given any content image. MSG-Net embeds style using a 2D representation and learns to match the Gram matrices of the style targets inherently during the training. SelfieArt uses the original MSG-Net implementation trained with the COCO dataset [24] and a set of 21 style images.

Although MSG-Net allows for real-time style transfer, it is restricted to the set of pre-trained styles. If one would like to use an arbitrary style image, our system also supports the slower style-transfer method (NST) from [10].

C. Parameterized Soft Masks

Given the stylized image g, the logits matrices $z_k, k \in \{0, K-1\}$ (produced by the face parsing model), and the segment maps U chosen by the user, the goal is to blend images c and g only in the regions defined by the segments contained in U. Since all K semantic segments are disjoint, a blending mask m can be constructed by combining the masks m_k for each segment $k \in U$:

$$m = \sum_{k \in U} m_k. \tag{5}$$

Then, the final image o can be obtained by compositing g and c according to an alpha mask m:

$$o = m * g + (1 - m) * c. \tag{6}$$

A naive approach for obtaining the mask m consists of representing each m_k as a *binary mask*, computed as:

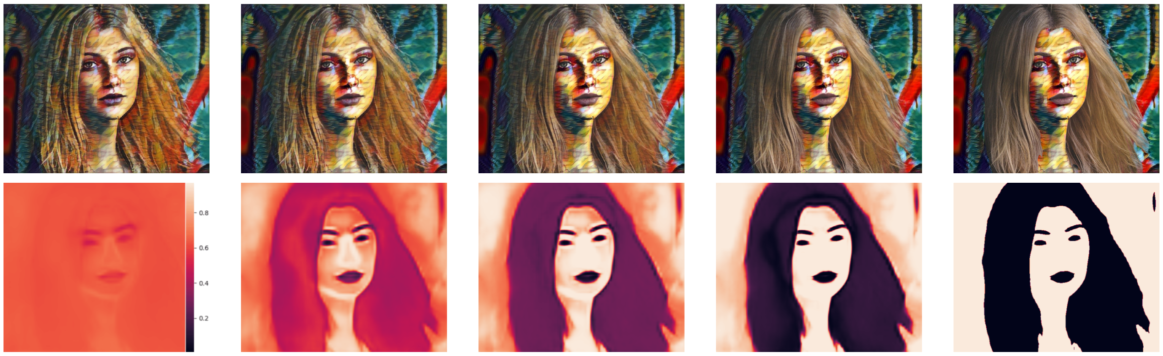
$$m_k(x, y) = \begin{cases} 1, & \text{if } (\arg\max_{i \in K} z_i(x, y)) = k; \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

This approach was used in [14], and although simple and fast, it tends to generate sharp, unnatural transitions at the boundaries of stylized regions (Figure 7 (d)).

In order to obtain smooth transitions at the boundaries, one can apply the softmax function to the logits z_k. The resulting *softmax masks* m_k can be interpreted as probabilities:

$$m_k = \frac{e^{z_k}}{\sum_i e^{z_i}}. \tag{8}$$

However, the softmax output of a CNN does not reliably estimate prediction uncertainties or confidences [25]. In addition, applying the mask m_k as defined in Eq. 8 can introduce distortions in regions far from the segment k in the image. For instance, consider pixels in the background for which the face parsing model mistakenly predicted a 20% of chance of them belonging to the hair segment. Such background pixels will be composited as 80% of the style applied to the background and 20% of the style applied to hair.



(a) $\tau = 0.1$ (b) $\tau = 0.5$ (c) $\tau = 1$ (d) $\tau = 2$ (e) Binary Mask

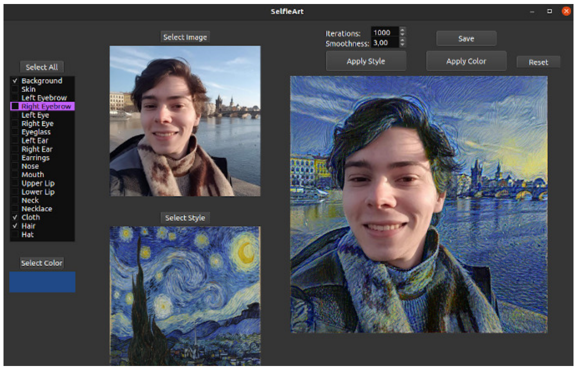
Fig. 3: Blending results and masks for different values of τ.

We obtain smooth transition masks while minimizing such artifacts. This is achieved by scaling the logits z_k using a single scalar parameter τ. Thus, our parameterized soft masks m_k are defined as:

$$m_k = \frac{e^{\tau z_k}}{\sum_i e^{\tau z_i}}, \quad \tau \in \mathbb{R}_+. \tag{9}$$

The parameter τ controls the amount of contribution (to the generated image) of the pixels for which the face parsing model is uncertain about. This parameter smooths the softmax (*i.e.*, raises the output entropy) when $\tau < 1$. When $\tau = 1$, one recovers the original softmax defined in Eq. 8. As $\tau \to \infty$, this approximates the binary mask in Eq. 7.

Figure 3 illustrates how the parameter τ affects the generated mask m. In this example, U contains all segments except for hair, eyes, eyebrows, and lips. For low values of τ, the values in the mask for the selected semantic segments are more uniform. This results in all pixels being partially stylized with similar blending factor – see Equation 6 (note the hair being stylized for values of $\tau \leq 1$). As the value of τ increases, the mask's values for the pixels in the user-selected semantic segments increase. For instance, for $\tau = 2$ the mask becomes more similar to the binary mask, but still presents differences for the pixels near the boundaries of the segments. This reduces unnatural transitions between regions containing at least one stylized segment.



Iterations: 1000
Smoothness: 3,00

☑ Background
☐ Skin
☐ Left Eyebrow
☐ Right Eyebrow
☐ Left Eye
☐ Right Eye
☐ Eyeglass
☐ Left Ear
☐ Right Ear
☐ Earrings
☐ Nose
☐ Mouth
☐ Upper Lip
☐ Lower Lip
☐ Neck
☐ Necklace
☑ Cloth
☑ Hair
☐ Hat

Fig. 4: SelfieArt application interface.

IV. Results

We have implemented the described technique using Python, OpenCV, and PyTorch, and applied it to a large number of images and videos. For semantic segmentation, it uses BiSeNet [21] pre-trained with CelebAMask-HQ [22], capable of segmenting a portrait in up to 19 distinct regions. For multi-style transfer, our application supports both MSG-Net [2] pre-trained on 21 classic artistic styles, and NST [10] for use with any user-selected style. SelfieArt successfully stylizes pre-defined regions of a portrait, providing real-time feedback to users and effectively expanding the possibilities for artistic expression. Figure 4 shows the interface of our current prototype, which we intend to make publicly available. It allows users to select a portrait to be stylized (center top), and a style image (one at a time – center bottom) to be applied to the semantic segments (left). The resulting stylized image is updated in real time (right). The interface also allows the user to interactively change the value of the smoothness parameter (τ) used to control the final compositing.

Creating a high-quality multi-style transferred portrait with SelfieArt takes about 25 seconds, and requires modest computational resources. The accompanying video shows live recordings of interactive sessions of multi-style transfer running on an Intel Core i5 CPU of a laptop with 8GB of RAM. Figure 5 illustrates the use of a two-style transfer to create a gallery of artistic portraits using our technique. The images at the first row and column correspond to styles that are transferred to photographs located at the corresponding row and column intersections. These results provide a clear demonstration of the potential of our technique. All results in Figure 5 were created using a smoothing factor $\tau = 3$.

Processing videos requires semantic segmentation and style transfer on a per-frame basis. Figure 6 shows frames of two multi-style videos generated by our technique. On the left, only the clothes and the background were stylized; on the right, different styles have been applied to the facial features, and to the clothes and background. For these examples, our technique processed the input video (1280 × 720) at 6.8 fps for the style on the left, and at 3.6 fps for the style on the

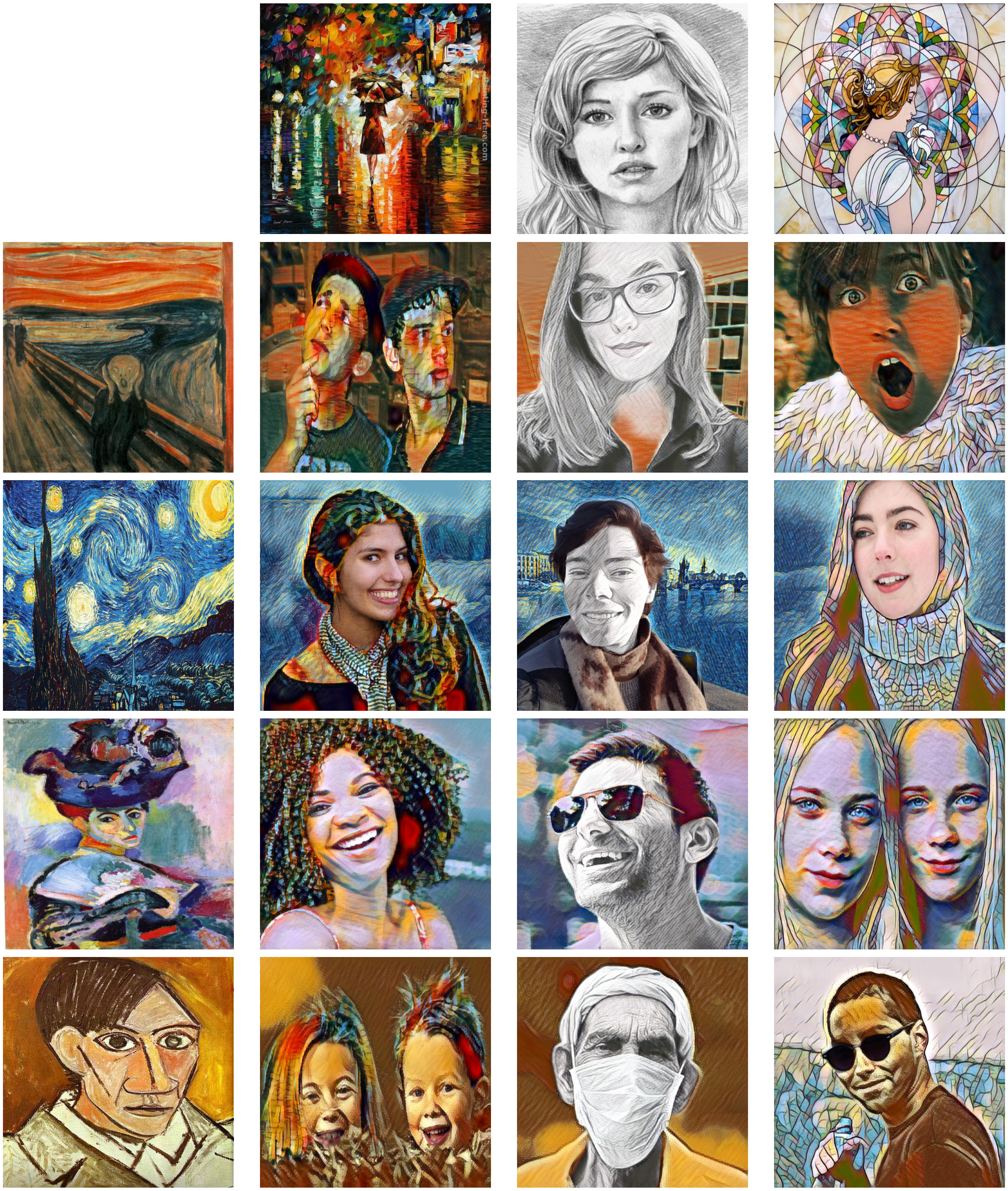


Fig. 5: Use of two-style transfer to create artistic portraits using our technique. The styles are defined by the images in the first row and column. Each portrait combines the styles of their corresponding rows and columns.

right, on a GeForce RTX 2060 mobile GPU with 4 GB of memory. The accompanying video shows several examples of multi-style transfer to videos generated by our technique.

Figure 7 compares the impact of the smoothness parameter τ on the resulting composites and with the use of binary masks. Binary masks produces unnatural, abrupt transitions at the boundaries of stylized regions (Figure 7 (d)). For $\tau = 1$, the style exceeds the segmented face patch and is also applied to the hair with low intensity. Using $\tau = 3$ produces some intermediate result. The user can continuously adjust the value of τ in real-time to create the desired effect.

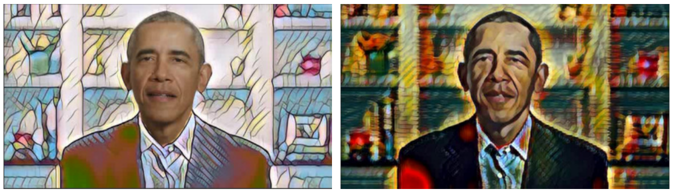
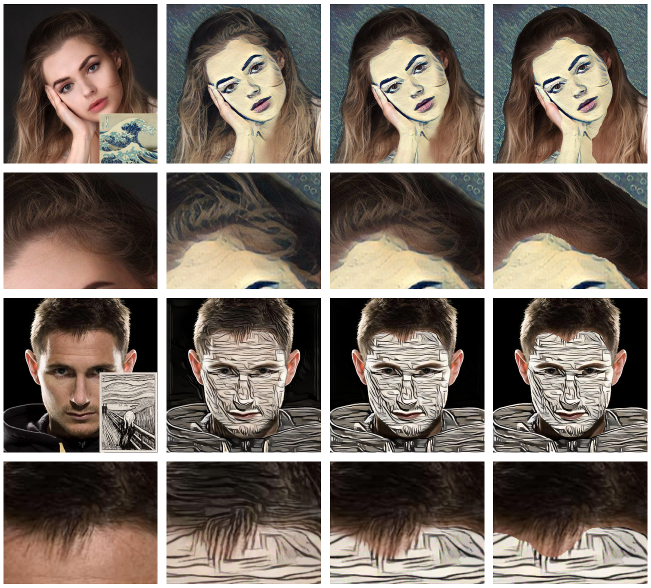


Fig. 6: Video stylization examples produced by our technique.



(a) Input (b) $\tau = 1$ (c) $\tau = 3$ (d) Binary mask

Fig. 7: Comparison of the impact of soft vs. binary mask.

Limitations The quality of the results produced by our technique depends on the quality of the semantic segmentation generated by the used face parsing method. Such methods are not intended for segmenting fine details, such as sparse hair strands (Figure 7). Face parsing methods may also fail to correctly segment facial elements if a face is partially occluded. This situation is illustrated in the last stylized sequence shown in the accompanying video.

V. Conclusion

We presented an interactive technique for performing multi-style transfer for portraits and videos. Our method provides a simple and intuitive way of producing exquisite artistic results that smoothly combine multiple styles. We demonstrate the effectiveness of our solution on a large set of selfies and videos. Given its flexibility, speed, and quality of results, our solution can be a valuable tool for creative exploration, allowing anyone to transform photographs and drawings in world-class artistic results.

Acknowledgments

This work was sponsored by CNPq-Brazil (fellowships and grants 312975/2018-0, 423673/2016-5), and financed in part by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. We thank Zhang et al. for making their source code available, and Alexis Jacq for providing an implementation of Gatys et al.'s algorithm in PyTorch. We also thank PixaBay for providing some of the portraits used in this work, and the Obama Foundation for providing Obama's speech video.

References

[1] C. Li and M. Wand, "Combining markov random fields and convolutional neural networks for image synthesis," in IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2479–2486.

[2] H. Zhang and K. Dana, "Multi-style generative network for real-time transfer," in Computer Vision – ECCV 2018 Workshops, L. Leal-Taixé and S. Roth, Eds., 2019, pp. 349–365.

[3] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in Proc. IEEE ICCV, vol. 2, 1999, pp. 1033–1038 vol.2.

[4] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in Proc. SIGGRAPH, 2001, p. 341–346.

[5] L.-Y. Wei and M. Levoy, "Fast texture synthesis using tree-structured vector quantization," in Proc. SIGGRAPH, 2000, p. 479–488.

[6] M. Ashikhmin, "Synthesizing natural textures," in Proc. Symposium on Interactive 3D Graphics, 2001, p. 217–226.

[7] D. J. Heeger and J. R. Bergen, "Pyramid-based texture analysis/synthesis," in Proc. SIGGRAPH, 1995, p. 229–238.

[8] J. S. De Bonet, "Multiresolution sampling procedure for analysis and synthesis of texture images," in Proc. SIGGRAPH, 1997, p. 361–368.

[9] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies," in Proc. SIGGRAPH, 2001, p. 327–340.

[10] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in IEEE CVPR, 2016, pp. 2414–2423.

[11] P. labs inc., "Prisma photo editor," 2016. [Online]. Available: https://prisma-ai.com/

[12] B. Li, Y. Shi, Z. Qi, and Z. Chen, "A survey on semantic segmentation," in 2018 IEEE International Conference on Data Mining Workshops (ICDMW), 2018, pp. 1233–1240.

[13] C. D. Castillo, S. De, X. Han, B. Singh, A. K. Yadav, and T. Goldstein, "Son of zorn's lemma: Targeted style transfer using instance-aware semantic segmentation," IEEE ICASSP, pp. 1348–1352, 2017.

[14] L. Kurzman, D. Vazquez, and I. Laradji, "Class-based styling: Real-time localized style transfer with semantic segmentation," in Proc. IEEE ICCV Workshops, 2019.

[15] J. Lin, H. Yang, D. Chen, M. Zeng, F. Wen, and L. Yuan, "Face parsing with roi tanh-warping," CVPR, pp. 5647–5656, 2019.

[16] A. A. S. Seleim, M. A. Elgharib, and L. Doyle, "Painting style transfer for head portraits using convolutional neural networks," ACM Trans. Graph., vol. 35, pp. 129:1–129:18, 2016.

[17] H. Zhao, P. Rosin, and Y.-K. Lai, "Automatic semantic style transfer using deep convolutional neural networks and soft masks," The Visual Computer, 08 2017.

[18] H. Zhao, J. Zheng, Y. Wang, X. Yuan, and Y. Li, "Portrait style transfer using deep convolutional neural networks and facial segmentation," Computers & Electrical Engineering, vol. 85, 2020.

[19] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in ECCV, 2016.

[20] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in ECCV, 2018, pp. 325–341.

[21] zllrunning, "face-parsing.pytorch," https://github.com/zllrunning/face-parsing.PyTorch, 2019.

[22] C.-H. Lee, Z. Liu, L. Wu, and P. Luo, "Maskgan: Towards diverse and interactive facial image manipulation," in CVPR, 2020.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in 3rd International Conference on Learning Representations (ICLR), Y. Bengio and Y. LeCun, Eds., 2015.

[24] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in Computer Vision – ECCV 2014, 2014, pp. 740–755.

[25] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in Proc. ICML - Volume 70. JMLR.org, 2017, p. 1321–1330.