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

Ambareesh Revanur∗ Debraj Basu Shradha Agrawal Dhwanit Agarwal Deepak Pai
Adobe
∗arevanur@adobe.com

Figure 1 The original and edited images using CoralStyleCLIP are shown in the first and second rows of images, respectively. The bottom row shows the regions and StyleGAN2 layer numbers automatically selected for editing. The driving text prompts are above every column.

Abstract

Edit fidelity is a significant issue in open-world controllable generative image editing. Recently, CLIP-based approaches have traded off simplicity to alleviate these problems by introducing spatial attention in a handpicked layer of a StyleGAN. In this paper, we propose CoralStyleCLIP, which incorporates a multi-layer attention-guided blending strategy in the feature space of StyleGAN2 for obtaining high-fidelity edits. We propose multiple forms of our co-optimized region and layer selection strategy to demonstrate the variation of time complexity with the quality of edits over different architectural intricacies while preserving simplicity. We conduct extensive experimental analysis and benchmark our method against state-of-the-art CLIP-based methods. Our findings suggest that CoralStyleCLIP results in high-quality edits while preserving the ease of use.

1. Introduction

Controlling smooth semantic edits to photorealistic images [1, 5, 34, 41] synthesized by well-known Generative Adversarial Networks (GANs) [13, 18, 19] has become simplified with guidance from independently trained contrastive models such as CLIP [36]. Using natural language as a rich medium of instruction for open-world image synthesis [39, 46–48] and editing [11, 12, 24, 26, 28, 45] has addressed many drawbacks of previously proposed methods.

As first demonstrated by StyleCLIP [34], the requirements for large amounts of annotated data [25] and manual efforts [14, 44] were considerably alleviated. Furthermore, the range of possible edits that were achievable significantly improved [34]. The underlying theme of related approaches involves CLIP-driven exploration [15,22,34] of the intermediate disentangled latent spaces of the GANs.
It is well understood by now that manipulating the latent code of a StyleGAN for aligning with a text prompt can be computationally intense, as seen in StyleCLIP latent optimization [34], as well as the latent mapper methods [34]. This presents a trade-off between the complexity and quality of edits leveraged by StyleCLIP global directions [34] and StyleMC [22].

In addition, these methods often result in undesirable edits to unexpected regions of an image (see [15]), addressed to some extent by FEAT [15]. However, FEAT requires manual intervention, as described in Section 2, and involves significant training complexity of the order of hours\footnote{With no official implementation available, we present comparisons with our reimplementation of FEAT denoted by FEAT$^*$ in this paper.}. Contributions. In this paper, we propose CoralStyleCLIP, which addresses these challenges by combining the ease of use [34] with efficient [22] high fidelity edits [15] into our approach. In particular, we propose a novel strategy which, for a given text prompt, jointly learns both the appropriate direction of traversal in the latent space, as well as which spatial regions to edit in every layer of the StyleGAN2 [19] (see Figure 1, Figure 2) without any mediation.

Our approach overcomes the need for manual effort in selecting an appropriate layer for FEAT by incorporating multi-layer feature blending to enable the joint learning process. As a result, the edits are very accurate, rendering our method simple and effective.

The co-optimized regions and layers jointly learned with appropriate latent edits typically select earlier layers for enacting coarse edits, such as shape and structural, compared to finer edits, such as color and texture, which are usually orchestrated through the latter layers of the StyleGAN2.

To alleviate the time complexity, we implement this strategy for segment selection (see Section 3.2), where we jointly learn a global direction [22,34] in the $\mathcal{W}^+$ space and limit the predicted areas of interest at every layer to segments from a pre-trained segmentation network. Doing so reduces the learning complexity significantly (see Table 1), albeit with potential pitfalls discussed in Section 4.3. We mitigate these pitfalls with a jointly trained attention network where we relax the areas of interest at every layer to spatial masks predicted by the network (see Section 3.2). As a result, the training time increases from a few minutes to about an hour while improving the quality of edits compared to the segment selection approach.

In summary, our contributions are as follows:

- We propose a novel multi-layer blending strategy that attends to features selectively at the appropriate StyleGAN layer with minimal hand-holding.

- A CORAL variant based on segment selection demonstrates high edit quality at a fraction of time cost.

- Through extensive empirical analysis, we find that CORAL outperforms recent state-of-the-art methods and is better equipped to handle complex text prompts.

2. Related Work

The use of generative models for high-quality image synthesis and manipulation has a rich history [7, 8, 17]. In particular, the disentangled latent spaces of StyleGAN provide robust interpretable controls for editing valuable semantic attributes of an image [3, 9, 14, 30, 41–44]. Desirable changes to attributes of interest were previously brought out by discovering the relevant channels [44] and curating principal components [14] either through manual inspection or otherwise driven by data-hungry attribute predictors.

StyleFlow [4] leverages normalizing flows to perform conditional exploration of a pre-trained StyleGAN for attribute-conditioned image sampling and editing. By learning to encode the rich local semantics of images into multi-dimensional latent spaces with spatial dimension, StyleMapGAN [20] demonstrates improved inversion quality and the benefits of spatially aware latent code interpolation between source and target images for editing purposes. The advent of CLIP [36] has re-ignited interest in open domain attribute-conditioned synthesis of images [34, 35, 50]. Text-driven edits have considerably reduced both the time and effort required for editing images and extended the range of possible edits significantly [34], all the more with increased interest in diffusion models [29, 37, 38].

The disentangled nature of the latent spaces of StyleGAN has facilitated heuristics such as a fixed global direction in StyleCLIP [34] and, more recently, StyleMC [22]. For training efficiency, StyleMC performs CLIP-driven optimization on the image generated at a low-resolution layer.
of the StyleGAN. Unfortunately, this limits the range of edits to only those possible by manipulating latent codes at the earlier layers.

For ameliorating edits in unexpected regions of an image, strategies for blending latent features have been an emerging theme in many recent papers [6, 15, 16, 20]. [6, 15, 20] interpolate spatial features more explicitly. In contrast, StyleFusion [16] realizes similar objectives through blended latent code extracted using a fusion network that combines disjoint semantic attributes from multiple images.

Our work is most closely related to [15, 22, 34]. FEAT [15] reduces undesirable edits by imposing sparsity in the number of spatial features modified by StyleCLIP at a manually selected layer \( l \) of the StyleGAN2. FEAT edits layers \( \leq l \) using a non-linear latent mapper, while the attention mechanism (see Section 3). The network consists of a mapper module comprising 18 convolutional blocks. Generating high-resolution images typically of sizes \( 1024 \times 1024 \) or \( 512 \times 512 \). The network consists of a mapper module that maps a random vector \( z \sim N(0, 1) \) to a vector in \( w \in \mathcal{W} \) space via a multi-layer perceptron (MLP), and a generator module comprising 18 convolutional blocks.

Furthermore, we argue that the required edits for aligning with a given text prompt arise from multiple layers of the StyleGAN2, necessitating a multi-layer feature interpolation mechanism (see Section 3.3). Our method percolates meaningful edits from the current layer onto subsequent layers, with restrictions on the number of spatial edits customized for each layer. As a result, we can automatically select the correct layers and regions for editing an image.

To correctly identify the region of interest at every layer, we discuss a lightweight segment-selection scheme (see Section 3.2) and contrast this with an involved convolutional mask prediction model motivated by FEAT. Recently, SAM [32] accomplished superior GAN inversion at the cost of editability by leveraging different latent spaces of the StyleGAN2 in a spatially adaptive manner. However, the edits performed on the inverted latent codes continue to modify irrelevant image regions and could benefit from CORAL (see Section 3).

With a focus on convenience and fidelity, CoralStyleCLIP learns global directions at every layer of the StyleGAN2, as done in [22], and exhibits high-quality edits with a significant reduction in the training time and manual effort (see Table 1). Borrowing inspiration from [34], we also implement our co-optimized region and layer selection strategies for a non-linear mapper-based latent edit and demonstrate additional customized and high fidelity edits.

### 3. Approach

An image edit is often spatially localized to a specific region of interest. For example, edits corresponding to the *mohawk* text prompt should affect only the hair region of the portrait image while preventing edits in other parts. In this work, we learn a latent edit vector and a soft binary mask at every layer of a StyleGAN2 to accurately edit the image according to the input text prompt. We achieve this by training them end-to-end while respecting the challenging but desirable minimal overall edit area constraint. Following a brief revisit to the StyleGAN architecture, we introduce two simple yet effective strategies to determine the region of interest given a text prompt. Finally, we introduce a novel multi-layer blending strategy that is vital for achieving high fidelity minimal edits.

#### 3.1. Background

StyleGAN2 [19] is a state-of-the-art model trained for generating high-resolution images typically of sizes \( 1024 \times 1024 \) or \( 512 \times 512 \). The network consists of a mapper module that maps a random vector \( z \sim N(0, 1) \) to a vector in \( w \in \mathcal{W} \) space via a multi-layer perceptron (MLP), and a generator module comprising 18 convolutional blocks.
The $\mathcal{W}^+$ space, first defined by [2], is a concatenation of 18 different $w_l^{(i)}$ vectors where $l \in \{1, 2, \ldots, 18\}$. The $w_l^{(i)}$ instance in $\mathcal{W}^+$-space is first transformed through a layer-specific affine operation to obtain stylecode $s_l^{(i)} \in \mathcal{S}$, at all layers of the generator module. The input to the generator module is a learned tensor of $4 \times 4$ resolution. It is gradually increased to a resolution of $1024 \times 1024$ as the input tensor is passed down through the layers of the generator.

We denote the constant input tensor as $c$ and the feature obtained at a layer $l$ as $f_l^{(i)}$. Further, we denote the $\mathcal{W}^+$ code at layer $l$ as $w_l^{(i)}$ and a layer in generator module as $\Phi_l^{(i)}$. Therefore, $f_l^{(i)}$ can be expressed as $f_l^{(i)} = \Phi_l^{(i)}(f_{l-1}^{(i)}), w_l^{(i)}$, where $l \in \{1, 2, \cdots, 18\}$, $c = f^{(0)}$ and the generated image $I = \sum_{l=1}^{18} RGB_l^{(i)}(f_l^{(i)})$.

In our work, we aim to find a latent vector $\Delta_l^{(i)}$ in the $\mathcal{W}^+$ such that the image generated by the latent code $w_l^{(i)} + \Delta_l^{(i)}$ applied to every layer of generator results in an edited image $I^*$. For simplicity, we denote $f^*$ and $w^* = w + \Delta$ as edited features and $\mathcal{W}^+$ latent code, respectively. Therefore, we have $f^* = \Phi_l^{(i)}(f^{(i-1)}), w_l^{(i)} + \Delta_l^{(i)}$. A recent study showed that StyleGAN2 learns global attributes such as position in earlier layers, structural changes in middle layers, and appearance changes (e.g., color) in the final set of layers [18, 45]. However, determining the right set of layers for a given text prompt is challenging and has been explored only empirically in FEAT [15].

3.2. Co-optimized region and layer selection (CORAL)

We aim to edit the image to match the text prompt with minimal changes. To this end, the first step is correctly identifying the region of interest. Further, given the diversity and richness of latent space at each layer in the generator, we posit that the edits to the image can come from multiple layers of the StyleGAN2 generator.

To address both requirements, we introduce CORAL, a co-optimized region and layer selection mechanism. In CORAL, we propose two simple-yet-effective approaches for learning a soft binary mask $m^{(i)} \in [0, 1]^{d_{lm}}$ at every layer of the generator module with the same height and width dimensions as the feature resolution at the given layer.

**CORAL based on segment-selection.** We can use any off-the-shelf pre-trained semantic segmentation network to determine the region of interest in this approach. Intuitively, existing image segmentation networks generally capture semantic parts of the image that we are interested in editing, such as eyes, mouth, and lips. Therefore in many cases, this problem can be posed as selecting the appropriate segments. To achieve this, we introduce a matrix $e$ of dimension $P \times 18$ where $P$ is the number of classes predicted by the segmentation network. Each entry in the matrix $e$ is in the range $[0, 1]$, where 1 represents a confident segment selection for the given text prompt $t$.

The matrix $e$ is converted into a spatial mask $m_l^{(i)}$ by masking the segments with the confidence values and resizing the segmentation map to the resolution of the feature maps at each layer. In the training phase, the parameters in the matrix $e$ are trained after applying a sigmoid, and during inference, we apply a prompt-specific threshold $\tau_t$ to the sigmoid. As depicted in Figure 4, the only trainable parameters in this pipeline are $e$. Therefore, this can achieve desirable edits with high accuracy up to 8x faster than FEAT [15].

**CORAL based on convolutional attention network.** Segment-selection-based CORAL is limited by the segments available in the pre-trained network. As shown in results Figure 6-F, the segment-selection method is prone to over-selection or under-selection of the region of interest. To overcome this limitation, we implement an attention network that directly predicts the masks $m_l^{(i)}$ at every layer of the generator as shown in Figure 4. In this architecture, we obtain a mask with the exact resolution as that of the corresponding feature in the layer. Unlike FEAT, we hypothesize that the mask at a layer $l$ should depend only on the features $f_l^{(i)}$ available at the current layer since we are interested in predicting the mask at every layer.

Despite incurring higher training costs from having to learn the convolutional layers, the masks produced with this approach are smoother and avoid over/under-selection issues by accurately predicting the correct region of interest.
3.3. Multi-layer feedforwarded feature blending

CORAL produces soft binary masks $m^{(l)}$ at every layer of the generator module. These masks blend features such that the features corresponding to the confident regions are borrowed from features $f^*$ generated with updated style code, and on similar lines, features from non-confident regions are borrowed from original features $f$ of the unedited image. This ensures that we only modify the regions corresponding to the text prompt and prevents modifications of non-masked regions. Unfortunately, a 0-mask (completely black mask) at any layer would throw away any updated feature information from the previous layers and would propagate the original features $f$ from that point onward.

To prevent this bottleneck, we design a novel multi-layer feature blending strategy (see Figure 3) that utilizes a parallel pathway where the feature obtained from layer $l - 1$ is passed through the generated block $\Phi$ twice - once with the original latent code $w$ and another pathway with updated latent code $w + \Delta$ to obtain two feature sets for blending. The former feature can be viewed as a feature that is not edited but has all the information propagated from previous layers. The multi-layer blending strategy expressed in (1), ensures that no feature information is lost along the way.

\[
\begin{align*}
\widetilde{f}^{s(l)} &= \Phi(l)(f^{s(l-1)}, w^{(l)} + \Delta^{(l)}) \\
\tilde{f}^{(l)} &= \Phi(l)(f^{s(l-1)}, w^{(l)}) \\
f^{s(l)} &= m^{(l)} \odot \tilde{f}^{s(l)} + (1 - m^{(l)}) \odot \tilde{f}^{(l)} \tag{1}
\end{align*}
\]

Intuitively, when the mask is completely blank (which is often desirable to keep the edits to a minimum), the features are feedforwarded simply with edits from previous layers.

3.4. Types of latent edits

For a given convolutional layer $l$, when the learned latent edit $\|\Delta^{(l)}\| > 0$, the corresponding feature $f^{s(l)}$ in (1) incorporates attributes which are desirable for semantic alignment with the given text prompt. The mask $m^{(l)}$ counteracts possible undesirable artifacts through a region-of-interest-aware interpolation strategy.

The $\Delta^{(l)}$ by itself is, however, well studied in [22, 34], both of which identify a single global direction that can semantically edit images for a given text prompt. Such a simple parameterization does result in accurate edits for simple text prompts, as discussed in [34].

Our findings suggest that training time is significantly reduced for prompts where a global direction can affect desirable changes. However, a more involved image-dependent non-linear mapper model $g(\cdot)$ as a function of $w^{(l)}$ at every layer can affect such changes with higher precision.

Therefore, we implemented CORAL for both versions of latent edits: (i) global direction; (ii) latent mapper. The latent mapper $g(\cdot)$ is an MLP-based model along the lines of [34, Section 5], where the $w^{(l)}$ are split into three groups: coarse ($l$ 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 multi-layer feature blending mechanism is independent of the parametrization of the latent edit, which is jointly learned with the mask $m^{(l)}$ predictors.

3.5. Loss formulation

We now describe our proposed methods’ training strategy and loss formulation. We are given a text prompt $t$ and an image with corresponding $\mathcal{W}^+$ code $w$. The goal is to find the right region of interest using a CORAL framework and determine the latent vector to help with the image edit. The only trainable components in our approach are the latent vector $\Delta$ and the parameters in the CORAL framework.

In the case of segment selection, the only trainable component in CORAL is the matrix, and in the case of convolutional attention networks, the Conv layers in the attention network are trainable.

**CLIP loss:** The first key loss component is the CLIP loss originally proposed in StyleCLIP [34]. The idea is to use the pre-trained CLIP model to edit the latent vector such that the embedding of the image $I^*$ produced aligns with the embedding of the text prompt $t$ in CLIP’s latent space.

In addition we also synthesize the image $\tilde{I}$, by setting $m_{i,j,l} = 1 \forall i, j, l$ in (1) and compute its CLIP loss. To understand this, we can envision $I^*$ as a sophisticated non-linear interpolation between $I_0$ and $\tilde{I}$ using strategies given in (1). Here $I_0$ is the original unedited image.

By simply imposing a CLIP loss on $I^*$, $\tilde{I}$ remains unrestricted and can potentially contain undesirable artifacts, as long as $I^*$ aligns with the text prompt $t$. However, our region selectors in Section 3.2 derive their supervision from $\tilde{I}$ and might also learn to include these artifacts. Our final semantic alignment loss is as follows:

\[
\mathcal{L}_{clip} = \frac{1}{2} \left( D_{CLIP}(I^*, t) + D_{CLIP}(\tilde{I}, t) \right) \tag{2}
\]

**$L_2$ loss:** Controlled perturbations to the latent spaces of a StyleGAN2 can result in smooth semantic changes to the generated image. As a result, we optimize the squared Euclidean norm of $\Delta$, i.e., $\mathcal{L}_{l_2} = \|\Delta\|_2^2$, in the $\mathcal{W}^+$ space to prefer latent edits with smaller $l_2$ norms.

**ID loss:** In order to prevent changes to the identity of a person during image manipulation, we impose an ID loss $\mathcal{L}_{id} = 1 - \langle R(I^*), R(I) \rangle$ using cosine similarity between the embeddings in the latent space of a pre-trained ArcFace network $\mathcal{R}$ [10, 22, 34, 40].

**Minimal edit-area constraint:** We encourage the network to find an edit with changes to compact image areas. In the case of segment selection, this is achieved by penalizing the CORAL matrix $e$ as follows:

\[e^{\text{Unlikely in [34], we remove the LeakyReLU activation after the final fully connected layer, as it empirically expedites the optimization.}}
}
\[
L_{area}^{ss} = \sum_{i,j} e_{i,j}
\]  

(3)

In the case of a convolutional attention network, this is achieved by imposing the minimal edit constraint directly on the masks \( m \) as follows:

\[
L_{area}^{can} = \sum_{l} n_l \left( \sum_{i,j} m_{i,j}^{(l)} \right)
\]  

(4)

where \( n_l \) is a normalizing constant defined per layer to account for the growing feature dimensions as the feature passes through the StyleGAN2 generator module.

**Smoothness loss:** In the case of the convolutional attention network, it would be desirable to predict a smooth mask. This is achieved by imposing a total variation loss \([15]\). In the case of a convolutional attention network, this is typically achieved by setting

\[
L_t v = \sum_{i,j,l} \| m_{i,j}^{(l)} - m_{i+1,j}^{(l)} \|^2_2 + \sum_{i,j,l} \| m_{i,j}^{(l)} - m_{i,j+1}^{(l)} \|^2_2
\]  

(5)

In summary, the loss formulations for the segment selection and convolutional attention mechanisms are as follows:

\[
L_{ss} = L_{clip} + \lambda_{t v} L_{area} + \lambda_{id} L_{id} + \lambda_{area} L_{area}^{can}
\]

\[
L_{can} = L_{clip} + \lambda_{t v} L_{area} + \lambda_{id} L_{id} + \lambda_{area} L_{area}^{can} + \lambda_{tv} L_{tv}
\]

(6)

Both the CORAL module and the latent editor are optimized in an end-to-end fashion using the above losses.

### 4. Experiments

We evaluate CORAL mainly in the context of human faces and demonstrate high-quality edits to photo-realistic faces of size \( 1024 \times 1024 \) generated by a StyleGAN2 pre-trained on the FFHQ dataset \([18]\). We present additional results on sketch and pixar domains as well as Cars dataset \([23]\) in Suppl. For both variants of CORAL in Section 3.2, we compare edits from the global direction and latent mapper in Section 3.4. All hyperparameter configurations for (6) and (7) are provided in Suppl.

#### 4.1. Training and inference

The loss functions corresponding to the two different variants of CORAL are given by (6) and (7). Our experiments were conducted on one NVIDIA Tesla P40 24 GB GPU with a batch size of 3. The latent editor and CORAL modules are jointly optimized using an Adam optimizer \([21]\) while keeping the StyleGAN2 fixed.

For a given text prompt \( t \), a data point is given by a randomly sampled standard normal vector \( z \sim \mathcal{Z} \) space, and the maximum number of iterations is set to 20,000. However, in Table 1, we note that the training time required for achieving the desired quality of edit increases as we switch from segment-selection to a convolutional attention network, the same as in going from global direction edits to training a latent mapper.

Furthermore, during inference, we limit the automatically selected regions for editing by setting \( m^{(t)} \leftarrow m^{(t)} \odot 1 \{ \tau (m^{(l)} \geq \tau) \} \) where \( \tau \) is typically 0.85. For applying desirable edits and reversing them (see Figure 6-G), we have a multiplying factor \( \alpha \in [-1.5, 1.5] \) for the edit direction \( \Delta \).

Out of the 18 convolutional blocks and the corresponding \( w \) code per layer, our CORAL strategy and the latent edits, as well as edits from our baselines, are only performed on the first 13 layers, which are known to span coarse and fine controls over diverse attributes \([44]\) such as expressions, age, style and color of facial hair, and eyes, among others.

**Segment selection:** Based on ideas from \([31]\), a pre-trained mixture model is used for performing unsupervised semantic segmentation of the StyleGAN2 generated images into 5 classes per pixel. This model is then used to determine the region of interest with CORAL based on segment selection. In Figure 6-E, we also compare with CORAL for a weakly
Figure 6 Each column in figures A to D demonstrates a text-driven edit on an input image along with the corresponding layers and regions selected. As a limitation of segment selection, we observe over-selection of the region of edit in figure F, which is absent in E. Figure G compares edits along both the positive and negative direction where we observe intuitive differences between removal and application of makeup, happy vs. unhappy and curly vs. smooth hair. Finally, Figure H demonstrates the edit regions selected by CORAL across different layers of the StyleGAN2 for a complex prompt.

Figure 7 Comparison of CORAL with FEAT* for multi-faceted prompts. supervised 34 class DatasetGAN [49] network, trained on the features of StyleGAN2 network using few shot labels.

Attention network: At each convolutional block $l \in [1, 2, \ldots, 13]$ of the StyleGAN2, the attention network first applies 32 different $1 \times 1$ convolutional filters upon the spatial features $f^{(l)}$ to reduce the number of channels to 32 followed by ReLU [27] activation, after which another $1 \times 1$ convolutional layer and sigmoid activation are applied to obtain $m^{(l)}$. We set $n_l = \frac{1}{\text{size}[l]}$ in (4), where size[l] is given based on the height and width of $f^{(l)}$, for example if the resolution is $32 \times 32$, then $n_l = \frac{1}{32}$.

4.2. Evaluation

Our method is most closely related to StyleCLIP [34], StyleMC [22] and FEAT [15]. For a comparison with CORAL, we run the official implementation of the latent mapper technique of StyleCLIP, as well as a re-
Table 1: Average Clean-FID [33] and training time to desirable quality. Text-prompt legend: T1 Happy; T2 Surprised; T3 Blue eyes; T4 Mohawk hairstyle. Method legend: 1) SS Global; 2) SS Mapper; 3) AttnNet-Global; 4) AttnNet-Mapper; 5) StyleCLIP; 6) StyleMC; 7) FEAT∗

<table>
<thead>
<tr>
<th></th>
<th>Clean-FID</th>
<th>Avg. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>T1</td>
<td>T2</td>
</tr>
<tr>
<td>CORAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4.23</td>
<td>4.42</td>
</tr>
<tr>
<td>2</td>
<td>3.75</td>
<td>9.28</td>
</tr>
<tr>
<td>3</td>
<td>2.22</td>
<td>2.26</td>
</tr>
<tr>
<td>4</td>
<td>1.85</td>
<td>6.38</td>
</tr>
<tr>
<td>others</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>6.98</td>
<td>18.14</td>
</tr>
<tr>
<td>6</td>
<td>2.93</td>
<td>25.12</td>
</tr>
<tr>
<td>7</td>
<td>2.51</td>
<td>9.46</td>
</tr>
</tbody>
</table>

implementation of StyleMC3 which optimizes a single global direction across multiple images, only for layers of the StyleGAN2 until resolution 256 × 256 (see [22]).

Without an official implementation of FEAT, we evaluate our re-implementation of FEAT denoted by FEAT∗ with l ∈ {7, 13}. We maintain equivalent settings in the design of the latent mapper and attention network and only intend to compare the single-layer FEAT-style blending with our multi-layer feedforwarded blending (see Figure 3).

4.3. Results

Merits: In Figure 5, we observe that both the StyleCLIP mapper method and StyleMC result in undesirable edits, such as irrelevant edits to the background. StyleCLIP also reduces the age in the first row and affects the neck region. In the fourth row, we see that in addition to applying the prompt surprised, it discards the white shirt. StyleMC affects the first three rows’ complexion, facial expression, and hair color. As also seen in [15], we find that for finer edits (row 1), FEAT-style blending at layer 13 (FEAT∗13) is preferable as also with FEAT∗7 for coarse edits (rows 2-4). We find that blue eyes results in unwanted edits when blended at l = 7, and so does mohawk hairstyle at l = 13.

CORAL, however (last four columns in Figure 5), only affects the relevant regions of interest, which would be the hair region for long curly hair and mohawk hairstyle, the eyes and mouth for blue eyes as well as surprised. These traits persist in Figure 1, and Figure 6-A to D wherein the edits are incorporated such that the editing area is minimal and is limited to only the relevant layers. CORAL learns the layers and regions to edit automatically with no domain knowledge or repeated trials. The edits are highly accurate. For example, the prompt mustache does not also affect the beard, as is apparent from the corresponding masks.

Under the minimality constraints given by (3) and (4), we observe that for enabling finer edits such as blue eyes and purple hair, only the latter higher resolution layers (typically l ≥ 8) are selected, whereas, for coarser structural edits, the earlier smaller layers (typically l ≤ 8) are automatically selected. We clearly see that when CORAL is trained for complex multi-faceted prompts such as curly hair and lipstick (see Figure 7 and Figure 6-H), the hair edits come from earlier layers whereas the lip edits come from last layers. Furthermore, for such prompts, we found that FEAT blending fails to preserve realism by introducing noise artifacts (see the example for FEAT∗13 under makeup and afro hairstyle in Figure 7). This is also seen in Figure 5 for mohawk using FEAT∗13.

From Table 1, we see that while the Clean-FID [33] of all our edits remains within acceptable bounds of the initially generated distribution, the time required to train CORAL to a desirable edit quality increases with the complexity of the region, layer selector, and the latent editor combined, from method 1 to 4. Segment-selection-based CORAL is significantly faster to train than the attention network.

We also observe that, on average, segment selection has a higher FID than attention network. Along similar lines, global edits have a higher FID than latent mapper, except for surprised, which we attribute to global edits predominantly affecting the eyes for this prompt, even for StyleMC, unlike the mapper method which also opens up the mouth.

Limitations: The segment-selection-based approach trains at a fraction of the time taken by its counterparts, as seen in Table 1. However, the defined segments of a pre-trained segmentation model can affect performance. For example, in Figure 6-F, our semantic segmentation model combines the region, layer selector, and the latent editor combined, to a desirable edit quality increases with the complexity of the generated distribution, the time required to train CORAL to a desirable edit quality increases with the complexity of the region, layer selector, and the latent editor combined, from method 1 to 4. Segment-selection-based CORAL is significantly faster to train than the attention network.

We also note that the quality of the mustache is superior in Figure 6-B compared to A. It turns out that unlike our non-linear mapper which succeeds, the global edits result in black coloration in the mustache region in many examples.

Ethical aspects: In line with current works, we benchmark our approach using publicly available celebrity images [34]. Although our approach demonstrates superior edits on diverse faces, our approach still inherits biases present in StyleGAN and CLIP models. Further, a generative model (e.g., CORAL) could be misused to create fake information.

5. Conclusion

CoralStyleCLIP leverages StyleGAN2 and CLIP models to co-optimize region and layer selection for performing high-fidelity text-driven edits on photo-realistic generated images. We demonstrate the efficacy of our generic multi-layer feature blending strategy across varying complexities of the latent editors and region selectors, addressing limitations regarding manual intervention, training complexity, and over- and under-selection of regions along the way. The CORAL strategy can also enhance interactive editing experience by utilizing the predicted masks at each layer.
References


