StyleSpace Analysis: Disentangled Controls for StyleGAN Image Generation

Zongze Wu
Hebrew University
zongze.wu@mail.huji.ac.il

Dani Lischinski
Hebrew University
danix@cs.huji.ac.il

Eli Shechtman
Adobe Research
elishe@adobe.com

Abstract

We explore and analyze the latent style space of StyleGAN2, a state-of-the-art architecture for image generation, using models pretrained on several different datasets. We first show that StyleSpace, the space of channel-wise style parameters, is significantly more disentangled than the other intermediate latent spaces explored by previous works. Next, we describe a method for discovering a large collection of style channels, each of which is shown to control a distinct visual attribute in a highly localized and disentangled manner. Third, we propose a simple method for identifying style channels that control a specific attribute, using a pretrained classifier or a small number of example images. Manipulation of visual attributes via these StyleSpace controls is shown to be better disentangled than via those proposed in previous works. To show this, we make use of a newly proposed Attribute Dependency metric. Finally, we demonstrate the applicability of StyleSpace controls to the manipulation of real images. Our findings pave the way to semantically meaningful and well-disentangled image manipulations via simple and intuitive interfaces.

1. Introduction

Modern Generative Adversarial Networks (GANs) are able to produce a wide variety of highly realistic synthetic images. The phenomenal success of these generative models underscores the need for a better understanding of "what makes them tick" and what kinds of control these models offer over the generated data. Of particular practical importance are controls that are interpretable and disentangled, as they suggest intuitive image manipulation interfaces.

In traditional GAN architectures, such as DCGAN [25] and Progressive GAN [16], the generator starts with a random latent vector, drawn from a simple distribution, and transforms it into a realistic image via a sequence of convolutional layers. Recently, style-based designs have become increasingly popular, where the random latent vector is first transformed into an intermediate latent code via a mapping function. This code is then used to modify the channel-wise activation statistics at each of the generator's convolution layers. BigGAN [6] uses class-conditional BatchNorm [14], while StyleGAN [17] uses AdaIN [13] to modulate channel-wise means and variances. StyleGAN2 [18] controls channel-wise variances by modulating the weights of the convolution kernels. It has been shown that the intermediate latent space is more disentangled than the initial one [17]. Additionally, Shen *et al*. [28] show that the latent space of StyleGAN [17, 18] is more disentangled than that of Progressive GAN [16].

Some control over the generated results may be obtained via conditioning [20], which requires training the model with annotated data. In contrast, style-based design enables discovering a variety of interpretable generator controls after training the generator. However, current methods require either a pretrained classifier [10, 28, 29, 34], a large set of paired examples [15], or manual examination of many candidate control directions [12], which limits the versatility of these approaches. Furthermore, the individual controls discovered by these methods are typically entangled, affecting multiple attributes, and are often non-local.

In this work, our goal is to understand to what degree disentanglement is inherent in style-based generator architectures. Perhaps an even more important question is to how to find these disentangled controls? In particular, can this be done in an unsupervised manner, or with only a small amount of supervision? In this paper we report several findings with respect to these questions.

Recent studies of disentangled representations [8, 27] consider a latent representation to be perfectly disentangled if each latent dimension controls a single visual attribute (*disentanglement*), and each attribute is controlled by a single dimension (*completeness*). Following this terminology, we explore the latent space of StyleGAN2 [18]. Unlike other works that analyze the (intermediate) latent space $\mathcal{W}$ or $\mathcal{W}+$ [1], we examine *StyleSpace*, the space spanned by the channel-wise style parameters, denoted $\mathcal{S}$. In Section 3 we measure and compare the disentanglement and completeness of these spaces using the metrics proposed for this purpose [8]. To our knowledge we are the first to apply this quantitative framework to models trained on real

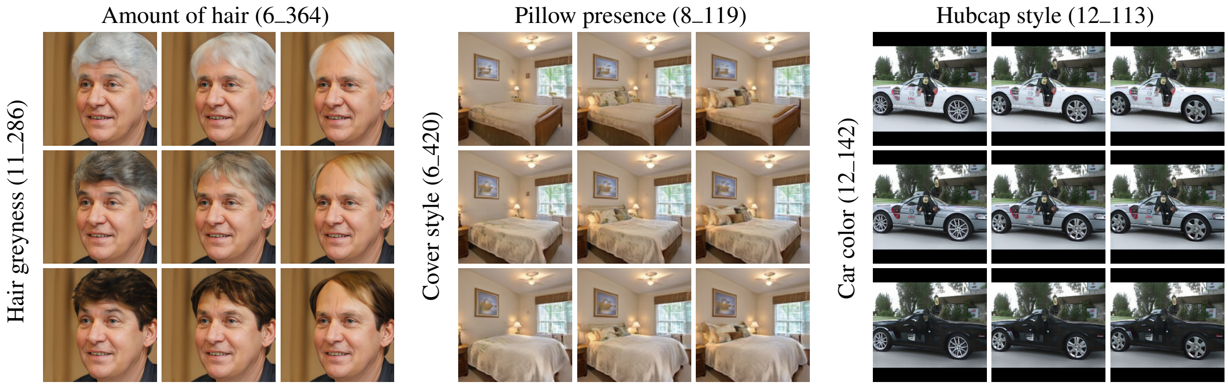

Figure 1. Disentanglement in style space, demonstrated using three different datasets. Each of the three groups above shows two manipulations that occur independently inside the same semantic region (hair, bed, and car, from left to right). The indices of the manipulated layer and channel are indicated in parentheses.

data. Our experiments reveal that $\mathcal{S}$ is significantly better disentangled than $\mathcal{W}$ or $\mathcal{W}+$.

In Section 4 we propose a simple method for detecting StyleSpace channels that control the appearance of local semantic regions in the image. By computing the gradient maps of generated images with respect to different style parameters, we identify those channels that are consistently active in specific semantic regions, such as hair or mouth, in the case of portraits. We demonstrate the effectiveness of this approach across three different datasets (FFHQ [17], LSUN Bedroom, and LSUN Car [36]). The StyleSpace channels that we detect are highly localized, affecting only a specific area without any visible impact of other regions. They are also surprisingly well disentangled from each other, as demonstrated in Figure 1.

Our next goal is to identify style channels that control a specific target attribute. To achieve this goal we require a set of exemplar images that exhibit the attribute of interest. The basic idea is to compare the average style vector across the exemplar set to the population average, thereby detecting dimensions that deviate the most. Our experiments indicate that such dimensions usually indeed control the target attribute, and reveal that a single attribute is typically controlled by only a few different StyleSpace channels.

To our knowledge, there is no metric to compare the disentanglement of different image manipulation controls. In Section 6 we propose Attribute Dependency (AD) as a measure for how manipulating a target attribute affects other attributes. Comparing manipulations performed in StyleSpace to those in $\mathcal{W}$ and $\mathcal{W}+$ spaces [12, 29], shows that our controls exhibit significantly lower AD.

Finally, we share our insights about the pros and cons of two major image inversion methods, latent optimization [18, 1, 2] and encoders [38]. We show that a combination of the two may be used in order to apply our StyleSpace controls to disentangled manipulation of real images.

2. Related Work

Understanding the latent representations of pretrained generators has attracted considerable attention, since it contributes to better GAN architecture design and facilitates controllable manipulation. Bau *et al*. [5, 3] utilized semantic segmentation to analyze Progressive GAN [16] and detect causal units that control the presence of certain objects through ablation. Shen *et al*. [29] and Yang *et al*. [34] utilize classifiers to analyze StyleGAN [17] and show that a linear manipulation in $\mathcal{W}$ space can control a specific target attribute. They further show that in $\mathcal{W}+$ space, early layers control layout, middle layers control the presence of objects, and late layers control final rendering. Collins *et al*. [7] transfer the appearance of a specific object part from a reference image to a target image, through swapping between style codes. Concurrent work by Xu *et al*. [33] shows that style space can be used for a variety of discriminative and generative tasks.

By utilizing the weights of pretrained generators, several works [1, 2, 9, 11, 22] design different latent optimization methods to do inpainting, style transfer, morphing, colorization, denoising and super resolution. Instead of latent optimization, Nitzan *et al*. [21] use the generator as a fixed decoder, and facilitate disentanglement by training an encoder for identity and another encoder for pose. Richardson *et al*. [26] do image translation by training encoders from sketches or semantic maps into StyleGAN's $\mathcal{W}$ space.

To facilitate attribute manipulations in an unsupervised manner, Voynov and Babenko [31] detect interpretable controls through training a direction matrix and a reconstructor simultaneously. Härkönen *et al*. [12] detect interpretable controls based on PCA applied either to the latent space of StyleGAN [17] or to the feature space of BigGAN [6]. Layerwise perturbations along the principle directions give rise to a variety of useful controls. Similarly, Shen *et al*. [30] do eigenvector decomposition in the affine transformation layer between $\mathcal{W}$ and $\mathcal{S}$ spaces, and use eigenvectors with the highest eigenvalues as manipulation directions. Peebles *et al*. [23] identify interpretable controls by minimizing a Hessian loss. However, in unsupervised settings, users must examine many different manipulation directions and manually identify meaningful controls.

In contrast, we discover a large amount of localized controls using semantic maps (Section 4). The controls are ranked making it easier to detect meaningful localized manipulations in each semantic region. Furthermore, our controls are surprisingly well disentangled and fine-grained. We also detect attribute-specific controls using a small number of examples (Section 5).

3. Disentanglement of StyleGAN latent spaces

The StyleGAN/StyleGAN2 generation process involves a number of latent spaces. The first latent space, $\mathcal{Z}$, is typically normally distributed. Random noise vectors $z \in \mathcal{Z}$ are transformed into an *intermediate* latent space $\mathcal{W}$ via a sequence of fully connected layers. The $\mathcal{W}$ space is claimed to better reflect the disentangled nature of the learned distribution [17]. Each $w \in \mathcal{W}$ is further transformed to channel-wise style parameters s, using a different learned affine transformation for each layer of the generator. We refer to the space spanned by these style parameters as *StyleSpace*, or $\mathcal{S}$. Some works make use of another latent space, $\mathcal{W}+$, where a different intermediate latent vector w is fed to each of the generator's layers. $\mathcal{W}+$ is mainly used for style mixing [17] and for image inversion [1, 18, 38].

In StyleGAN2 [18], there is a single style parameter per channel, which controls the feature map variances by modulating the convolution kernel weights. Additional style parameters are used by the tRGB blocks that transform feature maps to RGB images at each resolution [18]. Thus, in a 1024×1024 StyleGAN2 with 18 layers, $\mathcal{W}$ has 512 dimensions, $\mathcal{W}+$ has 9216 dimensions, and $\mathcal{S}$ has 9088 dimensions in total, consisting of 6048 dimensions applied to feature maps, and 3040 additional dimensions for tRGB blocks. See supp. Section 9 for more detail. Below we refer to individual *dimensions* of $\mathcal{S}$ as *StyleSpace channels*.

Our first goal is to determine which of these latent spaces offers the most disentangled representation. To this end, we use the recently proposed DCI (disentanglement / completeness / informativeness) metrics [8], which are suitable for comparing latent representations with different dimensions. The DCI metrics employ regressors trained using a set of latent vectors paired with corresponding attribute vectors (split into training and testing sets). *Disentanglement* measures the degree to which each latent dimension captures at most one attribute, *completeness* measures the degree to which each attribute is controlled by at most one latent dimension, while *informativeness* measures the classification accuracy of the attributes, given the latent representation.

	Comparison w/ $\mathcal{Z}$ and $\mathcal{W}$				Comparison with $\mathcal{W}+$		
	Disent.	**Compl.**	**Inform.**		**Disent.**	**Compl.**	**Inform.**
$\mathcal{Z}$	0.31	0.21	0.72				
$\mathcal{W}$	0.54	0.57	0.97	$\mathcal{W}+$	0.54	0.64	0.94
$\mathcal{S}$	**0.75**	**0.87**	**0.99**	$\mathcal{S}$	**0.63**	**0.81**	**0.98**

Table 1. Disentanglement, completeness and informativeness for different latent spaces (larger is better, maximum is 1). The two comparisons are performed using different sets of images; thus, the scores are not comparable between the two tables.

Rather than analyzing the degree of disentanglement using a synthetically generated dataset, where the factors of variations are few and known [8], we analyze StyleGAN2 trained on a real dataset, specifically FFHQ. To generate the training data for the DCI regressors, we employ 40 binary classifiers pretrained on the CelebA attributes [17]. The classifiers are trained to detect common features in portraits such as gray hair, smiling, and lipstick, and their logit outcome is converted to a binary one via a sigmoid activation.

We first randomly sample 500K latent vectors $z \in \mathcal{Z}$ and record their corresponding w and s vectors, as well as the generated images. Each image is then annotated by each of the 40 classifiers, where we record the logit, rather than just the binary outcome. Since not all attributes are well represented in the generated images (for example, there are very few portraits with a necktie), we only consider 31 attributes for which there are more than 5% positive and 5% negative outcomes. Similarly to Shen *et al*. [29], we reduce classifer uncertainty by using only the most positive 2% and most negative 2% examples, for each attribute, and split the examples equally into training and testing sets.

Finally, we compute the DCI metrics [8] to compare the latent spaces $\mathcal{Z}$, $\mathcal{W}$ and $\mathcal{S}$. As shown in Table 1 (left), while the informativeness of both $\mathcal{W}$ and $\mathcal{S}$ is high and comparable, $\mathcal{S}$ scores much higher in terms of disentanglement and completeness. This indicates that each dimension of $\mathcal{S}$ is more likely to control a single attribute and vice versa.

Since $\mathcal{W}+$ is often used for StyleGAN inversion [1, 18], we also perform a separate experiment to compare between $\mathcal{W}+$ and $\mathcal{S}$. Specifically, we first randomly sample 500K intermediate latent codes $w \in \mathcal{W}$, and construct each $w+$ by concatenating n_l random w codes ($n_l = 18$ for a 1024×1024 StyleGAN2). The resulting images are somewhat less natural than those obtained in the standard manner, resulting in a smaller number of considered attributes (25 instead of 31), which we use to evaluate $\mathcal{W}+$ and $\mathcal{S}$ as before. Table 1 (right) shows, again, that $\mathcal{S}$ scores higher than $\mathcal{W}+$.

To our knowledge, we are the first to perform a quantitative evaluation of latent space disentanglement for a GAN model trained on real data. Since our analysis indicates that the style space $\mathcal{S}$ is more disentangled than other latent spaces, we proceed to further analyze $\mathcal{S}$ below.

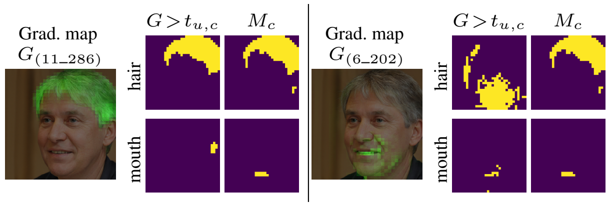

Figure 2. A gradient map with respect to each style channel u, e.g., (11_286), channel 286 of generator level 11, is thresholded against a category-specific threshold, chosen such that the resulting mask has the same size as the semantic mask M_c. The gradient mask of (11_286) has large overlap with the mask for hair, and no overlap with the mouth, while that of (6_202) has large overlap with the mask for mouth and almost none with the hair.

4. Detecting locally-active style channels

In this section we describe a simple method for detecting StyleSpace channels that control the visual appearance of local semantic regions. The intuition behind our approach is that by examining the gradient maps of generated images with respect to different channels, and measuring their overlap with specific semantic regions, we can identify those channels that are consistently active in each region. This is demonstrated in Figure 2 using two gradient maps for two different channels. If the overlap is consistent over a large number of images, these channels will be identified as locally-active for the overlapped semantic regions.

Specifically, for each image generated with style code $s \in \mathcal{S}$, we apply back-propagation to compute the gradient map of the image with respect to each channel of s. To save computation, the gradient maps are computed at a reduced spatial resolution $r \times r$ ($r = 32$ in our experiments). Next, a pretrained image segmentation network is used to obtain the semantic map M^s of the generated image. The map is resized to $r \times r$ by using the most abundant semantic category inside each bin as its semantic label. For each semantic category c and each channel u, we measure the overlap between the semantic region M^s_c and the gradient map G^s_u:

$$OC^s_{u,c} = \frac{|(G^s_u > t^s_{u,c}) \cap M^s_c|}{|M^s_c|^d}. \qquad (1)$$

Here $t^s_{u,c}$ is a threshold chosen such that gradient mask $(G^s_u > t^s_{u,c})$ has the same size as M^s_c (see Figure 2). The correction factor d gives more weight to small areas, since a large overlap between two small masks indicates precise localization. In practice, $d = 2$ gives us good balance between large and small areas.

To ensure consistency across a variety of images, we sample 1K different style codes, and compute for each code s and each channel u the semantic category with the highest overlap coefficient: $c^*_{s,u} = \arg\max OC^s_{u,c}$. Our goal is to detect channels for which the highest overlap category is the same for the majority of the sampled images. Furthermore, we require that the overlap with the second most commonly affected category is twice as rare.

4.1. Experiments

We analyze StyleGAN2 [18] pretrained on FFHQ 1024x1024, LSUN Car 512x384, and LSUN Bedroom 128x128 [36]. To obtain semantic maps, we use a BiSeNet model [35] pretrained on CelebAMask-HQ [19], and a unified parsing network [32] pretrained on Broden+ [4].

As explained in Section 3 and supp. Section 9, 3040 channels of $\mathcal{S}$ are used to control the tRGB blocks. None of these channels were found to have a localized effect. Rather, these channels have a global effect on the generated image, as shown in supp. Figure 10.

Among the remaining 6048 channels, 1871 were found to be locally-active (in the model trained on FFHQ). Most of the detected channels control clothes (34.9%) or hair (21%). For the model trained on LSUN bedroom, we found 421 locally-active channels, most of which control the bed region (27.6%). For StyleGAN2 pretrained on LSUN car, we found 913 locally-active channels, most of which control window (33.1%) and wheel (27.3%) regions. Most of the detected channels are spread among several middle layers, with barely any channels found in early or late layers. A detailed summary of the detected locally-active channels and their breakdown by different semantic regions is included in supp. Section 11.

Figures 3 and 4 demonstrate some of the localized manipulations obtained by modifying the values of the channels we detected. Surprisingly, each channel appears to only control a single attribute, and even channels affecting the same local region are well disentangled, as demonstrated in Figure 1 and supp. Figure 12. Unlike most controls detected by previous methods, these SpaceStyle channels provide an extremely fine-grained level of control. For example, the four channels for the ear region (last row of Figure 3), provide separate controls for the visibility of the ear, its shape, and the presence of an earring. A variety of fine-grained controls are also detected in the Car and Bedroom models (Figure 4). It should be noted that finding such interpretable disentangled local controls is very easy with our method: out of the top 10 most localized channels for each semantic area, we observe that 4–10 dimensions control (subjectively) meaningful visual attributes. A detailed breakdown by semantic category is reported in supp. Table 4.

In contrast, individual channels of $\mathcal{W}$ or $\mathcal{W}+$ space are usually entangled, with each channel affecting multiple attributes, as predicted by the DCI-based analysis from the previous section. We attribute this to the fact that each channel of $\mathcal{W}+$ affects the style parameters of an entire generation layer (via an affine transformation), rather than those of a single feature map channel.

Figure 3. Examples of manipulations, each controlled by a single style channel. Each pair of images shows the result of manipulation by decreasing (-) and increasing (+) the value of the style parameter (the original image is omitted). The layer index, channel index, and the direction of change is overlayed in the bottom left corner.

5. Detecting attribute-specific channels

In this section we propose a method for identifying StyleSpace channels which control a specific target attribute, specified by a set of examples. For example, given a collection of portraits of grey-haired persons, our goal is to find individual channels that control hair greyness. In contrast to InterFaceGAN [29], where around 10K positive and 10K negative examples are required, our approach typically requires only 10–30 positive exemplars. This is an important advantage, since for many attributes, negative examples can be highly varied. For example, while it is easy to find positive examples for blond hair, negative examples should ideally include all non-blond hair colors.

Our approach is based on the simple idea that the differences between the mean style vector of the positive examples (exemplar mean) and that of the entire generated distribution (population mean) reveal which channels are the most relevant for the target attribute.

Specifically, let μ^p and σ^p denote the mean and the standard deviation of the style vectors over the generated distribution. Given the style vector s^e of a specific positive example, we compute its normalized difference from the population mean: $\delta^e = \frac{s^e - \mu^p}{\sigma^p}$. Next, let μ^e and σ^e denote the mean and the standard deviation of the differences δ^e over the exemplar set. For each style channel u, the magnitude of the corresponding component μ^e_u indicates the extent to which u deviates from the population mean. Thus, we measure the relevance of u with respect to the target attribute as

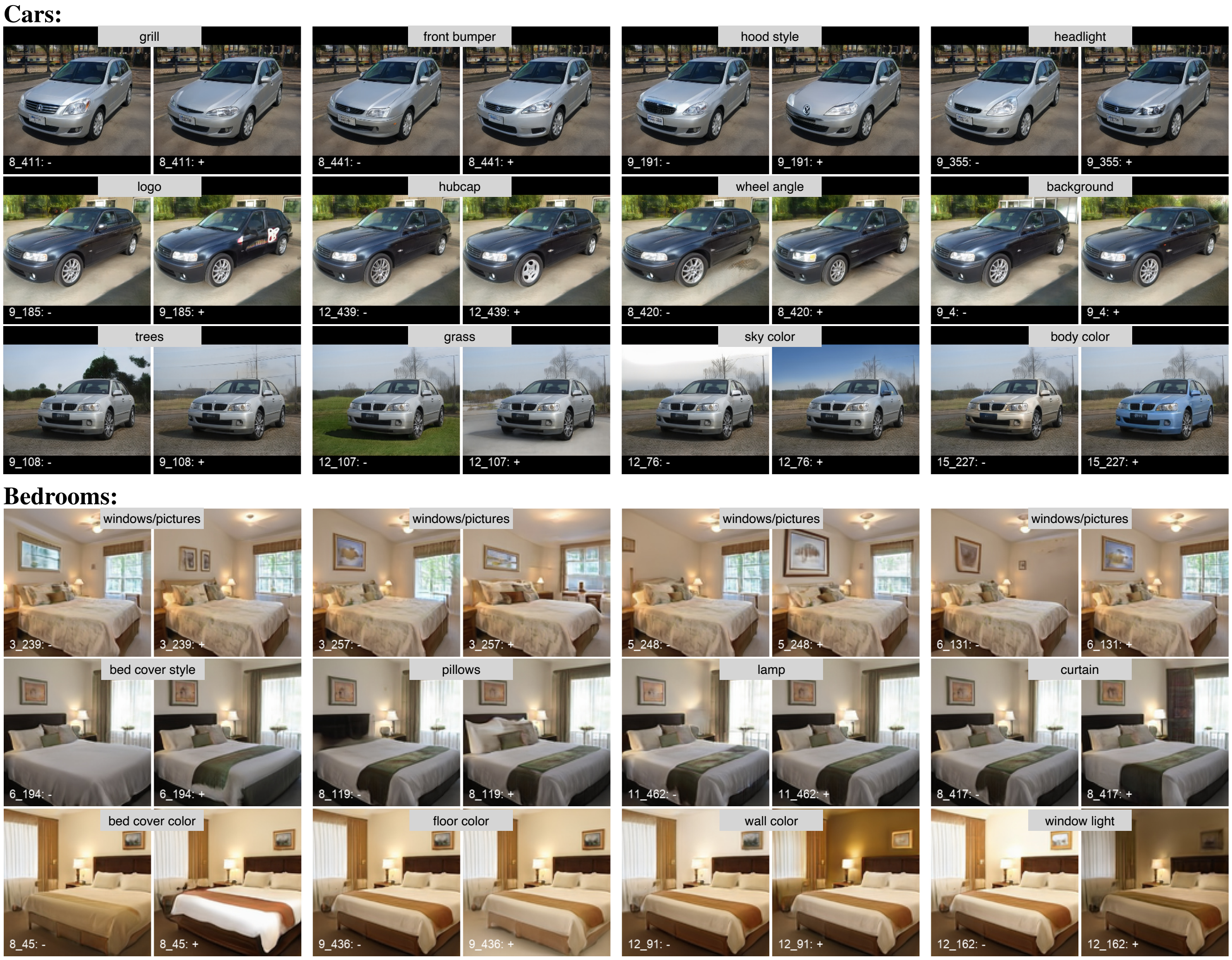

Figure 4. Examples of manipulations, each controlled by a single style dimension. Each pair of images shows the result of manipulation by decreasing (-) and increasing (+) the value of the style parameter (the original image is omitted). The layer index, channel index, and the direction of change is overlayed in the bottom left corner.

the ratio $\theta_u = \frac{|\mu_u^e|}{\sigma_u^e}$. Due to the high disentanglement of $\mathcal{S}$ (Section 3), a style channel u with a high θ_u value may be assumed to control the target attribute.

5.1. Experiments

We first use a large number (1K) of positive examples to verify that the simple method described above is indeed able to identify a set of attribute-specific control channels. Next, we demonstrate that as few as 10–30 positive examples are sufficient to detect most of these channels.

We first use the set of pretrained classifiers that were used in Section 3, to identify 1K highly positive examples for each of selected 26 attributes (see supp. Section 12 and Table 5). For each attribute, we rank all the style channels (except the 3040 tRGB ones) by their relevance θ_u, and manually examine the top 30 channels to verify that they indeed control the target attribute.

Our examination reveals that 16 out of the 26 attributes may be controlled by at least one single style channel (see supp. Table 5). The channels detected for each attribute and their ranks are reported in supp. Table 6. Interestingly, for well-defined visual attributes, such as gender, black hair, or gray hair, our method was able to find only one controlling channel. In contrast, for less specific attributes, especially those related to hair styles (bangs, receding hairline), we identified multiple controlling channels. We observe that these controls are not redundant, each controlling a unique hair style. The remaining 10 attributes are typically entangled (e.g., high cheekbones, young, or chubby), and thus no single control channels were detected for them. See supp. Section 12 for further discussion.

Most of the detected attribute-specific control channels were highly ranked by our proposed importance score θ_u. For example, for 14 out of 16 attributes, the top-ranked

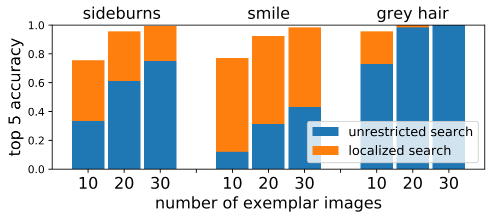

Figure 5. Top-5 detection accuracy for attribute-specific controls (for three target attributes) using 10, 20, or 30 positive examples.

channel was verified to indeed control the attribute (see supp. Table 6 for the ranks of all the attribute-specific channels that we detected). This suggests that a small number of positive examples provided by a user might be sufficient for identifying such channels.

To verify the above conjecture, we randomly select sets of 10, 20, and 30 positive examples for each of three attributes (sideburns, smile, gray hair) and identify the top 5 channels for each of these small exemplar sets. If the top 5 channels include any of the verified control channels (determined using 1K images), this is considered a success. The results are reported in Figure 5.

As shown in Figure 5, increasing the number of positive examples improves the detection accuracy. The accuracy may be further improved by only considering locally-active channels (found as described in Section 4) in areas related to the target attribute. For example, if smile is the target attribute, considering only channels that are active in the mouth area, greatly improves the chances of detection. As shown by the orange bars in Figure 5, the top-5 detection rate exceeds 92% using as few as 20 examples, if the search is restricted to channels locally-active in the target area.

In summary, our approach requires only 10–30 positive examples, and detects single StyleSpace control channels. In contrast, GANSpace [12] identifies manipulation controls via a manual examination of a large number of different manipulation directions, which typically involve all of the channels of one or several layers. InterFaceGAN [29] requires more than 10K positive and 10K negative examples for each manipulation direction, which is defined in $\mathcal{W}$ space, and thus affects all layers. Furthermore, the controls detected by these two approaches are more entangled that our control channels, as shown in the next section.

6. Disentangled attribute manipulation

In this section we compare the ability of our approach to achieve disentangled manipulation of visual attributes to that of two state-of-the-art methods, specifically GANSpace [12] and InterFaceGAN [29]. Comparisons to additional methods are included in supplementary material.

Figure 6 and supp. Figure 15 show a qualitative comparison between the three methods, showing the manipulation of three attributes for which the direction of manipulation is identified by all three approaches: *Gender*, *Gray hair*, and *Lipstick*. The step size along the manipulation direction is chosen such that it induces the same amount of change in the logit value l_t of the corresponding classifiers (pretrained on CelebA). Note that InterFaceGAN manipulations sometimes significantly change the identity of the person (esp. in the *Lipstick* manipulation), and some other attributes as well (added wrinkles in the *Gray hair* manipulation). GANSpace manipulations also exhibit some entanglement (*Lipstick* affects face lightness, *Gray hair* ages the rest of the face). In contrast, our approach appears to affect only the target attribute. Our *Gender* manipulation, for example, does not affect the hair style, and minimally changes the face, yet the gender unmistakably changes.

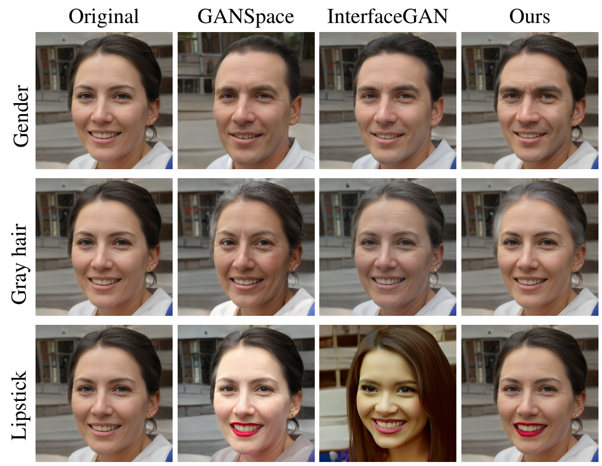

Figure 6. Comparison with state-of-the-art methods using the same amount of manipulation $\Delta l_t = 1.5\sigma(l_t)$.

To perform a more comprehensive and quantitative comparison between the three methods, we propose a general disentanglement metric for real images, which we refer to as *Attribute Dependency* (AD). Attribute Dependency measures the degree to which manipulation along a certain direction induces changes in other attributes, as measured by classifiers for those attributes (see supp. Section 13 for additional details). The use of classifiers here is necessary in order to cope with real images, where the exact factors of variation are not known, and we have no means to measure them. Intuitively, disentangled manipulations should induce smaller changes in other attributes.

To perform the comparison, we sample a set of images without the target attribute t (e.g., without gray hair), and manipulate them towards the target attribute, by a certain amount measured by the change in the logit outcome Δl_t of a classifier pretrained to detect attribute t. Next, we measure the change of logit Δl_i between the original images and the manipulated ones for other attributes $\forall i \in \mathcal{A} \backslash t$, where $\mathcal{A}$ is the set of all attributes. Each change is normalized by $\sigma(l_i)$, the standard deviation of the logit value for attribute i over a large set of generated images. We measure mean-AD, de-

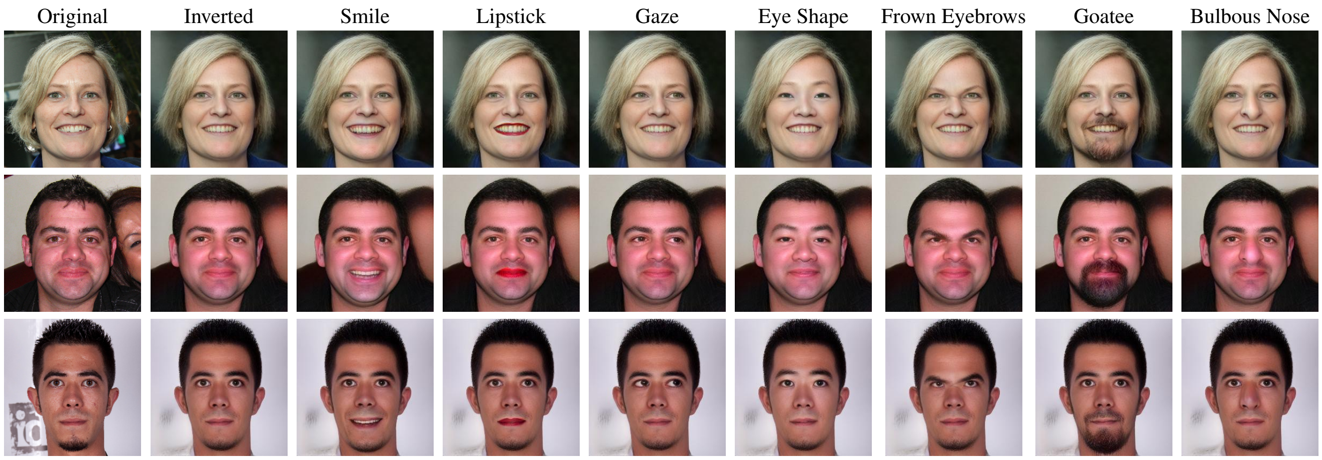

Figure 7. Manipulation of real images using encoder-based inversion. Original images are from FFHQ, and were not part of the encoder's training set. More results can be found in supplementary Figure 19.

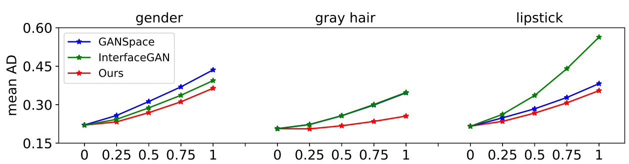

Figure 8. Mean-AD vs. the degree of target attribute manipulation ($\Delta l_t/\sigma(l_t)$). Lower mean-AD indicates better disentanglement.

fined as $E(\frac{1}{k}\sum_{i\in\mathcal{A}\setminus t}(\frac{\Delta l_i}{\sigma(l_i)}))$, where $k = |\mathcal{A}|-1$. Similarly, we measure max-AD, defined as $E(\max_{i\in\mathcal{A}\setminus t}(\frac{\Delta l_i}{\sigma(l_i)}))$.

Figure 8 plots the mean-AD of the three methods (GANSpace, InterFaceGAN, and ours) for a range of manipulations of the *Gender*, *Gray hair*, and *Lipstick* attributes. It may be seen that our method (in red) exhibits a smaller mean-AD, compared to the other two methods, for each of these three attributes and across the entire manipulation range. This is consistent with our qualitative visual observations, as demonstrated in Figure 6. Our method also achieves lower max-AD scores, as reported in the supplementary material.

7. Manipulation of Real Images

To manipulate real images, it is necessary to first invert them into latent codes. This may be done via latent optimization [1, 2] or by training an encoder [39, 38] based on reconstruction loss (LPIPS [37] or L2). We adapt the latent optimization algorithm of Karras *et al*. [18] to invert real images into $\mathcal{W}$, $\mathcal{W}+$, and $\mathcal{S}$ separately. Latent optimization in $\mathcal{W}+$ and $\mathcal{S}$ spaces has more flexibility than in $\mathcal{W}$, enabling a closer reconstruction of the input image. Indeed, we find that the visual accuracy of the reconstruction is the highest when optimizing in $\mathcal{S}$, followed by $\mathcal{W}+$, and is the lowest for $\mathcal{W}$ (see supp. Figure 17). Unfortunately, the extra flexibility may result in latent codes that do not lie on the generated image manifold, and attempting to manipulate such codes typically results in unnatural artifacts. Thus, conversely to reconstruction accuracy, we find that manipulation naturalness is best when the latent optimization is done in $\mathcal{W}$, followed by $\mathcal{W}+$, and the worst for $\mathcal{S}$ (see supp. Figure 18).

In order to achieve a satisfactory compromise between reconstruction accuracy and artifact-free manipulation, we train an encoder to $\mathcal{S}$ space following the training strategy of [38] using only reconstruction loss (LPIPS). The encoder's structure follows that of StyleALAE [24]. Due to limited computational resources, the encoder is trained on real images from FFHQ whose resolution was reduced to 128×128. The reconstructed images bear good similarity to the input images, but exhibit some compression artifacts. The encoder's result serves as the starting point for latent optimization [18] in $\mathcal{S}$ space, which proceeds for a small number of iterations (50 rather than a few thousands). We find that this process can efficiently remove compression artifacts, and the resulting inversions enable artifact-free manipulation, as demonstrated in Figure 7. We believe this is because the encoder learns to embed the input real images closer to the generated image manifold, and the few optimization iterations only fine-tune the embedding.

8. Conclusion

We have shown that StyleSpace is highly disentangled, and proposed simple methods for detecting meaningful manipulation controls in this space. Future work should focus on finding meaningful control directions that involve multiple style channels. We also plan to develop inversion techniques that can deliver both high reconstruction accuracy and manipulability.

Acknowledgments We thank the anonymous reviewers for their comments. This work was supported by a gift from Adobe and by the Israel Science Foundation (grant no. 2492/20).

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proc. ICCV*, pages 4432–4441, 2019.

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *Proc. CVPR*, pages 8296–8305, 2020.

[3] David Bau, Hendrik Strobelt, William Peebles, Jonas Wulff, Bolei Zhou, Jun Yan Zhu, and Antonio Torralba. Semantic photo manipulation with a generative image prior. *ACM Trans. Graph.*, 38(4), 2019.

[4] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network Dissection: quantifying interpretability of deep visual representations. In *Proc. CVPR*, pages 6541–6549, 2017.

[5] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B. Tenenbaum, William T. Freeman, and Antonio Torralba. GAN Dissection: visualizing and understanding generative adversarial networks. In *Proc. ICLR*, 2019.

[6] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[7] Edo Collins, Raja Bala, Bob Price, and Sabine Susstrunk. Editing in style: Uncovering the local semantics of GANs. In *Proc. CVPR*, pages 5771–5780, 2020.

[8] Cian Eastwood and Christopher KI Williams. A framework for the quantitative evaluation of disentangled representations, 2018. In *Proc. ICLR*, volume 5, page 8, 2018.

[9] Aviv Gabbay and Yedid Hoshen. Style generator inversion for image enhancement and animation. *arXiv preprint arXiv:1906.11880*, 2019.

[10] Lore Goetschalckx, Alex Andonian, Aude Oliva, and Phillip Isola. GANalyze: toward visual definitions of cognitive image properties. In *Proc. ICCV*, pages 5744–5753, 2019.

[11] Jinjin Gu, Yujun Shen, and Bolei Zhou. Image processing using multi-code GAN prior. In *Proc. CVPR*, pages 3012–3021, 2020.

[12] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. GANSpace: discovering interpretable GAN controls. *arXiv preprint arXiv:2004.02546*, 2020.

[13] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proc. ICCV*, pages 1501–1510, 2017.

[14] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[15] Ali Jahanian, Lucy Chai, and Phillip Isola. On the "steerability" of generative adversarial networks. *arXiv preprint arXiv:1907.07171*, 2019.

[16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[17] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proc. CVPR*, pages 4401–4410, 2019.

[18] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *Proc. CVPR*, pages 8110–8119, 2020.

[19] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *Proc. CVPR*, pages 5549–5558, 2020.

[20] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[21] Yotam Nitzan, Amit Bermano, Yangyan Li, and Daniel Cohen-Or. Disentangling in latent space by harnessing a pretrained generator. *arXiv preprint arXiv:2005.07728*, 2020.

[22] Xingang Pan, Xiaohang Zhan, Bo Dai, Dahua Lin, Chen Change Loy, and Ping Luo. Exploiting deep generative prior for versatile image restoration and manipulation. *arXiv preprint arXiv:2003.13659*, 2020.

[23] William Peebles, John Peebles, Jun-Yan Zhu, Alexei Efros, and Antonio Torralba. The hessian penalty: A weak prior for unsupervised disentanglement. *arXiv preprint arXiv:2008.10599*, 2020.

[24] Stanislav Pidhorskyi, Donald A Adjeroh, and Gianfranco Doretto. Adversarial latent autoencoders. In *Proc. CVPR*, pages 14104–14113, 2020.

[25] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[26] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. *arXiv preprint arXiv:2008.00951*, 2020.

[27] Karl Ridgeway and Michael C Mozer. Learning deep disentangled embeddings with the f-statistic loss. In *Advances in Neural Information Processing Systems*, pages 185–194, 2018.

[28] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of GANs for semantic face editing. In *Proc. CVPR*, pages 9243–9252, 2020.

[29] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. InterFaceGAN: interpreting the disentangled face representation learned by GANs. *arXiv:2005.09635*, 2020.

[30] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in GANs. *arXiv:2007.06600*, 2020.

[31] Andrey Voynov and Artem Babenko. Unsupervised discovery of interpretable directions in the GAN latent space. In *Proc. ICML*, pages 9786–9796, 2020.

[32] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proc. ECCV*, pages 418–434, 2018.

[33] Yinghao Xu, Yujun Shen, Jiapeng Zhu, Ceyuan Yang, and Bolei Zhou. Generative hierarchical features from synthesizing images. *arXiv preprint arXiv:2007.10379*, 2020.

[34] Ceyuan Yang, Yujun Shen, and Bolei Zhou. Semantic hierarchy emerges in deep generative representations for scene synthesis. *arXiv preprint arXiv:1911.09267*, 2019.

[35] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmen-

tation network for real-time semantic segmentation. In *Proc. ECCV*, pages 325–341, 2018.
[36] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.
[37] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proc. CVPR*, pages 586–595, 2018.
[38] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain GAN inversion for real image editing. *arXiv preprint arXiv:2004.00049*, 2020.
[39] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative visual manipulation on the natural image manifold. In *Proc. ECCV*, pages 597–613. Springer, 2016.