Reference-Based 3D-Aware Image Editing with Triplanes

Supplementary Material

6. Hyperparameters

We provide the mentioned hyperparameters of the main paper here.

Triplane gradient post-processing. We provide the code snippet for post-processing stage in Listing 1, and the default parameters for ϵ, μ, and σ. For the sake of simplicity, we omitted the batch dimension in $\mathbf{t}$.

Fusion of the triplanes. Eq. (6) describes the fusion operation, where $\mathbf{T}_{\text{imp}}$ is the implicitly fused triplane, and $\mathcal{E}$ is the morphological erosion operation applied on the post-processed triplane masks $\mathbf{M}$.

$$\mathbf{T}_{\text{f}} = \mathcal{E}(\mathbf{M}_{\text{ref}}) * \mathbf{T}_{\text{ref}} + \mathcal{E}(\mathbf{M}_{\text{src}}) * \mathbf{T}_{\text{src}} + (\mathcal{E}(\mathbf{M}_{\text{src}}) - \mathcal{E}(\mathbf{M}_{\text{ref}})) * \mathbf{T}_{\text{imp}} \tag{6}$$

Snippet for the masking operations are given in Listing 2. Morphological operations are realized with `max_pool2d` and Gaussian blurring. Note that `dilated_tp_mask` can be utilized when $\mathbf{M}_{\text{ref}} = 1 - \mathbf{M}_{\text{src}}$ is set. Batch size is set to 1.

7. Additional visual results

We showcase additional reference-based editing results in Figs. 14 to 18 for glasses, hairstyle, mouth, nose, and eyes on CelebA, and Figs. 19 and 20 for eyes and nose on AFHQ dataset. Reference and input images are encoded and reconstructed via EG3D-GOAE for human portraits, and synthesised for animal faces.

8. User study

We provide additional samples for the top 3 incorrect and correct responses in Fig. 21. For the correct responses, even though the users were able to distinguish between edited and non-edited samples, the editing is still faithful to the source images as there are no visible artifacts.

```python
import torchvision.transforms as T
def postprocess_tp_mask(t, epsilons=(0.9, 1.1), mu=7, sigmas=(0.1, 2.0)):
    blur_func = T.GaussianBlur(mu, sigma=sigmas)

    3,C_N,H,W = t.shape
    t = t.view(3*C_N,H,W) # 96 256 256 for EG3D and AG3D, 288 256 256 for PanoHead
    C = t.shape[0]
    for i in range(C):
        t[i] -= torch.min(t[i])
        t[i] /= torch.max(t[i])
    for i in range(C):
        mu = t[i].mean()
        # take around mean region
        t[i] = (t[i]<epsilons[1]*mu) * (t[i]>epsilons[0]*mu)
        t[i] = blur_func(t[i].unsqueeze(0)).squeeze(0)
    # get the inverse, we are interested in outside of the mean
    t = 1-t
    # binarize
    mean_grad = torch.mean(t, dim=0, keepdim=True).repeat(C,1,1)
    b_mask = torch.zeros_like(mean_grad)
    b_mask[mean_grad>mean_grad.mean()] = 1.0
    b_mask = blur_func(b_mask.unsqueeze(0)).squeeze(0)
    t = b_mask
    t = t.view(3,C_N,H,W)
    return t
```

Listing 1. Post-processing the triplane gradients.

```python
import torch.nn.functional as F
import torchvision.transforms as T
def create_dilated_eroded_tp_masks(self, tp_mask,
                                   blur_k_size=9, std_devs=(2.0,2.0), morph_k_size=11):
    blur = T.GaussianBlur(blur_k_size, sigma=std_devs)
    1,3,N_C,H,W = tp_mask.shape # N_C = 32 for EG3D and AG3D, 96 for PanoHead
    dilated_tp_mask = blur(F.max_pool2d(tp_mask.view(1,3*N_C,H,W), kernel_size=morph_k_size,
                    stride=1, padding=(morph_k_size - 1) // 2)).view(1,3,N_C,H,W)
    eroded_tp_mask = blur((1 - F.max_pool2d((1 - tp_mask.view(1,3*N_C,H,W)), kernel_size=morph_k_size,
                    stride=1, padding=(morph_k_size - 1) // 2))).view(1,3,N_C,H,W)
    return eroded_tp_mask, dilated_tp_mask
```

Listing 2. Dilating and eroding triplane masks.

Figure 14. Hair edits and 3D visualizations.

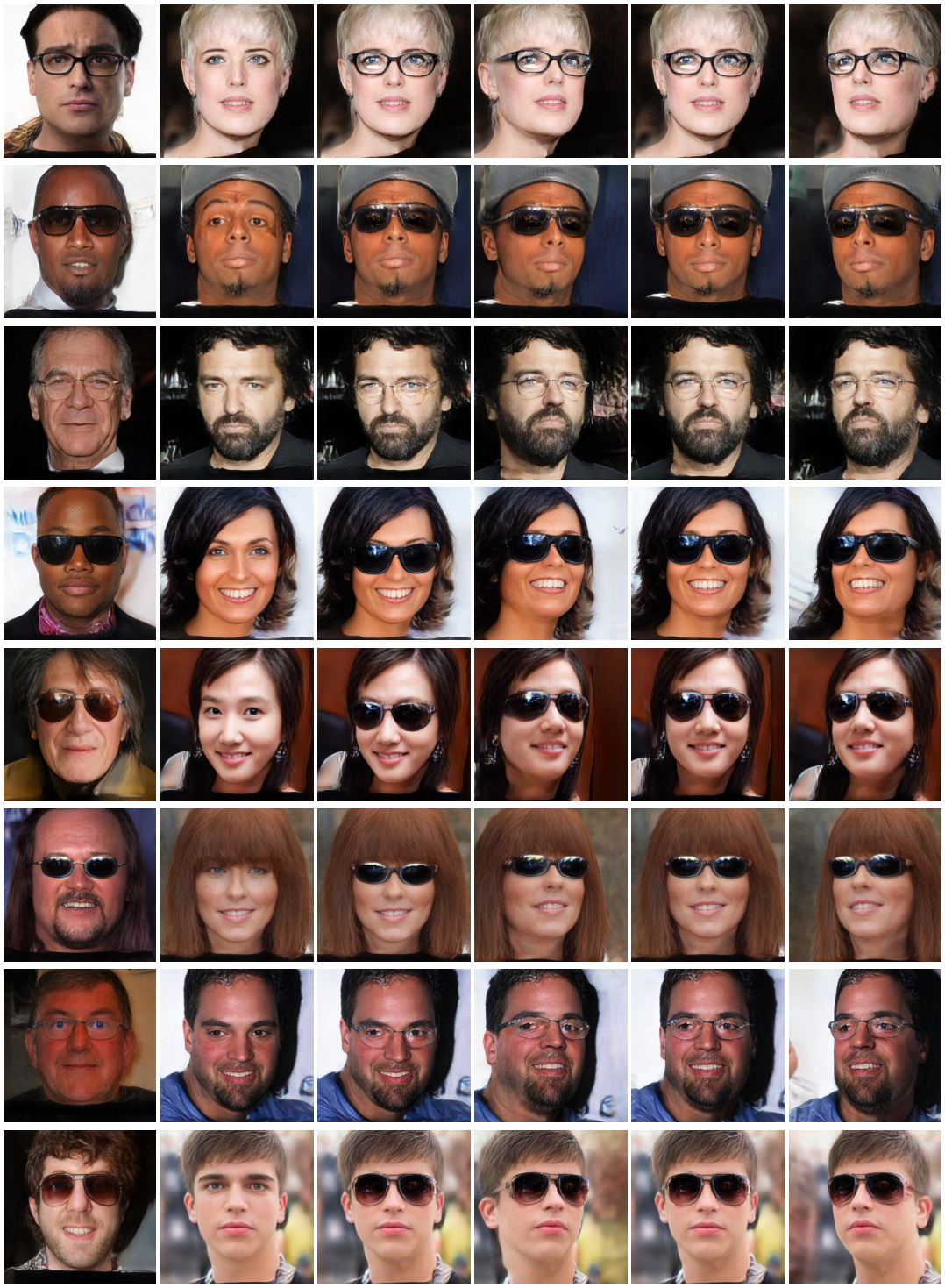

Figure 15. Glasses edits and 3D visualizations.

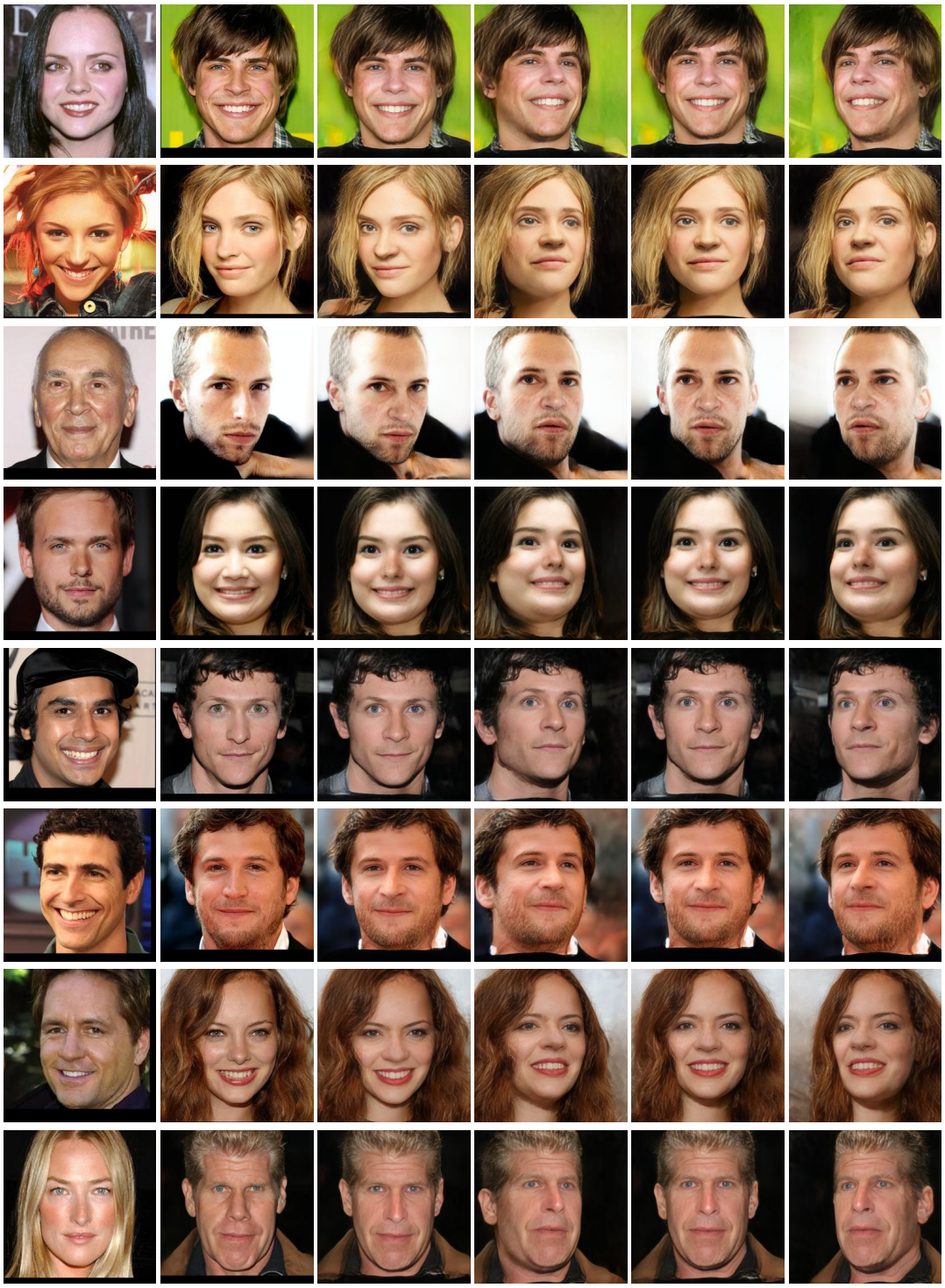

Figure 16. Nose edits and 3D visualizations.

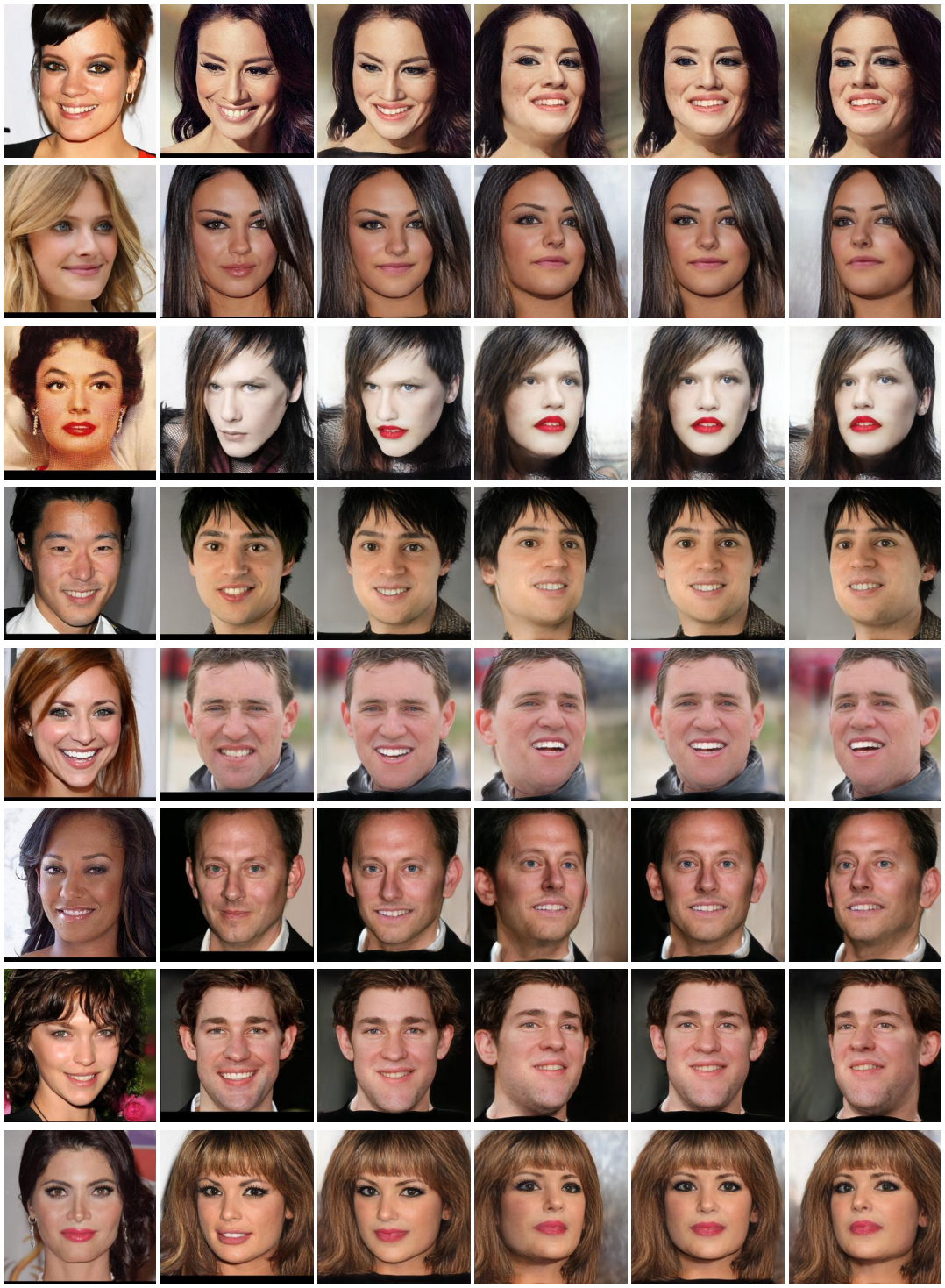

Figure 17. Mouth edits and 3D visualizations.

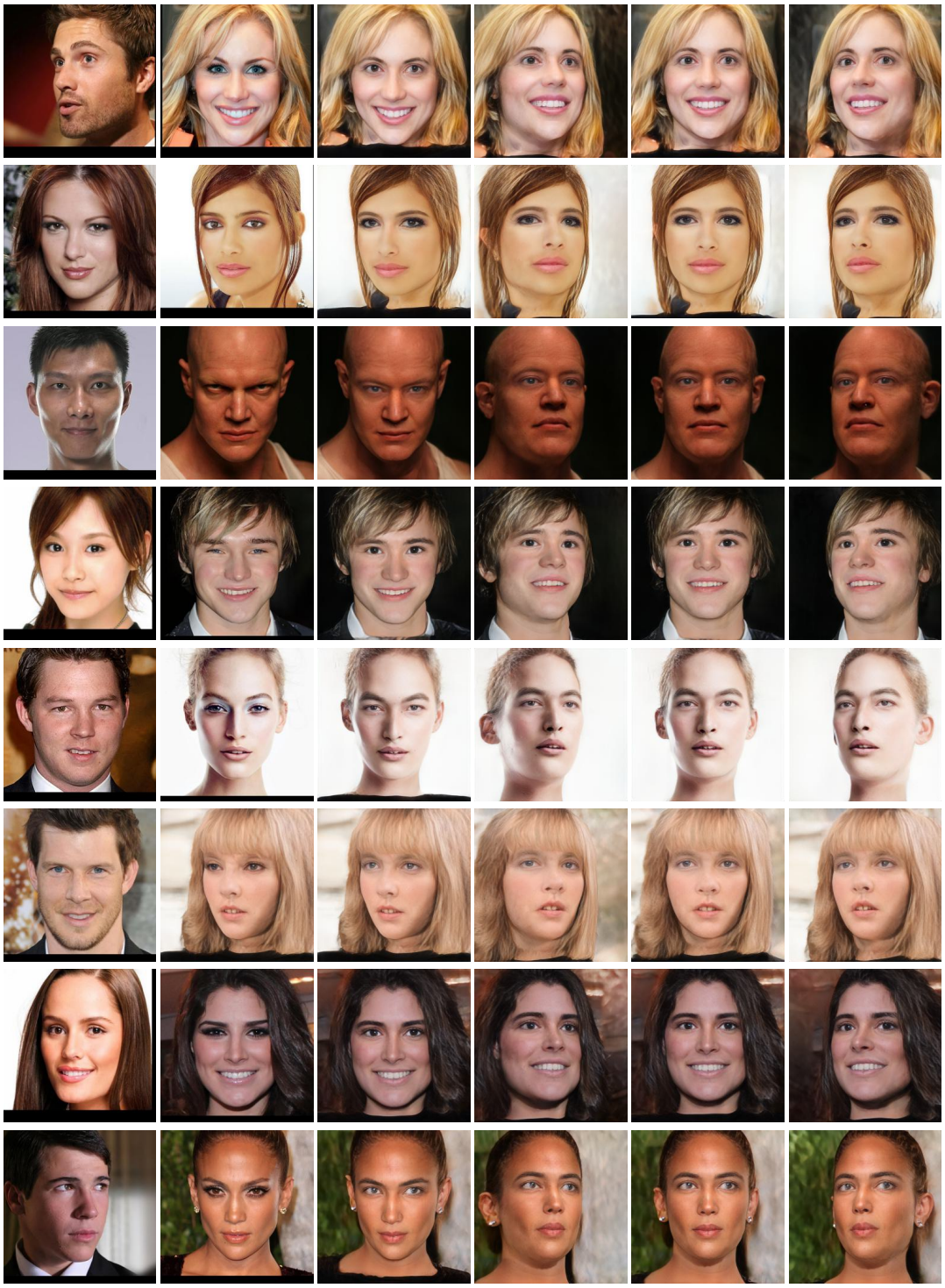

Figure 18. Eyes edits and 3D visualizations.

Figure 19. AFHQ eyes edits and 3D visualizations.

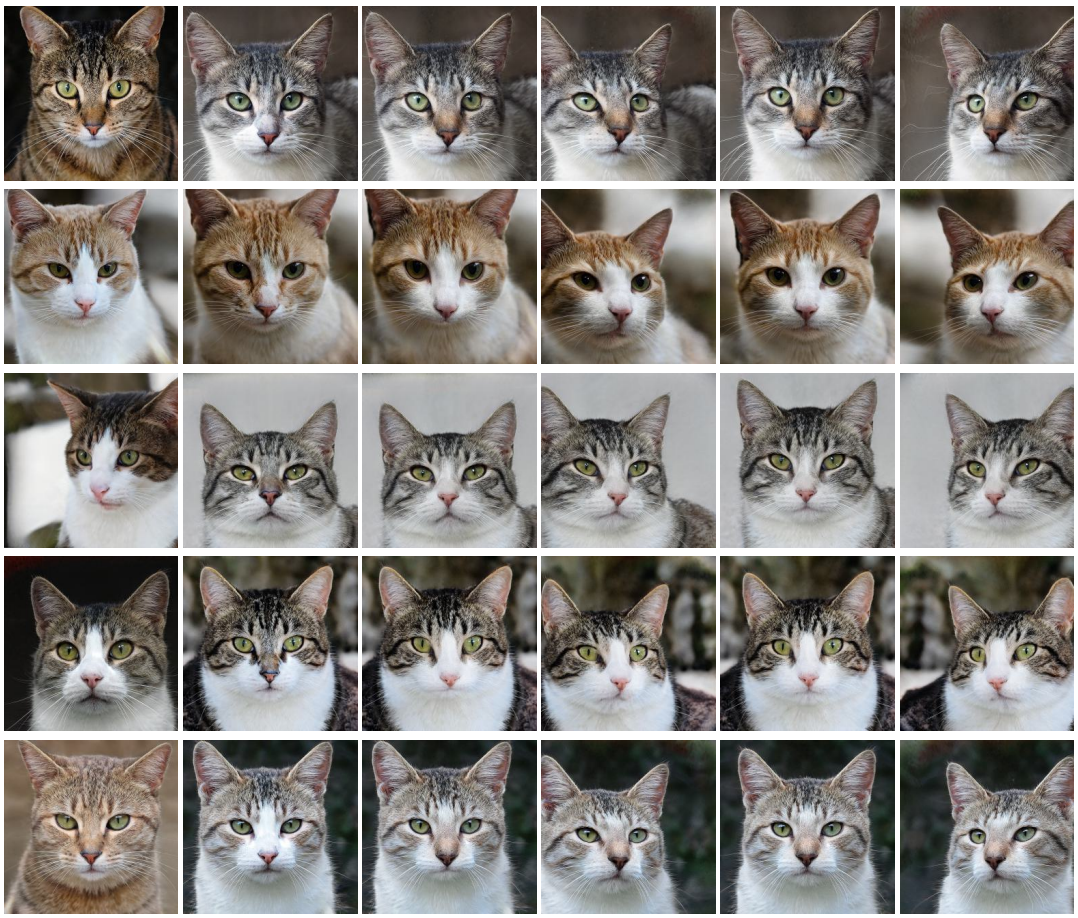

Figure 20. AFHQ mouth & nose edits and 3D visualizations.

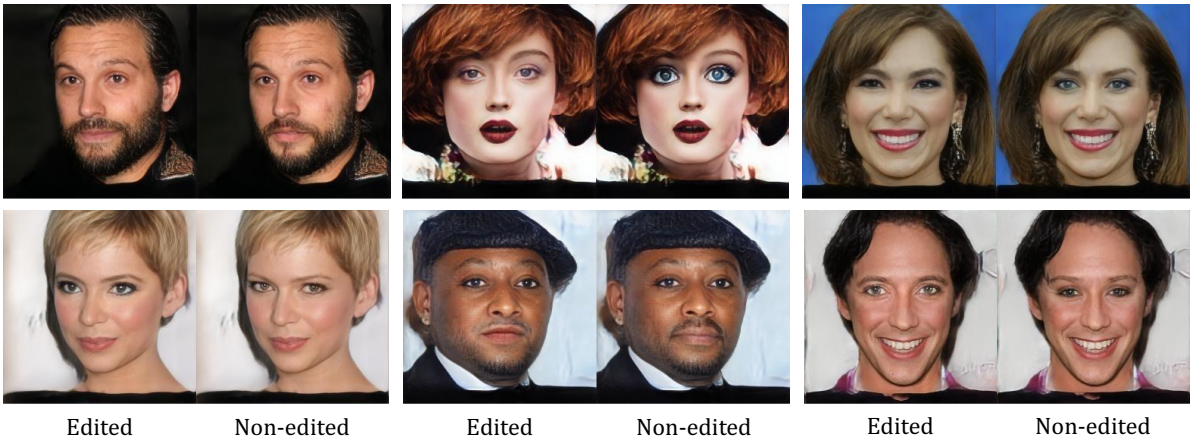

Figure 21. Samples for top 3 most incorrect (first row) and correct (second row) responses on the user study. Incorrect responses correspond to the user not being able to distinguish between edited and non-edited samples, and vice versa for the correct responses.