UP-FacE: User-predictable Fine-grained Face Shape Editing

Florian Strohm*,1,3 Mihai Bâce*,2 Andreas Bulling[3]
[1]Fraunhofer IPA [2]KU Leuven [3]University of Stuttgart

Abstract

We present User-predictable Face Editing (UP-FacE) – a novel method for predictable face shape editing. In stark contrast to existing methods for face editing using trial and error, edits with UP-FacE are predictable by the human user. That is, users can control the desired degree of change precisely and deterministically and know upfront the amount of change required to achieve a certain editing result. Our method leverages facial landmarks to precisely measure facial feature values, facilitating the training of UP-FacE without manually annotated attribute labels. At the core of UP-FacE is a transformer-based network that takes as input a latent vector from a pre-trained generative model and a facial feature embedding, and predicts a suitable manipulation vector. To enable user-predictable editing, a scaling layer adjusts the manipulation vector to achieve the precise desired degree of change. To ensure that the desired feature is manipulated towards the target value without altering uncorrelated features, we further introduce a novel semantic face feature loss. Additionally, we analyse the correlations between different facial features and incorporate this information into the transformer network's regularization process. This dynamic regularization allows naturally correlated facial features to be adjusted simultaneously in a coherent manner. Qualitative and quantitative results demonstrate that UP-FacE provides competitive face editing capabilities while offering precise and user-predictable control over 23 distinct facial shape features.

1. Introduction

Computational face editing is an active area of research with broad applications, such as digital image editing. Powerful generative models such as StyleGAN [23, 24] can generate high-quality face images and have been shown to learn disentangled latent image spaces that facilitate controlled face editing [8, 42, 50, 55]. Because of this, there has been a growing interest in methods that enable users to control the generative process. Various methods have been proposed for controllable face editing, from unsupervised ones that automatically discover disentangled latent dimensions to control specific parts of the face [16, 34, 41] over mask-based methods that use segmentation masks to control the face geometry [13, 29, 30, 44, 46, 47], to text-based methods that use natural language [3, 20, 22, 36, 45, 56], 3D-based methods that translate a 3D face model to a real image [10, 27, 33, 49, 50], or attribute-based approaches [1, 7, 12, 32, 43, 55, 57, 60] that can control specific appearance attributes.

While all of these methods can produce impressive results, they also have fundamental limitations that prevent users from easily and deterministically editing the face shape: Attribute- and mask-based methods are tedious as they require either manual annotations or sufficient skill to manipulate segmentation masks. While unsupervised methods overcome these limitations, they are often model-dependent, and the dimensions to manipulate the desired attributes might not be discovered or might be entangled with other attributes. 3D-based methods are effective for novel-view synthesis, lighting manipulation, and transferring expressions but require significant 3D modelling efforts for face shape editing. Most importantly, none of these methods permit users to know the outcome of a particular face edit upfront. Instead, users must manipulate facial features via trial and error until they are satisfied with the result, resulting in a tedious and error-prone face editing process.

To address these limitations, we present *User-predictable Face Editing* (UP-FacE). UP-FacE can manipulate 23 face features that describe key characteristics of human face geometry, such as the eye and mouth width and openness, or the chin and eyebrow shape (see Figure 1). Similar to action units from the Facial Action Coding System, which describe how face appearance changes for different emotions [11], we introduce *semantic face features* that are derived from and describe the relation between 2D facial landmarks and face characteristics. Un-

*Part of this work was done while at University of Stuttgart.

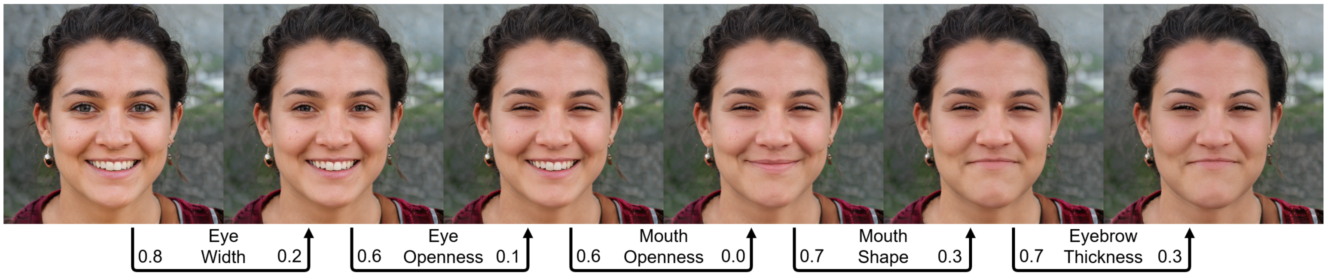

Figure 1. UP-FacE allows for fine-grained control over various face shape features, such as eye width, eye and mouth openness, or eyebrow thickness. Unlike existing methods that rely on trial-and-error feature editing, UP-FacE enables predictable edits by letting users set the desired degree of change directly and anticipate the resulting outcome. For example, setting *mouth openness* to 0.0 results in a fully closed mouth. In addition, our method enables both isolated progressive (i.e. of the same feature) and sequential (i.e. multiple different features) edits without altering other (unrelated) facial features.

like existing methods, these semantic features are derived from landmarks, enabling us to quantify the exact facial feature values. This ability allows us to train UP-FacE without the need for any manual attribute annotations. To train our model, we propose a novel *semantic face feature loss* that enforces the change in the desired face feature. Furthermore, we estimated the correlations of the semantic face features on real human faces to dynamically regularise UP-FacE allowing it to modify naturally correlated features jointly. That is, our method only manipulates the desired and related features, without altering other undesired characteristics of the face.

We report qualitative and quantitative results showing that UP-FacE allows for predictable manipulation of facial features with a high degree of disentanglement. Most crucially, our capability to measure and set precise facial feature values lets users specify target edits directly, instead of searching via trial and error. In addition, we developed a user interface that consists of multiple sliders, similar to common digital character editing tools [2, 40] that provide users with an easy-to-use interface with precise control over the face feature values (see video demonstration: `https://youtu.be/xSXAJP1M3ew`), paving the way for more user-friendly digital face editing tools.[1]

2. Related Work

Unsupervised Face Editing. Most unsupervised methods aim at decomposing the latent space [6, 16, 34] or weights [41] of a generative model to identify semantic editing dimensions. While such methods do not require labelled data, only a limited amount of meaningful semantic directions can be discovered and the level of entanglement is typically higher compared to supervised methods. DragGAN [35] allows users to edit the shape of a face by moving facial landmarks. Although this enables a variety of shape edits, it requires significant user effort, especially when multiple landmarks must be adjusted. Current landmarks can be detected automatically, but target landmark locations remain unknown and must be specified by users. Instead, we specify semantic features using landmarks, which enables UP-FacE to learn how to move these landmarks, thereby facilitating effortless user control.

Mask-based Face Editing. Mask-based methods condition a generative model on face masks to provide fine-grained control. One common approach is to allow users to erase parts of the face and sketch the desired outlines of facial features. Subsequently, a generative model performs image inpainting based on the sketch mask [4, 5, 37]. Another approach is to condition face segmentations, where facial features like the eyes and mouth are separated by masks. This allows users to edit faces by manipulating the face segmentation masks [13, 29, 30, 44, 46, 47]. Most mask-based methods still require significant user effort and skill to manipulate masks for the desired edits, whereas our slider-based interface does not require special skill.

Text-based Face Editing. Xia et al. [56] and Huang et al. [22] combined the ideas from mask-based techniques with text-guided image edits. Patashnik et al. [36] combined a pre-trained generator and a powerful CLIP [39] image-language encoder for improved text-guided editing capabilities. Subsequent work improved upon this to perform more localised edits [20] or to generalise to more diverse text prompts [45]. Text-based face editing methods provide a simple language interface to facilitate a large variety of editing possibilities. However, these methods do not allow for very precise and fine-grained edits.

Attribute-based Face Editing. Early work on attribute-based face editing focused on binary control of attributes defined during inference [7, 14, 18, 28, 32, 53, 57, 58, 62, 62], e.g. open or closed mouth. Other works used attribute classifiers during training for improved editing results [12, 17, 48]. Generative models conditioned on the attributes have typically limited progressive editing control. Therefore, Shen et al. [43] trained SVMs to separate face at-

[1] Project code and models will be released upon acceptance.

tributes in the generator's latent space with the SVM normal vectors providing global semantic directions along which the corresponding attributes could be manipulated smoothly. Han et al. [15] improved these global directions with a local iterative search based on the current image. Another approach is to learn how to modify the latent vectors of a pre-trained generative face model to perform semantically plausible face edits [1, 21, 25, 55, 60]. Yang et al. [59] proposed a method that did not require binary attributes but sets of images that each shared a particular attribute. Despite significant progress in attribute-based face editing, two main disadvantages remain: First, the latest methods can only control attributes for which enough manually labelled data exists. Manual labelling at scale is not only tedious, time-consuming, and costly, but it can also be challenging to provide binary labels for features like *chin shape* for which no clear boundary exists. Second, while current methods allow for progressive control over attributes, it remains unclear for users how far to move along a semantic dimension to achieve a desired result, e.g. to fully close the mouth. Our method addresses both limitations as it does not require any manually labelled data and provides semantically meaningful, user-predictable control over the face shape features.

3. Method

The core idea of our *User-predictable Face Editing* (UP-FacE) method is to allow the control of face characteristics through a set of *semantic face features* in a deterministic way. In the following, we first define our semantic face features and then present how UP-FacE can manipulate the shape of a face based on these features.

3.1. Landmark-based Face Features

We define semantic face features $\mathcal{M}$ based on commonly used 98 2D landmarks available in the Wider Facial Landmarks in the Wild (WFLW) dataset [54] (see Figure 2 right). Inspired by common digital face creation tools [2, 40] that provide an interface with multiple sliders to manipulate semantic face features, we define a total of 23 features $m \in \mathcal{M}$ (see Table 1). x and y indicate which dimension of a landmark we used for calculating the feature, while the subscripts refer to the corresponding landmark as shown in Figure 2 (right). The features fall into three different categories: absolute distance, relative distance, and relative anchor distance. Most features are defined based on the relative distance between two or more landmarks, such as *eye width* or *nose length*. The advantage of relative distance features is that they are invariant to head translations. Given a lack of clear reference landmarks for some facial features, we additionally define five absolute distance features that encode the absolute position of facial landmarks in the face image: horizontal and vertical *pupil position*, *eyebrow height*, *mouth height*, and *chin length*. The disadvantage of

Semantic Feature	Landmark Formula
Eye width	$(x_{64} - x_{60}) + (x_{72} - x_{68})$
Eye distance	$(x_{68} - x_{60}) + (x_{72} - x_{64})$
Eye openness	$(y_{66} - y_{62}) + (y_{74} - y_{70})$
Pupil position x	$x_{96} + x_{97}$
Pupil position y	$y_{96} + y_{97}$
Eyebrow height	$\sum_{i=33}^{50} y_i$
Eyebrow width	$(x_{37} - x_{33}) + (x_{46} - x_{42})$
Eyebrow thickness	$(y_{41} - y_{34}) + (y_{38} - y_{37}) + (y_{50} - y_{42}) + (y_{47} - y_{45})$
Eyebrow shape	$(y_{33} - y_{35}) + (y_{37} - y_{35}) + (y_{42} - y_{44}) + (y_{46} - y_{44})$
Nose width	$x_{59} - x_{55}$
Nose length	$y_{57} - y_{51}$
Nose pointiness	$y_{57} - y_{54}$
Mouth height	$\sum_{i=76}^{88} y_i$
Mouth width	$x_{92} - x_{88}$
Mouth openness	$y_{94} - y_{90}$
Mouth shape	$(y_{76} - y_{90}) + (y_{82} - y_{90})$
Upper lip thickness	$y_{90} - y_{79}$
Lower lip thickness	$y_{85} - y_{94}$
Chin length	y_{16}
Chin width	$x_{18} - x_{14}$
Chin shape	$(y_{14} - y_{16}) + (y_{18} - y_{16})$
Jaw width	$x_{23} - x_9$
Temple width	$x_{32} - x_0$

Table 1. List of the proposed semantic face features based on different facial landmarks. The characters x and y indicate if a particular landmark's x- or y-coordinate is used, while the subscripts refer to the landmarks shown in Figure 2 (right).

these features is that they are not translation-invariant. For example, translating the whole face in an image downwards also increases the *chin length*. However, as we will show later, these features are still effective at controlling the desired facial features. Although relative reference landmarks are possible, e.g. by defining *chin length* (y_{16}) relative to the mouth position, this introduces undesired side effects because the feature then depends on both chin length and mouth position. For *eyebrow shape*, *chin shape*, and *mouth shape*, we define translation-invariant features that relate two landmarks to a third one (anchor) and, as such, allow us to control the angle between them and the shape of the underlying facial feature. Any differentiable function to combine landmarks can be used, potentially allowing for many more interesting semantic face features.

3.2. User-predictable Face Editing

Our goal is to develop a method that allows the predictable manipulation of the latent vector of an image I in such a way that only a desired semantic face feature is changed towards a target value, thus generating a new image I_{edit}. Figure 2 shows the overall architecture of the

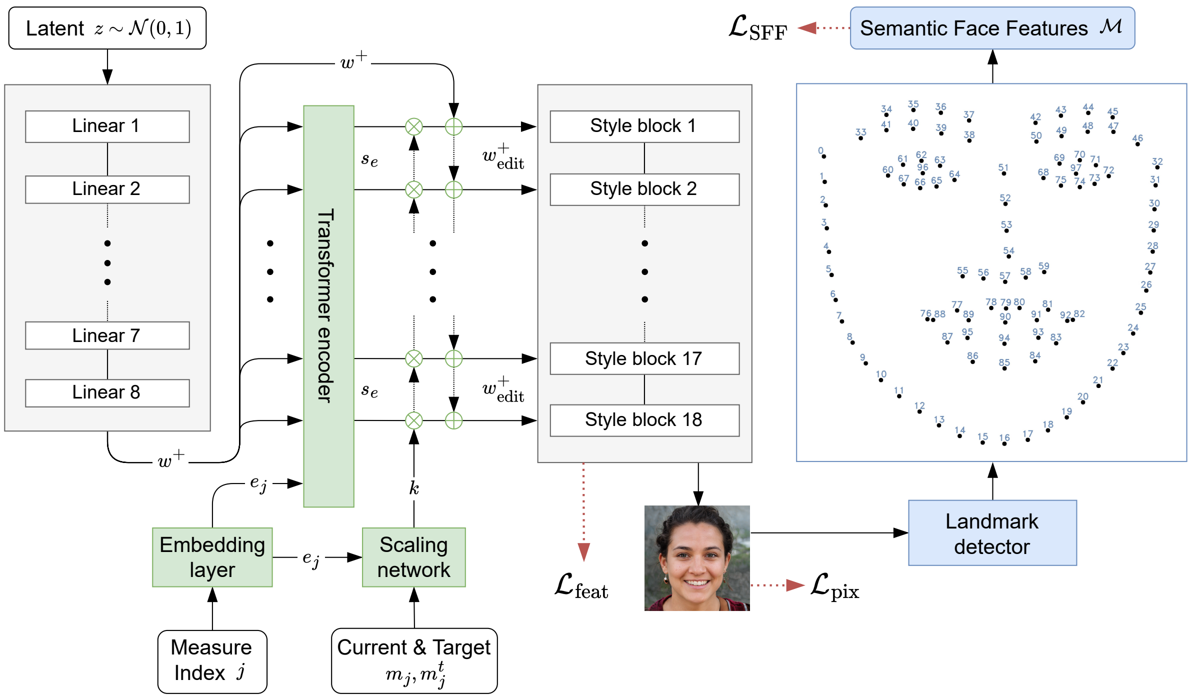

Figure 2. Overview of our method. Shown in grey is the StyleGAN2 architecture. We inject a transformer encoder network between the StyleGAN mapping and synthesis network (green) that can modify the latent vector w^+ based on the desired face feature embedding e_j by adding a semantic manipulation vector s_e. This manipulation vector is scaled by k, a scalar predicted by the scaling network based on the current and target face feature values m_j and m_j^t. The landmark detector and calculated face features (blue) are only required during the training of the components highlighted in green.

proposed method to achieve this goal. The grey components are from the state-of-the-art generative network StyleGAN2 [24] that was pre-trained on the Flickr-Faces-HQ dataset (FFHQ) [23] to generate high-quality images of human faces. A mapping network first maps latent vectors $z \in \mathbb{R}^{512}$ drawn from a standard normal distribution, $z \sim \mathcal{N}(0, 1)$, into $\mathcal{W}$ space. The vector $w \in \mathcal{W}$ is then repeated 18 times, $w^+ \in \mathcal{W}^+ \in \mathbb{R}^{512\times 18}$ and input to the generative network that subsequently generates the image. Prior work has shown that the $\mathcal{W}^+$ space is highly disentangled and can be modulated to perform various semantic image manipulations [8, 42, 50, 55].

Inspired by this, we propose to inject a Transformer encoder network [52], highlighted in green in Figure 2, between StyleGAN2's mapping and synthesis network. The objective of the Transformer encoder is to predict a semantic manipulation vector $s_e \in \mathbb{R}^{512\times 18}$ that is scaled by k and added to w^+:

$$w^+_{\text{edit}} = w^+ + k * s_e, \tag{1}$$

such that w^+_{edit} translates to the same image as w^+ except that the value of the selected semantic face feature m is changed to the target value. We input w^+ as a sequence of 18 w vectors with an additional embedding vector $e_j \in \mathbb{R}^{512}$ into the Transformer. The vector e_j encodes information about the semantic face feature $m_j \in \mathcal{M}$ to be manipulated and is produced by an additional embedding layer trained end-to-end on 23-dimensional one-hot face feature vectors based on our semantic features defined in Table 1. The Transformer consists of multiple layers of multi-head self-attention to learn how to manipulate w^+ based on e_j to generate s_e, which is extracted from the output sequence by ignoring the last element corresponding to the input embedding vector. A scaling network takes the embedding vector and the current and target values for the face feature we want to change, m_j and m_j^t, as input. It predicts a scaling factor k multiplied with s_e as defined in Equation (1), allowing us to change the face feature value by a desired amount deterministically.

During the training of the scaling network, embedding layer and Transformer encoder, our method requires a differentiable landmark detector to be able to calculate the face features M (highlighted in blue in Figure 2). The weights of StyleGAN2 and the landmark detector model are frozen and only used to calculate the gradient to update the weights of the components highlighted in green. We define the full loss function $\mathcal{L}$ to train the network as follows:

$$\mathcal{L} = \lambda_{\text{pix}} * \mathcal{L}_{\text{pix}} + \lambda_{\text{feat}} * \mathcal{L}_{\text{feat}} + \lambda_{\text{SFF}} * \mathcal{L}_{\text{SFF}}, \tag{2}$$

with λ_{pix}, λ_{feat} and λ_{SFF} representing scalars to weight the different loss terms. $\mathcal{L}_{\text{pix}}$ is defined as the pixel-based mean squared error (MSE) between the original image I and the modified image I_{edit}, which aims at preserving the original image as much as possible. This loss is zero if the Transformer predicts $s_e = \overrightarrow{0}$, i.e., the network should perform as few changes as possible on w^+ to fulfil the other constraints. $\mathcal{L}_{\text{feat}}$ is defined as the feature-based MSE between I and I_{edit}, where we consider the features of the last style block in the StyleGAN2 generator. Like $\mathcal{L}_{\text{pix}}$, this loss is zero for $s_e = \overrightarrow{0}$ and therefore also encourages minimal edits. Compared with pixel supervision, feature supervision is less rigid and gives the model more flexibility to change the face while preserving overall appearance. $\mathcal{L}_{\text{SFF}}$ is our novel semantic face feature loss and is responsible for allowing the network to learn how to modify the semantic face features. It is defined as:

$$\begin{aligned} \mathcal{L}_{\text{SFF}} = {} & \text{MSE}(m_j^{\text{p}}, m_j^{\text{t}}) + \lambda_{\text{reg}} * \\ & \sum_{i=1, i\neq j}^{23} \text{MSE}(m_i^{\text{p}}, m_i^{\text{t}}) * \left(1 - \lambda_{\text{cor}} * |c(m_i, m_j)|\right). \end{aligned} \tag{3}$$

The first term calculates the MSE between the face feature m_j^p of the edited face I_{edit} and the target value for the feature m_j^t. To calculate m_j^p, the generated image is passed through a differentiable landmark detector in the forward-pass as illustrated in Figure 2 to predict the landmarks for I_{edit}. While this term encourages the model to change w^+ to manipulate the feature m_j, we must ensure that all other face features remain unchanged. Therefore, we add a regularising term with weighting λ_{reg} that sums the MSE between predicted and target values for all other features where m_i^t is fixed to the original image I. Furthermore, the regularising term is multiplied by a correlation-relaxation weight $w_{ij} = 1 - \lambda_{\text{cor}}|c(m_i, m_j)|$, where $c(m_i, m_j)$ is the absolute Pearson correlation between the edited feature m_j and feature m_i to be kept fixed. Given that not all semantic features are fully disentangled due to human anatomy, we estimate these correlations on FFHQ [24]. This weighting is monotone decreasing with correlation magnitude through the absolute-correlation term. Hence, strongly correlated feature pairs are penalised less when they change together.

4. Experiments

4.1. Implementation Details

We use a Transformer-based editing module and scaling network as described in the Method section. During training, we sample $z \sim \mathcal{N}(0, 1)$, generate w^+ and images with pre-trained StyleGAN2 [24], and sample both a feature index j and target value m_j^t. Current and edited face-feature values are obtained with SPIGA [38]. We optimise UP-FacE end-to-end (with frozen StyleGAN2 and landmark detector) for 10^5 steps using Adam [26], learning rate 2^{-5}, and batch size 16. We set $\lambda_{\text{pix}} = 1$, $\lambda_{\text{feat}} = 3$, $\lambda_{\text{SFF}} = 0.005$, $\lambda_{\text{reg}} = 0.1$, and $\lambda_{\text{cor}} = 1$. Full architecture and training details are provided in the supplementary material.

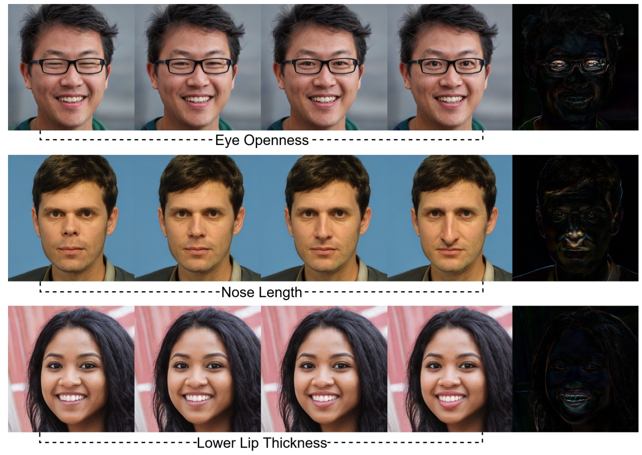

Figure 3. Example progressive edits performed with UP-FacE. UP-FacE allows to easily and predictably perform high-quality progressive edits along many different semantic dimensions with explicit control over the desired target feature values. For each demonstration of the progressive edits, we also show the difference between the first and last image, highlighting which parts of the image changed.

4.2. Interactive Face Editing Interface

Inspired by popular digital character creation tools such as FaceMaker [40] and MakeHuman [2], we developed an intuitive slider-based interface for UP-FacE (see supplementary material). For each of our 23 semantic face features, users can directly set a desired target value. Whenever a slider changes, we run a forward pass of UP-FacE to update the face image immediately. Because naturally correlated features may change jointly, we re-estimate feature values after each edit and update the related sliders accordingly. To make controls predictable, we manually define per-feature min-max bounds from pilot edits and map all sliders to the range $[0, 1]$. For example, *Mouth Openness* at 0.0 yields a fully closed mouth, while 1.0 yields a wide but still realistic opening. This design avoids exaggerated edits, improves reproducibility across faces, and reduces trial-and-error interaction compared with prior methods.

4.3. Qualitative Results

Figure 1 shows sample sequential edits performed using UP-FacE. We first generate a face by sampling a random

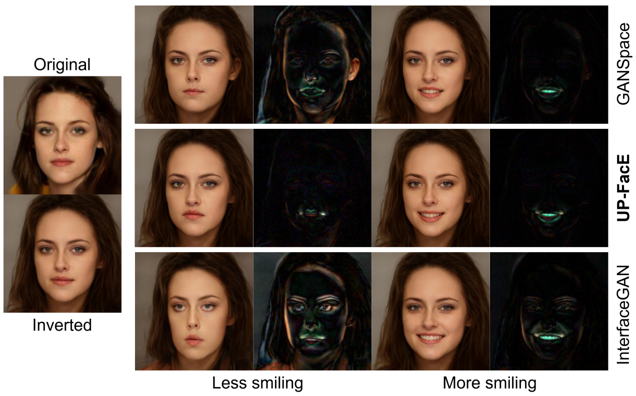

Figure 4. Sample face editing results on a real face image of UP-FacE in comparison with the two state-of-the-art methods GANSpace [16] and InterFaceGAN [43]. The original image was inverted into latent space using the e4e framework [51], and subsequently, the degree of smiling was edited. Also shown are the difference images between the edited and original images. UP-FacE is the only method that allows for fine-grained and user-predictable control of the degree of smiling without distortions.

latent vector and decoding it with StyleGAN2 (left). Afterward, we sequentially edit specific features by changing the current value to the desired target value, as indicated by the arrows. Our method performs multiple fine-grained edits on the desired features while preserving the visual appearance of the rest of the face, suggesting a high degree of disentanglement. In contrast to prior work, we can calculate the exact value of our face features at any point and set desired values to perform precise and user-predictable editing operations. This allows users to anticipate the outcome of an edit before execution, e.g. setting *mouth openness* to 0.0 always results in a closed mouth.

Because our features are defined on a continuous scale, UP-FacE naturally enables progressive edits as shown in Figure 3. For each example, the figure also shows the difference between the first and last image in the edit sequence. UP-FacE operates highly locally, as significant changes are visible only in the relevant regions. Other changes visible in the difference images are mostly due to high-frequency features, such as the outline of the face. Minor variations along these lines are visible in the difference images but are not noticeable in the actual images.

Although the semantic face features defined in Table 1 relate only a small number of landmarks, we can still model complex face dynamics accurately. For example, although *mouth openness* only relates the y-coordinate of two landmarks in the centre of the mouth, the progressive editing results in Figure 3 show that the whole mouth opens naturally. We hypothesise that the effectiveness of our semantic face features is largely due to implicit biases and correlations learned by StyleGAN2. To move the two landmarks

Model	Pred. Edit.	# Labels	Attr. ACC	FID	ID Score
GANSpace [16]	×	0	76.1%	36.89	<u>0.82</u>
IFGAN [43]	×	30K	<u>83.2%</u>	<u>32.58</u>	0.80
EditGAN [30]	×	16	**84.2%**	35.11	0.77
UP-FacE	✓	0	81.8%	**27.15**	**0.83**

Table 2. Quantitative results of GANSpace, InterFaceGAN, EditGAN and UP-FacE on the smile edit benchmark. Only UP-FacE allows for user-predictable editing and does not require any manual attribute labels. We report the smile attribute accuracy (ACC), the FID and ID Score. The best results are highlighted in bold, while the second best are underlined.

used for *mouth openness* apart, UP-FacE learns to open the whole mouth, as this is the only plausible option under the learned human face distribution of StyleGAN2.

Figure 4 shows editing results on a real face that we inverted into StyleGAN2's latent space using the e4e encoder [51]. We edit the facial expression twice, once to show less smiling and once to show more smiling. As the UP-FacE results in the middle row show, real faces can also be manipulated with high precision, which is generally harder because the inverted latent code may not lie in well-defined regions of StyleGAN2 (see supplementary material for more real-face editing results). Because smiling is one attribute that previous methods can also control, we compare our results with GANSpace [16] and InterFaceGAN [43]. Both methods fail to accurately reduce the degree of smiling and begin to distort other parts of the face, which is most noticeable in the difference images. When increasing the degree of smiling, all three methods can edit the face accurately, while UP-FacE remains more local. Additionally, UP-FacE provides more fine-grained control over the mouth by editing *mouth shape* and *mouth openness* separately, allowing us to add a smile without necessarily opening the mouth. Importantly, UP-FacE does not require manually labelled training data, while methods such as InterFaceGAN rely on large face datasets with smiling attribute annotations.

4.4. Quantitative Results

We quantitatively evaluate UP-FacE on the Smile Edit benchmark [29, 30], comparing its performance against three strong baselines: GANSpace [16], InterFaceGAN [43], and EditGAN [30]. For consistency and fairness, all methods utilise the same StyleGAN2 model. The task of this benchmark is to convert faces with neutral expressions into smiling faces. The performance is measured with three metrics: *attribute accuracy*, which measures whether a face is smiling after editing using an attribute

Model	Pixel Error ↓	LPIPS ↓	Edit Error ↓	Entangle-ment ↓
$\lambda_{\text{pix}} = 0$	0.192	0.258	0.664	0.624
$\lambda_{\text{feat}} = 0$	0.041	0.045	0.551	0.238
$\lambda_{\text{reg}} = 0$	0.043	0.047	0.549	0.267
$\lambda_{\text{cor}} = 0$	**0.040**	**0.041**	0.575	**0.225**
UP-FacE	0.041	0.045	0.529	0.234
UP-FacE 3x	0.065	0.087	**0.318**	0.340

Table 3. Ablation experiments to evaluate the impact of the choice of loss term on different error metrics: pixel λ_{pix}, feature λ_{feat}, regularisation λ_{reg}, and feature correlation relaxation λ_{cor}.

classifier pre-trained on the CelebA [31] dataset, *Fréchet Inception Distance (FID)* [19], calculated between 4000 edited images and the FFHQ dataset to evaluate image quality [24], and *Identity Score (ID Score)*, which measures if the faces' identity is preserved after editing by calculating the cosine similarity between embeddings extracted from a pre-trained ArcFace model [9]. To make a neutral face smile with UP-FacE we adjust *mouth shape* and *mouth openness* as shown in Figure 4. The results of the smile edit benchmark are reported in Table 2. We find that UP-FacE significantly outperforms all baselines regarding FID and ID scores. While InterFaceGAN achieves a slightly higher attribute accuracy than UP-FacE, it is important to note that it was trained on the same 30.000 attribute labels as the classifier used to calculate the accuracy metric. In contrast, UP-FacE did not require any attribute labels and did not explicitly learn to manipulate the smiling attribute while still being competitive with InterFaceGAN and outperforming GANSpace. EditGAN [30] achieves the highest attribute accuracy. However, it also has the lowest Identity Score and a FID significantly lower than that of UP-FacE. Furthermore, unlike previous methods, UP-FacE allows for precise and predictable control over face shape features. This could potentially be one reason for the significant improvement in FID and ID scores, as UP-FacE can automatically adjust the editing intensity based on the measured values of the face features. This allows UP-FacE for example, to only slightly modify faces that are almost smiling, whereas previous methods would move a fixed amount along a semantic editing direction, which can lead to unnecessary over-editing and distortion.

4.5. Ablation Experiments

We perform ablation experiments for different versions of UP-FacE and report four metrics: *pixel error*, *LPIPS*, *edit error*, and *entanglement*. *Pixel error* is the mean absolute error (MAE) between the original and edited image; lower is better, but a value of zero would indicate no edit. *LPIPS* [63] is a perceptual similarity metric in which small-

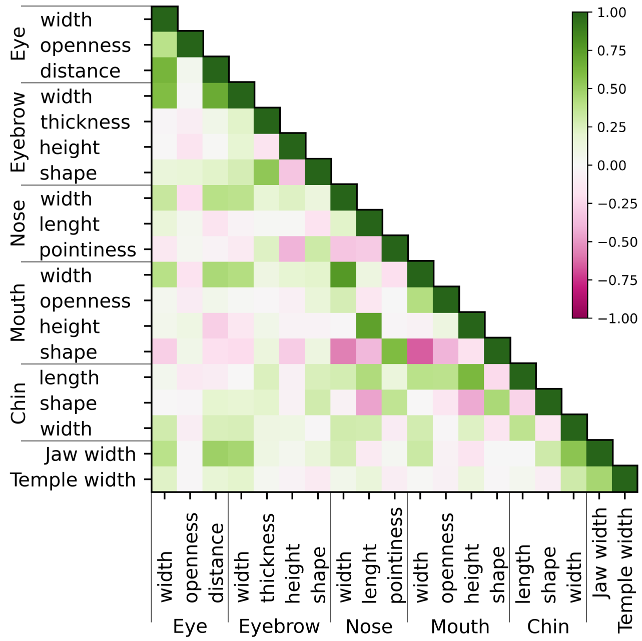

Figure 5. Correlation matrix showing the Pearson correlation between two semantic face features in each cell.

er values indicate higher similarity. *Edit error* is the MAE between predicted and target values (m_j^{p} and m_j^{t}) for features explicitly edited, whereas *entanglement* is the same MAE for features that should remain unchanged. We report results in Table 3, computed from 10,000 random images with edits on five random semantic features per face.

We can see that the model is unstable during training without the pixel loss ($\lambda_{\text{pix}} = 0$), which is reflected in the lower performance. Training without the feature loss ($\lambda_{\text{feat}} = 0$) slightly increases the edit error, indicating that the feature supervision adds some flexibility compared to pure pixel supervision, improving editing precision. Removing the regularisation term with $\lambda_{\text{reg}} = 0$ results in a higher entanglement, which is expected since we do not penalise the network when changing features other than implicitly through the pixel and feature loss. Training without the feature correlation relaxation in the regularising term ($\lambda_{\text{cor}} = 0$) significantly increases the edit error as it constrains the model more. At the same time, we can also see that this model achieves a better entanglement score compared to UP-FacE, which is expected since this term allows for the entanglement of correlated features in UP-FacE. Finally, we compare the results with UP-FacE 3x, which iteratively applies UP-FacE three times by feeding w_{edit}^{+} back as the input to the transformer. UP-FacE may sometimes over- or undershoot the desired target value for a semantic face feature. However, by iteratively applying the method multiple times, the edit error can be significantly reduced, resulting in more precise and user-predictable editing.

4.6. Correlation of the Face Features

The ablation results in Table 3 show that relaxing the regularising term based on feature correlations, as defined in Equation (3), is important for improving edit precision by purposefully entangling correlated features. Figure 5 presents the Pearson correlation coefficient matrix for a subset of our features (see supplementary material for the full matrix and evaluation). We estimate these correlations by calculating all semantic face features for each face in the FFHQ dataset. The matrix shows that most features are nearly uncorrelated, while only a small set of pairs is strongly correlated. For example, *eye width* and *eyebrow width* are strongly correlated, which is intuitive because small eyes with very wide eyebrows (and vice versa) are uncommon. A similarly strong correlation appears between *mouth width* and *nose width*. By reducing the loss penalty when the network edits such correlated features together, we give the model more freedom to perform the change while keeping edited faces aesthetically plausible. The matrix also shows that *jaw width* correlates with other width-related features. Since a larger *jaw width* usually corresponds to a wider face, other facial features can scale accordingly without incurring a high penalty. Likewise, a wide face with small facial features is uncommon. Another strong correlation appears between *nose length* and *mouth height*, which can be explained by the nose-to-upper-lip distance (the philtrum), typically around 2 cm with only slight variation [61]. Therefore, these two features tend to change together, allowing the philtrum to stay within a natural range.

While most feature correlations can be intuitively explained by natural correlations in human faces, some are likely caused by limitations of our absolute-distance semantic face features. One example is the correlation between *chin length* and *mouth height*. Because both are absolute-distance measures, they are sensitive to y-direction translations and rotations, which can produce a false positive correlation. This suggests that future work could further improve results by using more invariant features.

5. Conclusion

We proposed UP-FacE – a novel method that enables user-predictable and precise face shape editing with little effort. The key idea behind UP-FacE is the use of *23 semantic face features*, determined from different facial landmarks groups. Unlike existing methods, landmarks have the advantage that they allow the calculation of the values of these face features dynamically, which allows training UP-FacE without requiring any manual labelling. We also introduced a novel *semantic face feature loss* that encourages the model to manipulate only the desired face features while keeping unrelated features unchanged. Given that these features are continuous, we trained a scaling network that could learn how to scale the manipulation vector to achieve the desired change in facial appearance. This represents a significant advance: unlike prior methods where users had to move towards the desired appearance via trial and error, UP-FacE allows for user-predictable face editing and paves the way for more user-friendly digital face editing tools.

Acknowledgements

Florian Strohm and Andreas Bulling were funded by the European Research Council (ERC) under the grant agreement 801708. Mihai Bâce was funded by a Swiss National Science Foundation (SNSF) Postdoc.Mobility Fellowship (grant number 214434) while at the University of Stuttgart.

References

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

[2] Leyde Briceno and Gunther Paul. Makehuman: a review of the modelling framework. In *Proceedings of the 20th Congress of the International Ergonomics Association (IEA 2018) Volume V: Human Simulation and Virtual Environments, Work With Computing Systems (WWCS), Process Control 20*, pages 224–232. Springer, 2019.

[3] Hila Chefer, Sagie Benaim, Roni Paiss, and Lior Wolf. Image-based clip-guided essence transfer. In *European Conference on Computer Vision*, pages 695–711. Springer, 2022.

[4] Shu-Yu Chen, Wanchao Su, Lin Gao, Shihong Xia, and Hongbo Fu. Deepfacedrawing: Deep generation of face images from sketches. *ACM Transactions on Graphics (TOG)*, 39(4):72–1, 2020.

[5] Shu-Yu Chen, Feng-Lin Liu, Yu-Kun Lai, Paul L Rosin, Chunpeng Li, Hongbo Fu, and Lin Gao. Deepfaceediting: deep face generation and editing with disentangled geometry and appearance control. *ACM Transactions on Graphics (TOG)*, 40(4):1–15, 2021.

[6] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in neural information processing systems*, 29, 2016.

[7] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018.

[8] Edo Collins, Raja Bala, Bob Price, and Sabine Susstrunk. Editing in style: Uncovering the local semantics of gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5771–5780, 2020.

[9] 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.

[10] Yu Deng, Jiaolong Yang, Dong Chen, Fang Wen, and Xin Tong. Disentangled and controllable face image generation via 3d imitative-contrastive learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5154–5163, 2020.

[11] E Friesen and Paul Ekman. Facial action coding system: a technique for the measurement of facial movement. *Palo Alto*, 3(2):5, 1978.

[12] Yue Gao, Fangyun Wei, Jianmin Bao, Shuyang Gu, Dong Chen, Fang Wen, and Zhouhui Lian. High-fidelity and arbitrary face editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16115–16124, 2021.

[13] Shuyang Gu, Jianmin Bao, Hao Yang, Dong Chen, Fang Wen, and Lu Yuan. Mask-guided portrait editing with conditional gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3436–3445, 2019.

[14] Jingtao Guo, Zhenzhen Qian, Zuowei Zhou, and Yi Liu. Mulgan: Facial attribute editing by exemplar. *arXiv preprint arXiv:1912.12396*, 2019.

[15] Yuxuan Han, Jiaolong Yang, and Ying Fu. Disentangled face attribute editing via instance-aware latent space search. *IJCAI*, 2021.

[16] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. *Advances in neural information processing systems*, 33:9841–9850, 2020.

[17] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want. *IEEE transactions on image processing*, 28(11):5464–5478, 2019.

[18] Zhenliang He, Meina Kan, Jichao Zhang, and Shiguang Shan. Pa-gan: Progressive attention generative adversarial network for facial attribute editing. *arXiv preprint arXiv:2007.05892*, 2020.

[19] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[20] Xianxu Hou, Linlin Shen, Or Patashnik, Daniel Cohen-Or, and Hui Huang. Feat: Face editing with attention. *arXiv preprint arXiv:2202.02713*, 2022.

[21] Xianxu Hou, Xiaokang Zhang, Hanbang Liang, Linlin Shen, Zhihui Lai, and Jun Wan. Guidedstyle: Attribute knowledge guided style manipulation for semantic face editing. *Neural Networks*, 145:209–220, 2022.

[22] Ziqi Huang, Kelvin CK Chan, Yuming Jiang, and Ziwei Liu. Collaborative diffusion for multi-modal face generation and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6080–6090, 2023.

[23] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[24] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020.

[25] Siavash Khodadadeh, Shabnam Ghadar, Saeid Motiian, Wei-An Lin, Ladislau Bölöni, and Ratheesh Kalarot. Latent to latent: A learned mapper for identity preserving editing of multiple face attributes in stylegan-generated images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3184–3192, 2022.

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[27] Marek Kowalski, Stephan J Garbin, Virginia Estellers, Tadas Baltrušaitis, Matthew Johnson, and Jamie Shotton. Config: Controllable neural face image generation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 299–315. Springer, 2020.

[28] Jeong-gi Kwak, David K Han, and Hanseok Ko. Cafe-gan: Arbitrary face attribute editing with complementary attention feature. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 524–540. Springer, 2020.

[29] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5549–5558, 2020.

[30] Huan Ling, Karsten Kreis, Daiqing Li, Seung Wook Kim, Antonio Torralba, and Sanja Fidler. Editgan: High-precision semantic image editing. *Advances in Neural Information Processing Systems*, 34:16331–16345, 2021.

[31] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018):11, 2018.

[32] Yongyi Lu, Yu-Wing Tai, and Chi-Keung Tang. Attribute-guided face generation using conditional cyclegan. In *Proceedings of the European conference on computer vision (ECCV)*, pages 282–297, 2018.

[33] Safa C Medin, Bernhard Egger, Anoop Cherian, Ye Wang, Joshua B Tenenbaum, Xiaoming Liu, and Tim K Marks. Most-gan: 3d morphable stylegan for disentangled face image manipulation. In *Proceedings of the AAAI conference on artificial intelligence*, pages 1962–1971, 2022.

[34] Yongjie Niu, Mingquan Zhou, and Zhan Li. Disentangling the latent space of gans for semantic face editing. *Plos one*, 18(10):e0293496, 2023.

[35] Xingang Pan, Ayush Tewari, Thomas Leimkühler, Lingjie Liu, Abhimitra Meka, and Christian Theobalt. Drag your gan: Interactive point-based manipulation on the generative image manifold. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023.

[36] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of

stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021.

[37] Tiziano Portenier, Qiyang Hu, Attila Szabó, Siavash Arjomand Bigdeli, Paolo Favaro, and Matthias Zwicker. Faceshop: deep sketch-based face image editing. *ACM Transactions on Graphics (TOG)*, 37(4):1–13, 2018.

[38] Andrés Prados-Torreblanca, José M Buenaposada, and Luis Baumela. Shape preserving facial landmarks with graph attention networks. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*. BMVA Press, 2022.

[39] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[40] Valentin Schwind, Katrin Wolf, and Niels Henze. Facemaker—a procedural face generator to foster character design research. *Game Dynamics: Best Practices in Procedural and Dynamic Game Content Generation*, pages 95–113, 2017.

[41] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1532–1540, 2021.

[42] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9243–9252, 2020.

[43] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. Interfacegan: Interpreting the disentangled face representation learned by gans. *IEEE transactions on pattern analysis and machine intelligence*, 44(4):2004–2018, 2020.

[44] Linsen Song, Jie Cao, Lingxiao Song, Yibo Hu, and Ran He. Geometry-aware face completion and editing. In *Proceedings of the AAAI conference on artificial intelligence*, pages 2506–2513, 2019.

[45] Jianxin Sun, Qiyao Deng, Qi Li, Muyi Sun, Min Ren, and Zhenan Sun. Anyface: Free-style text-to-face synthesis and manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18687–18696, 2022.

[46] Jingxiang Sun, Xuan Wang, Yichun Shi, Lizhen Wang, Jue Wang, and Yebin Liu. Ide-3d: Interactive disentangled editing for high-resolution 3d-aware portrait synthesis. *ACM Transactions on Graphics (ToG)*, 41(6):1–10, 2022.

[47] Jingxiang Sun, Xuan Wang, Yong Zhang, Xiaoyu Li, Qi Zhang, Yebin Liu, and Jue Wang. Fenerf: Face editing in neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7672–7682, 2022.

[48] Qiushi Sun, Jingtao Guo, and Yi Liu. Pattgan: Pluralistic facial attribute editing. *IEEE Access*, 10:68534–68544, 2022.

[49] Ayush Tewari, Mohamed Elgharib, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhöfer, and Christian Theobalt. Pie: Portrait image embedding for semantic control. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020.

[50] Ayush Tewari, Mohamed Elgharib, Gaurav Bharaj, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhofer, and Christian Theobalt. Stylerig: Rigging stylegan for 3d control over portrait images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6142–6151, 2020.

[51] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4): 1–14, 2021.

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[53] Yi Wei, Zhe Gan, Wenbo Li, Siwei Lyu, Ming-Ching Chang, Lei Zhang, Jianfeng Gao, and Pengchuan Zhang. Maggan: High-resolution face attribute editing with mask-guided generative adversarial network. In *Proceedings of the Asian Conference on Computer Vision*, 2020.

[54] Wayne Wu, Chen Qian, Shuo Yang, Quan Wang, Yici Cai, and Qiang Zhou. Look at boundary: A boundary-aware face alignment algorithm. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2129–2138, 2018.

[55] Zongze Wu, Dani Lischinski, and Eli Shechtman. Stylespace analysis: Disentangled controls for stylegan image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12863–12872, 2021.

[56] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2256–2265, 2021.

[57] Taihong Xiao, Jiapeng Hong, and Jinwen Ma. Elegant: Exchanging latent encodings with gan for transferring multiple face attributes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 168–184, 2018.

[58] Guoxing Yang, Nanyi Fei, Mingyu Ding, Guangzhen Liu, Zhiwu Lu, and Tao Xiang. L2m-gan: Learning to manipulate latent space semantics for facial attribute editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2951–2960, 2021.

[59] Huiting Yang, Liangyu Chai, Qiang Wen, Shuang Zhao, Zixun Sun, and Shengfeng He. Discovering interpretable latent space directions of gans beyond binary attributes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12177–12185, 2021.

[60] Xu Yao, Alasdair Newson, Yann Gousseau, and Pierre Hellier. A latent transformer for disentangled face editing in images and videos. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13789–13798, 2021.

[61] Andreas Zankl, Lukas Eberle, Luciano Molinari, and Albert Schinzel. Growth charts for nose length, nasal protrusion,

and philtrum length from birth to 97 years. *American journal of medical genetics*, 111(4):388–391, 2002.

[62] Gang Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Generative adversarial network with spatial attention for face attribute editing. In *Proceedings of the European conference on computer vision (ECCV)*, pages 417–432, 2018.

[63] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.