The Devil is in the Details: StyleFeatureEditor for Detail-Rich StyleGAN Inversion and High Quality Image Editing

Supplementary Material

7. Training details

The training of the StyleFeatureEditor consists of two phases: Phase 1 – training of the Inverter and Phase 2 – training of the Feature Editor. A batch size of 8 is used for both phases. We used Ranger optimiser with a learning rate of 0.0002 to train each part of our model, and Adam optimiser with a learning rate of 0.0001 to train the Discriminator.

Phase 1. During this phase, we duplicate the batch of source images X and synthesize the reconstruction $\widehat{X}$ and the reconstruction from w-latents only $\widehat{X}_w$ for the same images. The loss is computed for both pairs $(X, \widehat{X}), (X, \widehat{X}_w)$ and consists of L_2, LPIPS, ID, adversarial loss and regularization loss for predicted feature tensor F_k with corresponding coefficients λ_{loss}. We used $\lambda_{L_2} = 1, \lambda_{lpips} = 0.8, \lambda_{id} = 0.1$ for face domain and $\lambda_{id} = 0.5$ for car domain, $\lambda_{adv} = 0.01, \lambda_{reg} = 0.01$. We start applying adversarial loss and training the discriminator only after 14'000 steps. The full training duration of the first phase is 37'500 steps.

Phase 2. As in phase 1, the batch of source images X is duplicated and the same images are used for both inversion and editing loss. First, training samples X_E and X'_E are synthesized, then X_E is passed through StyleFeatureEditor which tries to reconstruct and edit it to $\widehat{X}'_E$, the editing loss is calculated between X'_E and $\widehat{X}'_E$. For inversion loss, the reconstruction $\widehat{X}$ is synthesized for the same images used for sampling (X_E, X'_E). The inversion loss is calculated between X and $\widehat{X}$. For both losses we use L_2, LPIPS and ID with the corresponding coefficients $\lambda_{L_2} = 1, \lambda_{lpips} = 0.8, \lambda_{id} = 0.1$ for face domain and $\lambda_{id} = 0.5$ for car domain. For inversion loss we additionally apply adversarial loss with $\lambda_{adv} = 0.01$. The duration of this phase is 20'000 steps.

During the second phase, we fix a set $\mathcal{D}$ of possible editing directions that we apply to compute the editing loss. $\mathcal{D}$ consists of InterfaceGAN[35] directions ("Age", "Smile", "Pose Rotation", "Glasses", "Make-up"), GANSpace[16] direction "Face Roundness", StyleClip[27] directions ("Afro", "Angry", "Bobcut Hairstyle", "Mohawk Hairstyle", "Purple Hair") and Stylespace[42] directions ("Blonde Hair", "Gender"). For car domain we used InterfaceGAN[35] ("Cube Shape", "Grass", "Colour Change") and Stylespace[42] ("Trees", "Headlights"). For each direction, we empirically choose several editing powers in such a way that E produces non-artefacting edits.

8. Architecture details

Our architecture has 2 parts: Inverter I and Feature Editor H. Inverter consists of Feature-Style-like encoder I_{fse} and Fuser I_{fus}. I_{fse} has been slightly changed compared to original version. The original Iresnet-50 backbone consists of 4 blocks (Figure 8 (a)), where each block increases the number of channels and reduces the spatial resolution of the input tensor. Each block consists of several layers, whose typical architecture is shown in Figure 7. As far as we increased k from 5 to 9 (which increases spatial resolution of predicted tensor from 16×16 to 64×64) it is necessary to extract features from the backbone with corresponding to the new spatial resolution. However, in the original Iresnet-50 architecture, such a tensor can only be gathered after block 2, which means that the original image is only passed through 3 + 4 = 7 layers, which is not enough to extract finer detail information. To fix this, we reduced stride in one of the layers in block 3, so that the resolution of the predicted tensor is changed as shown in Figure 8 (b), and the source image is processed through 21 layers.

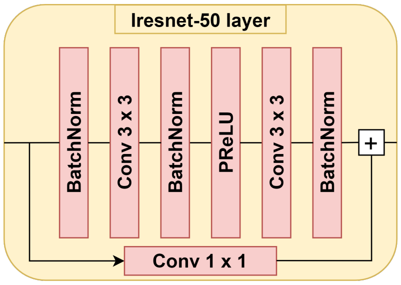

Figure 7. Scheme of the typical layer of Iresnet-50. H and I_{fus} also consist of such layers.

It is important to note that in the car domain, the information of some editings consists only in high-dimensional features with a spatial resolution of 128×128. To take this into account, during Δ computation, we synthesize outputs of the 11-th generator layer $F_{w_E}, F'_{w_E} \in \mathcal{F}_{11}$ instead of $\mathcal{F}_9$. To transform such Δ to size of $512 \times 64 \times 64$, we first apply an additional trainable Irensnet-50 layer, which reduces the resolution, and only then pass processed Δ to H.

I_{fus} and H have the same architecture. They both consist of 6 Iresnet-50 layers (Figure 7) with skip connections. During passing through I_{fus} or H spatial resolution of input tensor is not changed. When applying skip connections, we also use 1×1 convolution in case the number of feature map channels changes.

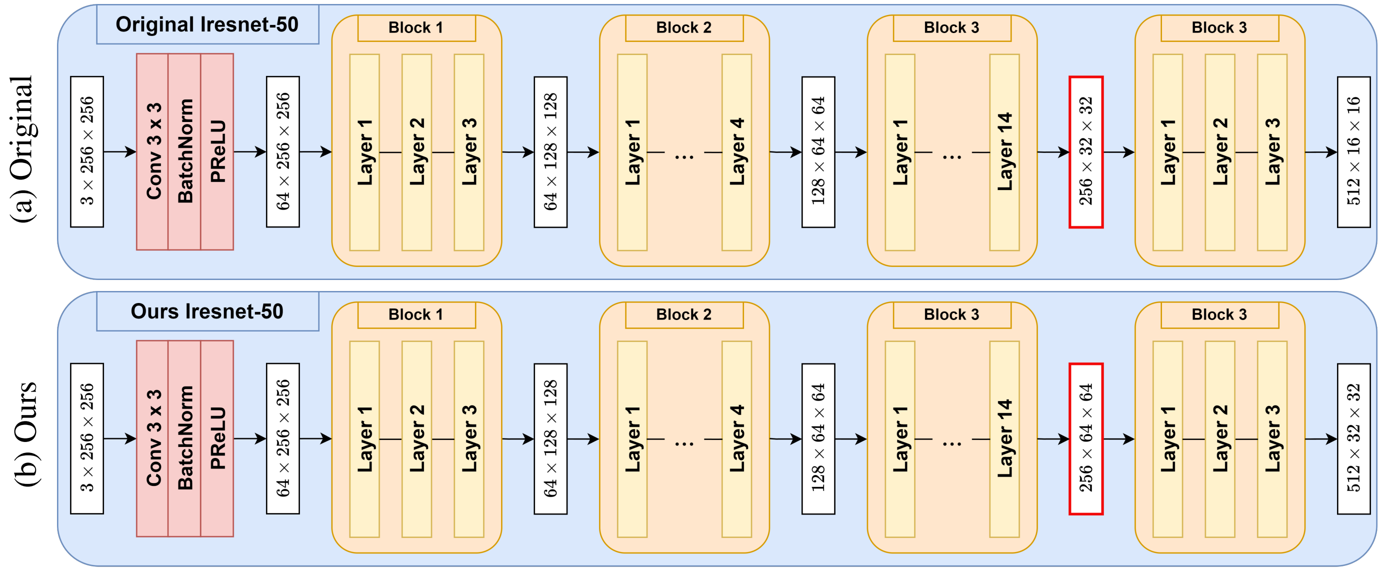

Figure 8. Architecture of Iresnet-50 backbone. Red-framed output is the one that is then passed to Feature predictor to predict F_{pred}.

Table 4. Additional editing results for StyleRes, SFE and checkpoint of SFE trained on a restricted set of editing directions $\mathcal{D}_{xmall}$ (see Ablation Study 4.4 and Appendix 10) on Celeba HQ. The technique used to calculate the editing metric is described in 4.3. *However, since Celeba HQ does not have a rotation attribute, we used a different technique for this direction. We randomly divided Celeba HQ into 2 equal parts, applied rotation to one of them and calculated the FID between them to evaluate the realism of the edited images.

Model	Glasses(+)	Rotation*(-)	Rotation*(+)	Bangs (+)	Beard(+)
StyleRes	**73.089**	26.004	27.492	45.497	77.084
$\mathcal{D}_{small}$	74.855	<u>25.429</u>	<u>26.371</u>	<u>40.006</u>	**76.168**
SFE	<u>73.098</u>	**23.541**	**24.084**	**39.319**	<u>77.529</u>

9. Masking

To edit images, StyleFeatureEditor uses editing information from the additional encoder E, which allows Inverter to focus only on reconstruction features. However, this is also a disadvantage of our method: if w_E would have artefacts during editing, Feature Editor will mostly inherit these artefacts. Therefore, it is important to choose E carefully. Unfortunately, there is a more general problem: some directions may not only change the attribute to which they refer, but also influence others.

Typically, 2 types of artefacts appear. First, while editing one attribute, another face attribute may be changed. For example, when adding glasses with a higher editing power, the mouth starts to open. The second type is that because E is only a w-latent encoder, it cannot reconstruct background well and make it smooth, so during editing, such background could also be affected (for example directions of bob cut and bowl cut hairstyles).

Our method is able to fix the second type of such artefacts. To do this, we propose to use an additional pre-trained model M, which is able to predict the face mask of the source image. The mask is scaled to a resolution of 64×64 and applied to Δ so that all features outside the face zone become zeros. As Δ preserves positional information, this means that the part of the image outside the face zone is not edited. The results of this approach are shown in Figure 11.

However, such a simple technique could lead to artefacts in cases where editing should be applied outside the face zone, such as pose rotation or afro hairstyle. Therefore, we left this technique as an optional feature.

10. Editings generaization

In this section we provide additional results from the Ablation Study checkpoint $\mathcal{D}_{small}$. This checkpoint was trained on a restricted set of editing directions $\mathcal{D}_{small}$ (see Ablation Study 4.4). In Figure 9 we compare this checkpoint with StyleRes and our main model (SFE) on directions not presented in $\mathcal{D}_{small}$. Both of our models outperform StyleRes, preserving more image detail and providing comparable editing, while the restricted and unrestricted checkpoints are only slightly different. This proves that our method is generalisable to any direction, even those not represented in the training set. The numerical results in Tab. 4 also confirm this.

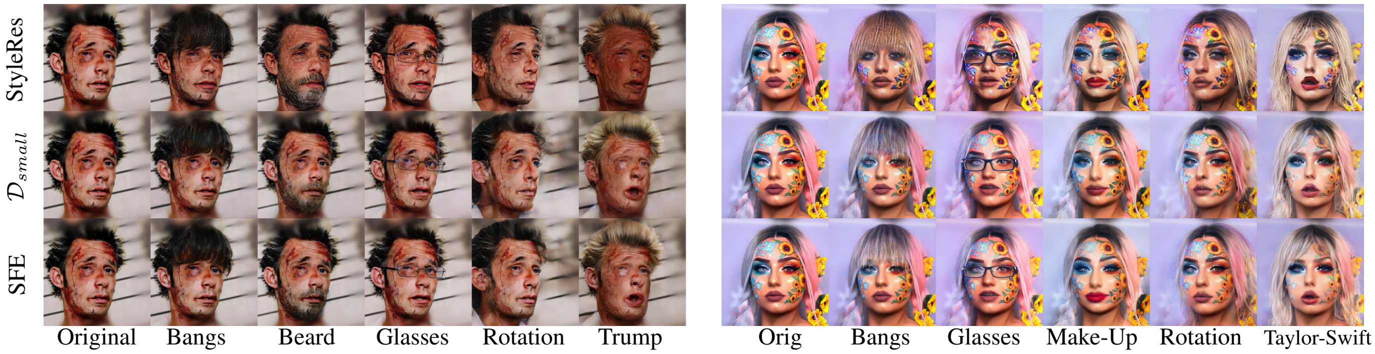

Figure 9. Additional editing results for SFE, StyleRes and SFE trained on stricted set of edits $\mathcal{D}_{small}$ (see question 1). Better zoom-in.

11. Additional results

In this section we provide additional visual examples of the StyleFeatureEditor. In Figure 10 we compare our method with StyleRes on out-of-domain MetaFaces dataset. Our method is able to preserve the original image style, while StyleRes makes it more realistic. In Figure 12 we show the work of our method in the face domain for several additional editing directions. In Figure 13 we present an additional comparison between StyleFeatureEditor and previous approaches in the face domain, and in Figure 14 we present more results for the car domain.

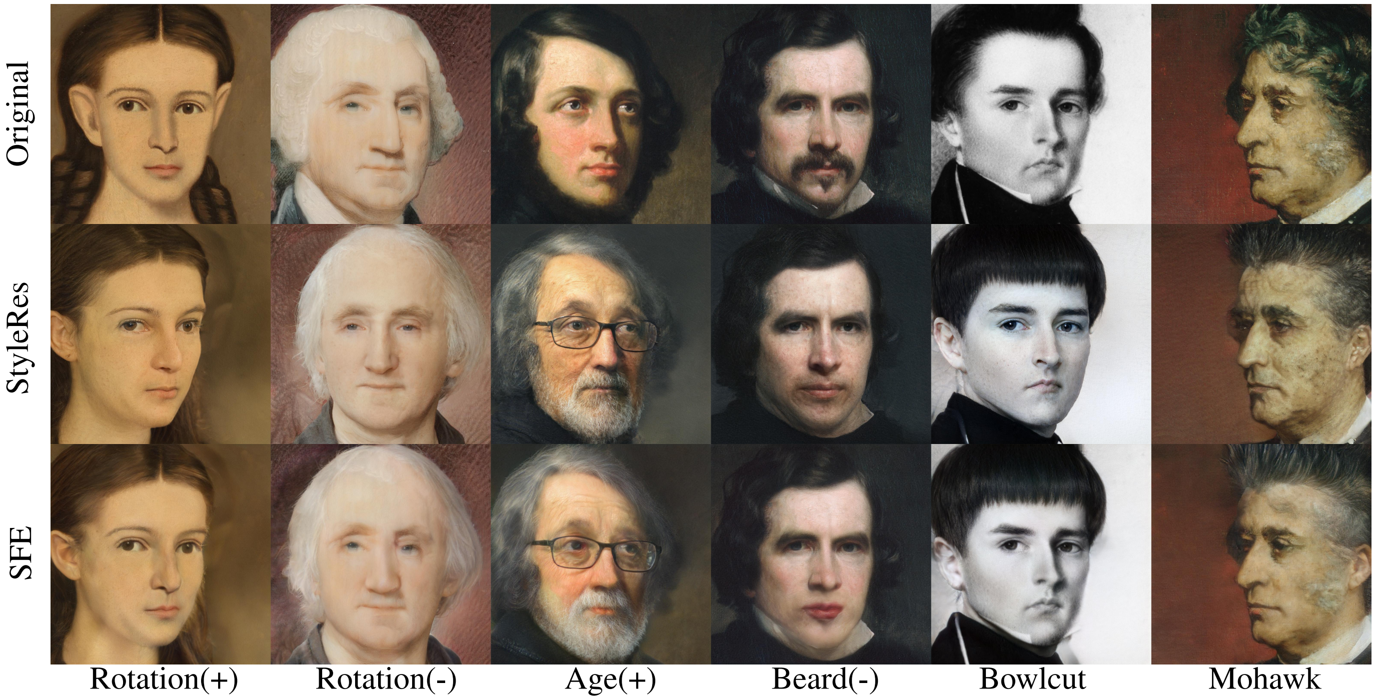

Figure 10. Results for SFE and StyleRes on MetaFaces. SFE preserves the original image style, while StyleRes makes it more photorealistic (see 5th and 6th columns).

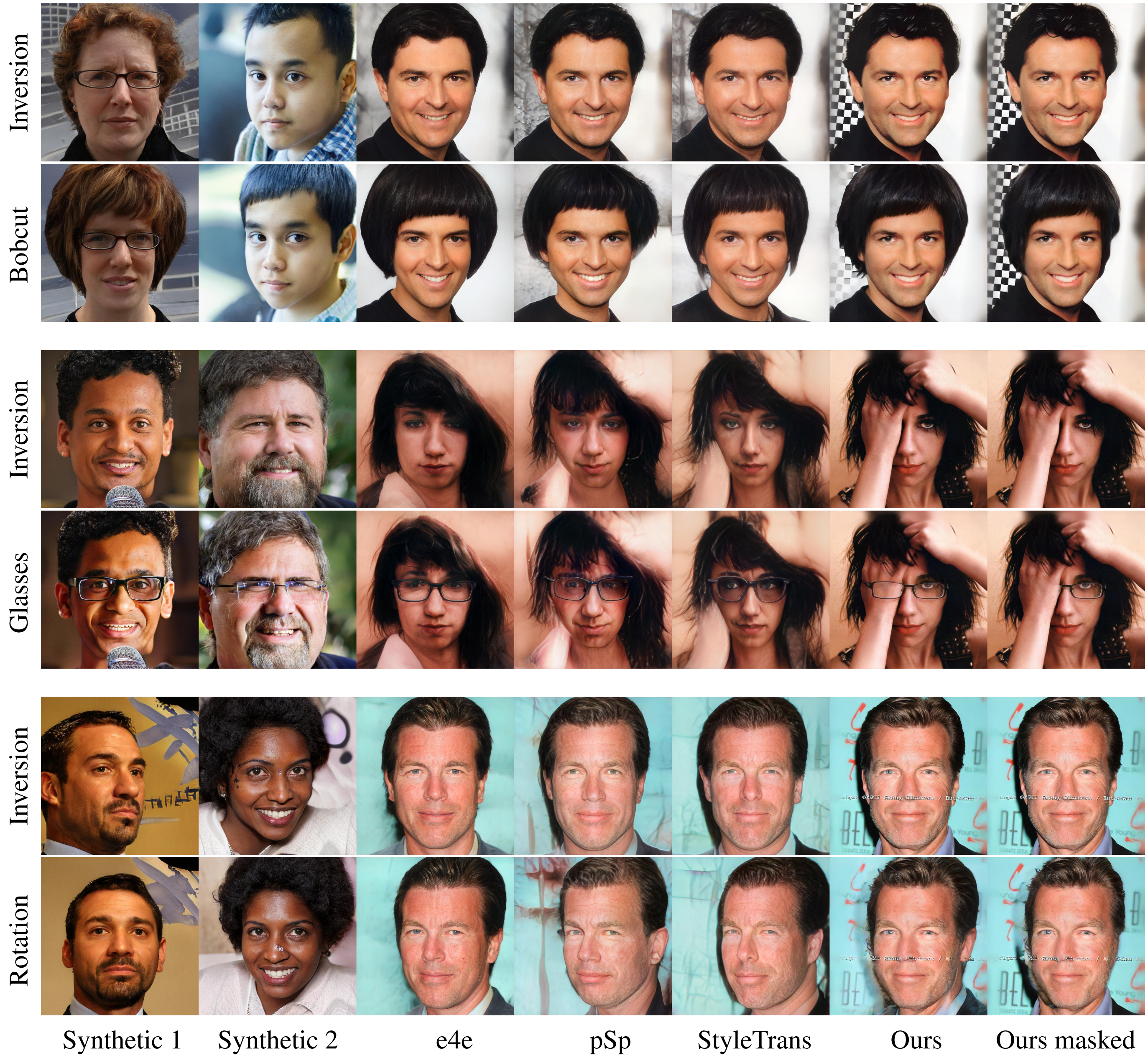

Figure 11. Examples of artefacts created by inaccurate editing directions. The first two columns represent synthetic images synthesized from w (Inversion) or its edited version w' (Corresponding Editing Direction), where w is obtained by randomly sampling z and passing it through the StyleGAN Mapping Network. Other columns represent inversion of real images (they are not represented here, but they are visually indistinguishable from the inversion of Ours method) by different encoders and its edited versions. During Bobcut editing, the background starts to disappear (even for synthetic images); during Glasses editing, the mouth starts to open. The masking technique (last column, for more details see Section 9) allows our method to avoid artefacts that appear during editing within the face zone (Bobcut, Glasses), but does not allow editing correctly while regions outside the face zone should be edited (Rotation).

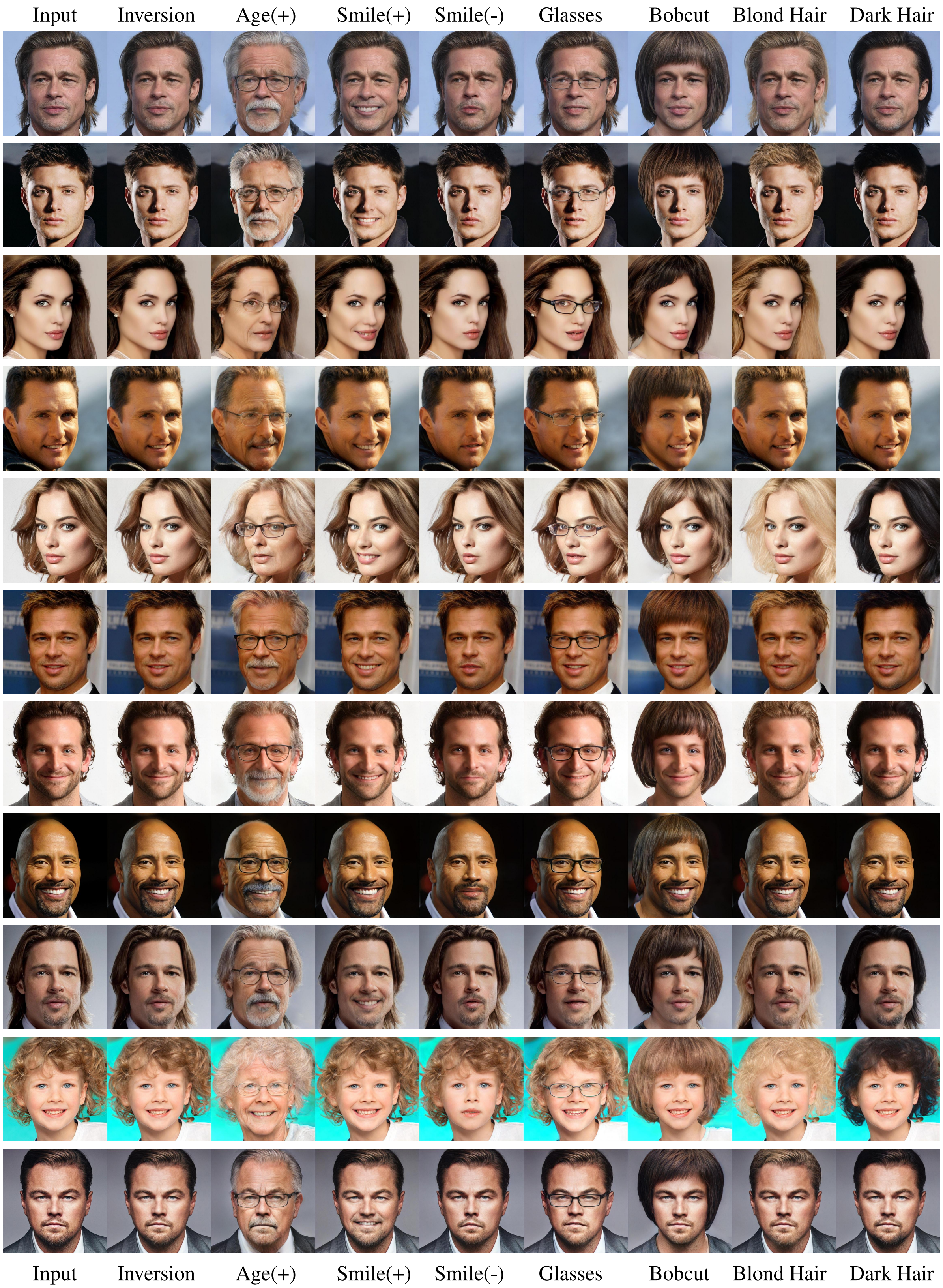

Figure 12. Additional visual example of StyleFeatureEditor in face domain.

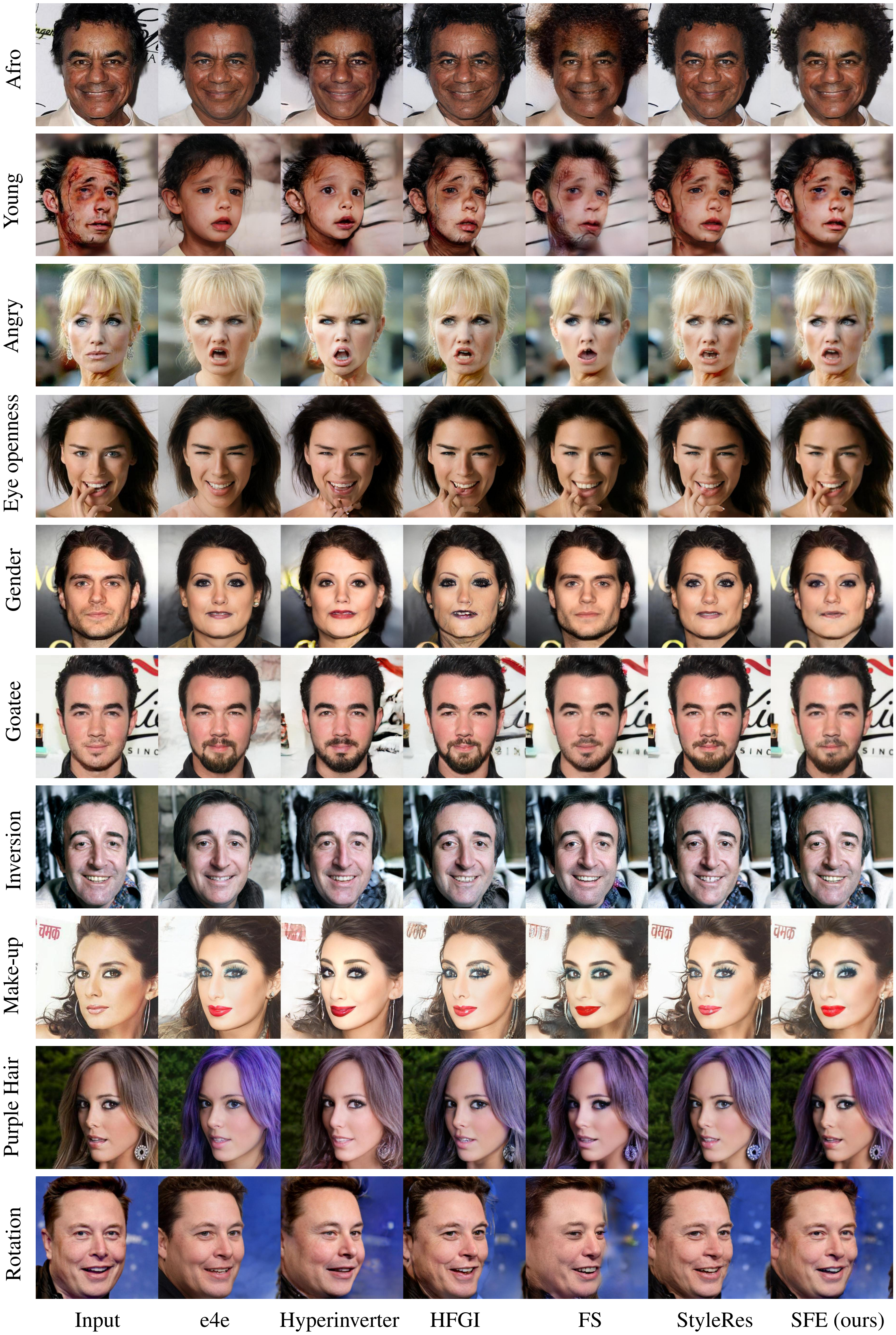

Figure 13. Additional visual comparison of FaetureStyleEditor with previous approaches in face domain.

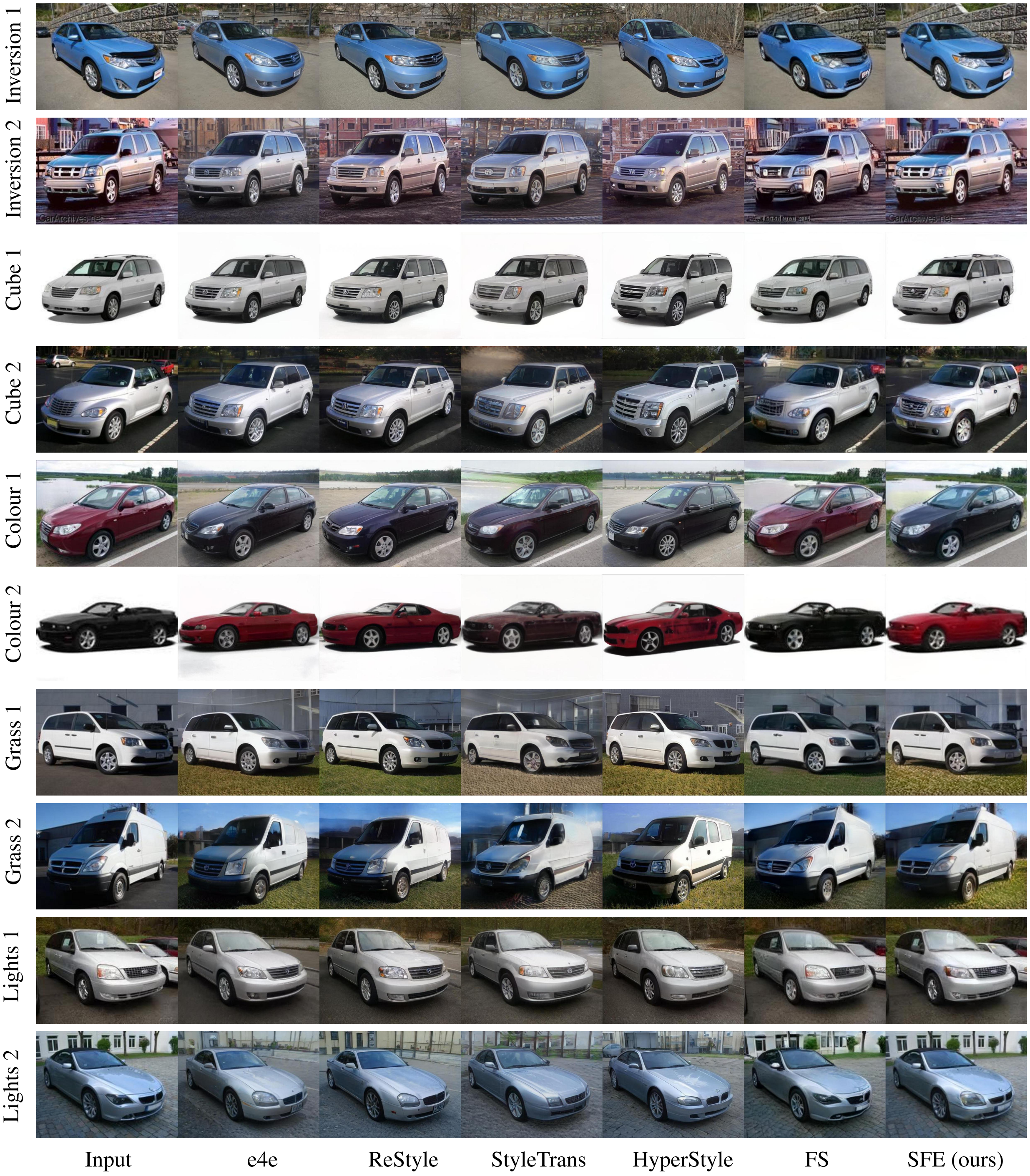

Figure 14. Additional visual comparison of StyleFaetureEditor with previous approaches in car domain.