Supplementary material
CoralStyleCLIP: Co-optimized Region and Layer Selection for Image Editing

A. Notation

$X \sim \mathcal{N}(\mu, \Sigma)$ denotes a Gaussian random vector X with mean μ and covariance Σ, and $\mathbf{I}$ denotes the identity matrix. Out of the four well known latent spaces of the StyleGAN2 [20], denoted by $\mathcal{Z}, \mathcal{W}, \mathcal{W}^+$ and $\mathcal{S}$ or the *StyleSpace*, CORAL extensively utilizes the $\mathcal{W}^+$ space. MLP abbreviates multi-layer perceptron in this paper.

For simplicity, $f^{(l)}$ is used to denote original features while $f^{*(l)}$ denotes edited features at each convolutional layer of StyleGAN2. Correspondingly, I denotes the original image and I^* denotes the edited image. $\langle \mathbf{x}, \mathbf{y} \rangle$ represents the dot product between vectors $\mathbf{x}$ and $\mathbf{y}$. $\|\mathbf{x}\|_2$ is the Euclidean norm of vector $\mathbf{x}$. For any score, $\uparrow$ is used to denote that a higher value is more desirable. The definition for $\downarrow$ follows along similar lines.

B. Pseudocode for Section 3

StyleGAN2 (Section 3.1). We will first present an overview of the StyleGAN2 [20] architecture which can be modularized into three parts.

1. Mapper $\underset{\mathcal{Z} \to \mathcal{W}^+}{\text{MLP}}(\cdot)$ from $z \in \mathbb{R}^d$ in $\mathcal{Z}$ space to $\mathbf{w} := [w^{(1)}, w^{(2)}, \ldots, w^{(18)}] \in \mathbb{R}^{18 \times d}$ in $\mathcal{W}^+$ space,

2. 18 learned convolutional blocks $\Phi^{(l)}$ where $l \in \{1, 2, \ldots, 18\}$, used as $f^{(l)} = \Phi^{(l)}(f^{(l-1)}, w^{(l)})$ emitting features $f^{(l)} \in \mathbb{R}^{H_l \times W_l \times d_l}$ as outputs. Here $H_l \times W_l$ is the resolution at which the features are generated at layer l. In addition, a fixed pre-trained tensor $f^{(0)} := c \in \mathbb{R}^{4 \times 4}$ is learned when training the StyleGAN2 backbone network on a dataset.

The progressive nature of StyleGAN2 architecture implies that $H_l \leq H_{l+1}$ and $W_l \leq W_{l+1}$. In our experiments, we also have $H_l = W_l$. Furthermore, layers with smaller l synthesize coarser attributes, while the latter layers are seen to control finer attributes, as evident in multiple figures in our paper.

3. RGB image constructed as $I = \sum_{l \in \widetilde{L}} RGB^{(l)}(f^{(l)})$ where $I \in \mathbb{R}^{H \times W \times 3}$ and $\widetilde{L} := \{2, 4, 6, \ldots, 18\}$.

Given the latent code $z \in \mathcal{N}(\mathbf{0}, \mathbf{I})$ for a particular image I, we can obtain the corresponding $w^{(l)}$ vectors by,

$$[w^{(1)}, w^{(2)}, \ldots, w^{(18)}] = \underset{\mathcal{Z} \to \mathcal{W}^+}{\text{MLP}}(z)$$

after which the forward pass of the StyleGAN2 generator is given by Algorithm 1.

Algorithm 1 StyleGAN2 forward pass

```
Input $\{w^{(l)}\}_{l=1}^{18} \in \mathcal{W}^+$ space
Output Generated image $I$, features $\{f^{(l)}\}_{l=1}^{18}$ at every layer
function FORWARD($\mathbf{w}$)
    Set $f^{(0)} = c$
    for $l \in \{1, 2, \ldots, 18\}$ do
        $f^{(l)} = \Phi^{(l)}(f^{(l-1)}, w^{(l)})$
        if $l \in \widetilde{L}$ then
            $I^{(l)} = RGB^{(l)}(f^{(l)})$
        end if
    end for
    $I = \sum_{l \in \widetilde{L}} I^{(l)}$
    return $I, \{f^{(l)}\}_{l=1}^{18}$
end function
```

Algorithm 2 StyleGAN2 blended forward pass

```
Input $\mathbf{w_1}, \mathbf{w_2} \in \mathcal{W}^+$, blending masks $m^{(l)} \in [0,1]^{H_l \times W_l} \forall\, l \in \{1, 2, \ldots, 18\}$
Output Generated image $I$
function BLENDEDFORWARD($\mathbf{w_1}, \mathbf{w_2}, \{m^{(l)}\}_{l=1}^{18}$)
    $\Delta^{(l)} = w_2^{(l)} - w_1^{(l)}$
    Set $f^{*(0)} = c$
    for $l \in \{1, 2, \ldots, 18\}$ do
        $\widehat{f^{*(l)}} = \Phi^{(l)}(f^{*(l-1)}, w_1^{(l)} + \Delta^{(l)})$
        $\widehat{f^{(l)}} = \Phi^{(l)}(f^{*(l-1)}, w_1^{(l)})$
        $f^{*(l)} = m^{(l)} \odot \widehat{f^{*(l)}} + (1 - m^{(l)}) \odot \widehat{f^{(l)}}$
        if $l \in \widetilde{L}$ then
            $I^{*(l)} = RGB^{(l)}(f^{*(l)})$
        end if
    end for
    $I^* = \sum_{l \in \widetilde{L}} I^{*(l)}$
    return $I^*$
end function
```

Multi-layer feedforwarded blending (Section 3.3). Alternatively, if instead we have $\mathbf{w_1} := \{w_1^{(l)}\}_{l=1}^{18} \in \mathcal{W}^+$,$\mathbf{w_2} := \{w_2^{(l)}\}_{l=1}^{18} \in \mathcal{W}^+$, obtained as

$$\mathbf{w_1} = \underset{\mathcal{Z} \to \mathcal{W}^+}{\text{MLP}}(z_1) \quad \mathbf{w_2} = \underset{\mathcal{Z} \to \mathcal{W}^+}{\text{MLP}}(z_2)$$

corresponding to two images, a blended forward pass can be performed as described in Algorithm 2. Note that the blended forward pass is a sophisticated non-linear spatial interpolation mechanism between the two images generated by z_1 and z_2.

CORAL (Section 3.2). Under the assumption that we can make use of two trainable modules SEGMENTSELECTOR$(\cdot)$ and CONVATTNNETWORK$(\cdot)$ described for segment-selection-based and attention network-based CORAL re-

Algorithm 3 CORAL based on segment-selection

```
    **Input** $z \in \mathcal{N}(\mathbf{0}, \mathbf{I})$ in $\mathcal{Z}$ space, text prompt $t$
    **Output** Loss $\mathcal{L}$
function Loss$(z, t)$
    $\mathbf{w} := \{w^{(l)}\}_{l=1}^{18} = \underset{\mathcal{Z}\to\mathcal{W}+}{\mathrm{MLP}}(z)$
    $\Delta := \{\Delta^{(l)}\}_{l=1}^{18} = g(\mathbf{w})$
    $\mathbf{w_1} = \mathbf{w}, \mathbf{w_2} = \mathbf{w} + \Delta$
    $I, \{f^{(l)}\}_{l=1}^{18} = \textsc{Forward}(\mathbf{w_1})$
    $\widetilde{I}, \{\widetilde{f}^{(l)}\}_{l=1}^{18} = \textsc{Forward}(\mathbf{w_2})$
    $e, \{m^{(l)}\}_{l=1}^{18} = \textsc{SegmentSelector}(I)$
    $I^* = \textsc{BlendedForward}(\mathbf{w_1}, \mathbf{w_2}, \{m^{(l)}\}_{l=1}^{18})$
    Loss $\mathcal{L} = \mathcal{L}_{ss}(I, I^*, \widetilde{I}, e, \Delta, t)$
    return $\mathcal{L}$
end function
```

Algorithm 4 CORAL based on attention network

```
    **Input** $z \in \mathcal{N}(\mathbf{0}, \mathbf{I})$ in $\mathcal{Z}$ space, text prompt $t$
    **Output** Loss $\mathcal{L}$
function Loss$(z, t)$
    $\mathbf{w} := \{w^{(l)}\}_{l=1}^{18} = \underset{\mathcal{Z}\to\mathcal{W}+}{\mathrm{MLP}}(z)$
    $\Delta := \{\Delta^{(l)}\}_{l=1}^{18} = g(\mathbf{w})$
    $\mathbf{w_1} = \mathbf{w}, \mathbf{w_2} = \mathbf{w} + \Delta$
    $I, \{f^{(l)}\}_{l=1}^{18} = \textsc{Forward}(\mathbf{w_1})$
    $\widetilde{I}, \{\widetilde{f}^{(l)}\}_{l=1}^{18} = \textsc{Forward}(\mathbf{w_2})$
    $\{m^{(l)}\}_{l=1}^{18} = \textsc{ConvAttnNetwork}(\{f^{(l)}\}_{l=1}^{18})$
    $I^* = \textsc{BlendedForward}(\mathbf{w_1}, \mathbf{w_2}, \{m^{(l)}\}_{l=1}^{18})$
    Loss $\mathcal{L} = \mathcal{L}_{can}(I, I^*, \widetilde{I}, \{m^{(l)}\}_{l=1}^{18}, \Delta, t)$
    return $\mathcal{L}$
end function
```

spectively in Section 3.2, we can compute the loss functions in both settings. In the case of segment selection, we use a pre-trained frozen segmentation network and a matrix e. The matrix $e \in [0, 1]^{P \times 18}$ is used to model the selection strength of each image segment, where P is the number of semantic segments predicted by the pre-trained segmentation network. In the case of the attention network, we employ a convolution network at each layer of StyleGAN2. More details about the network architecture are described in Appendix D.

The function $g(\mathbf{w})$ is used to represent the latent edit direction which could either be a *global direction* or an output from a non-linear *latent mapper*. Finally, the only parameters optimized for minimizing the loss $\mathcal{L}$ are e in segment selection, the parameters of the convolutional attention network for CORAL based on attention network, and finally, those of the latent edit predictor $g(\cdot)$ or Δ in both variants of CORAL.

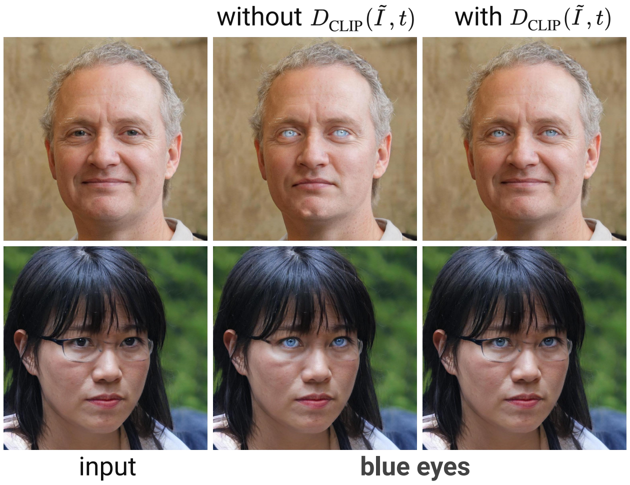
without $D_{\text{CLIP}}(\tilde{I}, t)$
with $D_{\text{CLIP}}(\tilde{I}, t)$
input
blue eyes

Figure 8 Results with and without additional CLIP loss

C. CLIP loss for semantic alignment

In Figure 8, we see that the additional $D_{\text{CLIP}}(\widetilde{I}, t)$ loss in (2) is essential for obtaining high-quality edits. In particular, for the text prompt *blue eyes*, we see that without this loss, there are unwanted white patches near the chin in the first row and second column, and the expression of the face is also affected. In the second row, the glasses are removed, and the complexion becomes fairer, thus affecting unrelated attributes.

With $D_{\text{CLIP}}(\widetilde{I}, t)$, however, the edits are more precise, and only the eye region is affected. This is happening because the image corresponding to $\mathbf{w_2}$ in Algorithm 2 is also explicitly driven to be semantically aligned with the text prompt t and therefore provides a better reference image for guiding the interpolation.

D. Architecture diagram

In Figure 9, we present the architecture diagram of CoralStyleCLIP with all components. In this paper, we demonstrated CORAL with four different variants. We demonstrated two approaches for predicting CORAL masks - segment selection and convolutional attention network. We also demonstrated results on two different variations of the latent direction - global direction Δ and latent mapper network $g(\cdot)$. Therefore, in total, we have four combinations of CORAL variants. For segment selection, we have a matrix e, which is used to modulate the weights of each image segment. In the case of the Convolutional Attention Network, we employ a CNN at each layer of the StyleGAN network. Architecture details of the network are mentioned in Section 4 of the main paper.

Latent Mapper: As also discussed in Section 3.4, the latent mapper $g(\cdot)$ is an MLP-based model along the lines of [36, Section 5], where the $w^{(l)}$ are split into three groups: coarse

Table 2 Metrics for Text Prompt - *Blue eyes*

	ID (↑)	LPIPS (↓)	MS-SSIM (↑)	L_2 (↓)
SS Global	0.776	0.0160	0.972	0.0029
SS Mapper	0.799	0.0082	0.969	0.0023
AttnNet-Global	0.868	0.0067	0.988	0.0015
AttnNet-Mapper	0.896	0.0060	0.989	0.0012
StyleCLIP	0.741	0.0856	0.871	0.0265
StyleMC	0.522	0.1670	0.596	0.2550
FEAT*	0.904	0.0017	0.997	0.0004

Table 3 Metrics for Text Prompt - *Happy*

	ID (↑)	LPIPS (↓)	MS-SSIM (↑)	L_2 (↓)
SS Global	0.633	0.0313	0.935	0.0080
SS Mapper	0.651	0.0308	0.933	0.0084
AttnNet-Global	0.830	0.0136	0.961	0.0050
AttnNet-Mapper	0.847	0.0155	0.959	0.0053
StyleCLIP	0.644	0.0904	0.835	0.0301
StyleMC	0.821	0.0244	0.940	0.0080
FEAT*	0.846	0.0150	0.957	0.0064

Table 4 Metrics for Text Prompt - *Mohawk hairstyle*

	ID (↑)	LPIPS (↓)	MS-SSIM (↑)	L_2 (↓)
SS Global	0.828	0.1980	0.756	0.0760
SS Mapper	0.882	0.0970	0.868	0.0303
AttnNet-Global	0.945	0.0849	0.919	0.0289
AttnNet-Mapper	0.922	0.0971	0.899	0.0337
StyleCLIP	0.522	0.2940	0.597	0.1530
StyleMC	0.651	0.0704	0.898	0.0180
FEAT*	0.953	0.0746	0.924	0.0236

Table 5 Metrics for Text Prompt - *Surprised*

	ID (↑)	LPIPS (↓)	MS-SSIM (↑)	L_2 (↓)
SS Global	0.747	0.0173	0.972	0.0037
SS Mapper	0.519	0.0412	0.902	0.0122
AttnNet-Global	0.819	0.0167	0.973	0.0035
AttnNet-Mapper	0.654	0.0297	0.927	0.0092
StyleCLIP	0.633	0.1390	0.765	0.0496
StyleMC	0.602	0.0952	0.728	0.0713
FEAT*	0.719	0.0252	0.938	0.0088

(l in 1 to 4), medium (l in 5 to 8) and fine (l in 9 to 18); and each of these groups is processed by a different MLP. Our latent mapper network consists of four BiEqual linear layers followed by a multi-layer perceptron. Each BiEqual layer consists of two MLPs followed by a LeakyReLU [49] activation function and a differencing operation (Figure 9).

E. Additional experiment details

We provide the hyperparameters used for training CoralStyleCLIP for making edits to images generated by a StyleGAN trained on the FFHQ dataset. Note that for both segment selection and attention network, the user can potentially decrease the λ_{id} by 20-40% depending on how likely the prompt is to make any edit to the facial region. This is essential for text prompts such as *kid*, *elderly*, and *asian*, where the transformation can alter the identity of a person significantly.

Segment selection: For experiments based on global directions (Segment Selection - Global Direction), we set $\lambda_{l_2} = 0.0007, \lambda_{id} = 0.015, \lambda_{area} = 0.10$, whereas for latent mapper based edits (Segment Selection - Mapper), we set $\lambda_{l_2} = 0.0002, \lambda_{id} = 0.020, \lambda_{area} = 0.08$.

Attention network: For experiments based on global directions (Attention Network - Global Direction) we set $\lambda_{l_2} = 0.0009, \lambda_{id} = 0.08, \lambda_{area} = 0.00009, \lambda_{tv} = 0.00003$, whereas for latent mapper based edits (Attention Network - Mapper), we set $\lambda_{l_2} = 0.0006, \lambda_{id} = 0.08, \lambda_{area} = 0.00002, \lambda_{tv} = 0.00003$

In Section F.2, Section F.3 and Section F.4, we also present results for attention network-based CORAL with *latent mapper* edits. For the Stanford cars dataset, λ_{l_2} is set to 0.0002 while keeping other hyperparameters the same. For the remaining two domains, which are adaptations of FFHQ, λ_{l_2} is set to 0.0004.

F. Additional results

In addition to presenting results for editing images using CoralStyleCLIP on the FFHQ dataset [19] (see Figure 10), we also demonstrate the benefits of our method for the Stanford Cars dataset [24] (see Figure 11), and for face generators which were adapted to the following domains: *sketch*, and *pixar* (see Figure 12 and Figure 13 respectively), using StyleGAN-NADA [13].

For experiments other than those with FFHQ, we disable the ID loss $\mathcal{L}_{id}$. Nonetheless, we do observe high-fidelity edits in these settings as well. Also note that as mentioned in Section 4, for $l > 13$, we set the masks $m^{(l)} = \mathbf{0}$ for CoralStyleCLIP.

F.1. FFHQ [19]

In Figure 10, we present more examples where CoralStyleCLIP executes a range of edits for human faces with high precision and minimal hand-holding. We choose a variety of text prompts that demand challenging structural and color edits. These results show that our method can accurately select the correct region and layer.

F.2. Cars [24]

We trained CoralStyleCLIP based on convolutional attention network and the *latent mapper* edit for the 512 × 512 size StyleGAN2 model trained on Stanford Cars dataset [24] available from [6]. For text prompts *classic*, *sports*, and *yellow car*, we observe that only the car is edited while the background is not selected in Figure 11. Furthermore, we also observe that CoralStyleCLIP automatically selects earlier layers for executing the first two prompts while it utilizes the latter layers to change the car's color. It is also interesting to note that the network generally selects layer 8 for wheel modification while it selects layers 5-7 for editing the car's body. This indicates that the car's wheels will likely have a more significant structural disentanglement and edit flexibility in layer 8. Furthermore, for *yellow*

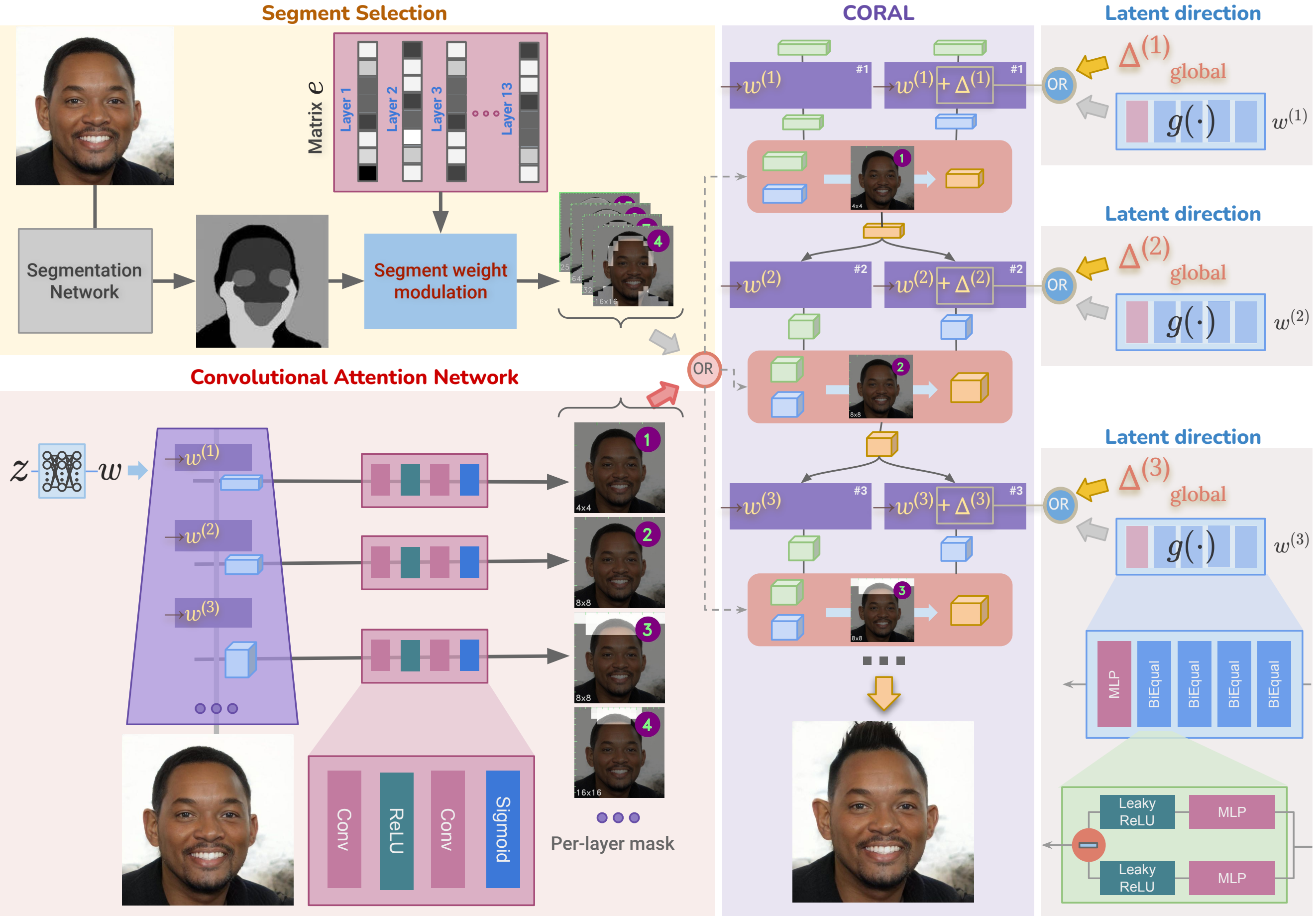

Figure 9 Architecture diagram of CoralStyleCLIP. **Segment selection network** consists of a pre-trained segmentation network and matrix e. The weights of each segment are modulated to produce a CORAL mask. **Convolutional attention network** consists of a CNN which predicts the CORAL mask at each layer of StyleGAN. The CORAL mask can either come from Segment selection *or* Convolutional Attention Network. **Latent direction** can either come from the learnt global direction Δ *or* via a mapper $g(\cdot)$ at each layer of CoralStyleCLIP. The layers of CoralStyleCLIP blend the features using the mask and latent direction (See Suppl. Pseudocode). There are three $g(\cdot)$ modules for coarse ($l \in [1, 4]$), medium ($l \in [5, 8]$) and fine layers ($l \in [9, 18]$) each. In this figure, the result of *mohawk hairstyle* used a convolutional attention network and global direction.

car, CORAL prioritizes the car's body over the wheels and windows. We see that the car's color remains preserved for *classic car* and *sports car*.

F.3. Sketch [13]

We trained CoralStyleCLIP based on convolutional attention network and the *latent mapper* edit for a StyleGAN that was pre-trained on FFHQ [19], and then domain adapted to *sketches* using StyleGAN-NADA [13]. For both prompts *kid* and *frown*, we see that CORAL successfully identifies the appropriate regions and layers for editing and executes them in Figure 12. In the case of *kid*, the network selects the facial region, and in *frown*, the network selects the eyes and mouth regions. Similar to the observation made in *Cars* subsection above (Section F.2), the network selects layer 8 for structural changes to the eyes and selects other layers for making facial edits to achieve *frown*.

F.4. Pixar [13]

Along the lines of Section F.3, in Figure 13, we also apply edits corresponding to *glasses* and *scared* for StyleGAN2 generated images adapted for the *pixar* domain using StyleGAN-NADA [13].

While the *glasses* emerge from layers 1 to 4, leaving other attributes undisturbed, *scared* affects both the eyes and mouth regions, their corresponding edits emerging from layers 5 and 8, respectively. In both cases, the edits are primarily structural and executed through the earlier layers. These results demonstrate that CoralStyleCLIP can determine the correct regions to edit at the correct set of StyleGAN2 layers with minimal hand-holding.

F.5. Additional quantitative results (Table 2 to 5)

In Table 1, we observed that CORAL edits on human faces achieve a reasonable Clean-FID [35], thereby preserv-

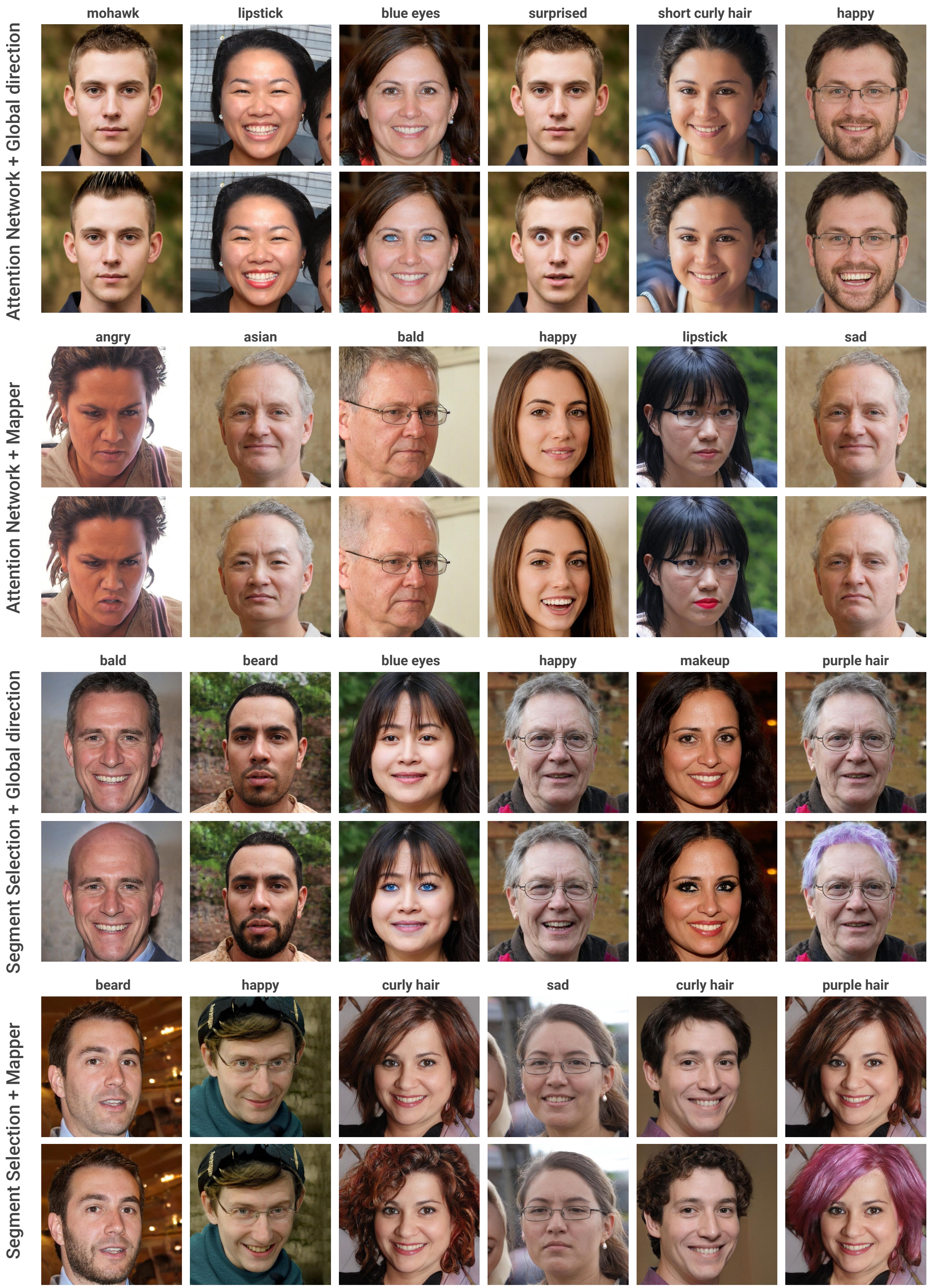

Figure 10 Additional results on FFHQ [19] generated images.

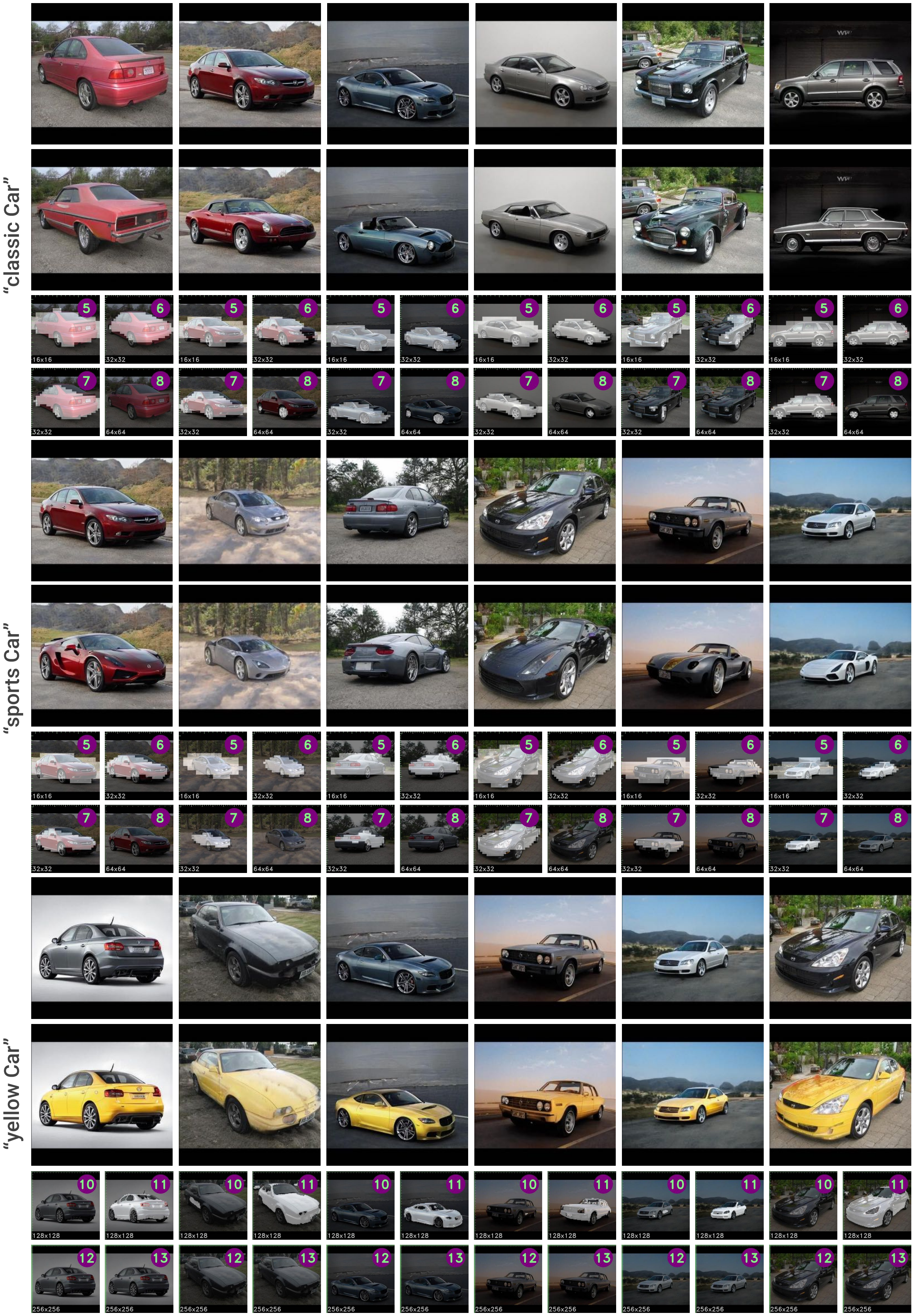

Figure 11 Additional results on Cars [24] generated images.

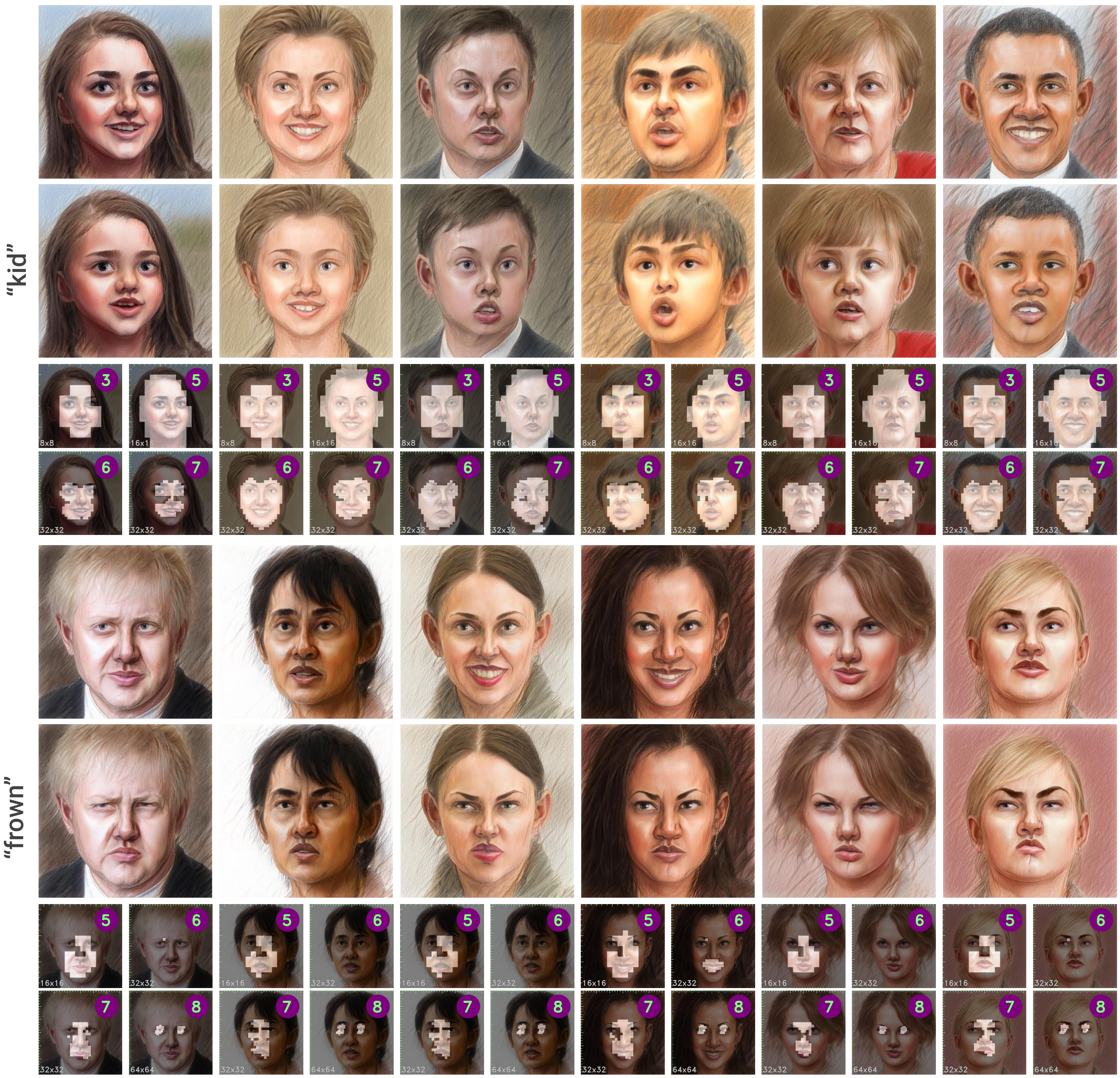
"kid"
3 8x8 5 16x16 3 8x8 5 16x16 3 8x8 5 16x16 3 8x8 5 16x16 3 8x8 5 16x16 3 8x8 5 16x16
6 32x32 7 32x32 6 32x32 7 32x32 6 32x32 7 32x32 6 32x32 7 32x32 6 32x32 7 32x32 6 32x32 7 32x32
"frown"
5 16x16 6 32x32 5 16x16 6 32x32 5 16x16 6 32x32 5 16x16 6 32x32 5 16x16 6 32x32 5 16x16 6 32x32
7 32x32 8 64x64 7 32x32 8 64x64 7 32x32 8 64x64 7 32x32 8 64x64 7 32x32 8 64x64 7 32x32 8 64x64

Figure 12 Additional results on Sketch [13] generated images.

ing the *realism* of the edited images.

We also compare CORAL for a StyleGAN2 trained on FFHQ with our baselines across the four text prompts from Table 1 based on other quantitative metrics. In particular, we compute the identity similarity (ID) based on the cosine similarity between ArcFace embeddings [11], LPIPS [53] distance, MS-SSIM [46] score and the pixel-wise Euclidean distance, between each edited image and the original.

For comparison with FEAT [16], we only compare CORAL with results from our reimplementation FEAT$^*_7$ for *happy*, *mohawk* and *surprised*, and FEAT$^*_{13}$ for *blue eyes* so that the edits do occur with high precision. On average, the ID is higher in Table 2 and Table 4 as compared to Table 3 and Table 5. This is because edits to the facial regions, such as the mouth and eyes, diminish the facial recognition capabilities of ArcFace [11].

In general, CORAL, based on the attention network, has higher similarity and lower dissimilarity scores than the segment-selection-based methods, which can be attributed to higher precision in the region of interest selection at ev-

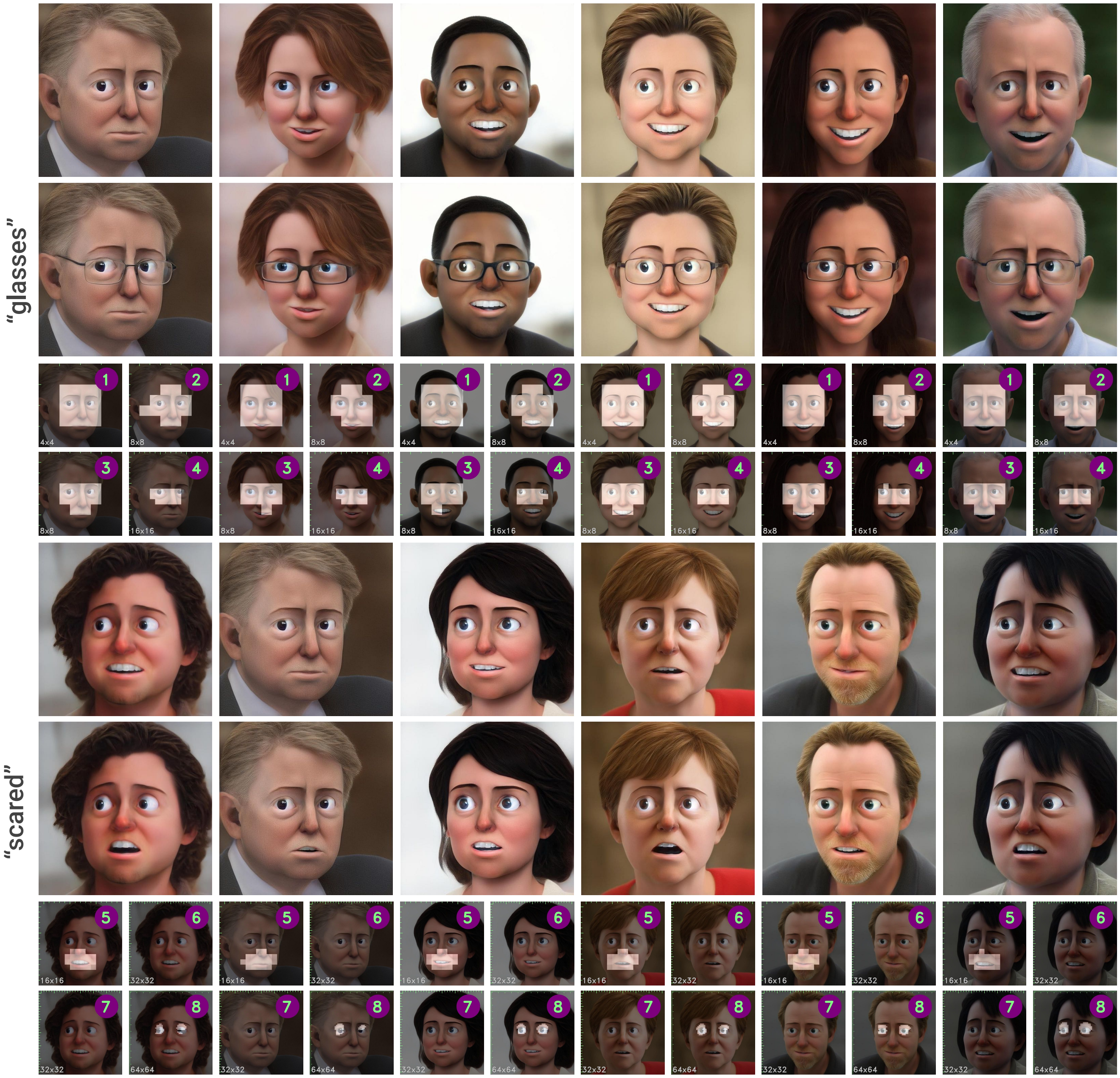

Figure 13 Additional results on Pixar [13] generated images.

ery layer of the generator. However, the same cannot be said for StyleCLIP and StyleMC, which affect irrelevant regions, consequently underperforming CORAL and FEAT.

G. CORAL v/s multi-layer FEAT

CORAL method applies the proposed feedforwarded feature blending strategy at each of the StyleGAN layers (Section 3.3). On similar lines, an interesting and a direct extension of FEAT would be to apply their proposed feature blending strategy at every StyleGAN layer. The resulting method, which we refer to as "multi-layer FEAT", would have a similar high level architecture with two parallel pathways but would differ in the blending strategies (Figure 14). In the case of the multi-layer FEAT, the blended features would be propagated through the edited pathway ($w + \Delta$) while the original features would continue to propagate through the original unedited (w) pathway. In other words, the FEAT blending strategy draws relevant edits from a particular layer and learns to discard edits from all the previous layers. Therefore, multi-layer FEAT cannot propagate the edits effectively. However, in the case of CORAL, blended features are passed parallelly to both

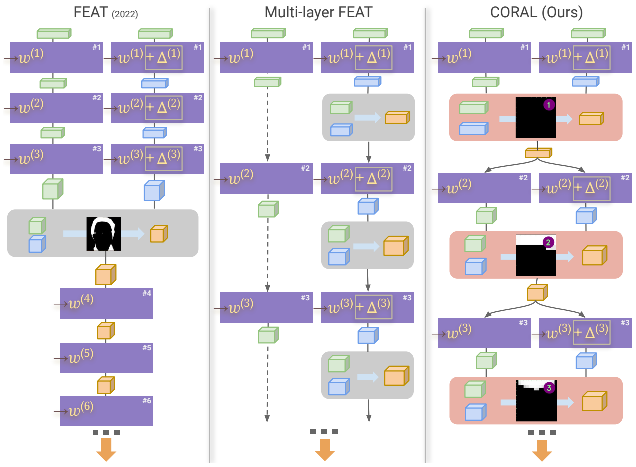

Figure 14 FEAT vs Multi-layer FEAT vs CORAL

pathways (Section 3.3). As a result, CORAL can discard irrelevant edits at the current layer and propagate the updated edits effectively.

H. Future applications

The edits outside the regions of interest are explicitly prevented by the feed-forwarded blending strategy proposed in Section 3.3. As a result, methods such as [4, 16, 17, 23, 27, 36] could directly benefit from applying our strategy for inference with separately constructed masks.

Furthermore, the synergy between the co-optimized region and layer selection (CORAL) in Section 3.2 and the feed-forwarded blending may provide increased control to content designers who may find it easier to refine an accurate initial predicted region of interest.

References

[1] Yuval A., Or Patashnik, and Daniel Cohen-Or. Only a matter of style: age transformation using a style-based regression model. *ACM Trans. Graph.*, 40(4):45:1–45:12, 2021.

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *ICCV*, 2019.

[3] Rameen Abdal, Peihao Zhu, John Femiani, Niloy J. Mitra, and Peter Wonka. Clip2stylegan: Unsupervised extraction of stylegan edit directions. In *SIGGRAPH*. ACM, 2022.

[4] Rameen Abdal, Peihao Zhu, Niloy J. M., and Peter Wonka. Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. *ACM Trans. Graph.*, 40(3):21:1–21:21, 2021.

[5] Yuval Alaluf, Or Patashnik, Zongze Wu, Asif Zamir, Eli Shechtman, Dani Lischinski, and Daniel Cohen-Or. Third time's the charm? image and video editing with stylegan3. *ECCVw*, 2022.

[6] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit Bermano. Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *CVPR*, 2022.

[7] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *CVPR*, 2022.

[8] Amit H. B., Rinon Gal, Yuval Alaluf, Ron Mokady, Yotam Nitzan, Omer Tov, Or Patashnik, and Daniel Cohen-Or. State-of-the-art in the architecture, methods and applications of stylegan. *Comp. Graph. Forum*, 41(2):591–611, 2022.

[9] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*, 2019.

[10] Edo Collins, Raja Bala, Bob Price, and Sabine Süsstrunk. Editing in style: Uncovering the local semantics of GANs. *CVPR*, 2020.

[11] Jiankang Deng, Jia Guo, Jing Yang, Niannan Xue, Irene Kotsia, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. *IEEE TPAMI*, 2022.

[12] H. Dong, Simiao Yu, Chao Wu, and Y. Guo. Semantic image synthesis via adversarial learning. *ICCV*, pages 5707–5715, 2017.

[13] Rinon Gal, Or Patashnik, Haggai Maron, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Trans. Graph.*, 41(4):141:1–141:13, 2022.

[14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi M., Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C., and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

[15] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable GAN controls. In *NeurIPS*, 2020.

[16] Xianxu Hou, Linlin Shen, Or Patashnik, Daniel Cohen-Or, and Hui Huang. FEAT: face editing with attention. *CoRR*, abs/2202.02713, 2022.

[17] Omer Kafri, Or Patashnik, Yuval A., and Daniel Cohen-Or. Stylefusion: Disentangling spatial segments in stylegan-generated images. *ACM Trans. Graph.*, Mar 2022.

[18] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *ICLR*, 2018.

[19] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *IEEE TPAMI*, 43(12):4217–4228, 2021.

[20] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *CVPR*, 2020.

[21] Hyunsu Kim, Yunjey Choi, Junho Kim, Sungjoo Yoo, and Youngjung Uh. Exploiting spatial dimensions of latent in GAN for real-time image editing. In *CVPR*, 2021.

[22] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *ICLR*, 2015.

[23] Umut Kocasari, Alara Dirik, Mert Tiftikci, and Pinar Yanardag. Stylemc: Multi-channel based fast text-guided image generation and manipulation. In *WACV*, 2022.

[24] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *2013 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2013, Sydney, Australia, December 1-8, 2013*, pages 554–561. IEEE Computer Society, 2013.

[25] Seung H. L., Won Kyoung Roh, Wonmin Byeon, Sang Ho Yoon, Chan Y. K., Jinkyu Kim, and Sangpil Kim. Sound-guided semantic image manipulation. *CVPR*, 2022.

[26] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip H. S. Torr. Manigan: Text-guided image manipulation. In *CVPR*, 2020.

[27] Huan Ling, Karsten Kreis, Daiqing Li, Seung Wook Kim, Antonio Torralba, and Sanja Fidler. Editgan: High-precision semantic image editing. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 16331–16345, 2021.

[28] Yahui Liu, Marco De Nadai, Deng Cai, Huayang Li, Xavier Alameda-Pineda, N. Sebe, and Bruno Lepri. Describe what to change: A text-guided unsupervised image-to-image translation approach. *ACMMM*, 2020.

[29] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In Johannes Fürnkranz and Thorsten Joachims, editors, *ICML*, 2010.

[30] Seonghyeon Nam, Yunji Kim, and S. Kim. Text-adaptive generative adversarial networks: Manipulating images with natural language. In *NeurIPS*, 2018.

[31] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*, 2022.

[32] Weili Nie, Tero Karras, Animesh Garg, Shoubhik Debnath, Anjul Patney, Ankit B. Patel, and Animashree Anandkumar. Semi-supervised stylegan for disentanglement learning. In *ICML*, 2020.

[33] Daniil Pakhomov, Sanchit Hira, Narayani Wagle, Kemar E. Green, and Nassir Navab. Segmentation in style: Unsupervised semantic image segmentation with stylegan and CLIP. *CoRR*, abs/2107.12518, 2021.

[34] Gaurav Parmar, Yijun Li, Jingwan Lu, Richard Zhang, Jun-Yan Zhu, and Krishna Kumar Singh. Spatially-adaptive multilayer selection for GAN inversion and editing. In *CVPR*, 2022.

[35] Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in GAN evaluation. In *CVPR*, 2022.

[36] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *ICCV*, 2021.

[37] Justin N. M. Pinkney and Chuan Li. clip2latent: Text driven sampling of a pre-trained stylegan using denoising diffusion and CLIP. *BMVC*, 2022.

[38] Alec Radford, Jong W. K., Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[39] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *CoRR*, abs/2204.06125, 2022.

[40] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021.

[41] S. Reed, Zeynep Akata, Xinchen Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *ICML*, 2016.

[42] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: A stylegan encoder for image-to-image translation. In *CVPR*, 2021.

[43] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *CVPR*, 2020.

[44] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *CVPR*, 2021.

[45] Ayush Tewari, Mohamed Elgharib, Gaurav Bharaj, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhöfer, and Christian Theobalt. StyleRig: Rigging StyleGAN for 3d control over portrait images. *CVPR*, 2020.

[46] Z. Wang, E.P. Simoncelli, and A.C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402 Vol.2, 2003.

[47] Zongze Wu, Dani Lischinski, and Eli Shechtman. Stylespace analysis: Disentangled controls for stylegan image generation. In *CVPR*, 2021.

[48] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. TediGAN: Text-guided diverse face image generation and manipulation. *CVPR*, 2021.

[49] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *CoRR*, abs/1505.00853, 2015.

[50] T. Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and X. He. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. *CVPR*, 2018.

[51] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017.

[52] Han Zhang, T. Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N. Metaxas. StackGAN++: Realistic image synthesis with stacked generative adversarial networks. *IEEE TPAMI*, 41:1947–1962, 2019.

[53] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 586–595. Computer Vision Foundation / IEEE Computer Society, 2018.

[54] Yuxuan Zhang, Huan Ling, Jun Gao, Kangxue Yin, Jean-Francois Lafleche, Adela Barriuso, Antonio Torralba, and Sanja Fidler. Datasetgan: Efficient labeled data factory with minimal human effort. In *CVPR*, 2021.

[55] Yufan Zhou, Ruiyi Zhang, Jiuxiang Gu, Chris Tensmeyer, Tong Yu, Changyou Chen, Jinhui Xu, and Tong Sun. Tigan: Text-based interactive image generation and manipulation. In *AAAI*, 2022.