PFStorer: Personalized Face Restoration and Super-Resolution
Supplementary Material

This supplementary material contains the following sections. First, background is provided for the used models. Next, further details of the user study are provided. Then, full experimental details and additional experiments of the personalized model are shown. Finally, the training details and further experiments of the non-personalized base restoration model are discussed. As the last section, societal impact is discussed.

Background

In this section we provide sufficient background to keep the paper self-sustained. We first introduce latent diffusion models, namely Stable Diffusion [12], as both the used methods, StableSR [14] and ViCo [3] are based off of it. Next we provide further details of the base model, StableSR and the personalization technique ViCo.

Latent Diffusion Models As oppose to diffusion models [5], LDMs [12] (latent diffusion models) perform the diffusion steps in a latent space. In Stable Diffusion [12] an encoder $\mathcal{E}$ is first trained to map input images $x \in \mathbb{R}^{H\times W\times 3}$ to a latent code $z = \mathcal{E}(x) \in \mathbb{R}^{(H/8)\times(W/8)\times 4}$, which can be approximately reconstructed with a decoder $\mathcal{D}$. The diffusion forward and backward steps are then performed within the latent space. To perform conditional generation using text y, it is first transformed to an embedding $c(y)$ with a text embedder. The training loss is then given by:

$$\mathcal{L} = \mathbb{E}_{z\sim\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\left[||\epsilon - \epsilon_\theta(z_t, c(y), t)||_2^2\right]. \quad (1)$$

Above, at timestep t the diffusion model ϵ_{theta} denoises the added noise from z_t conditioned on the text embedding $c(y)$ and the timestep t. After training the model can be used to generate images with a text prompt y and starting from a $z_t \sim \mathcal{N}(0, 1)$ iteratively until all noise is removed at $t = 0$.

StableSR To exploit the rich generative prior in Stable Diffusion [12] for image restoration, StableSR [14] uses conditioning of low-quality images. The entire Stable Diffusion is kept frozen, while a time-aware encoder and spatial feature transformations are added as adapters to condition the low-quality images. The encoder takes in a low-quality image and the current time-step of the diffusion model and outputs feature maps at different resolutions, corresponding to the ones in Stable Diffusion's UNet. The features are then fed through spatial feature transforms and added to the output of the original UNet layer's intermediate outputs.

The model is trained on high-quality generic natural images of 2k and 8k resolutions, which are randomly cropped to 512×512. This training enables the use of arbitrary size super-resolution using aggregation sampling. Here, the input is split to overlapping tiles, which are processed by the model independently. To avoid border artifacts a Gaussian kernel is used for the fusion of the tiles.

ViCo For efficient and accurate personalization, ViCo [3] also uses Stable Diffusion as its base. Similarly to StableSR, ViCo keeps the entire Stable Diffusion frozen. Only added adapter blocks and a single text-embedding are trained. Similar to [2], a text-embedding is made learnable that can be associated with the subject. Additionally, image cross-attention adapters are added to four blocks of the UNet. These cross-attention layers take the current time-step's intermediate result and the intermediate features of a reference image, which has also gone through the UNet. To enhance the result, a mask is used to ignore the background. The mask is obtained from an attention map $A = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)$, which is a product of the reference image and the text-embedding. A regularizer

$$\mathcal{L}_{Pers} = ||A_\star/\max(A_\star) - A_{\mathrm{EOT}}/\max(A_{\mathrm{EOT}})||_2^2, \quad (2)$$

where $A_\star$ are the similarity logits corresponding to the learnable token of the text-embedding and A_{EOT} is the end-of-text token, is used to avoid overfitting. The end-of-text token $< |\mathrm{EOT}| >$ captures a global representation, which retains good semantics of the personalizable object through training.

User Study Details

From the light and heavy partitions we randomly select two images for each identity. From the real data, we select all of the images. In total we have $20 \times 2 + 20 \times 2 + 20 \times 1 = 100$ images. With 40 users and two tasks we have a total of 8000 unique answers. As identifying fine-grained details of an identity, especially not a familiar one, can be difficult, we chose four images from each identity (light and heavy). This way the users can get more accustomed to the identities and make more accurate evaluations for the similarity of the identity features.

To ensure that the users are being accurate with their annotations, we use five control tasks. Here a ground-truth image is paired with extremely poor quality images. If the user fails in these tasks, their annotations are likely to be inaccurate and their results can be potentially invalidated. To avoid biases with users always choosing A, B or C, we randomly shuffle the model's positions.

Figure 1 displays the user interface used for the study. Users have to read the full instructions before taking the

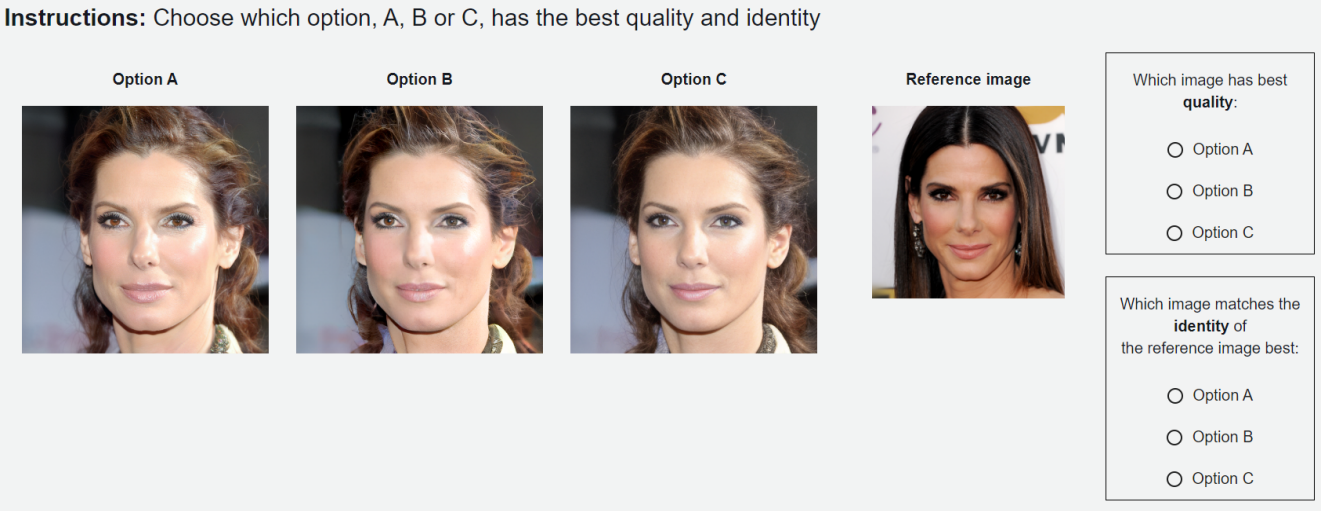

Figure 1. User interface used in the user study.

study. The instructions detail the two different tasks and how they should be evaluated.

Amazon Mechanical Turk is used for the study. We follow principles from [11]. We filter users based on the Master certificate to ensure quality annotations. For each task we pay $ 0.04 as suggested in [11].

Personalized Model Additional Experiments

In this section we detail the full experimental settings and conduct additional experiments of hyperparameters for the personalized model. In all experiments, excluding the parameter to be studied other hyperparameters are kept constant. With further fine-tuning of hyperparameters, results for specific individuals and inputs can be improved.

Settings Followed by community findings that prompts can improve quality of the restored image, we use both a positive and a negative prompt. For the positive prompt we use *"a Photo of * , masterpiece, best quality, realistic, very clear, professional"* and for the negative prompt we use *"3d, cartoon, anime, sketches, worst quality, low quality"*. We note that including semantic changes in the prompt like *"red hair"* does not have an effect. This is due to the restoration blocks fusing the low-quality image with the denoised image directly. This is in line with the goal of the paper, as it is a restoration method, not an editing method.

For the classifier-free guidance value we set 4 as a default for all experiments. Standard DDPM [5] sampling is used with 200 steps as in StableSR [14]. As the personalization fine-tuning approach is about learning a single identity and not multiple parts of an identity, we find that not using the 50% random crops improves the results slightly. For the baseline method DR2 + SPAR [16] we empirically experimented with several hyperparameters values of N and τ that are crucial for the performance of the method. N is a downsampling factor and τ is the output step after which generation is started. We set $N = 8$ and $\tau = 40$ as we found it performed the best across different levels of degradations. For LMSE (Landmark MSE) [18] was used to obtain landmarks. In cases where landmarks could not be found due to the image being severely degraded the MSE was set to 128. Similarly in cases where the MSE was for an image was more than 128 it was capped to 128 to avoid outliers due to numerical errors or other errors.

Degradations During testing we synthesized a light and a heavy degradation to better evaluate our algorithm in different situations. During training we use the heavy setting. We use the settings from StableSR as a base and modify them. To better suit for real-world applications we include motion and median blur, as well as adding ISP (Image Signal Processing) noise [17].

To ensure our method works in less severe cases, we also include a light partition during testing. Here, we only include a first-order noise similar to CodeFormer.

The light portion follows:

$$I_D = \{[(I \circledast k_\sigma)_{\downarrow_r} + n_\delta]_{\text{JPEG}_q}\}_{\uparrow_r}, \tag{3}$$

where k_σ is Gaussian blur kernel, $\downarrow_r$ and $\uparrow_r$ are the downsamping and upsampling operators, n_δ additive Gaussian noise and $[\cdot]_{\text{JPEG}_q}$ is JPEG compression. We sample uniformly σ, r, δ and q from $[0.1, 10]$, $[1, 4]$, $[0, 2]$ and $[30, 100]$, respectively. The additive Gaussian noise has a probability of 40% and downsampling a probablity of 70%, while filtering and JPEG compression occur always.

The heavy portion first applies ISP model [17] with a 50% probability, followed by motion and median blur with

5% and 10% probabilities. Next, we use equation 3 and the same settings except, r and δ are chosen from $[1, 10]$ and $[0, 15]$, respectively, followed by a sinc filter [14]. Finally, equation 3 is applied a second time with a 90% probability.

Classifier-Free Guidance Value To further emphasize the conditional element, CFG [4] can be used to guide the denoising process. As mentioned earlier, we use a negative prompt instead of a null one. The formula is given by

$$\tilde{X} = X + \lambda_{cfg}(X(p_{pos}, I_{LQ}) - X(p_{neg}, I_{LQ})), \quad (4)$$

where p_{neg} and p_{pos} correspond to the positive and negative prompts. We also experimented with null conditioning the low-quality image

$$\tilde{X} = X + \lambda_{cfg}(X(p_{pos}, I_{LQ}) - X(p_{neg}, \varnothing)), \quad (5)$$

but found the results to be of lower-quality, as emphasizing the low-quality image may exaggerate blurry features.

We experiment using Eq. (4) with different CFG values λ_{cfg} in Fig. 2 and note that $\lambda_{cfg} = 1$ corresponds to not using guidance at all. It can be seen that a higher λ_{cfg} can oversature, as in the top row. In the lower row, a low λ_{cfg} loses identity features, whereas in the top row it is more subtle. From our experiments we observe that different identities behave differently with different λ_{cfg}. A common value in text-to-image applications is $\lambda_{cfg} = 7.5$, but to avoid saturation we default to $\lambda_{cfg} = 4$ in all of our other experiments.

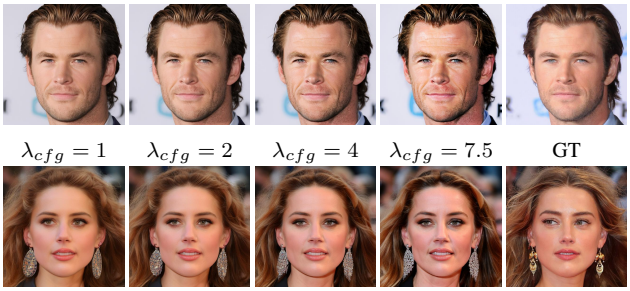

Figure 2. Experiments with different classifier-free guidance values.

Controlling Identity The used personalization technique consists of two parts. 1) a learnable text-embedding and 2. image cross-attention. We experiment with different values of λ_{att}, which controls the weight of the cross-attention layers, in Fig. 3. The sample with $\lambda_{att} = 0$ corresponds to only using the learnable text-embedding. We can see that it contains some identity at a high-level but is missing details such as wrinkles and dip in the chin. With increasing λ_{att} the identity features become more prominent, even to a degree of exaggeration. Similar to CFG values, we have observed that for different individuals and depending on the noise levels of the input, λ_{att} acts differently. We chose $\lambda_{att} = 1$ as a default value, although in some cases, like this sample, the optimal results can be something different.

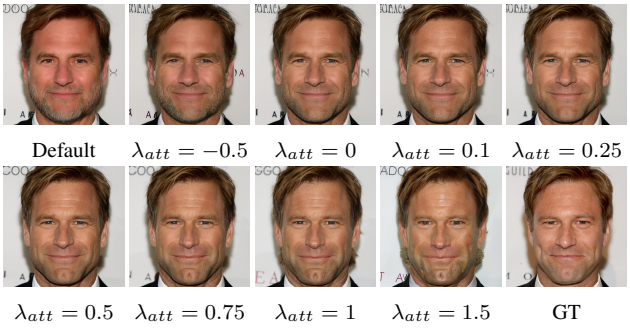

Figure 3. Controlling identity. The default corresponds to non-personalized output. Samples with λ_{att}, use the personalized token in the prompt.

Number of Reference Images In Fig. 4 we experiment with how many reference images are required to accurately capture the identity. With $n_{img} = 0$, *i.e.* no personalization, high-level features matching the input can be observed. With just one reference image, the eyes, eyebrows and other finer details start to appear. We default to using $n_{img} = 5$ as it often performs sufficiently and the addition of more images has less noticeable effect. For some individuals we found that even three images can be sufficient, but it should be noted that the similarities between input image and the reference images affect the results.

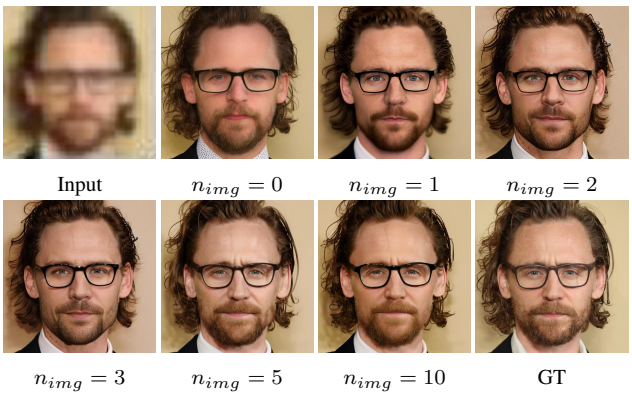

Figure 4. Number of images used for personalization. $n_{img} = 0$ refers to no personalization.

Randomness with Different Seeds Diffusion models are stochastic and notorious for unsatisfactory results with different random initializations. Figure 5 contains results for

an image with light and heavy degradations with four different seeds. For the light portion, the outputs tend to be mostly similar with small differences like skin texture. With the heavy portion, there are noticeable differences in the mouth, eyes and colors, although the identity is kept the same. Interestingly the background logo and text deviate largely, as they are not part of the learned personalization.

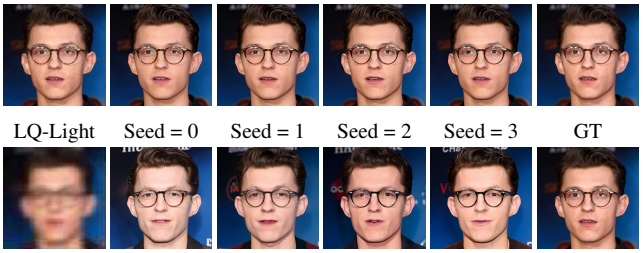

Figure 5. Random seed effect.

Additional Qualitative Results Here we provide additional results on the light, heavy and real portions of the Celeb-Ref [9] dataset. From Fig. 7 row three, we can see that the DMDNet is able to better preserver the identity compared to CodeFormer, but compared to ours it is still missing fine-grained details such as the notch in the chin. Despite achieving good results with light degradations, DMDNet struggles with the heavy and real degradations in Figs. 8 and 9. Although DR2 provides poor results in several cases, it works well on row 4 of Fig. 8.

Despite the input being very noisy and small in size, our result is faithful with the identity, while codeformer struggles due to requiring alignment. Figure 6 contains a qualitative comparison between different personalization techniques and more results are provided in Figs. 10 and 11.

Additional Quantitative Results To complete the quantitative results of heavy portion from **??**, the results of light and real portions are presented.

Table 1 tabulates the results for the real portion. As no GT is available, we only use MUSIQ [8] and ID [1] as metrics. For the ID we use a reference image of the same person. As can be seen from the results, the ID metric drops significantly compared to the heavy portion, where ID used GT image. Despite this the rankings of the results remain similar with ours as first and DMDNet and CodeFormer being close with similar results and DR2 achieving the lowest due to blurry results. Base Model + DreamBooth [13] achieves the best result in ID which is likely due to overfitting to the identity, with poor restoration results. Table 2 presents results for the light partition. Our method is consistently among the second best performers, although the differences between the methods are minor.

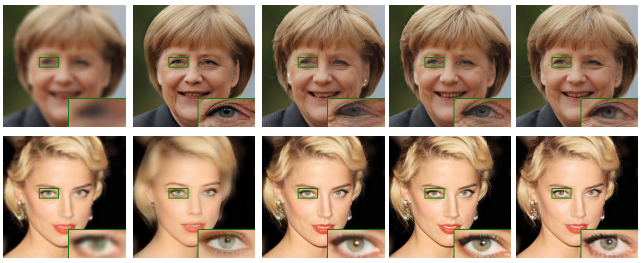

Figure 6. Results using different personalization techniques combined with a base restoration model. DreamBooth [13] is not able to capture all the details and can result in poor-quality images. ViCo is better able to capture most details, but can still result in blurry images. Ours is able to capture fine-grained details without hurting the restoration performance of the Base Model. Zoom in for best view.

Table 1. Quantitative results for the real portion of the data. Red indicates the best and blue indicates the second best

Methods	Ref	MUSIQ ↑	*ID ↑
Input		24.84	20.37
StableSR [14]		51.59	23.39
Base Model		60.27	24.29
Base Model + DreamBooth	✓	55.31	29.78
Base Model + ViCo	✓	57.67	24.71
DMDNet [9]	✓	58.36	22.29
DR2 [16]		29.18	18.38
CodeFormer [19]		44.60	22.90
PFStorer (Ours)	✓	60.11	25.01

* Compare with a reference image.

Value of Learnable γ after training Each layer l has vector γ^l with the size depending on the layer hidden dimension. The values of the mean of the vector for each layer is around 0.2 and 0.5. The higher importance of 0.5 values is from the middle of the UNet layers, where the resolution is lowest and the lower values of 0.2 at the higher resolution layers.

Non-Personalized Base Model Experiments

In this section we cover the training details and results with the Base Model without personalization. The model is a pre-trained StableSR [14] without any modifications to the architecture. The personalized models all use the fine-tuned Base Model described in this section as their starting point.

Training The training is performed on a facial dataset FFHQ [7], which contains 70,000 facial images in the resolution of 1024×1024. 50% of the data is resized randomly to 512×512 and the other 50% are taken as ran-

Table 2. Quantitative results for the light portion of the data. Red indicates the best and blue indicates the second best

Methods	Ref	PSNR ↑	SSIM ↑	LPIPS ↓	MUSIQ ↑	LMSE ↓	ID ↑
Input		22.56	0.719	0.615	58.83	9.74	21.85
StableSR		27.18	0.767	0.337	62.96	5.73	70.62
Base Model		27.72	0.767	0.318	64.16	4.93	72.43
Base Model + DreamBooth	✓	24.77	0.721	0.419	62.01	14.29	62.57
Base Model + ViCo	✓	27.58	0.765	0.325	63.04	4.88	73.26
DMDNet [9]	✓	27.72	0.780	0.312	63.07	6.43	72.66
DR2 [16]		22.17	0.701	0.449	47.36	13.13	30.01
CodeFormer [19]		27.19	0.759	0.293	66.00	5.91	69.13
PFStorer (Ours)	✓	27.71	0.767	0.309	63.31	4.79	75.39
GT		∞	1	0	62.37	0	100

dom crops of the same resolution. Fine-tuning is performed for 12 epochs. At this moment the personalizion adapter is not attached to the model. We synthesize training data in the same manner as the personalized model with the heavy degradation.

Qualitative Results For qualitative results on synthetic degradation on CelebA-Test split [10], see Fig. 12. The synthetic degradation for Celeba-Test is obtained from [15]. Compared to CodeFormer [19] our method is able to generate more fine-grained details, while being more faithful to the low-quality image, *e.g.*, the color of facial hair on top. Results from real-world datasets, LFW [6], WebPhoto [15] and Wider-Test [19] are shown in Fig. 13. In LFW, which contains less severe degradations, compared to CodeFormer our method is able to generate more details with sharper textures. Our method struggles with WebPhoto, as it contains old images with scratches, color degradation and other untypical degradations. With severe degradation on Wider-Test, our method is able to generate realistic images, while CodeFormer struggles with artifacts.

Quantitative Results We provide quantitative results with standard metrics. Table 3 tabulates results from CelebA-Test, where the results are taken from [15], except for CodeFormer and ours. In most of the metrics the results are similar between GFP-GAN, CodeFormer and ours. In the real-world datasets, Tab. 4, our method obtains the best FID for LFW and WIDER.

Table 3. Quantitative results for CelebA-Test with non-personalized model. Red indicates the best and blue indicates the second best

Methods	PSNR ↑	SSIM ↑	LPIPS ↓	MUSIQ ↑	FID ↓	ID ↑
Input	25.35	0.684	0.486	58.83	143.98	52.06
DFDNet	23.68	0.662	0.4341	N/A	59.08	59.69
PULSE	21.61	0.620	0.4851	N/A	67.56	30.45
GFP-GAN	25.08	0.677	0.3646	N/A	42.62	65.40
CodeFormer	26.77	0.719	0.343	66.54	52.44	62.73
Base Model	26.03	0.680	0.392	66.57	40.36	63.89
GT	∞	1	0	63.43	43.43	1

Table 4. Quantitative results for real-world datasets with non-personalized model. Red indicates the best and blue indicates the second best

Dataset	**LFW-Test**		**WebPhoto-Test**		**WIDER-Test**	
Degradation	mild		medium		heavy	
Methods	FID↓	MUSIQ↑	FID↓	MUSIQ↑	FID↓	MUSIQ↑
Input	137.56	25.05	170.11	19.24	202.06	15.57
PULSE	64.86	66.98	86.45	66.57	73.59	65.36
DFDNet	62.57	67.95	100.68	63.81	57.84	59.34
GFP-GAN [15]	49.96	68.95	87.35	68.04	40.59	68.26
CodeFormer [19]	52.02	71.43	78.87	70.51	39.06	69.31
Base Model	44.11	66.57	80.90	62.69	34.72	63.91
Light degradation	44.02	62.69	84.81	57.64	82.93	51.66

Ablation: Heavy Degradation We show that with more complex degradations the method is able to perform better in cases with severe degradation. The results are tabulated in bottom of Tab. 4. *Base Model* uses the heavy degradation, where as the *Light degradation* does not. For LFW, which has relatively mild degradations, the performance between Base Model and simple degradation does not change drastically as expected. However, for WIDER-Test we can see a large difference as the FID more than doubles from 34.72 to 82.93, meaning a significant decrease in quality. Using heavy degradation results in higher quality outputs under severe degradation, while having minimal effect on mild cases.

Societal Impact

Machine learning models can learn biases from their datasets. We show that our model is capable of working with different ethnicities and skin tones, while acknowledging that the testing is limited. We also note that since our model is built upon previous models, it inherits any biases these models may contain. To avoid misunderstanding of the capabilites of our models, *e.g.*, using it for enhancing security footage for criminal investigations, we have shown the limitations in experiments and emphasize that the identity of the restored individual needs to be known beforehand. Malicious users may want to mislead viewers with generated images, which is a common common issue with existing similar methods. However, recent approaches in detecting fake imagery are improving rapidly.

Privacy and Image Copyrights In this paper we showcase several pictures of individuals. Several images are from the publicly available Celeb-Ref dataset [9]. Images shown from the collected 20 image dataset are of well-known celebrities and are under a Creative Commons license. Real world images not part of the collected dataset are under public domain or a Creative Commons license.

Figure 7. Qualitative comparison with state-of-the-art restoration models on Celeb-Ref dataset [9] with synthetic light degradation.

References

[1] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 4

[2] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 1

[3] Shaozhe Hao, Kai Han, Shihao Zhao, and Kwan-Yee K Wong. Vico: Detail-preserving visual condition for personalized text-to-image generation. *arXiv preprint*

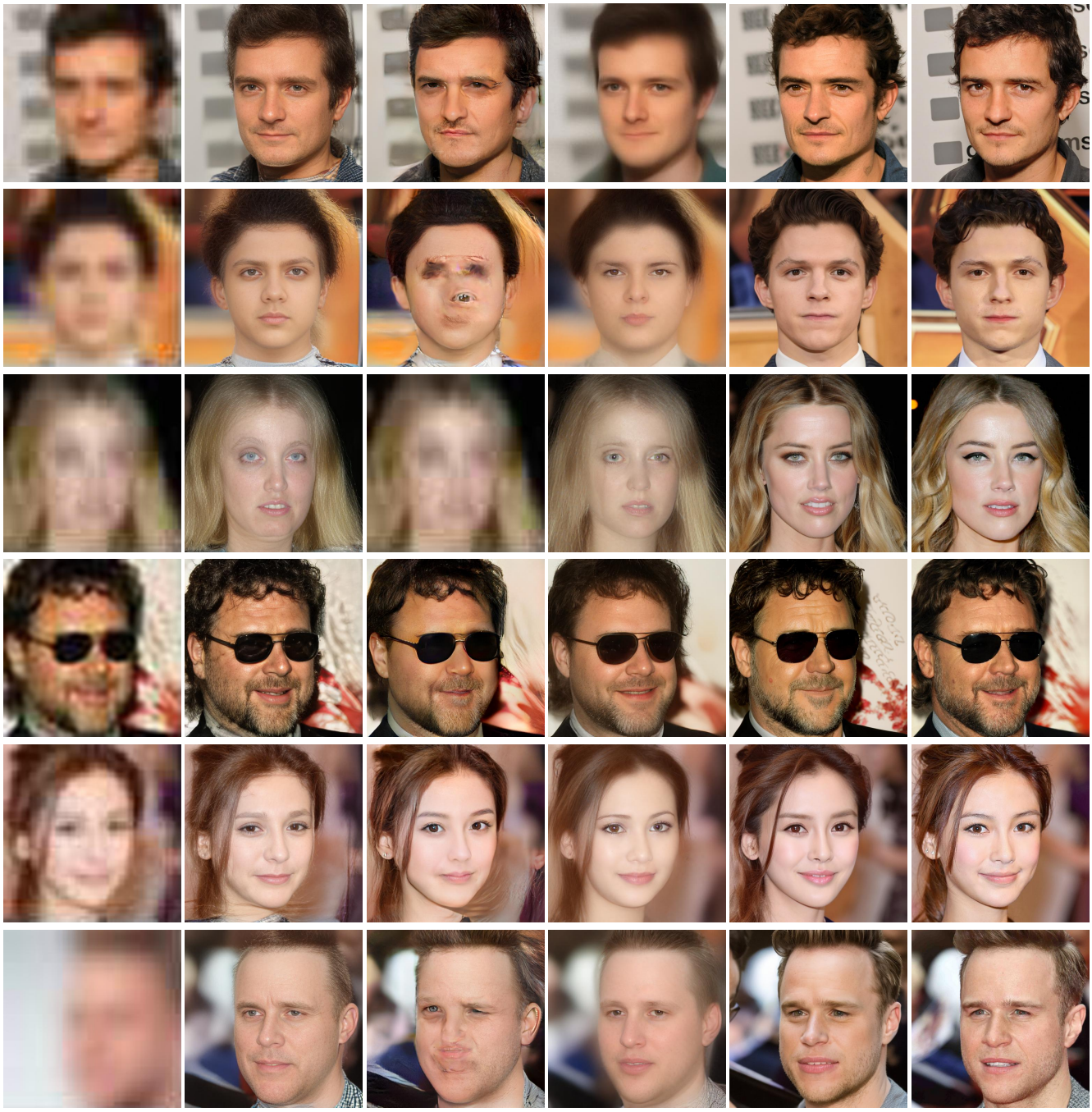

Figure 8. Qualitative comparison with state-of-the-art restoration models on Celeb-Ref dataset [9] with synthetic heavy degradation.

arXiv:2306.00971, 2023. 1

[4] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 2

[6] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008. 5, 12

[7] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of*

Figure 9. Qualitative comparison with state-of-the-art restoration models on Celeb-Ref dataset [9] with real degradation.

the IEEE/CVF conference on computer vision and pattern recognition, pages 8110–8119, 2020. 4

[8] Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5148–5157, 2021. 4

[9] Xiaoming Li, Shiguang Zhang, Shangchen Zhou, Lei Zhang, and Wangmeng Zuo. Learning dual memory dictionaries for blind face restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5904–5917, 2022. 4, 5, 6, 7, 8

[10] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages

3730–3738, 2015. 5, 12

[11] Mayu Otani, Riku Togashi, Yu Sawai, Ryosuke Ishigami, Yuta Nakashima, Esa Rahtu, Janne Heikkilä, and Shin'ichi Satoh. Toward verifiable and reproducible human evaluation for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14277–14286, 2023. 2

[12] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1

[13] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023. 4

[14] Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin C. K. Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution, 2023. 1, 2, 3, 4

[15] Xintao Wang, Yu Li, Honglun Zhang, and Ying Shan. Towards real-world blind face restoration with generative facial prior. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9168–9178, 2021. 5, 12

[16] Zhixin Wang, Ziying Zhang, Xiaoyun Zhang, Huangjie Zheng, Mingyuan Zhou, Ya Zhang, and Yanfeng Wang. Dr2: Diffusion-based robust degradation remover for blind face restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1704–1713, 2023. 2, 4, 5

[17] Kai Zhang, Yawei Li, Jingyun Liang, Jiezhang Cao, Yulun Zhang, Hao Tang, Deng-Ping Fan, Radu Timofte, and Luc Van Gool. Practical blind image denoising via swin-conv-UNet and data synthesis. *Machine Intelligence Research*, 2023. 2

[18] Yinglin Zheng, Hao Yang, Ting Zhang, Jianmin Bao, Dongdong Chen, Yangyu Huang, Lu Yuan, Dong Chen, Ming Zeng, and Fang Wen. General facial representation learning in a visual-linguistic manner. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18697–18709, 2022. 2

[19] Shangchen Zhou, Kelvin Chan, Chongyi Li, and Chen Change Loy. Towards robust blind face restoration with codebook lookup transformer. *Advances in Neural Information Processing Systems*, 35:30599–30611, 2022. 4, 5, 12

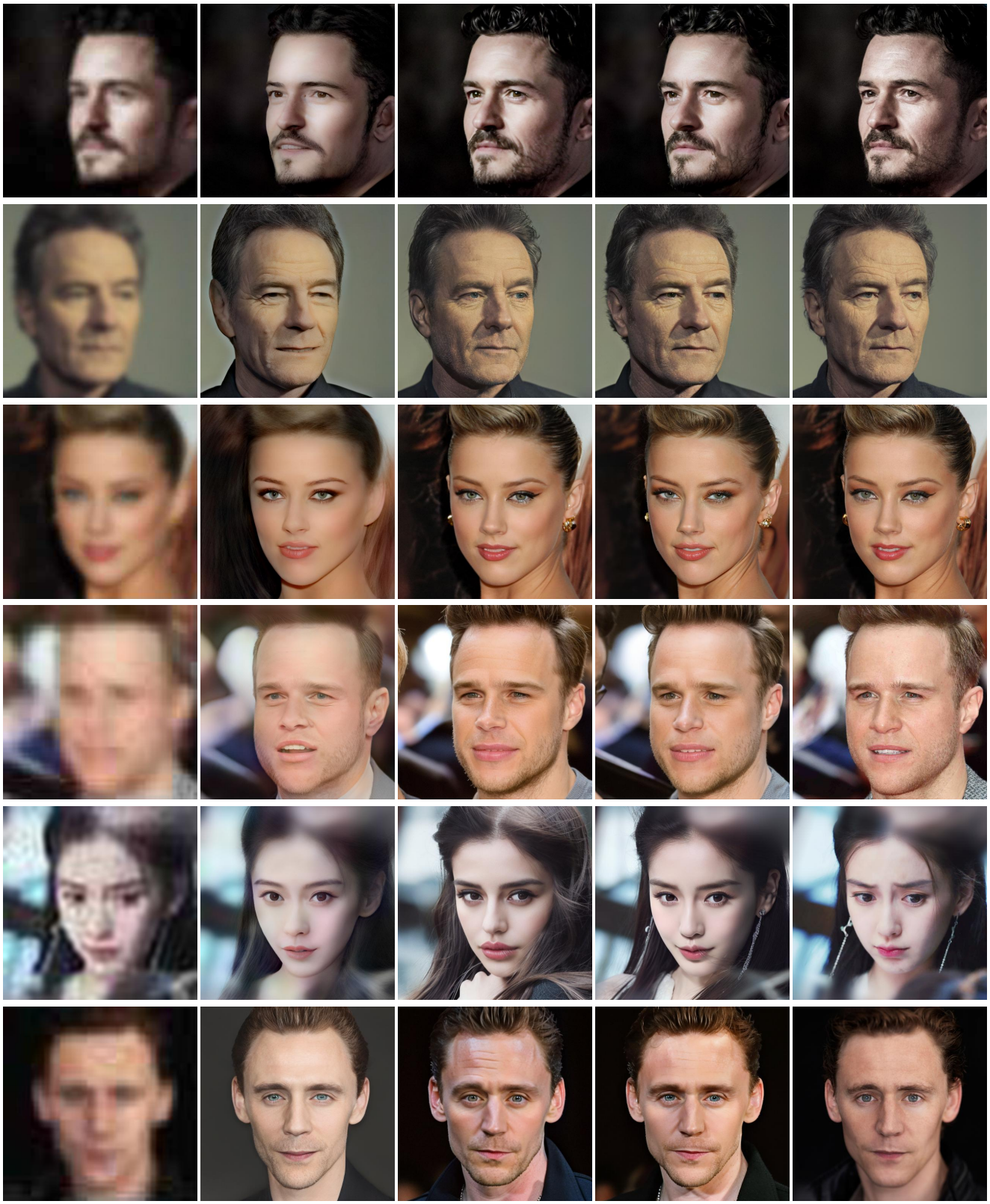

Figure 10. Results using different personalization techniques combined with a base restoration model with heavy degradation.

Figure 11. Results using different personalization techniques combined with a base restoration model with light degradation.

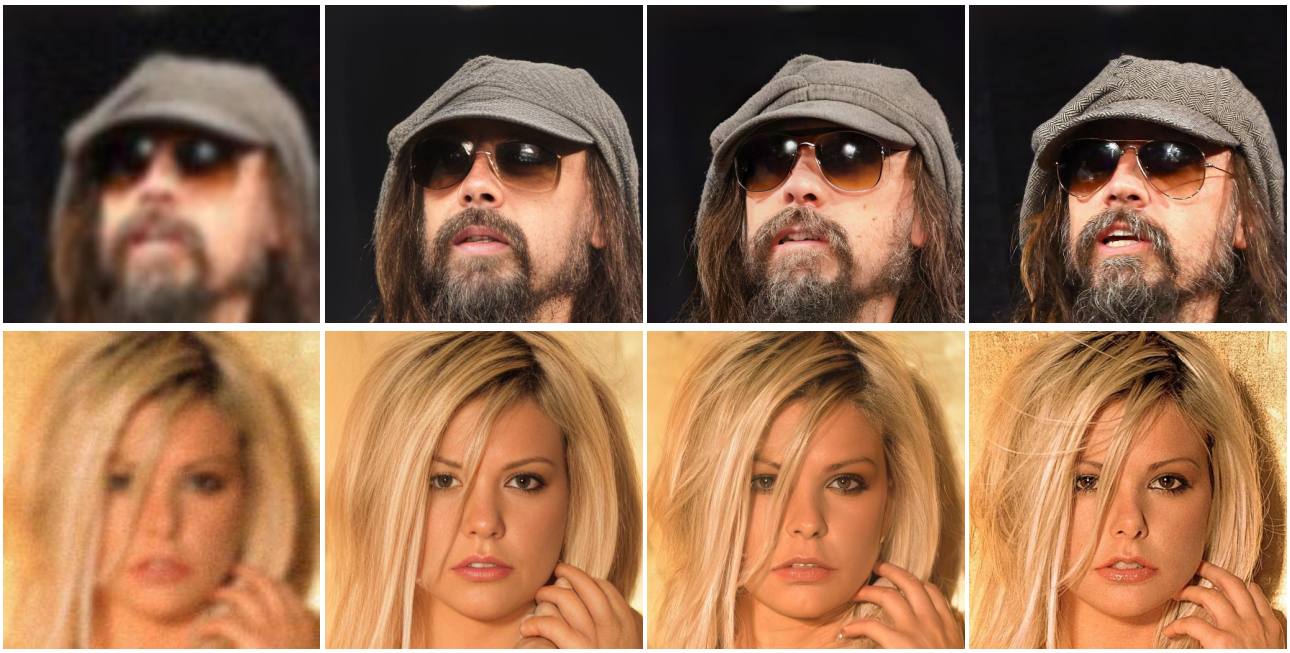

Figure 12. Qualitative comparison with state-of-the-art restoration models on CelebA-Test [10] with synthetic degradation.

Figure 13. Qualitative comparison with state-of-the-art restoration models on LFW [6], WebPhoto [15] and Wider-Test [19].