Automatic Animation of Hair Blowing in Still Portrait Photos - Supplementary Materials -

Wenpeng Xiao[1], Wentao Liu[1], Yitong Wang[1], Bernard Ghanem[2], Bing Li[2]
[1] ByteDance Intelligent Creation Lab
[2] King Abdullah University of Science and Technology (KAUST)
{xiaowenpeng.com, liuwentao.canon}@bytedance.com, wangyitong@pku.edu.cn, {Bernard.Ghanem, bing.li}@kaust.edu.sa

In this supplementary, we provide additional details about hair wisp extraction and hair wisp dataset in Sec. 1.1, and hair wisp animation in Secs. 1.2. We also provide more experimental results in Sec. 3. In Sec. 3, we show hair wisps extraction results, and provide additional comparison results with Chuang *et al*., Halperin *et al*. and Endo *et al*. on various portrait photos. In addition, we also compare with more image-to-video generation methods.

1. More Implementation Details

1.1. Instance-based Hair Wisp Extraction

Training Details. Following ISTR [9], we adopt ResNet50[8] as the backbone, FPN[10] for pyramid feature extraction, and RoIAlign[7] for instance feature calculation, a 12-layer swin-transformer[11] encoder to model the inter-relationships among instances, and a final prediction head. We use the AdamW[12] optimizer with a learning rate of 1e-6 and a weight decay of 1e-4, where the batch size is set to 16.

Additional Details of Hair Wisp Dataset. The portrait photos and corresponding hair sketches are collected from SketchHairSalon [14]. Since the ground-truth annotations of hair wisps are not available, we design a sketch-filled algorithm to automatically generate pseudo annotations from hair sketches. Our sketch-filled algorithm progressively floods the annotated sketch to the left, top, and bottom, as shown in Fig. 1. We observe that it is crucial to ensure that every two wisps have clear boundaries between them in a generated instance mask, which is helpful for training instance segmentation models and learning to discriminate each wisp region. To this end, we erode the filled mask for 5 pixels given the image resolution of 512×512, and discard those with zero sizes.

1.2. Additional Details of Hair Wisp Animation

We set the mass m of a particle to 1 in our wisp motion prediction. Given a mesh of a hair wisp, if its spring is completely inside the hair wisp, the spring constant K is 100. Otherwise, $K = 1$. Such non-uniform spring constant values can enable more flexible shape transformation than uniform ones.

We assign an initial wind velocity to particles in a hair wisp mesh, where the absolute velocity value is in the range [0 4], where the velocity values of particles linearly decayed from bottom to top in each hair wisp. The wind direction depends on the global growing tendency of the original hair from top to bottom. Furthermore, we additionally introduce a damping force with the damping constant of 5e-5 for dynamic convergence. Note that Heun's method and Runge-Kutta method can also be employed to solve the ODEs in wisp motion prediction.

2. More details of FVD

FVD [13] measures the difference between the real video distribution P_r and generated video distribution P_g :

$$d(P_r, P_g) = |\mu_r - \mu_g|^2 + \mathrm{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{\frac{1}{2}})$$

where μ and Σ are the means and the co-variance matrices of a distribution, respectively. Since FVD doesn't require ground truth, a set of real videos containing hair blowing is employed P_r for calculating FVD, following [46, 18]

3. More Experimental Results

Hair Wisp Extraction Results. Fig. 3 show the hair wisp extraction results of our method. As shown in Fig. 3, thanks to our hair wisp datasets and annotations of hair wisps, instance segmentation networks extract meaningful hair wisps from original images containing various hairstyles. Yet, the contours of hair wisps extracted by instance segmentation networks are not smooth. With our hair refinement, the coarse hair wisp results are refined, where hair wisp contours are smoothed and wisp tips become sharpened.

Additional comparison results Figs. 5 and 6 demonstrate more qualitative comparison results with Chuang *et*

(a) (b)

Figure 1: Our method automatically generates pseudo-ground-truth annotations of hair wisps from hair sketches. (a) Original images with hair sketches, (b) Generated annotations of hair wisps for hair wisp extraction.

al. [3], Halperin *et al*. [6] and Endo *et al*. [5]. As shown in Figs. 5 to 6, compared with these state-of-the-art approaches, our method achieves the best hair animation performance on images containing various hairstyles and face poses.

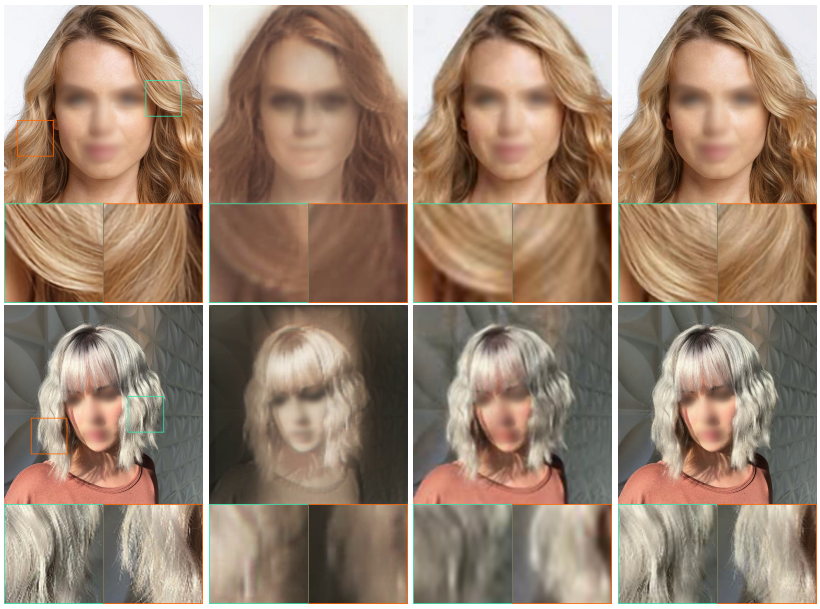
We also compare our methods with more state-of-the-art approaches in Figs. 4 and 2. Please refer to supplemented videos for spatiotemporal comparisons. Fig. 4 compares our method with Chai *et al*. [2] which animates hair based on strand-based motion simulation. As shown in Fig. 4, Chai *et al*. [2] introduces grainy artifacts in the moving hair and distorts the left face region of the girl. Chai *et al*. [2] generate motions using guided strand-based motion simulation, however, the generated motions look like water waving, which is not natural. In contrast, our method generates the animation of hair blowing, which provides better and more natural animation performance, thanks to our hair wisp animation and hair wisp extraction modules.

We also compare with two recent single-image-to-video generation methods: Dorkenwald *et al*. [4] and Blattmann *et al*. [1]. Fig. 2, Tab. 1 and supplemental videos show our method achieves the best performance, compared with Dorkenwald *et al*. and Blattmann *et al*. Both Dorkenwald *et al*. and Blattmann *et al*. synthesize a video from an image based on autoencoder, where the size of input image is set to be 128×128. However, it is difficult for an autoencoder to learn the complex motions of thin hairs from training data with low video resolutions, e.g., 128×128. As a result, Dorkenwald *et al*. and Blattmann *et al*. tend to generate global appearance changes, rather than animating hair. In addition, Dorkenwald *et al*. and Blattmann *et al*. leads to frame blurring and degradation, since they iteratively synthesize a video frame from its previous one and the artifacts/errors are accumulated along the temporal dimension. In contrast, we design a wisp motion prediction based on physical models, which enables our method to perform well on diverse portrait photos including high-resolution ones without relying on training data of hair animation.

Metric	FVD ↓	E_{warp} ↓
Dorkenwald *et al*. [4]	1294.05	841.08
Blattmann *et al*. [1]	1235.52	562.87
Ours	**1153.98**	**521.96**

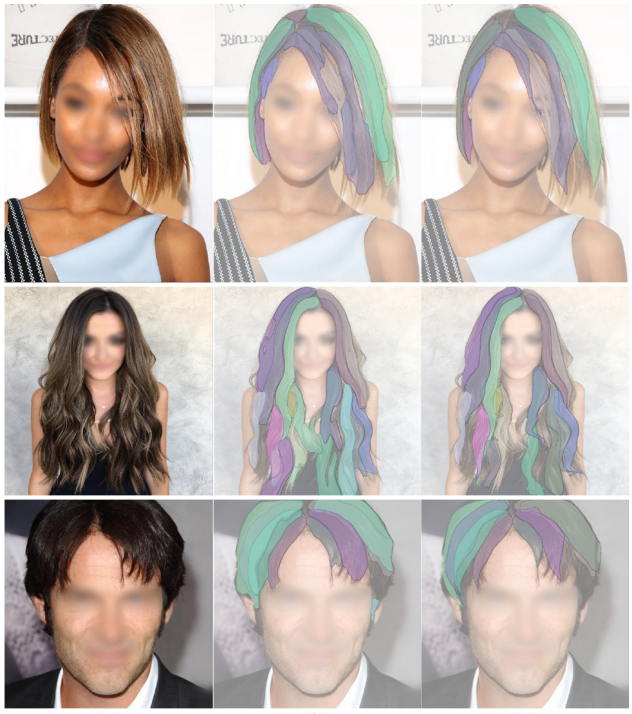
Table 1: Quantitative comparison results on 114 portrait images of SketchHairSalon[14]. Our method outperforms state-of-the-art approaches.

Runtime on GPU. Our method take 5.0s per generated video averagely on a single RTX 3090 GPU.



Input image Dorkenwald *et al*. [4] Blattmann *et al*. [1] Ours

Figure 2: Qualitative comparisons of our approach with methods for video generation. For a testing image, we show a full frame at the top and zoom-in rectangle regions marked by red/green at the bottom.

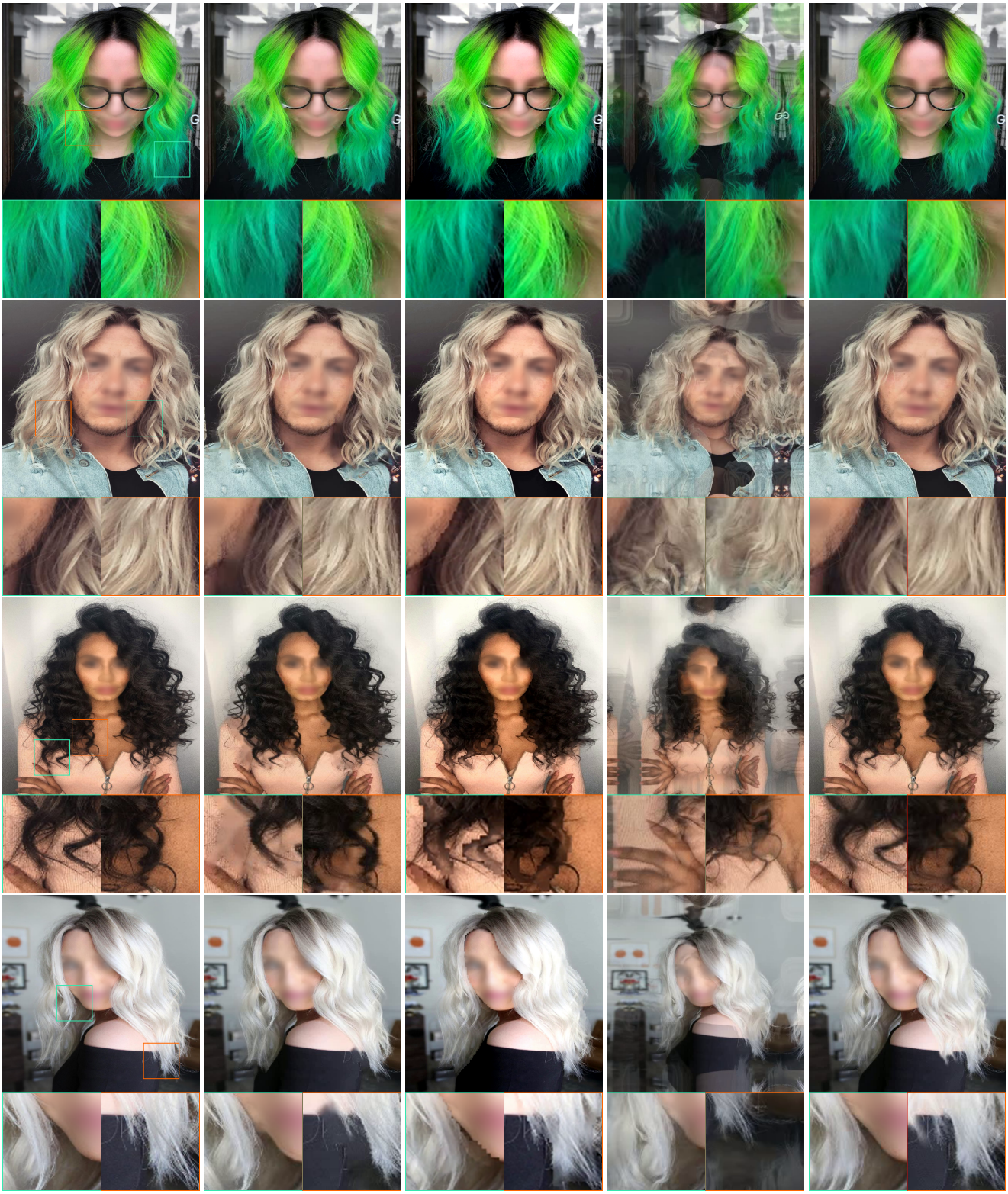


(a) (b) (c)

Figure 3: Illustraton of hair wisp extraction results. (a) input images, (b) coarse hair wisp extraction results by instance segmentation, (c) final hair wisp extraction results by refining coarse ones.

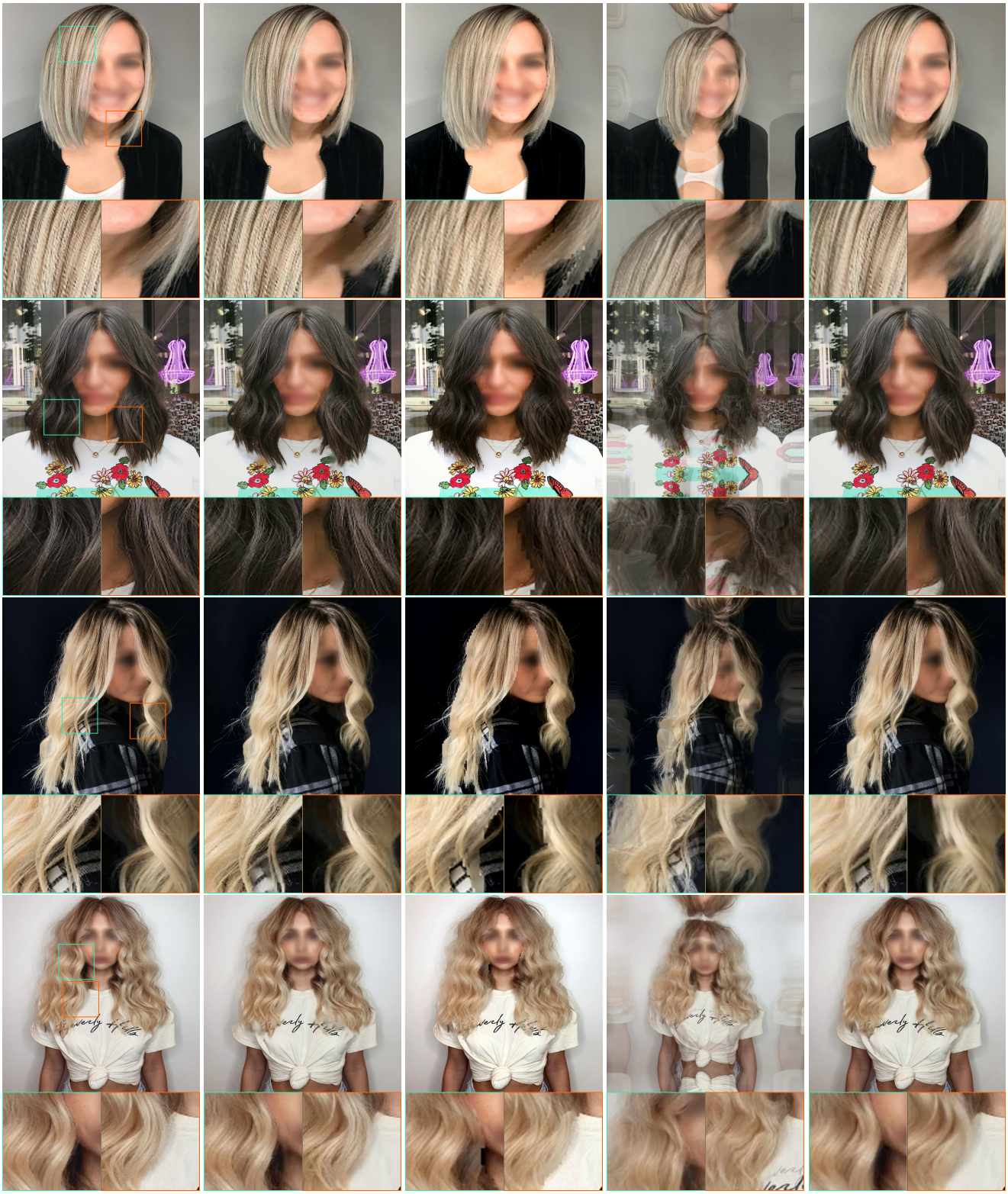
Input image Chai *et al*. [2] Ours

Figure 4: Qualitative comparisons of our approach with Chai *et al*. For a testing image, we show a full frame at the top and zoom-in rectangle regions marked by red/green at the bottom.



Input image | Chuang *et al*. [3] | Halperin *et al*. [6] | Endo *et al*. [5] | Ours

Figure 5: Qualitative comparisons of our approach with previous methods. For a testing image, we show a full frame at the top and zoom-in rectangle regions marked by red/green at the bottom.



Input image | Chuang *et al*. [3] | Halperin *et al*. [6] | Endo *et al*. [5] | Ours

Figure 6: Qualitative comparisons of our approach with previous methods. For a testing image, we show a full frame at the top and zoom-in rectangle regions marked by red/green at the bottom.

References

[1] Andreas Blattmann, Timo Milbich, Michael Dorkenwald, and Björn Ommer. ipoke: Poking a still image for controlled stochastic video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14707–14717, 2021. 2, 3

[2] Menglei Chai, Lvdi Wang, Yanlin Weng, Xiaogang Jin, and Kun Zhou. Dynamic hair manipulation in images and videos. *ACM Transactions on Graphics (TOG)*, 32(4):1–8, 2013. 2, 3

[3] Yung-Yu Chuang, Dan B Goldman, Ke Colin Zheng, Brian Curless, David H Salesin, and Richard Szeliski. Animating pictures with stochastic motion textures. In *ACM SIGGRAPH 2005 Papers*, pages 853–860. 2005. 2, 4, 5

[4] Michael Dorkenwald, Timo Milbich, Andreas Blattmann, Robin Rombach, Konstantinos G. Derpanis, and Bjorn Ommer. Stochastic image-to-video synthesis using cinns. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3742–3753, June 2021. 2, 3

[5] Yuki Endo, Yoshihiro Kanamori, and Shigeru Kuriyama. Animating landscape: self-supervised learning of decoupled motion and appearance for single-image video synthesis. *ACM Transactions on Graphics (Proc. of SIGGRAPH ASIA 2019)*, 38(6):175:1–175:19, 2019. 2, 4, 5

[6] Tavi Halperin, Hanit Hakim, Orestis Vantzos, Gershon Hochman, Netai Benaim, Lior Sassy, Michael Kupchik, Ofir Bibi, and Ohad Fried. Endless loops: Detecting and animating periodic patterns in still images. *ACM Trans. Graph.*, 40(4), Aug. 2021. 2, 4, 5

[7] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[9] Jie Hu, Liujuan Cao, Yao Lu, ShengChuan Zhang, Yan Wang, Ke Li, Feiyue Huang, Ling Shao, and Rongrong Ji. Istr: End-to-end instance segmentation with transformers. *arXiv preprint arXiv:2105.00637*, 2021. 1

[10] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 1

[11] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 1

[12] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 1

[13] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 1

[14] Chufeng Xiao, Deng Yu, Xiaoguang Han, Youyi Zheng, and Hongbo Fu. Sketchhairsalon: Deep sketch-based hair image synthesis. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2021)*, 40(6):1–16, 2021. 1, 2