Region-Based Semantic Factorization in GANs

Jiapeng Zhu [1] **Yujun Shen** [2] **Yinghao Xu** [3] **Deli Zhao** [4] **Qifeng Chen** [1]

Abstract

Despite the rapid advancement of semantic discovery in the latent space of Generative Adversarial Networks (GANs), existing approaches either are limited to finding global attributes or rely on a number of segmentation masks to identify local attributes. In this work, we present a highly efficient algorithm to factorize the latent semantics learned by GANs concerning an *arbitrary* image region. Concretely, we revisit the task of local manipulation with pre-trained GANs and formulate region-based semantic discovery as a dual optimization problem. Through an appropriately defined generalized Rayleigh quotient, we manage to solve such a problem *without any annotations or training*. Experimental results on various state-of-the-art GAN models demonstrate the effectiveness of our approach, as well as its superiority over prior arts regarding precise control, region robustness, speed of implementation, and simplicity of use. Our source code can be found at here.

1. Introduction

Recent studies have shown that versatile semantics emerge in the latent space of pre-trained Generative Adversarial Networks (GANs) (Goetschalckx et al., 2019; Shen et al., 2020a; Jahanian et al., 2020; Yang et al., 2021b). Identifying these variation factors, which are typically devised as some directions in the latent spaces (Shen et al., 2020a), facilitates a wide range of downstream tasks (Xu et al., 2021; Zhang et al., 2021b; Tan et al., 2020), especially image editing (Jahanian et al., 2020; Yang et al., 2021b; Menon et al., 2020; Gu et al., 2020; Zhu et al., 2020; Ling et al., 2021). In particular, moving latent codes along a certain direction can cause corresponding semantic changes in the synthesized images. Accordingly, it is of great use to discover these steerable directions diversely and precisely.

To interpret the latent space learned by GANs, many attempts have been made, including both supervised ones (Shen et al., 2020b; Jahanian et al., 2020; Yang et al., 2021b) and unsupervised ones (Shen & Zhou, 2021; Voynov & Babenko, 2020; Härkönen et al., 2020). Most prior arts, however, target at finding global attributes (Shen et al., 2020b; Yang et al., 2021b; Voynov & Babenko, 2020; Härkönen et al., 2020) such that altering the latent code with these attributes will manipulate the output image as a whole. Researchers have given recent attention to detecting local semantics due to their more practical usage, but they usually require a number of images labeled with segmentation masks for the discovery process (Suzuki et al., 2018; Collins et al., 2020; Wu et al., 2021; Ling et al., 2021). A very recent work manages to relate a local image region to a GAN latent subspace independent of annotations (Zhu et al., 2021), yet it turns to depend on some sensitive hyper-parameters, resulting in insufficient robustness to the selected region.

In this work, we propose a surprisingly simple algorithm, termed as ReSeFa, for region-based semantic factorization in GANs. Unlike existing methods that only treat image editing as an application and take no account of the manipulation model for semantic exploration, we re-examine the task of local editing using pre-trained GANs as the prior. Specifically, given a region of interest, a robust manipulation method should take effect on the contents within this area only and preserve the remaining contents as much as possible. In other words, after altering the latent code, we expect the pixels located in the target region to change while the outside pixels remain the same. Such an analysis helps define an optimization problem based on the derivative of pixel values with respect to the latent code (*i.e.*, Jacobian). Solving this problem can help identify the variation factors corresponding to a particular image region. We further notice that the optimization objective can be formulated as a generalized Rayleigh quotient (Horn & Johnson, 2012) such that the aforementioned problem can be solved efficiently.

To summarize, our proposed algorithm has the following advantages over prior work. First, our algorithm does not

[1]Department of CSE, The Hong Kong University of Science and Technology, Hong Kong, China. [2]ByteDance, Beijing, China [3]Department of IE, The Chinese University of Hong Kong, Hong Kong, China. [4]Ant Research, Hangzhou, China. Correspondence to: Qifeng Chen <cqf@ust.hk>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

rely on detailed spatial masks. Taking mouth editing with a face synthesis GAN model as an instance, ReSeFa only needs a rough bounding box around the mouth of a single synthesized image, making it sufficiently easy to use in practice. Second, our method is purely based on solving an eigen-decomposition problem, which is independent of any hyper-parameters or model structures. Consequently, it is fairly flexible so that users can customize the regions of their interests arbitrarily with any pre-trained GAN model. Third, thanks to the adequately defined generalized Rayleigh quotient, our approach enables a fast implementation, especially when the latent space is in high dimensions (*e.g.*, $\mathcal{W}^+$ space of StyleGAN (Abdal et al., 2020)). Extensive experimental results suggest that our ReSeFa shows precise controllability and strong robustness to the selected image local region, while it can be easily generalized to state-of-the-art GAN variants, including StyleGAN2 (Karras et al., 2020b) and BigGAN (Brock et al., 2019).

2. Related Work

Generative Adversarial Networks. GANs (Goodfellow et al., 2014) have significantly advanced high-fidelity image synthesis with different objective functions (Arjovsky et al., 2017), novel training schedules (Karras et al., 2018; Brock et al., 2019), carefully designed network architectures (Karras et al., 2019; 2020b; 2021), and improved data efficiency (Zhao et al., 2020; Karras et al., 2020a; Yang et al., 2021a). Through properly reusing the knowledge learned in the GAN pre-training, prior arts have demonstrated a wide range of downstream applications of GANs, such as image classification (Xu et al., 2021), image segmentation (Zhang et al., 2021b), visual alignment (Peebles et al., 2021), image editing (Gu et al., 2020; Menon et al., 2020), *etc.*

Local Editing with GANs. Among all the applications of GANs, image local editing earns a number of audiences considering its interactivity and practical usage. One straightforward way of controlling the synthesis of a certain image region is to make the GAN generator spatially aware during training (Lee et al., 2020; Kim et al., 2021). An alternative way is to first segment the synthesis results and then manipulate (*e.g.*, swap) the intermediate feature maps at the region of interest (Suzuki et al., 2018; Bau et al., 2020b; Collins et al., 2020; Zhang et al., 2021a). However, all these approaches tend to perform editing only from the instance level instead of the semantic level. Taking face local manipulation as an example, these methods are capable of harmonizing the eyes of one person to another (Lee et al., 2020; Kim et al., 2021; Suzuki et al., 2018; Collins et al., 2020) yet fail to make a person close the eyes. Meanwhile, they require users to specify spatial masks for each editing (*e.g.*, the eyes of a person may not always locate at the same spatial position in different images), making them hard to generalize to all samples.

Semantic Discovery in GANs. Interpreting the generation mechanism of GANs helps us understand the rules about how the generator renders an image. In this way, we can utilize such rules for image editing once for all (Bau et al., 2019; 2020a). A typical way to control the GAN generation is to identify some steerable directions within the latent space (Jahanian et al., 2020). These latent directions usually correspond to some high-level semantics, like the age of a person, and can be faithfully used for attribute manipulation of any synthesized image (Goetschalckx et al., 2019; Plumerault et al., 2020; Shen et al., 2020b; Yang et al., 2021b; Voynov & Babenko, 2020; Härkönen et al., 2020; Shen & Zhou, 2021; Spingarn-Eliezer et al., 2021; Cherepkov et al., 2021; He et al., 2021). Nevertheless, most directions found by previous methods are targeted at the entire image, and how to discover the semantics for some image regions remains unsolved.

It has recently been shown that some latent subspaces of GANs can be directly used for image local editing without operating the feature maps (Wu et al., 2021; Lang et al., 2021; Zhu et al., 2021; Ling et al., 2021). Wu et al. (2021) propose StyleSpace, which uncovers the relationship between some convolutional units in the generator and the objects within the output image, however, identifying the object-oriented channels requires a number of object masks as the ground-truth and is only applicable to style-based network structure (Karras et al., 2019). Ling et al. (2021) propose a novel local editing approach by controlling the segmentation mask. However, the manipulation pipeline requires manually editing the segmentation mask, which needs skilled personnel and precise ground truth, and requires optimizing the latent code to meet the expected semantic change, which can be time-consuming. In addition, using a segmentation mask for semantic discovery would fail to find appearance-related attributes. Zhu et al. (2021) propose low-rank subspaces in GANs for image local editing, but the discovery of these subspaces relies on low-rank factorization with a relaxation factor. Such a hyper-parameter turns out to be sensitive to the model structure and the selected local region, and an inadequate value may lead to unsatisfying manipulation results.

Different from existing methods, our algorithm has the following ***advantages:*** (1) Our method is based on derivative (Ramesh et al., 2019; Chiu et al., 2020; Wang & Ponce, 2021) and has no requirements on the model structure as long as it is differentiable. Hence, unlike some approaches that are particularly designed for StyleGAN (Wu et al., 2021; Ling et al., 2021), ReSeFa can be easily generalized to different GAN variants. (2) Our method can be directly solved by maximizing a properly defined generalized Rayleigh quotient, making it independent of any

annotations, hyper-parameters, or training. Such a robust formulation also enables fast implementation, significantly outperforming other alternatives. (3) Our approach enables more precise local control, which will be verified in the experiment section.

3. Methodology

In this section, we introduce our proposed method for region-based semantic factorization in GANs. As mentioned above, we revisit the task of local editing with pre-trained GANs and takes the manipulation model into account for identifying the variation factors regarding a particular image region. Base on our analysis, the semantic discovery process can be formulated as an optimization problem, whose objective happens to be a well-defined generalized Rayleigh quotient. Consequently, such a problem holds a super efficient solver.

3.1. Manipulation Model with GAN Priors

Using prior knowledge learned by GANs for image editing has been widely explored (Shen et al., 2020b; Jahanian et al., 2020; Yang et al., 2021b). Concretely, given a well-trained generator $G(\cdot)$ that maps the latent space $\mathcal{Z}$ to the image space $\mathcal{X}$, we would like to find a latent direction $\boldsymbol{n} \in \mathcal{Z}$ such that altering a latent code $\boldsymbol{z}$ through the direction can cause the corresponding semantic change in the output image $\boldsymbol{x} = G(\boldsymbol{z})$. Such a process can be formulated as

$$\texttt{edit}(\boldsymbol{x}) \triangleq \boldsymbol{x}^{\text{edit}} = G(\boldsymbol{z} + \alpha \boldsymbol{n}), \tag{1}$$

where α indicates the degree of editing. Here, $\boldsymbol{n}$ is usually assumed to be a unit vector (Shen & Zhou, 2021), *i.e.*, $\boldsymbol{n}^T \boldsymbol{n} = 1$.

3.2. Region-based Semantic Discovery

As shown in Equation (1), there is no explicit constraint on the relationship between $\boldsymbol{x}$ and $\boldsymbol{x}^{\text{edit}}$, hence the entire image may get changed, resulting in a global editing. However, for the case of local editing, we would like to only change the image content within a certain region, denoted as $\boldsymbol{x}_f$, while the surroundings, denoted as $\boldsymbol{x}_b$, keep untouched. Here, $\boldsymbol{x}_f$ and $\boldsymbol{x}_b$ form a partition of all pixels within $\boldsymbol{x}$, *i.e.*, $\boldsymbol{x}_f \cup \boldsymbol{x}_b = \boldsymbol{x}$ and $\boldsymbol{x}_f \cap \boldsymbol{x}_b = \varnothing$, where the subscripts f and b are short for "foreground" and "background" respectively. Accordingly, we reformulate Equation (1) to fit the local editing task as

$$\begin{cases} \boldsymbol{x}^{\text{edit}} = G(\boldsymbol{z} + \alpha \boldsymbol{n}), \\ \text{s.t. } ||\boldsymbol{x}_b^{\text{edit}} - \boldsymbol{x}_b||_2^2 = 0, \end{cases} \tag{2}$$

where $||\cdot||_2$ denotes the ℓ_2 norm.

Based on the local manipulation model described in Equation (2), the constraint $||\boldsymbol{x}_b^{\text{edit}} - \boldsymbol{x}_b||_2^2$ can be approximated using the first-order Taylor expansion

$$\begin{aligned} ||\boldsymbol{x}_b^{\text{edit}} - \boldsymbol{x}_b||_2^2 &= ||\{G(\boldsymbol{z} + \alpha \boldsymbol{n})\}_b - \{G(\boldsymbol{z})\}_b||_2^2 \\ &\approx \alpha^2 \boldsymbol{n}^T \boldsymbol{J}_b^T \boldsymbol{J}_b \boldsymbol{n}. \end{aligned} \tag{3}$$

Here, $\boldsymbol{J}_b$ is the derivative of pixel values with respect to the latent code (*i.e.*, Jacobian) (Ramesh et al., 2019; Zhu et al., 2021). Particularly, we have

$$(\boldsymbol{J}_b)_{j,k} = \frac{\partial \{G(\boldsymbol{z})\}_j}{\partial \boldsymbol{z}_k}, \tag{4}$$

where j stands for a pixel position within the image, and k stands for a dimension of the latent space $\mathcal{Z}$. In practice, $\boldsymbol{J}_b$ can be easily computed as long as the generator $G(\cdot)$ is differentiable, regardless of its architecture.

In other words, Equation (3) can be used to characterize the variation of the background region when altering the latent code $\boldsymbol{z}$ towards the latent direction $\boldsymbol{n}$. Similarly, we can also measure the foreground change with

$$||\boldsymbol{x}_f^{\text{edit}} - \boldsymbol{x}_f||_2^2 \approx \alpha^2 \boldsymbol{n}^T \boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n}. \tag{5}$$

We argue that, given any arbitrary pixel partition $\{\boldsymbol{x}_f, \boldsymbol{x}_b\}$, an adequate local editing should take sufficient effect on the pixels within the region of interest (Shen & Zhou, 2021), $\boldsymbol{x}_f$, yet maintain the remaining pixel values, $\boldsymbol{x}_b$. Therefore, we are able to factorize the region-based semantics via solving the following optimization problem

$$\begin{cases} \arg\max_{\boldsymbol{n}} \ \boldsymbol{n}^T \boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n}, \\ \arg\min_{\boldsymbol{n}} \ \boldsymbol{n}^T \boldsymbol{J}_b^T \boldsymbol{J}_b \boldsymbol{n}. \end{cases} \tag{6}$$

3.3. Computational Solution

Reformulation. The problem defined in Equation (6) can be solved via convex optimization (Boyd & Vandenberghe, 2004). Alternatively, we unify this dual-objective optimization into a single criterion as

$$\arg\max_{\boldsymbol{n}} \frac{\boldsymbol{n}^T \boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n}}{\boldsymbol{n}^T \boldsymbol{J}_b^T \boldsymbol{J}_b \boldsymbol{n}}, \tag{7}$$

where the new object happens to be a generalized Rayleigh quotient (Horn & Johnson, 2012), $R(\boldsymbol{J}_f, \boldsymbol{J}_b, \boldsymbol{n})$. Equation (7) can be solved as a generalized eigenvalue problem

$$\boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n} = \lambda \boldsymbol{J}_b^T \boldsymbol{J}_b \boldsymbol{n}, \tag{8}$$

where those eigenvectors $\boldsymbol{n}$ corresponding to the largest eigenvalues λ are just the local semantics we expect.

Standard solution. To solve Equation (8), a standard solution is to perform Cholesky decomposition (Higham, 2002) on $\boldsymbol{J}_b^T \boldsymbol{J}_b$ as $\boldsymbol{J}_b^T \boldsymbol{J}_b = \boldsymbol{L}\boldsymbol{L}^T$. $\boldsymbol{L}$ is a lower triangular

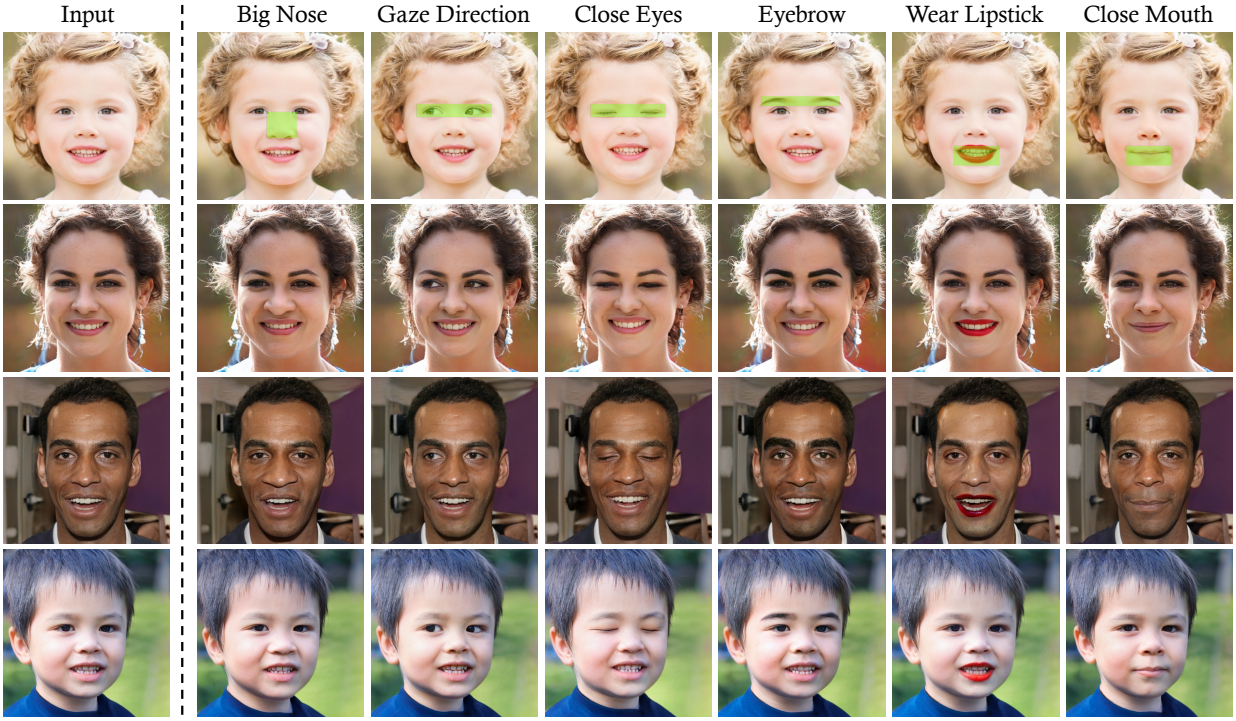

Figure 1. **Precise local editing results** achieved by our ReSeFa on the StyleGAN2 generator (Karras et al., 2020b) trained on FFHQ dataset (Karras et al., 2019). The regions of interest are highlighted with **green** boxes in the first row, while all rows share the same latent directions found by solving Equation (11). Note that our algorithm does *not* require the region masks to be precise, and can identify diverse semantics (*both appearance and shape*) corresponding the same region, like "wearing lipstick" and "closing mouth" for mouth.

matrix with real diagonal entries due to the symmetry of $\boldsymbol{J}_b^T\boldsymbol{J}_b$. Let $\tilde{\boldsymbol{n}} = \boldsymbol{L}^T\boldsymbol{n}$. Equation (8) can be reorganized as

$$\boldsymbol{L}^{-1}\boldsymbol{J}_f^T\boldsymbol{J}_f(\boldsymbol{L}^{-1})^T\tilde{\boldsymbol{n}} = \lambda\tilde{\boldsymbol{n}}, \tag{9}$$

which can be easily solved by performing eigen decomposition on $\boldsymbol{L}^{-1}\boldsymbol{J}_f^T\boldsymbol{J}_f(\boldsymbol{L}^{-1})^T$.

Handling singular case. However, in real cases, there is no guarantee that $\boldsymbol{L}$ is invertible. Recall that the pixel partition $\{\boldsymbol{x}_f, \boldsymbol{x}_b\}$ can be arbitrary, making it possible that the variation factors regarding region $\boldsymbol{x}_b$ are limited. In other words, $\boldsymbol{J}_b^T\boldsymbol{J}_b$ can be rank-deficient. To handle such a case, we make a slight modification on $\boldsymbol{J}_b^T\boldsymbol{J}_b$ to make it non-singular as

$$\boldsymbol{J}_b^T\boldsymbol{J}_b \leftarrow \boldsymbol{J}_b^T\boldsymbol{J}_b + \tau\text{tr}(\boldsymbol{J}_b^T\boldsymbol{J}_b)\boldsymbol{I}, \tag{10}$$

where $\boldsymbol{I}$ is the identity matrix and $\text{tr}(\cdot)$ denotes the trace. $\tau = 1e^{-3}$ is a small scaling factor. For simplicity, we set $a = \tau\text{tr}(\boldsymbol{J}_b^T\boldsymbol{J}_b)$. Now, Equation (8) can be converted to

$$(\boldsymbol{J}_b^T\boldsymbol{J}_b + a\boldsymbol{I})^{-1}\boldsymbol{J}_f^T\boldsymbol{J}_f\boldsymbol{n} = \lambda\boldsymbol{n}. \tag{11}$$

Fast implementation. Even though Equation (11) has provided an elegant formulation for region-based semantic factorization, solving it can be time consuming especially when the latent space $\mathcal{Z}$ is with extremely high dimensions. For example, the $\mathcal{W}^+$ space (Abdal et al., 2020) for a StyleGAN (Karras et al., 2019) generator with 18 layers is $512\times18 = 9216$ dimensional. To speed up the factorization process, we provide an efficient scheme by virtue of the Sherman-Morrison-Woodbury formula (Higham, 2002). To be specific, let $\boldsymbol{J}_b = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$ be the truncated Singular Value Decomposition (SVD) (Horn & Johnson, 2012) of $\boldsymbol{J}_b$, where $\boldsymbol{U}$, $\boldsymbol{V}$, $\boldsymbol{D}$ are the left singular matrix, the right singular matrix, and a diagonal matrix respectively. $\boldsymbol{D}$ is of size $r \times r$, where r is the rank of $\boldsymbol{J}_b$. Then we can derive

$$\begin{aligned}
&(\boldsymbol{J}_b^T\boldsymbol{J}_b + a\boldsymbol{I})^{-1} = (\boldsymbol{V}\boldsymbol{D}(\boldsymbol{V}\boldsymbol{D})^T + a\boldsymbol{I})^{-1} \\
&= a\boldsymbol{I} - \boldsymbol{V}\boldsymbol{D}(a\boldsymbol{I} + (\boldsymbol{V}\boldsymbol{D})^T\boldsymbol{V}\boldsymbol{D})^{-1}(\boldsymbol{V}\boldsymbol{D})^T \\
&= a\boldsymbol{I} - \boldsymbol{V}\boldsymbol{D}(a\boldsymbol{I} + \boldsymbol{D}^2)^{-1}\boldsymbol{D}\boldsymbol{V}^T \\
&= a\boldsymbol{I} - \boldsymbol{V}\tilde{\boldsymbol{D}}\boldsymbol{V}^T,
\end{aligned} \tag{12}$$

where $\tilde{\boldsymbol{D}} = \boldsymbol{D}(a\boldsymbol{I} + \boldsymbol{D}^2)^{-1}\boldsymbol{D}$ is reduced to the diagonal-wise operation, which is far more efficient.

Summary. With the above analysis, given a generator $G(\cdot)$ and a partition $\{\boldsymbol{x}_f, \boldsymbol{x}_b\}$, the semantic vectors related to region $\boldsymbol{x}_f$ can be efficiently obtained by solving Equation (11) using Equation (12) as an intermediate step.

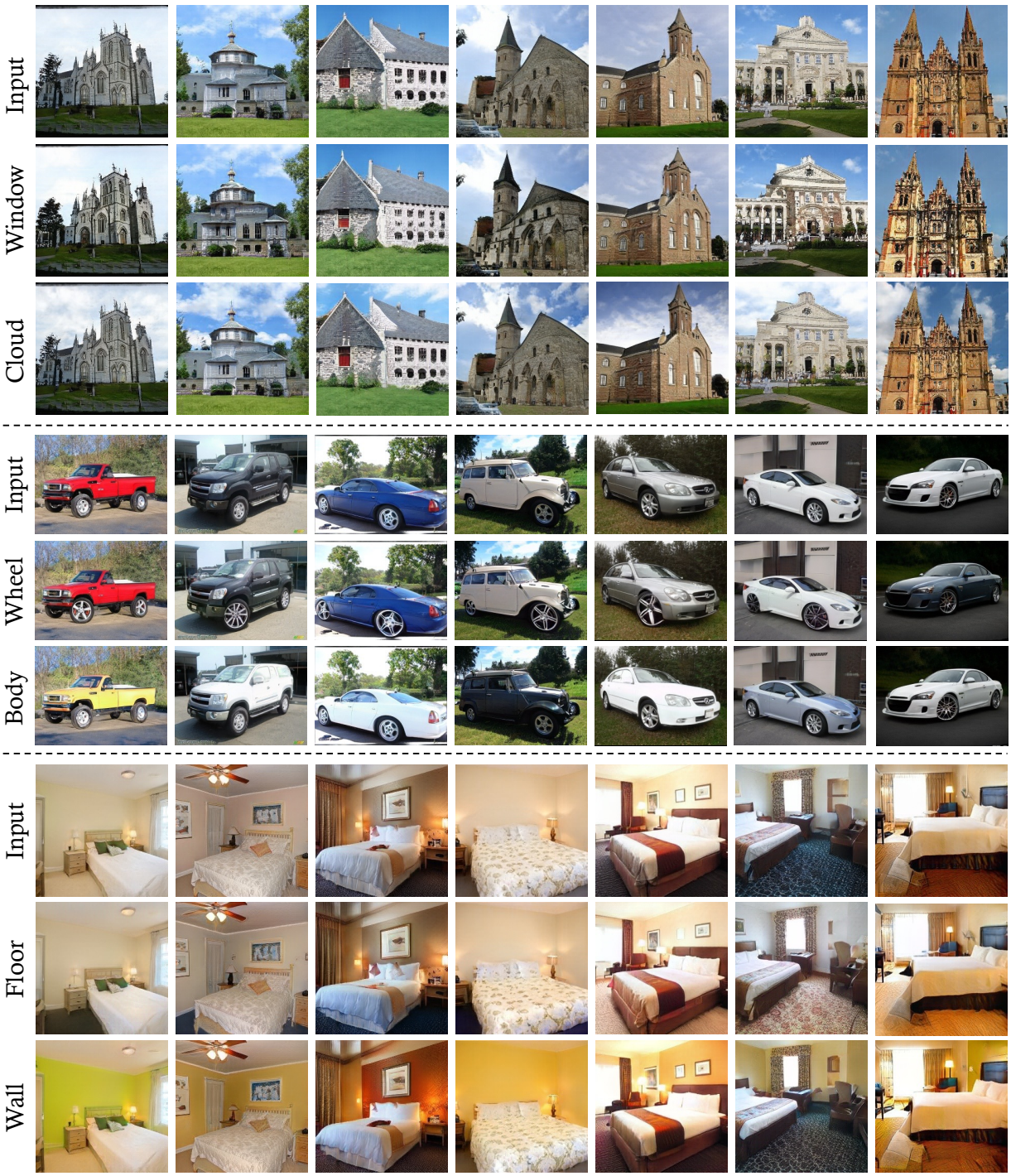

Figure 2. **Versatile local semantics** found by our algorithm using the StyleGAN2 models (Karras et al., 2020b) trained on various datasets, including LSUN churches (indoor scene), LSUN cars (general object), and LSUN bedrooms (indoor scene) (Yu et al., 2015).

4. Experiments

4.1. Experimental Setup

We conduct extensive experiments to evaluate our proposed method, mainly on two types of models, *i.e.*, StyleGAN2 (Karras et al., 2020b) and BigGAN (Brock et al., 2019). And the datasets we use are diverse, including FFHQ (Karras et al., 2019), LSUN bedroom, church, car (Yu et al., 2015), and ImageNet (Deng et al., 2009). For StyleGAN2, we use the models released by the authors, and for BigGAN, we use the model from TensorFlow Hub. For metrics, we use Fréchet Inception Distance (FID) (Heusel

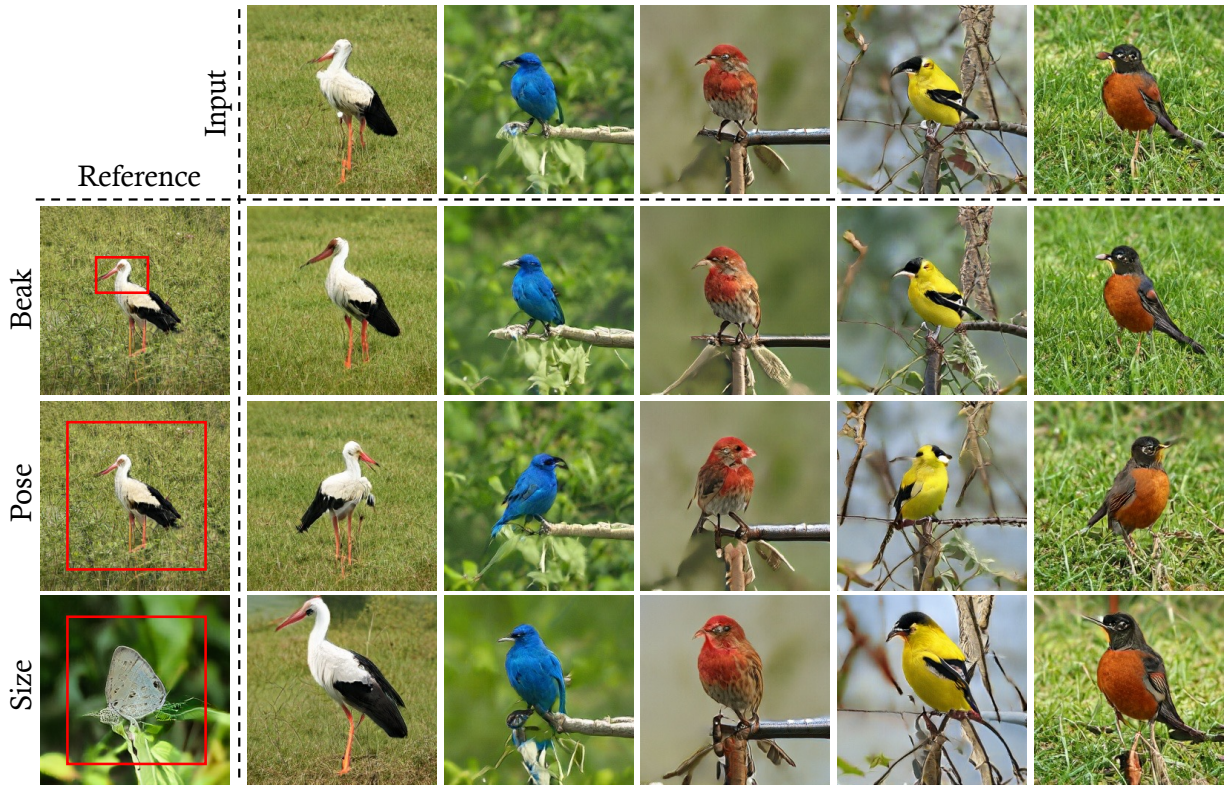

Figure 3. **Precise local editing on a conditional generative model**, *i.e.*, BigGAN (Brock et al., 2019), where we can conclude three observations. First, our algorithm is able to control the synthesis of only a part of the object with a small region of interest (*e.g.*, "beak" in the second row), or control the synthesis of the entire object with a large region of interest (*e.g.*, "pose" in the third row). Second, when altering the pose of birds in the third row, the background (*e.g.*, the grass in the second column and the branch in the third column) are barely affected, demonstrating the *precise control* achieved by our method. Third, the semantics found from one category can be convincingly applied to other categories.

et al., 2017), masked Mean Squared Error (MSE), and Identity loss (ID). FID is used to evaluate the image fidelity after editing. For the masked MSE, we used it to qualify the editing precision. Specifically, we can evaluate the change in the edited region or the rest region using a mask. We use the ArcFace model to evaluate identity similarity between the edited images and the original images. The experiments are organized as follows. First, we demonstrate that our method could easily find semantically meaningful directions when given a specific region of a generated image on various datasets in Section 4.2. Second, we compare our method with the existing methods in Section 4.3 and demonstrate that our methods have strong control over the local region of the synthesized images even when editing in the latent space. All the experiments are conducted on a single RTX 2080 Ti GPU.

4.2. Local Semantic Discovery and Manipulation

Recall that our method is rather simple and can be divided into three steps. First, we need to compute the Jacobian of a synthesized image to the latent code. Second, obtaining $\boldsymbol{J}_f$ and $\boldsymbol{J}_b$ according to the masked region (By default, in each figure, the masked region and the remaining part are used to compute $\boldsymbol{J}_f$ and $\boldsymbol{J}_b$, respectively.). Third, solving Equation (8) or Equation (11) to get the attribute vectors, which are used to edit the images. We first conduct experiments on the official FFHQ 1024×1024 model released in StyleGAN2. The eyes, eyebrows, mouth, and nose are chosen as the local regions to factorize their semantics. As shown in Figure 1, our method could uncover the semantics that enables fine-grained and precise local control over the synthesized images. For example, when factorizing the semantics in the nose-related region, we could find an attribute that changes the size of the nose. When the region is taken from the eyes, as the green masks are shown in the third and fourth columns in Figure 1, several eye-related semantics could be found, such as look askance, close eyes, etc. We could manipulate the eyebrows, as the fifth column shows when the region is chosen as the eyebrows. The last two columns show the found semantics when given the mouth region. For instance, the penultimate column shows that all the images have lipstick regarding their gender. The last column shows that semantics found in this region can close the mouth of the images. More

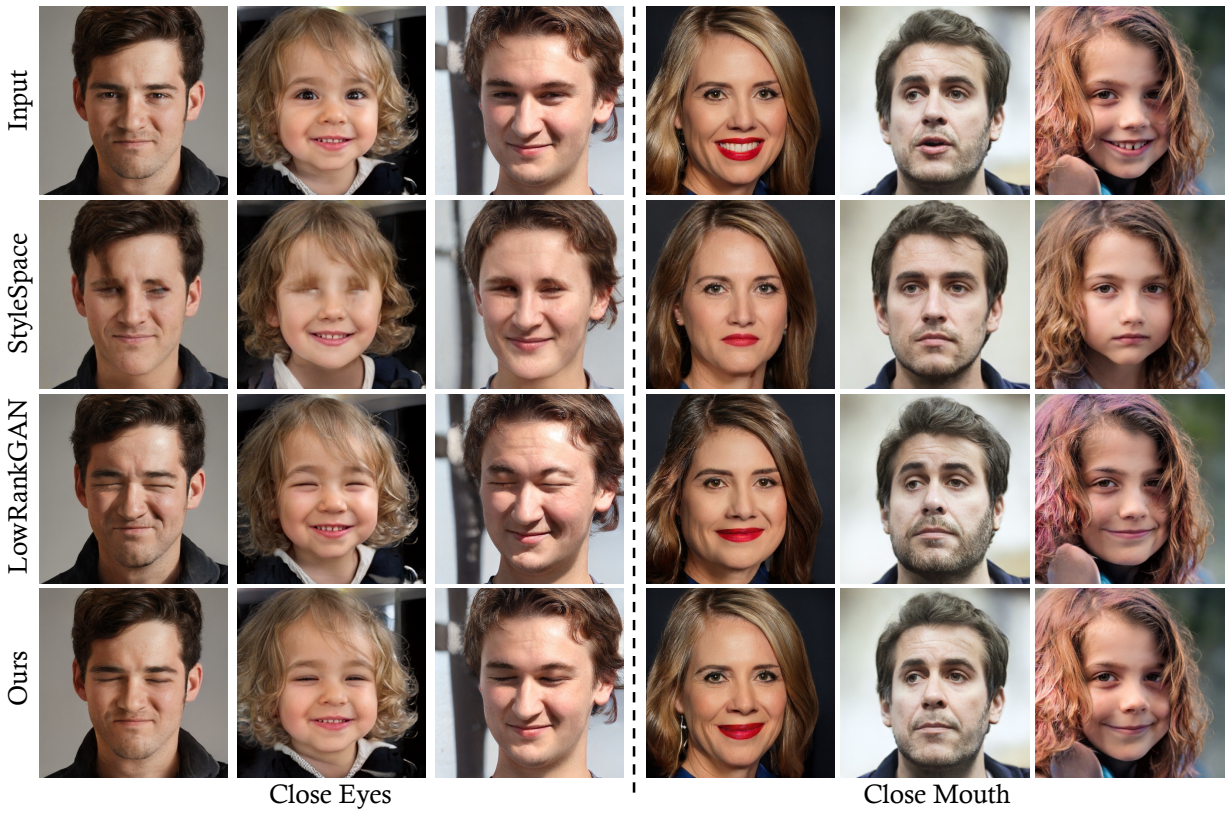

Figure 4. **Qualitative comparison** between different local editing approaches on face synthesis model (Karras et al., 2020b), including StyleSpace (Wu et al., 2021), LowRankGAN (Zhu et al., 2021), and ReSeFa. Our approach produces more realistic results and suggests more precise controllability in maintaining the image contents beyond the region of interests, *i.e.*, eyes and mouth.

semantic meaningful edit results on each face region can be found in Appendix C.

Besides the face model, we further validate our proposed method on other models. Figure 2 shows the results on LSUN-church, car, and bedroom, which demonstrates that our method could control either a large region of an image or a small area of an image. For the control on the small region, we could observe that the number of windows on the church are modified, and the wheel type of the car are changed as well. For the control on the large region, the cloud is added in the sky of the church, the color of the car body can be changed, and the color of the wall in bedroom is changed as well. In all, we could say that our proposed method perform well on StyleGAN2. For more results, please refer to the Appendix C.

For BigGAN, the most commonly changed attributes are the size, pose, and background color of the objects (Plumerault et al., 2020; Jahanian et al., 2020; Voynov & Babenko, 2020), which always result in a global change of an image. Seldom had they shown the local control over the synthesized images. However, our proposed method not only could control the size or pose of the object, but also could control over a small area of the synthetic images. Figure 3 displays the strong ability of our method to edit the local region. For example, the beak of these birds changed, which is a very small part of an image. Still, our method could edit them, which shows the great power of our method for local controllability. Except for the control on the small region, Figure 3 also demonstrates that our method could edit the attributes related to a large region as well. For instance, the size and pose can be varied successfully regrading the categories.

4.3. Comparison with Existing Alternatives

In this section, we compare our method with the state-of-the-art methods both qualitatively and quantitatively. We compare our method with StyleSpace (Wu et al., 2021) and LowRankGAN (Zhu et al., 2021), which are two state-of-the-art methods for local control on the generated images. In the main paper, we show the attributes of closing eyes and mouth, and for the comparison of the other attribute, please see the the Appendix C.

First, we compare with StyleSpace as shown in Figure 4, in which we could find that our method could achieve more photo-realistic results. For instance, the artifacts appeared when closing the eyes of the man, and the eyebrow

Table 1. **Quantitative comparison** between different local editing approaches on face synthesis model (Karras et al., 2020b), including StyleSpace (Wu et al., 2021), LowRankGAN (Zhu et al., 2021), and ReSeFa. We use FID (lower is better) to evaluate the image quality after editing, MSE (lower is better, scaled by $1e^4$ for good readability) to calculate the value change of pixels outside the region of interest, and ID similarity (higher is better) to measure the identity change before and after manipulation. The interpretation speed using each method is also reported in the last column.

	Close Eyes			Close Mouth			With Lipstick			Big Nose			
Method	**FID↓**	**MSE↓**	**ID↑**	**FID↓**	**MSE↓**	**ID↑**	**FID↓**	**MSE↓**	**ID↑**	**FID↓**	**MSE↓**	**ID↑**	**Speed**
StyleSpace	26.32	**2.31**	0.51	24.83	2.43	0.51	57.12	**0.63**	0.84	25.65	**0.98**	0.75	10.0s
LowRankGAN	25.43	5.61	0.53	24.91	4.96	0.73	**32.33**	8.44	0.63	25.37	5.52	0.56	393s
ReSeFa (Ours)	**24.40**	2.51	**0.83**	**23.35**	**2.11**	**0.85**	38.41	1.53	**0.89**	**24.82**	1.64	**0.85**	0.5s

disappeared simultaneously. The hair and background of the second image are changing as well. Sometimes, it is hard to close the images' eyes, as the second column shows. When closing the mouth, the jaw becomes smaller in all these three images, and the beard of the man decreases as well. On the contrary, the editing results are significantly improved when using our method.

Second, we compared our method with recently proposed LowRankGAN (Zhu et al., 2021). As shown in Figure 4, we could see that both the LowRankGAN and our method could successfully close the eyes or mouth of the images. Nevertheless, there are some differences. For one thing, when closing the eyes of the first and third images, the brightness changes, the face of the second image is smaller after editing. Instead, our method could well preserve these changes. For another, when closing the mouth, the jaw of the first woman is widened, the beard of the man is increased, and the hair color and the background of the girl are changed. Again, our method could well preserve these changes. We draw from the above experiment that our method has a stronger ability to precisely manipulate the local regions than those two baselines.

We also give the quantitative comparison results in Table 1 and Figure 5. Table 1 reports quantitative results on different attributes. As shown, our method could get the best identity similarity (ID) after editing for all these attributes. Regrading the speed to find the semantics in a specific region, we also report the time of discovery using different methods in Table 1 (The time). It can be observed that our ReSeFa owns the fastest implementation, thanks to the analysis on the local manipulation model. Also, it is worth noting that our algorithm does not require any annotations like StyleSpace (Wu et al., 2021).

Figure 5 gives the MSE on both the edited region and the remaining region when gradually increasing the manipulation strength (*i.e.*, α in Equation (1)) on closing mouth or eyes. Figure 5a shows that the MSE of StyleSpace is larger than the other two methods in the mouth region, presumably because the jaw has a large shift when closing the mouth,

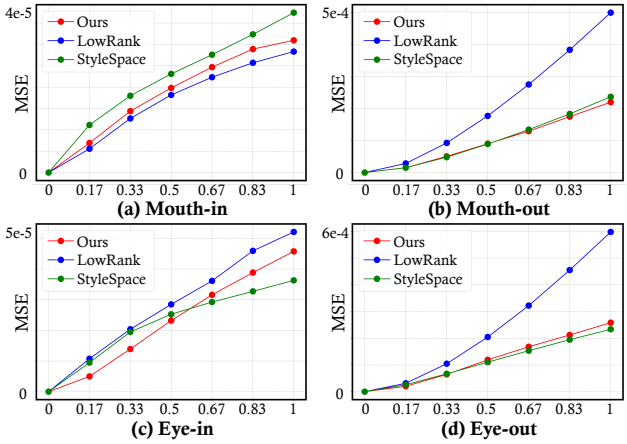

Figure 5. **Quantitative results of pixel change** by *gradually* making people close mouth and eyes using StyleSpace (Wu et al., 2021), LowRankGAN (Zhu et al., 2021), and ReSeFa. "in" and "out" refer to region of interest and its surroundings, respectively. A higher "in" change and a lower "out" change are expected for a promising local editing.

as shown in Figure 4. Figure 5b shows that the MSE of LowRankGAN is far better than the other two methods. When it comes to Figure 5c and Figure 5d, the MSE of StyleSpace in the eye region is small, while LowRankGAN gets the largest MSE again. Hence, our method could better control the specific region since the change in the region is large while the change in the remaining region is small compared to the state-of-the-art methods.

4.4. Discussion

We have demonstrated the impressive ability of our method in local control, but there are still some limitations. For example, our approach would fail to control extremely small regions embedded in large area of holistically uniform textures (*e.g.*, one tooth), or regions with multiple similar components in the image (*e.g.*, a single nostril) very well. It also shares a common limitation as existing methods in editing only one object of a symmetric pair (*e.g.*, one eye of a human face). Meanwhile, it is hard to discover the

global semantic directions in StyleGAN (*e.g.*, face pose), even choosing a sufficiently large region. Future research will focus on how to generalize our region-based semantic factorization on more fine-grained regions as well as how to unify global and local semantic exploration within the same algorithm.

5. Conclusion

In this work, we propose a simple algorithm to factorize the semantics learned by GANs regarding some particular regions. We re-examine the task of local editing and take the manipulation model into account for semantic discovery. By appropriately formulating this process as a dual-objective optimization problem, we enable an efficient and robust algorithm to find region-based variation factors without relying on any annotations or training.

References

Abdal, R., Qin, Y., and Wonka, P. Image2stylegan++: How to edit the embedded images? In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2, 4

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *ICML*, 2017. 2

Bau, D., Zhu, J.-Y., Strobelt, H., Zhou, B., Tenenbaum, J. B., Freeman, W. T., and Torralba, A. GAN dissection: Visualizing and understanding generative adversarial networks. In *Int. Conf. Learn. Represent.*, 2019. 2

Bau, D., Liu, S., Wang, T., Zhu, J.-Y., and Torralba, A. Rewriting a deep generative model. In *Eur. Conf. Comput. Vis.*, 2020a. 2

Bau, D., Strobelt, H., Peebles, W., Wulff, J., Zhou, B., Zhu, J.-Y., and Torralba, A. Semantic photo manipulation with a generative image prior. *ACM Trans. Graph.*, 2020b. 2

Boyd, S. and Vandenberghe, L. *Convex Optimization*. Cambridge University Press, 2004. 3, 12

Brock, A., Donahue, J., and Simonyan, K. Large scale GAN training for high fidelity natural image synthesis. In *Int. Conf. Learn. Represent.*, 2019. 2, 5, 6

Cherepkov, A., Voynov, A., and Babenko, A. Navigating the GAN parameter space for semantic image editing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

Chiu, C.-H., Koyama, Y., Lai, Y., Igarashi, T., and Yue, Y. Human-in-the-loop differential subspace search in high-dimensional latent space. *ACM Trans. Graph.*, 2020. 2

Collins, E., Bala, R., Price, B., and Susstrunk, S. Editing in style: Uncovering the local semantics of GANs. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1, 2

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009. 5

Ghojogh, B., Karray, F., and Crowley, M. Eigenvalue and generalized eigenvalue problems: Tutorial, 2019. 12

Goetschalckx, L., Andonian, A., Oliva, A., and Isola, P. GANalyze: Toward visual definitions of cognitive image properties. In *Int. Conf. Comput. Vis.*, 2019. 1, 2

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. In *Adv. Neural Inform. Process. Syst.*, 2014. 2

Gu, J., Shen, Y., and Zhou, B. Image processing using multi-code gan prior. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1, 2

Härkönen, E., Hertzmann, A., Lehtinen, J., and Paris, S. GANSpace: Discovering interpretable GAN controls. In *Adv. Neural Inform. Process. Syst.*, 2020. 1, 2

He, Z., Kan, M., and Shan, S. EigenGAN: Layer-wise eigen-learning for GANs. In *Int. Conf. Comput. Vis.*, 2021. 2

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Adv. Neural Inform. Process. Syst.*, 2017. 5

Higham, N. J. *Accuracy and Stability of Numerical Algorithms*. SIAM, 2002. 3, 4

Horn, R. A. and Johnson, C. R. *Matrix Analysis*. Cambridge University Press, 2012. 1, 3, 4

Jahanian, A., Chai, L., and Isola, P. On the" steerability" of generative adversarial networks. In *Int. Conf. Learn. Represent.*, 2020. 1, 2, 3, 7

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of GANs for improved quality, stability, and variation. In *Int. Conf. Learn. Represent.*, 2018. 2

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2, 4, 5

Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., and Aila, T. Training generative adversarial networks with limited data. In *Adv. Neural Inform. Process. Syst.*, 2020a. 2

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. Analyzing and improving the image quality of StyleGAN. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020b. 2, 4, 5, 7, 8, 13, 14

Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., and Aila, T. Alias-free generative adversarial networks. In *Adv. Neural Inform. Process. Syst.*, 2021. 2

Kim, H., Choi, Y., Kim, J., Yoo, S., and Uh, Y. Exploiting spatial dimensions of latent in gan for real-time image editing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

Lang, O., Gandelsman, Y., Yarom, M., Wald, Y., Elidan, G., Hassidim, A., Freeman, W. T., Isola, P., Globerson, A., Irani, M., et al. Explaining in style: Training a gan to explain a classifier in stylespace. In *Int. Conf. Comput. Vis.*, 2021. 2

Lee, C.-H., Liu, Z., Wu, L., and Luo, P. Maskgan: Towards diverse and interactive facial image manipulation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

Ling, H., Kreis, K., Li, D., Kim, S. W., Torralba, A., and Fidler, S. EditGAN: High-precision semantic image editing. In *Adv. Neural Inform. Process. Syst.*, 2021. 1, 2

Menon, S., Damian, A., Hu, S., Ravi, N., and Rudin, C. Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1, 2

Peebles, W., Zhu, J.-Y., Zhang, R., Torralba, A., Efros, A., and Shechtman, E. Gan-supervised dense visual alignment. *arXiv preprint arXiv:2112.05143*, 2021. 2

Plumerault, A., Borgne, H. L., and Hudelot, C. Controlling generative models with continuous factors of variations. In *Int. Conf. Learn. Represent.*, 2020. 2, 7

Ramesh, A., Choi, Y., and LeCun, Y. A spectral regularizer for unsupervised disentanglement. In *Int. Conf. Mach. Learn.*, 2019. 2, 3

Shen, Y. and Zhou, B. Closed-form factorization of latent semantics in GANs. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2, 3

Shen, Y., Gu, J., Tang, X., and Zhou, B. Interpreting the latent space of GANs for semantic face editing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020a. 1

Shen, Y., Yang, C., Tang, X., and Zhou, B. InterFaceGAN: Interpreting the disentangled face representation learned by GANs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020b. 1, 2, 3

Spingarn-Eliezer, N., Banner, R., and Michaeli, T. GAN steerability without optimization. In *Int. Conf. Learn. Represent.*, 2021. 2

Suzuki, R., Koyama, M., Miyato, T., Yonetsuji, T., and Zhu, H. Spatially controllable image synthesis with internal representation collaging. *arXiv preprint arXiv:1811.10153*, 2018. 1, 2

Tan, S., Shen, Y., and Zhou, B. Improving the fairness of deep generative models without retraining. *arXiv preprint arXiv:2012.04842*, 2020. 1

Voynov, A. and Babenko, A. Unsupervised discovery of interpretable directions in the GAN latent space. In *Int. Conf. Mach. Learn.*, 2020. 1, 2, 7

Wang, B. and Ponce, C. R. The geometry of deep generative image models and its applications. In *Int. Conf. Learn. Represent.*, 2021. 2

Wu, Z., Lischinski, D., and Shechtman, E. StyleSpace analysis: Disentangled controls for StyleGAN image generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2, 7, 8, 14

Xu, Y., Shen, Y., Zhu, J., Yang, C., and Zhou, B. Generative hierarchical features from synthesizing images. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2

Yang, C., Shen, Y., Xu, Y., and Zhou, B. Data-efficient instance generation from instance discrimination. In *2021*, 2021a. 2

Yang, C., Shen, Y., and Zhou, B. Semantic hierarchy emerges in deep generative representations for scene synthesis. *Int. J. Comput. Vis.*, 2021b. 1, 2, 3

Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., and Xiao, J. LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 5, 13

Zhang, C., Xu, Y., and Shen, Y. Decorating your own bedroom: Locally controlling image generation with generative adversarial networks. *arXiv preprint arXiv:2105.08222*, 2021a. 2

Zhang, Y., Ling, H., Gao, J., Yin, K., Lafleche, J.-F., Barriuso, A., Torralba, A., and Fidler, S. Datasetgan: Efficient labeled data factory with minimal human effort. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021b. 1, 2

Zhao, S., Liu, Z., Lin, J., Zhu, J.-Y., and Han, S. Differentiable augmentation for data-efficient gan training. In *Adv. Neural Inform. Process. Syst.*, 2020. 2

Zhu, J., Shen, Y., Zhao, D., and Zhou, B. In-domain GAN inversion for real image editing. *Eur. Conf. Comput. Vis.*, 2020. 1

Zhu, J., Feng, R., Shen, Y., Zhao, D., Zha, Z., Zhou, J., and Chen, Q. Low-rank subspaces in GANs. In *Adv. Neural Inform. Process. Syst.*, 2021. 1, 2, 3, 7, 8, 14

Appendix

A. Overview

This paper proposed ReSeFa to precisely control the local regions of the synthesized images on pre-trained GANs. This appendix is organized as follows. First, we give some proofs of the equations in the main paper in Appendix B. Second, we provide more qualitative results in Appendix C to demonstrate the effectiveness of our method.

B. Proof

The proof of Equation (7) $\rightarrow$ Equation (8) is listed as follows.

According to the Rayleigh-Ritz quotient method (Ghojogh et al., 2019), the optimization problem in Equation (7) can be cast as

$$\begin{cases} \arg\max_{\boldsymbol{n}} \boldsymbol{n}^T \boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n}, \\ \quad \text{s.t. } \boldsymbol{n}^T \boldsymbol{J}_b^T \boldsymbol{J}_b \boldsymbol{n} = 1. \end{cases} \tag{13}$$

Hence, the Lagrangian (Boyd & Vandenberghe, 2004) is

$$\mathcal{L} = \boldsymbol{n}^T \boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n} - \lambda(\boldsymbol{n}^T \boldsymbol{J}_b^T \boldsymbol{J}_b \boldsymbol{n} - 1), \tag{14}$$

where λ is the Lagrange multiplier. Performing the derivatives of Equation (14) on $\boldsymbol{n}$, we can get

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{n}} = 2\boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n} - 2\lambda \boldsymbol{J}_b^T \boldsymbol{J}_b \boldsymbol{n} \tag{15}$$

Setting Equation (15) equals to zero, we have

$$\boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n} = \lambda \boldsymbol{J}_b^T \boldsymbol{J}_b \boldsymbol{n}. \tag{16}$$

The deduction of Eq. (8) $\rightarrow$ Eq. (9) is outlined as follows.

Substituting $\boldsymbol{J}_b^T \boldsymbol{J}_b = \boldsymbol{L}\boldsymbol{L}^T$ into Eq. (8), we have

$$\boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n} = \lambda \boldsymbol{L}\boldsymbol{L}^T \boldsymbol{n}. \tag{17}$$

Multiplying both sides with $\boldsymbol{L}^{-1}$ results in

$$\boldsymbol{L}^{-1} \boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n} = \lambda \boldsymbol{L}^T \boldsymbol{n}. \tag{18}$$

We can further write

$$\boldsymbol{L}^{-1} \boldsymbol{J}_f^T \boldsymbol{J}_f (\boldsymbol{L}^{-1})^T \boldsymbol{L}^T \boldsymbol{n} = \lambda \boldsymbol{L}^T \boldsymbol{n}. \tag{19}$$

Letting $\tilde{\boldsymbol{n}} = \boldsymbol{L}^T \boldsymbol{n}$ delivers Eq. (9).

C. Additional Results

Figure A1 gives some extra attributes on the church and car except for those shown in the main paper. Figure A2 shows the comparison results with the baselines on wearing lipstick and changing the nose size. Recall that our attribute vectors are found by solving an eigen-decomposition problem. Namely, the eigenvectors corresponding to larger eigenvalues are the attribute vectors we need. Thus, there exists a bunch of meaningful edits using different eigenvectors. Here we show some edits using different eigenvectors from the same region. Specifically, Figure A3, Figure A4, Figure A5, and Figure A6 show the various semantics found in one image in the regions of the nose, eyes, eyebrow, and mouth, respectively. Figure A7 shows the editing results on the car wheels, Figure A8 shows the editing results on the church, and Figure A9 shows the editing results on the bedroom floor.

Figure A1. **Versatile local semantics** found by our algorithm using the StyleGAN2 models (Karras et al., 2020b) trained on various datasets, including LSUN churches (indoor scene) and LSUN cars (general object) (Yu et al., 2015).

Figure A2. **Qualitative comparison** between different local editing approaches on face synthesis model (Karras et al., 2020b), including StyleSpace (Wu et al., 2021), LowRankGAN (Zhu et al., 2021), and ReSeFa. Our approach produces more realistic results and suggests more precise controllability in maintaining the image contents beyond the region of interests, *i.e.*, mouth and nose.

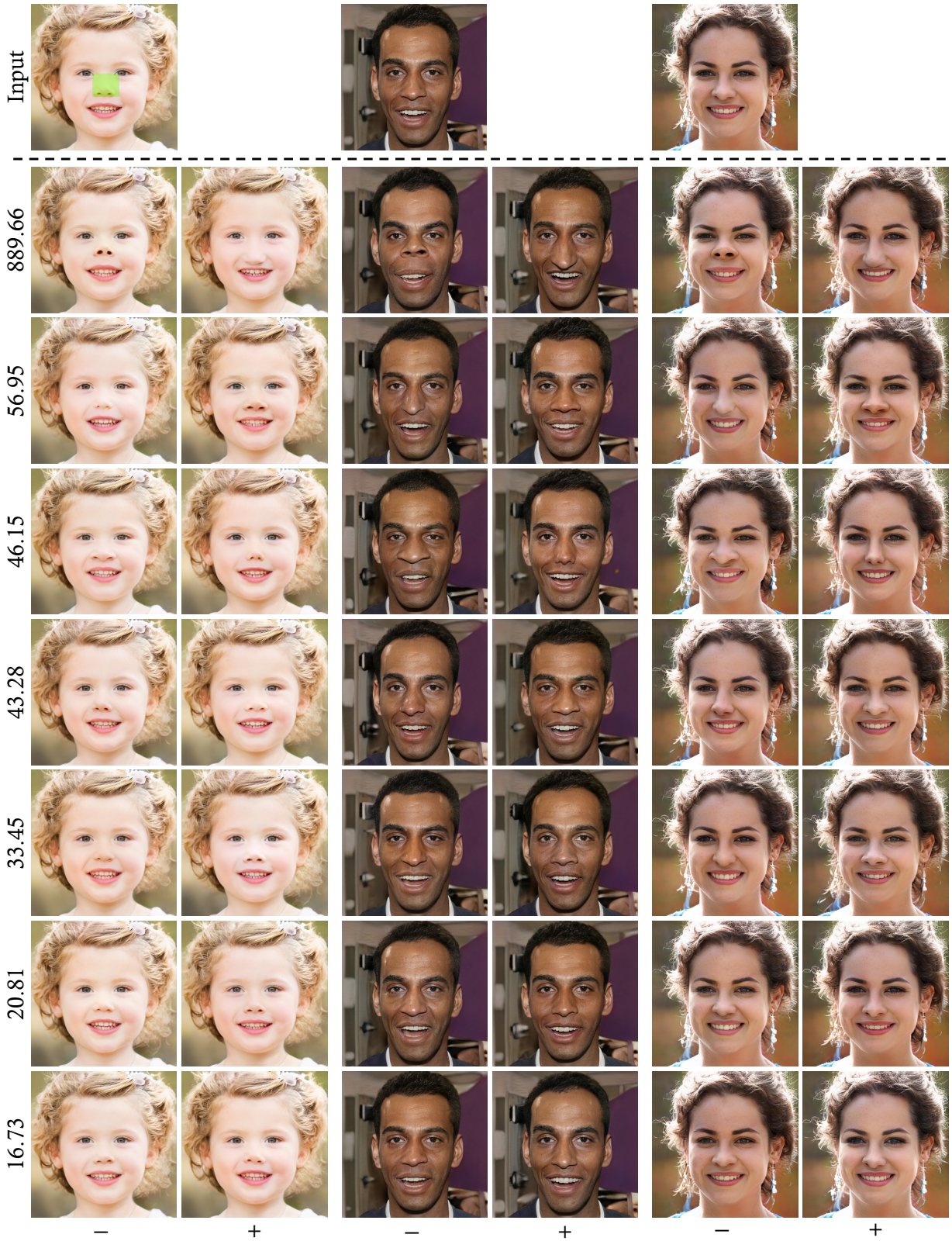

Figure A3. Visualization results of the first seven principal directions on the nose region of the human faces. The numbers asides the pictures are the eigenvalues corresponding to each direction. The **green** mask on the top left image is the region of interest used to compute the directions.

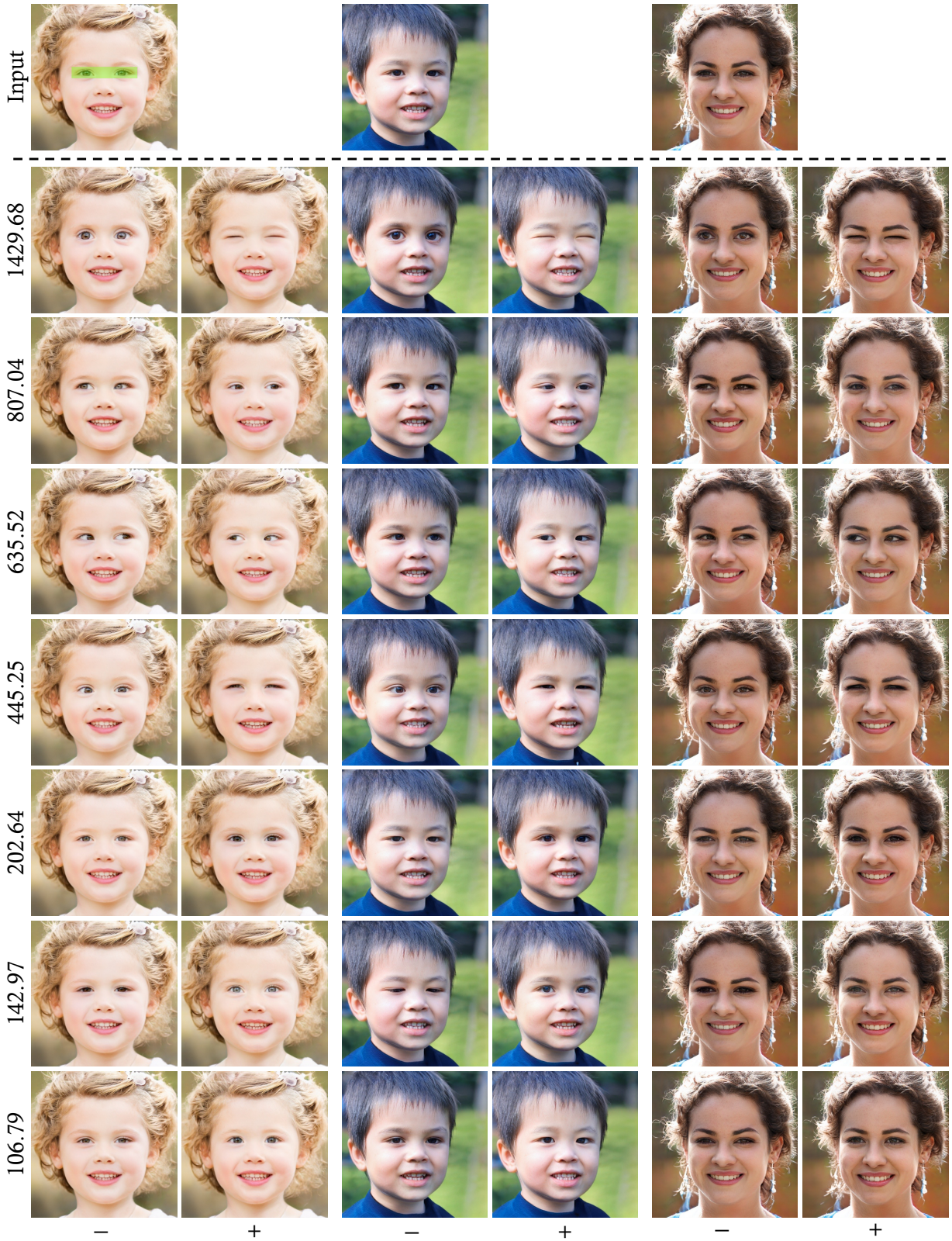

Figure A4. Visualization results of the first seven principal directions on the eyes region of the human faces. The numbers asides the pictures are the eigenvalues corresponding to each direction. The **green** mask on the top left image is the region of interest used to compute the directions.

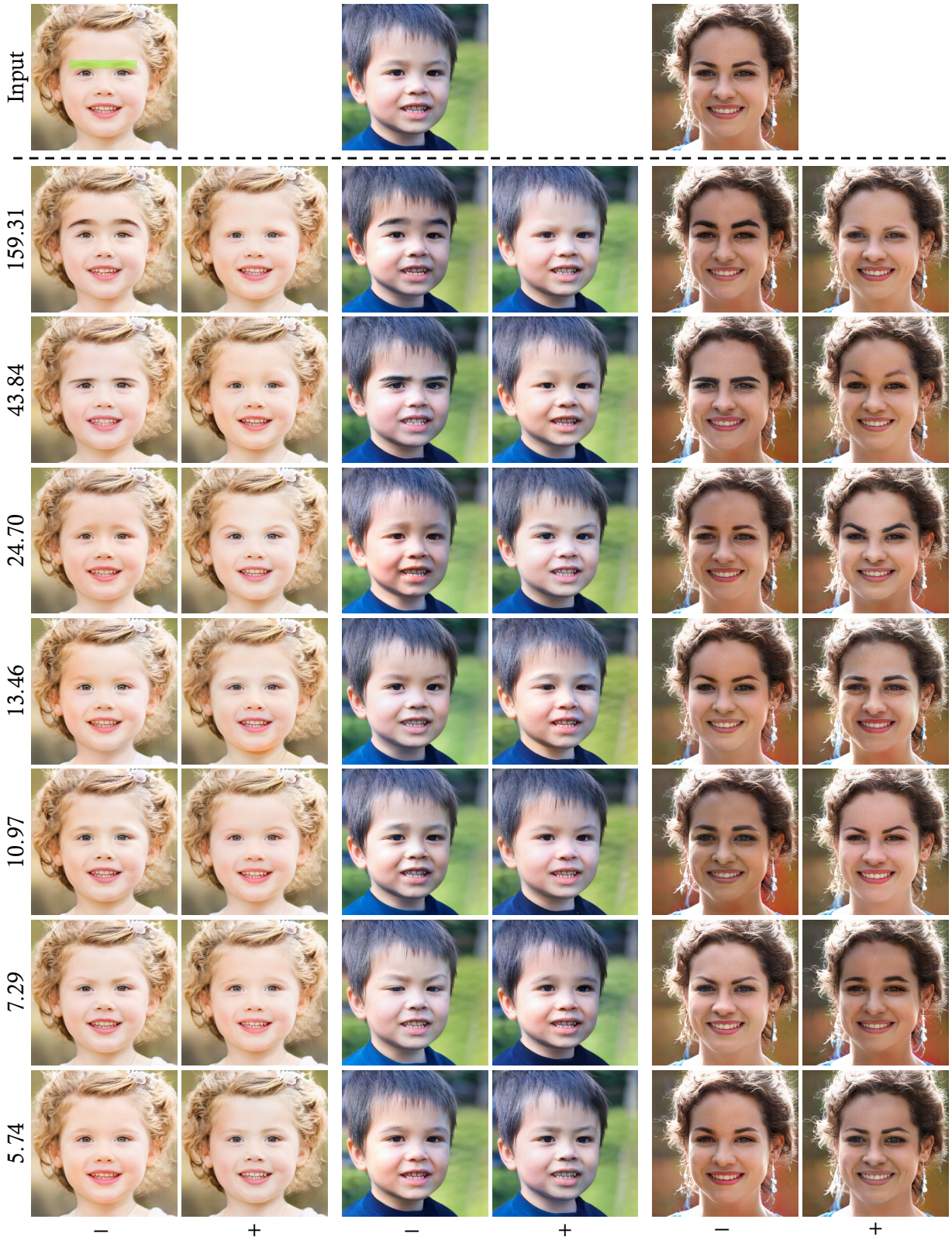

Figure A5. Visualization results of the first seven principal directions on eyebrows region of the human faces. The numbers asides the pictures are the eigenvalues corresponding to each direction. The **green** mask on the top left image is the region of interest used to compute the directions.

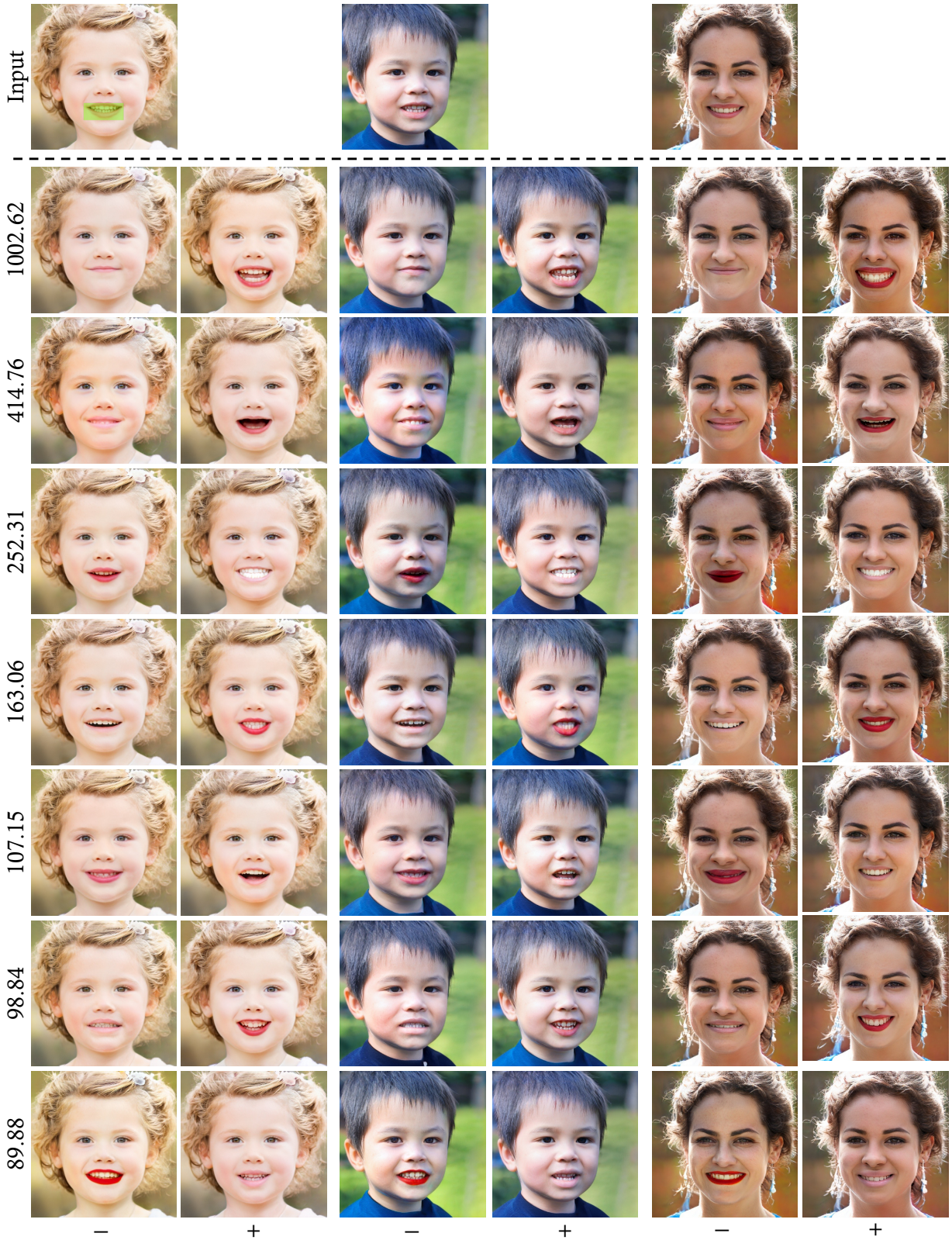

Figure A6. Visualization results of the first seven principal directions on the mouth region of the human faces. The numbers asides the pictures are the eigenvalues corresponding to each direction. The **green** mask on the top left image is the region of interest used to compute the directions.

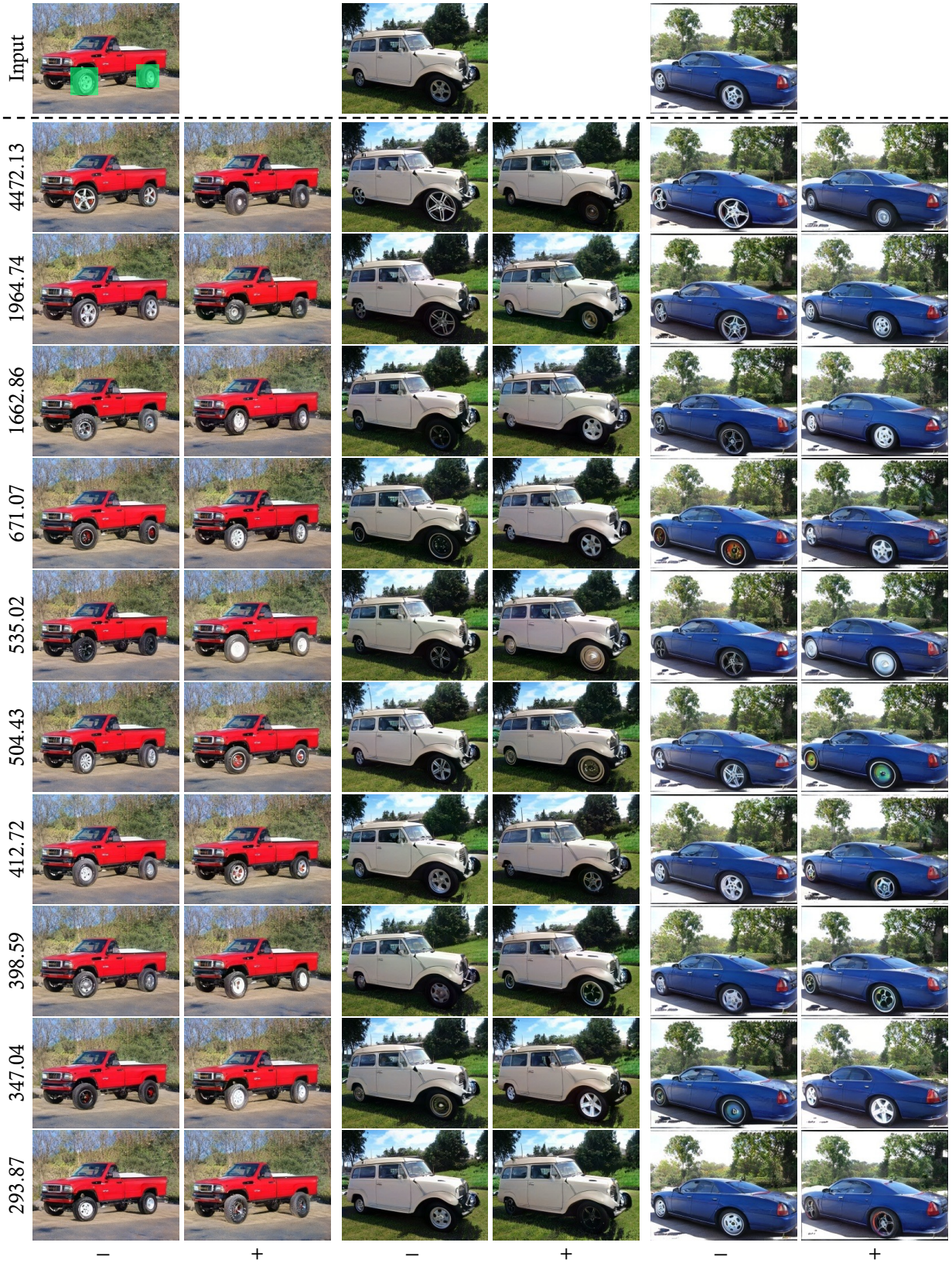

Figure A7. Visualization results of the first ten principal directions on the wheel region of the car. The numbers asides the pictures are the eigenvalues corresponding to each direction. The **green** mask on the top left image is the region of interest used to compute the directions.

Figure A8. Visualization results of the first seven principal directions on the church. The numbers asides the pictures are the eigenvalues corresponding to each direction. The **green** mask on the top left image is the region of interest used to compute the directions.

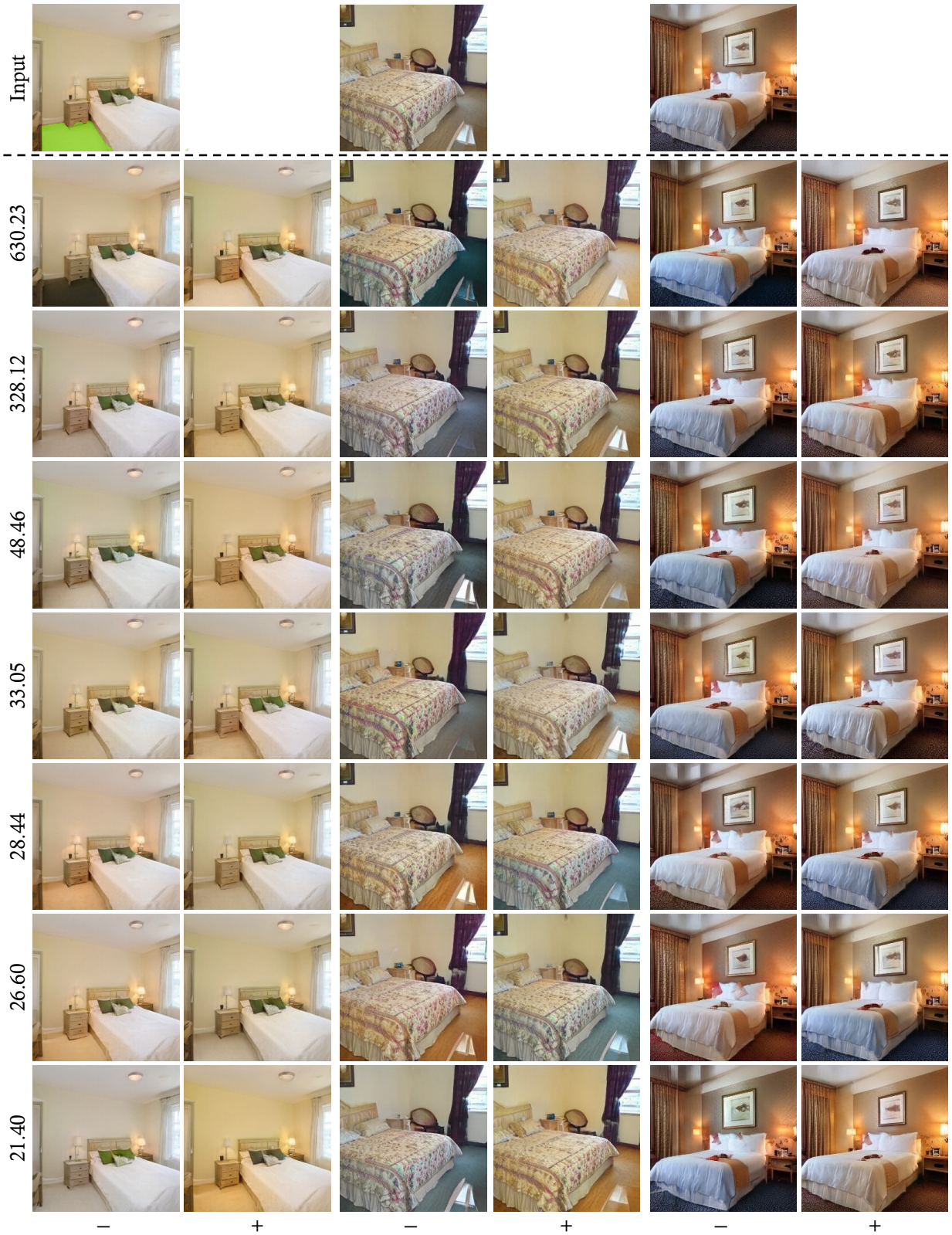

Figure A9. Visualization results of the first seven principal directions on the floor region of the bedroom. The numbers asides the pictures are the eigenvalues corresponding to each direction. The **green** mask on the top left image is the region of interest used to compute the directions.