Generative Adversarial Networks for Diverse and Explainable Text-to-Image Generation
Zhang, Zhenxing

DOI: 10.33612/diss.507581028

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date: 2023

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):
https://doi.org/10.33612/diss.507581028

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license. More information can be found on the University of Groningen website: https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment.

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Download date: 01-08-2024
Text-to-image generation can be potentially employed in the fields of art creation, data augmentation, photo-editing, etc. Although many efforts have been dedicated to this task, it remains particularly challenging to yield believable, realistic and natural scenes. To facilitate the real-world applications of text-to-image synthesis, we focus on studying the following issue: How to increase the likelihood that such an algorithm produces more natural, realistic and believable images? In this chapter, we explore the semantic relationship between a realistic image and an inadequate sample in the space of generated images by linearly interpolating a successful starting-point latent code and an unsuccessful end-point latent vector. After that, we constructed two novel data sets, i.e., the Good & Bad bird and face data sets, consisting of successful as well as unsuccessful synthesized samples selected by following strict criteria. To effectively and efficiently acquire high-quality pictures by increasing the probability of generating Good latent codes, we use a dedicated Good/Bad classifier for synthesized images. It is based on a pretrained front end and fine-tuned on the basis of the proposed Good & Bad data set. Experimental results on the designed DiverGAN generator pre-trained on two benchmark data sets demonstrate that our classifier achieves a better than 98% accuracy in predicting Good/Bad classes for synthetic samples, confirming the effectiveness of our presented techniques and data sets. Our data set is available at https://zenodo.org/record/6283798#.YhkN_ujMJzw.

This chapter is based on


4.1 Introduction

With the advances in a GAN and a cGAN [71], text-to-image generation has achieved promising progress in both image quality and semantic consistency. Nevertheless, it remains extremely challenging to coerce a conditional text-to-image GAN model to generate, with high probability, believable, realistic and natural images, since the space of generated images contains a large number of blurry and inadequate samples.

One particular disadvantage of synthetic image-generation algorithms is that the performance evaluation is more difficult than is the case in classification problems where a 'hard' accuracy can be computed. In case of the cGAN this issue is most clearly present for end users: How to ensure that generated images are believable, realistic or natural? In current literature, the good examples are often cherry picked while occasionally also the less successful samples are shown. However, for actual use in data augmentation or in artistic applications, one would like to guarantee that generated images are good, i.e., of a sufficiently believable natural quality. Given the high dimensionality of latent codes, there is a very high prior probability of non-successful patterns to be generated for a given input noise probe. How to construct a random latent-code generator with an increased probability of drawing successful samples? After the generator/discriminator pair has done its best effort, apparently additional constraints are necessary.

To address the above-mentioned issue, we attempt to study the semantic relationship between high-quality synthesized samples and implausible generated images in the space of synthesized samples by performing the pairwise linear interpolation of latent codes. We empirically discover that the generator is likely to yield a series of high-resolution images when linearly interpolating two Good latent codes. In addition, interpolation results often look blurry if the generator cannot generate plausible samples according to these two latent codes. This would imply that there may exist a close relationship between successful generated pictures and unsuccessful synthesized images in the distribution of generated images.

To further explore the semantic correspondence between a plausible sample and an inadequate picture in the space of synthesized image, we visualize the samples generated by linearly interpolating a successful starting-point latent vector and an unsuccessful end-point latent code. In addition, we compute the LPIPS score and the perceptual loss between two close interpolation images, in order to quantitatively measure if there is a smooth transition from a successful generated sample to an unsuccessful synthesized image. We find that the first part of interpolation results is usually realistic but the final part is not plausible when conducting the linear interpolation between a Good latent code and a Bad latent vector. Moreover, the visual appearance and the semantics do not always change gradually with the variations of latent codes. Fig. 4.1 visualizes multiple
examples of the pairwise linear interpolation between a Good latent vector and a Bad latent code. We therefore make the assumption that there is a non-linear boundary separating high-resolution images from inadequate samples in the space of generated samples.

Here, we intend to train a classifier to accurately distinguish successful synthesized samples from unsuccessful generated pictures after training a text-to-image generation framework. To this end, we created a Good & Bad data set (shown in Fig. 4.4), both for a bird and a face-image collection, which consists of a large number of realistic as well as implausible samples synthesized by DiverGAN that was pre-trained on the CUB bird [108] data set and the Multi-Modal CelebA-HQ data set [116], respectively. We choose these samples by following strict principles in order to ensure the quality of the selected images. To acquire a superior classifier, we train the CNN model (e.g., ResNet [37]) from the pretrained weights on our Good & Bad data set. We expect that the well-trained network can correctly predict the quality classes of synthesized images. Therefore, we are able to effectively and efficiently derive photo-realistic images from the synthesized samples while also obtaining corresponding Good latent vectors. More importantly, the discovery of Good latent codes provides a strong basis for further research, such as data augmentation and latent-space manipulation which will be discussed in the next chapter.

The developed DiverGAN in Chapter 3 has the ability to adopt a generator/discriminator pair to synthesize diverse and high-quality samples, given a textual description and different injected noise on the latent vector. We therefore carry out a serious of experiments on the DiverGAN generator that was trained on two popular text-to-image data sets, i.e., the CUB bird [108] and Multi-Modal CelebA-
hq [116] data sets. Experimental results show that our well-trained classifier achieves impressive classification accuracy (bird: 98.09% and face: 99.16%) on the Good & Bad data set, which validates our claim. The contributions of this chapter can be summarized as follows:

- We perform the pairwise linear interpolation of latent codes to explore the semantic relation between successful generated samples and unsuccessful synthesized pictures in the space of generated images.
- We constructed two new Good & Bad data sets to study how to ensure that generated images are natural, realistic and believable.
- We train two classifiers on the proposed Good & Bad data sets to well separate successful generated images from unsuccessful synthetic samples.

4.2 Proposed Methodology

In this section, we briefly introduce the single-stage text-to-image GAN model and the corresponding latent space. Subsequently, we study the semantic relationship between successful generated images and unsuccessful synthesized samples in the space of produced instances using the pairwise linear interpolation of latent codes. Afterwards, we elaborate on the proposed procedure automatically finding successful synthetic samples from generated pictures while acquiring corresponding Good latent codes.

4.2.1 Preliminary

In this subsection, we briefly describe a single-stage text-to-image synthesis architecture and analyze the corresponding latent space to help understanding the issue we attempt to address.

**Single-stage text-to-image pipeline.** A single-stage text-to-image generation framework (illustrated in Fig. 4.2) is composed of a generator network and a discriminator net, which are perceived as playing a minmax zero-sum game. Let $S = \{(C_i, I_i)\}_{i=1}^N$ denote a collection of $N$ text-image pairs for training, where $I_i$ is a picture and $C_i = (c^{i_1}, c^{i_2}, ..., c^{i_k})$ comprises $K$ textual descriptions. Word-embedding vectors $w$ and a sentence-embedding vector $s$ are commonly acquired by applying a bidirectional LSTM network [92] on a natural-language description $c_i$ randomly picked from $C_i$. After that, the generator $G(z, (w, s))$ is trained to produce a perceptually realistic and semantically related image $\hat{I}_i$ according to a latent code $z$ randomly sampled from a frozen distribution and word/sentence embedding vectors $(w, s)$. To be specific, $G(z, (w, s))$ consists of multiple layers where the first layer $F_0$ maps a latent code into a feature map and intermediate blocks typically leverage modulation modules (e.g., attention models [132], [133]) to reinforce the visual feature map to ensure image quality.
and semantic consistency. The last layer \( G_c \) transforms the feature map into the ultimate sample. Mathematically,

\[
\begin{align*}
    h_0 &= F_0(z) \\
    h_1 &= B_1(h_0, (w,s)) \\
    h_i &= B_i(h_{i-1} \uparrow, (w,s)) \quad \text{for } i = 2, 3, \ldots, 7 \\
    \hat{I} &= G_c(h_7)
\end{align*}
\]

where \( F_0 \) denotes a fully-connected layer and \( B_i \) is a modulation block that facilitates the feature map with textual features.

Compared with \( G(z, (w,s)) \), the discriminator of the single-stage text-to-image pipeline aims at distinguishing the real text-image pair \((c_i, I_i)\) from the fake text-image pair \((\hat{c}_i, \hat{I}_i)\).

**Latent-space analysis.** For a pretrained and fixed generator \( G(z, (w,s)) \), the quality of the generated sample depends on the random latent code \( z \), word embeddings \( w \) and the corresponding sentence vector \( s \). Consequently, the output of the network only relies on \( z \) when determining the input text description. It implicitly means that if we ignore the linguistic space of the conditional input-text probes, \( G(z, (w,s)) \) can be regarded as a deterministic function \( G: \mathcal{Z} \rightarrow \mathcal{X} \). Here, \( \mathcal{Z} \) represents the latent space, in which the latent code \( z \in \mathbb{R}^l \) is commonly sampled from a \( l \)-dimension normal distribution. \( \mathcal{X} \) denotes the space of synthesized images including visually realistic samples as well as implausible generated pictures. Moreover, the map from \( \mathcal{Z} \) to \( \mathcal{X} \) is not surjective [57]. Accordingly, even a superior text-to-image generation generator fails to ensure the quality of a synthesized sample, given random latent vectors. It implicitly means that the space of synthesized samples consists of high-resolution pictures from \textit{Good} latent vectors as well as unreasonable images from \textit{Bad} latent codes. However, we do not know how to construct a successful sample that would be accepted by human users. In order to promote the applicability of text-to-
4.2.2 Pairwise linear interpolation of latent codes

It has been extensively observed [94], [133] that when performing the pairwise linear interpolation of latent codes, the appearance and the semantics of generated samples change continuously along with latent vectors. It demonstrates that linear interpolation analysis between two random latent vectors plays a significant role in understanding the space of synthesized images. We therefore conduct the linear interpolation between a successful starting-point latent vector \( z_0 \) and a successful end-point latent code \( z_1 \) in order to analyze the relation between successful synthesized images in the space of generated data. To be specific, the pairwise linear interpolation of latent codes is defined as:

\[
f(\gamma) = G((1 - \gamma)z_0 + \gamma z_1, (w, s)) \quad \text{for} \quad \gamma \in [0, 1]
\]  \hspace{1cm} (4.5)

where \( \gamma \) is a scalar mixing parameter and \((w, s)\) are word/sentence embedding vectors. In an attempt to quantitatively measure if there is a smooth transition from a perceptually plausible sample to a successful generated image, we calculate the LPIPS score [128] and the perceptual loss [46] which reflect the diversity between two close interpolation samples.

We discover that the generator is likely to synthesize a set of high-resolution pictures based on the pairwise linear interpolation between two Good latent codes.
This would imply that there may be a close relationship between successful synthesized images in the space of generated data. That is to say, we may acquire a range of visually realistic pictures by sampling the latent vectors around a *Good* latent code.

To further explore the semantic relationship between a plausible sample and an inadequate image in the space of synthesized images, we visualize the samples generated by linearly interpolating a successful starting-point latent vector and an unsuccessful end-point latent code. In addition, we compute the LPIPS score and the perceptual loss between two close interpolation images. We empirically observe that although the first and last part of interpolation results change gradually with the variations of the latent vectors, both the LPIPS score and the perceptual loss between intermediate samples are the largest and considerably increase, which we detail in Section 4.3.2. In other words, when linearly interpolating an unsuccessful latent code and a *Good* latent vector, the appearance and the semantics do not always vary smoothly along with the latent vectors. We therefore make the assumption that there exists a non-linear boundary separating successful generated images from unsuccessful synthesized samples in the space of synthetic images, which is visualized in Fig. 4.3. It implicitly means that image quality in the space of synthesized samples may be distinguished. Suppose we have a non-linear image-quality function \( f_q : I \rightarrow t \), where \( I \) denotes a generated image sample and \( t \) represents the corresponding quality score. We are able to classify a synthesized sample as realistic or unsuccessful.

### 4.2.3 *Good & Bad* data set creation

Our goal is to train a powerful classifier that can distinguish successful generated samples from unsuccessful synthetic images. To this end, we built two novel data sets (i.e., the *Good & Bad* bird and face data sets) conditioned on the CUB bird data set [108] and the Multi-Modal CelebA-HQ data set [116], respectively. The *Good & Bad* data set is a collection of perceptually realistic as well as implausible samples generated by a well-trained and fixed text-to-image GAN architecture. The construction of the data set is based on a pilot study on initial manual labeling (210+210 samples), which was used as the training set for automatic good vs bad binary classification. However, such a data set is too small to obtain a training set suitable for end-to-end, deep-learning based quality classification of generated images. Using strict criteria, an extended collection of mixed manual and automatic ‘good’ and ‘bad’ samples was constructed, within one day. Specifically, the used *Good & Bad* bird data set consists of 6,700 synthesized samples, i.e., 2,700 *Good* and 4,000 *Bad* birds. The *Good & Bad* face data set contains 2,000 successful generated faces as well as 2,000 unsuccessful synthetic faces. A summary of the *Good & Bad* data set is reported in Table 4.1. We visualize
Table 4.1: Statistics of the Good & Bad bird and face data sets. ‘Bird’ represents the Good & Bad bird data set and ‘Face’ denotes the Good & Bad face data set.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>train</th>
<th>test</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bird</td>
<td>5,200</td>
<td>1,500</td>
<td>6,700</td>
</tr>
<tr>
<td>Face</td>
<td>3,200</td>
<td>800</td>
<td>4,000</td>
</tr>
</tbody>
</table>

a snapshot of our data set in Fig. 4.4. Below, we describe the procedure followed to construct the Good & Bad data set.

**Image collection.** The first stage of creating the Good & Bad data set involves producing a large set of candidate samples for each data set. The proposed DiverGAN has the ability to adopt a generator/discriminator pair to produce diverse, perceptually plausible and semantically consistent pictures, given a textual description and different injected noise on the latent vector. We therefore choose a pretrained DiverGAN generator to acquire candidate images. We generated 30,000 synthesized samples as the basis for the selection of a Good & Bad bird data set and a Good & Bad face data set, respectively.

**Image selection.** Given a variety of candidate pictures, we choose images according to the following criteria:

1) A successful generated image is supposed to have vivid shape, rich color distributions, clear background as well as realistic details. For the face data set, photo-realistic images should also have pleasing, undistorted facial attributes (e.g., eyes, hair, makeup, head and mouth) and expressions.

2) A synthetic picture with strange shape, blurry background or unclear color is viewed as Bad. Meanwhile, we reject faces with an implausible facial appearance or ornamentation (e.g., hat and glasses) as unsuccessful samples.

3) We exclude ambiguous images of the type where also for the human judge, the classification as Good or Bad is difficult. For instance, a bird with only a slightly strange body (e.g., lacking legs) is judged as an ambiguous-quality picture.

For the Good & Bad bird data set, we find it inefficient to manually choose thousands of plausible birds from 30,000 collected samples. To reduce the selection labor, we propose a process to obtain the desired birds as follows (depicted in Fig. 4.5):

1) Based on the principles mentioned before, we select 420 synthesized samples (i.e., 210 Good and 210 Bad birds) as the initial Good & Bad bird data set, which is split into a training set (i.e., 150 Good and 150 Bad birds) and a testing set (i.e., 60 Good and 60 Bad birds). We intend to use these labeled samples to train a simple classification model to try to predict the quality classes of synthesized images. However, it is difficult to directly apply a traditional classifier (e.g., a linear SVM) to separate realistic images from inadequate samples, since the image
4.2 Proposed Methodology

(a) The Good & Bad bird dataset

(b) The Good & Bad face dataset

Figure 4.4: A snapshot of the Good & Bad bird (three top rows) and face (three bottom rows) data sets: the left column is from (a) the Good data set; the right column is from (b) the Bad data set. These samples are synthesized by our proposed DiverGAN generator [133].

instances exist in a non-linear manifold [112]. In the meantime, we cannot train a deep neural network (e.g., VGG [97]) from scratch to label a synthetic sample as Good or Bad due to the small number of the samples in the initial Good & Bad training set. Bengio et al. [6] postulate that deep convolutional networks have the ability to linearize the manifold of pictures into a Euclidean subspace of deep features. Inspired by this hypothesis, we expect that Good and Bad samples can be classified by an approximately linear boundary in such deep-feature space.

2) We adopt the publicly available VGG-16 network trained on ImageNet to transform the image samples from the training set (i.e., 150 Good and 150 Bad samples) into the deep-feature representation of layer VGG-16/Conv5_1. The obtained deep features and the corresponding labels (i.e., Good and Bad) are used to fit a linear SVM model for automatic labeling of the samples in the deep-feature space. To evaluate the performance of the model, we transform the testing samples (i.e., 60 Good and 60 Bad birds) into deep-feature vectors while applying the learned SVM boundary to predict the classes for the unseen samples.

3) In order to harvest an expanded set of Good or Bad samples, we use the trained SVM model to automatically label the 30,000 collected birds. We manually
choose 2,700 Good and 2,000 Bad birds from the images that are classified as Good, which is not a laborious task due to the performance of the SVM. Moreover, to boost the diversity of Bad birds on our data set, we select 2,000 Bad birds from the samples that are predicted as Bad. Finally, 2,700 Good and 4,000 Bad birds are acquired as the final, expanded Good & Bad bird data set. Also for the faces, we discovered that it is easy to label the synthesized samples as Good or Bad. We therefore manually select 2,000 Good and 2,000 Bad samples from 30,000 synthetic faces for the Good & Bad face data set. The manual selection was realized in one day.

**Splitting of the data set.** The Good & Bad face data set is randomly divided into the training and test sets with a ratio of 4:1. After the splitting, the training set comprises 3,200 images, i.e., 1,600 Good and 1,600 Bad faces. The test set consists of 800 samples including 400 Good and 400 Bad faces. The Good & Bad bird data set contains 6,700 birds, where 5,200 images (i.e., 2,200 Good and 3,000 Bad birds) belong to the training set and the other 1,500 images (i.e., 500 Good and 1,000 Bad birds) belong to the test set.

### 4.2.4 Synthetic samples classification

Given the extensive training set obtained in this manner, it is now possible to do the quality classification by end-to-end deep learning instead of using an unmodified, pretrained CNN and a SVM. To fully automatically distinguish successful synthesized samples from unrealistic images, we attempt to fine-tune a pretrained CNN model (e.g., ResNet [37]) on the proposed Good & Bad data set, which we will detail in Section 4.3.4. We expect that this approach is able to achieve the best results. We therefore have the ability to effectively and
efficiently identify photo-realistic samples from generated images while acquiring corresponding Good latent vectors. These Good latent codes can be exploited for further research, facilitating and extending the applicability of text-to-image generation in practice. For instance, we can produce a wealth of high-quality samples by conducting the pairwise linear interpolation between Good latent codes, e.g., for the purpose of data augmentation. Given a Good latent vector, we can synthesize several similar but semantically-diverse pleasing generated samples via latent-space navigation, which will be discussed in the next chapter.

4.3 EXPERIMENTS

In this section, we evaluate the proposed Good & Bad data sets, i.e., the Good & Bad bird data set and the Good & Bad face data set, both qualitatively and quantitatively. The details of experimental settings are described in Section 4.3.1. Afterwards, we visualize the results of the pairwise linear interpolation of latent codes in Section 4.3.2. Subsequently, the results on the Good & Bad data sets are presented in Section 4.3.3 and Section 4.3.4.

4.3.1 Experimental settings

Datasets. We perform a set of experiments on two broadly utilized text-to-image data sets, i.e., the CUB bird [108] and Multi-Modal CelebA-HQ [116] data sets. The Multi-Modal CelebA-HQ data set is composed of 24,000 and 6,000 faces for training and testing, respectively. Each face is annotated with 10 sentences.

Implementation details. We take the presented DiverGAN generator as the backbone generator, which was pre-trained on the CUB bird and Multi-Modal CelebA-HQ data sets. The image size of the proposed Good & Bad data set is set to 256 × 256 × 3. We set the output dimension of the CNN models (e.g., ResNet [37] and VGG [97]) to 2. We adopt the Adam optimizer [49] with a batch size of 64 to fine-tune the classification network pre-trained on ImageNet. We utilize the learning-rate finder technique [98] to acquire a suitable learning rate. The one cycle learning rate scheduler [99] is leveraged to dynamically alter the learning rate whilst the model is training. The steps of the pairwise linear interpolation of latent codes are set to 10. Our methods are implemented by PyTorch [81]. We conduct all the experiments on a single NVIDIA Tesla V100 GPU (32 GB memory).

4.3.2 Results of the pairwise linear interpolation of latent codes

The interpolation results between two Bad latent codes are presented in Fig. 4.6. We can see that although the visual appearances of birds and the background
change gradually along with latent codes, intermediate images generated by linearly interpolating two unsuccessful latent vectors look blurry. It indicates that the generator produces a set of unrealistic images when the samples based on the first and the last latent vectors are not perceptually plausible. That is to say, there may exist a significant relationship between unsuccessful synthesized samples in the distribution of generated images.

To further explore the semantic relation between successful generated samples in the space of synthesized images, we show the results of the pairwise linear interpolation between a successful starting-point latent code and a successful end-point latent vector in Fig. 4.7. We can observe that DiverGAN yields a series of high-quality pictures when performing the pairwise linear interpolation between two Good latent codes. It demonstrates that we may obtain numerous high-resolution images by sampling latent vectors around a Good latent code.

The pictures generated by linearly interpolating a Good starting-point latent code and a Bad end-point latent vector are depicted in Fig. 4.8. It can be seen that the first part of interpolation results is visually realistic, whereas the final part is perceptually unclear. In addition, the background (2nd row), the visual appearances of footholds, the orientations, the shapes and the positions of birds (1st – 4th row) do not vary continuously along with latent codes, which suggests that successful generated samples and unsuccessful synthesized images may be separated in the distribution of synthetic samples. A non-linear classifier may be trained to classify a synthetic sample as Good or Bad.
Good Latent code

Figure 4.7: Four image generation results with linear interpolation between two Good latent vectors in DiverGAN on the CUB data set. The generator synthesizes a series of visually realistic samples.

To quantitatively compare the diversity between two close samples, we plot the interpolation samples between two successful latent vectors as well as the corresponding LPIPS score and the perceptual loss in Fig. 4.9. We can see that both the LPIPS score and the perceptual loss in Fig. 4.9 (1b), Fig. 4.9 (1c), Fig. 4.9 (2b) and Fig. 4.9 (2c) are always low and do not largely increase. Furthermore, all the points in Fig. 4.9 are below the red line, which is an approximate boundary distinguishing smooth changes from discontinuous variations and is determined by our observations. As shown in Fig. 4.9 (1a) and Fig. 4.9 (2a), both the background and the visual appearances of footholds in Fig. 4.9 (1a), both the shapes and the positions of birds in Fig. 4.9 (2a) change gradually with the variances of latent codes. The quantitative results are in line with the qualitative analysis.

To better understand the transition process from a successful synthesized sample to an unsuccessful generated image, we visualize the results of the pairwise linear interpolation between a Good latent code and a Bad latent vector and the corresponding LPIPS score and perceptual loss in Fig. 4.10. As can be observed in Fig. 4.10 (1a), for the first five and the last two pictures, both the background and the visual appearances of footholds vary gradually along with latent vectors. However, the background, the visual appearances of footholds, the positions, the shapes and even the orientations (7th → 8th sample) of the birds do not change continuously from the 6th image to the 8th sample. As can be seen in Fig. 4.10 (1b) and Fig. 4.10 (1c), the increase of the 6th point (6th → 7th sample) is the largest and the 7th point (7th → 8th sample) obtains the highest score for both the LPIPS score and the perceptual loss. Meanwhile, both points are over the
Figure 4.8: Four examples of the pairwise linear interpolation between a Good latent code and a Bad latent vector. The red line emphasizes a discontinuous transition within the linear-interpolation samples.

red line. The results of Fig. 4.10 (1b) and Fig. 4.10 (1c) match what observe in Fig. 4.10 (1a), indicating that the visual appearances of the birds does not always vary smoothly along with latent codes.

As shown in Fig. 4.10 (2a), the visual appearances of the birds and the background of the first three and the last four images change gradually with the variations of latent vectors, whereas the visual appearances of footholds, the shapes and the orientations of birds do not change continuously from the 4th image to the 6th sample. Fig. 4.10 (2b) and Fig. 4.10 (2c) show the corresponding quantitative results, where the increase of the 4th point (4th → 5th sample) is the largest for both the LPIPS and the perceptual loss. The 4th point and the 5th point (5th → 6th sample) obtain the highest score for the LPIPS score and the perceptual loss, respectively. Moreover, both points are over the red line. Both the quantitative and qualitative results show a discontinuous transition from the 4th image to the 6th sample.

The above analysis suggests that there may exist a non-linear boundary separating Good samples from Bad images in the space of generated images. In order to effectively draw successful generated samples from all synthesized images, we will attempt to discover the non-linear image-quality boundary in the next section.
Figure 4.9: Two examples of the pairwise linear interpolation of latent vectors (Good → Good). The interpolation results in (1a) and (2a) show a fluent transition. The dashed red line in (1b), (1c), (2b) and (2c) is an approximate boundary distinguishing smooth changes from discontinuous variations, determined by our observations. The index number represents the comparison, starting with 0, i.e., the comparison between the first and the second image on the left. The continuity is quantitatively revealed both in LPIPS and in perceptual loss.
(a) The results of pairwise linear interpolation of latent codes (good → bad)

(b) The LPIPS difference score

(c) The perceptual difference loss

(1) The first example

(2) The second example

Figure 4.10: Two examples of the pairwise linear interpolation of latent codes (Good → Bad / Bad → Good). The red bounding box in (1a) and (2a) emphasizes a discontinuous range within the linear-interpolation results. The dashed red line in (1b), (1c), (2b) and (2c) is an approximate boundary distinguishing smooth changes from discontinuous variations, determined by our observations. The index number represents the comparison, starting with 0, i.e., the comparison between the first and the second image on the left. The discontinuity is quantitatively revealed both in LPIPS and in perceptual loss. Please note that the points in the graph represent \( n \) successive pairwise difference scores, hence a red marking of \( n + 1 \) images in the bars.
4.3 Experiments

Table 4.2: Classification accuracy on the separation boundary with respect to image quality. *Image* refers to a direct application of SVM on the image pixels. *PCA-Image* refers to using PCA on the image pixels after reducing the dimensionality to 128 and applying SVM to identify realistic samples. *Latent Code* refers to the direct application of SVM in the latent space.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image</td>
<td>70.0</td>
</tr>
<tr>
<td>PCA-Image</td>
<td>73.3</td>
</tr>
<tr>
<td>Latent Code</td>
<td>75.8</td>
</tr>
<tr>
<td>VGG-16(conv5_3)</td>
<td>94.2</td>
</tr>
<tr>
<td>VGG-16(conv5_2)</td>
<td>96.7</td>
</tr>
<tr>
<td>VGG-16(conv5_1)</td>
<td>97.5</td>
</tr>
</tbody>
</table>

4.3.3 Results on the initial *Good & Bad* bird data set

We try different methods to classify a synthetic sample as *Good* or *Bad* on the initial *Good & Bad* bird data set, i.e., 210 *Good* and 210 *Bad* birds. The results are reported in Table 4.2. Here, we discover that all methods using the learned feature vectors of a well-trained VGG-16 network achieve over 94%, suggesting that there exists a (almost) linear boundary in the deep-feature space which can accurately distinguish *Good* samples from *Bad* samples. In addition, the conv5_1 activation in the pretrained network obtains the best performance (accuracy: 97.5%). We also attempted to employ the SVM with radial basis function (RBF) kernel to classify deep features, acquiring the same result as the linear SVM. Moreover, it can be observed that directly operating on the image pixels (accuracy: 70.0%) and the latent space (accuracy: 75.8%) does not work well for the classification of *Good* and *Bad* samples/latent codes. To boost the accuracy, we conduct PCA on the image pixels to reduce the dimension to 128 and apply a linear SVM to identify realistic samples. However, the accuracy is only improved by 3.3%. The above results confirm the effectiveness of our proposed framework for automatically acquiring the successful synthesized birds.

We visualize some typical output samples selected from the test set (Ngood=60, Nbad=60) in Fig. 4.11 according to their distance to the decision boundary of the trained SVM. It can be observed that *Good* samples are distinguishable from *Bad* samples. Meanwhile, the *Bad* birds around the boundary may have higher quality than the *Bad* birds far from the decision boundary. It should be noted that in non-ergodic problems, where there is not a natural single signal source for the *Good* (or the *Bad*) images, but there rather exists a partitioning of space, the SVM discriminant value for a sample is not guaranteed to be consistent with
the intuitive prototypicality of the heterogeneous underlying class [79] due to the lack of a central density for that class.

### 4.3.4 Results on the Good & Bad data set

The classification results. We fine-tune the pretrained CNN models (i.e., ResNet and VGG) on the Good & Bad data set in order to accurately predict the quality classes of generated images. The comparison between VGG-11, VGG-16, VGG-19, ResNet-18, ResNet-50 and Res-Net-101 with respect to the classification performance on the Good & Bad bird and face data sets is shown in Table 4.3. We can observe that ResNet-50 achieves the best result (accuracy: 98.09%) on the Good & Bad bird data set and ResNet-101 impressively acquires the accuracy of 99.16% on the Good & Bad face data set. It can also be seen that ResNet performs better than VGG and all the networks obtain a better than 95% accuracy on both the Good & Bad bird data set and the Good & Bad face data set. The above results demonstrate that the Good and Bad samples in the space of synthetic images can be effectively distinguished by a well-trained deep convolutional network.
Table 4.3: Classification performance of the deep convolutional networks on the Good & Bad bird and face data sets. Bird refers to the Good & Bad bird data set and Face refers to the Good & Bad face data set. The best results are in bold.

<table>
<thead>
<tr>
<th>Method</th>
<th>Dataset</th>
<th>Bird</th>
<th>Face</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGG-11</td>
<td></td>
<td>96.53</td>
<td>98.08</td>
</tr>
<tr>
<td>VGG-16</td>
<td></td>
<td>95.70</td>
<td>97.84</td>
</tr>
<tr>
<td>VGG-19</td>
<td></td>
<td>97.85</td>
<td>98.20</td>
</tr>
<tr>
<td>ResNet-18</td>
<td></td>
<td>97.59</td>
<td>98.68</td>
</tr>
<tr>
<td>ResNet-50</td>
<td></td>
<td>98.09</td>
<td>98.56</td>
</tr>
<tr>
<td>ResNet-101</td>
<td></td>
<td>97.79</td>
<td><strong>99.16</strong></td>
</tr>
</tbody>
</table>

Figure 4.12: The visualization for the samples on the Good & Bad data set by utilizing the PCA [115] in (1a) and (2a) and t-SNE [67] in (1b) and (2b). In this figure, the yellow color represents the Good sample and the purple color represents the Bad image.

Visualization of the learned representation. To visually investigate the distribution of the features learned by the CNN models, i.e., ResNet-50 for the Good & Bad bird data set and ResNet-101 for the Good & Bad face data set, we exploit the PCA [115] and t-SNE [67] approaches to embed the samples on the Good & Bad data set into a 2-dimensional space as shown in Fig. 4.12. From this figure, we can see that the learned representations of the classification networks from different classes (i.e., Good and Bad) are well separated, indicating that the image classification models can project the plausible and unrealistic samples into two diverse latent spaces. Therefore, discovering photo-realistic samples from synthesized images is feasible. It can also be observed that the samples of different categories on the Good & Bad face data set are more scattered than the Good & Bad bird data set, which demonstrates that ResNet-101 trained on the Good & Bad face data set.
set performs better than ResNet-50 trained on the Good & Bad bird data set. In other words, faces are easier to recognize than birds, which is consistent with the classification score.

**Explaining the classification prediction.** We leverage three different methods (i.e., Layer-CAM [45], integrated gradient [102] and extremal perturbation [23]) to explain the image classification prediction obtained by ResNet-50 trained on the Good & Bad bird data set and ResNet-101 trained on the Good & Bad face data set. Fig. 4.13 shows the explanation for the top 1 predicted class achieved by ResNet-50 trained on the Good & Bad bird data set, suggesting that the classification network derives the results by concentrating on the discriminative regions of the birds. For instance, Layer-CAM visualization (2\textsuperscript{nd} and 6\textsuperscript{th} column) localizes the heads and belly of the birds. Meanwhile, integrated gradient (3\textsuperscript{rd} and 7\textsuperscript{th} column) and extremal perturbation (4\textsuperscript{th} and 8\textsuperscript{th} column) correctly highlight the branches and the whole bodies of the birds, pinpointing the reason why the synthesized samples are classified into the corresponding categories. More importantly, the

**Figure 4.13:** Explaining the image classification prediction made by ResNet-50 on the Good & Bad bird data set using Layer-CAM [45], integrated gradient [102] and extremal perturbation [23]. The left half of the grid is from (a) the Good data set; the right half of the grid is from (b) the Bad data set, separated by the dashed line.
4.3 experiments

Figure 4.14: Explaining the image classification prediction made by ResNet-101 on the Good & Bad face data set using Layer-CAM [45], integrated gradient [102] and extremal perturbation [23]. The left column is from (a) the Good data set; the right column is from (b) the Bad data set, separated by the dashed line.

<table>
<thead>
<tr>
<th>Original image</th>
<th>Layer CAM</th>
<th>Integrated Gradient</th>
<th>Extremal Perturbation</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Original image" /></td>
<td><img src="image2" alt="Layer CAM" /></td>
<td><img src="image3" alt="Integrated Gradient" /></td>
<td><img src="image4" alt="Extremal Perturbation" /></td>
</tr>
<tr>
<td><img src="image5" alt="Original image" /></td>
<td><img src="image6" alt="Layer CAM" /></td>
<td><img src="image7" alt="Integrated Gradient" /></td>
<td><img src="image8" alt="Extremal Perturbation" /></td>
</tr>
<tr>
<td><img src="image9" alt="Original image" /></td>
<td><img src="image10" alt="Layer CAM" /></td>
<td><img src="image11" alt="Integrated Gradient" /></td>
<td><img src="image12" alt="Extremal Perturbation" /></td>
</tr>
<tr>
<td><img src="image13" alt="Original image" /></td>
<td><img src="image14" alt="Layer CAM" /></td>
<td><img src="image15" alt="Integrated Gradient" /></td>
<td><img src="image16" alt="Extremal Perturbation" /></td>
</tr>
<tr>
<td><img src="image17" alt="Original image" /></td>
<td><img src="image18" alt="Layer CAM" /></td>
<td><img src="image19" alt="Integrated Gradient" /></td>
<td><img src="image20" alt="Extremal Perturbation" /></td>
</tr>
<tr>
<td><img src="image21" alt="Original image" /></td>
<td><img src="image22" alt="Layer CAM" /></td>
<td><img src="image23" alt="Integrated Gradient" /></td>
<td><img src="image24" alt="Extremal Perturbation" /></td>
</tr>
<tr>
<td><img src="image25" alt="Original image" /></td>
<td><img src="image26" alt="Layer CAM" /></td>
<td><img src="image27" alt="Integrated Gradient" /></td>
<td><img src="image28" alt="Extremal Perturbation" /></td>
</tr>
<tr>
<td><img src="image29" alt="Original image" /></td>
<td><img src="image30" alt="Layer CAM" /></td>
<td><img src="image31" alt="Integrated Gradient" /></td>
<td><img src="image32" alt="Extremal Perturbation" /></td>
</tr>
</tbody>
</table>

(a) Good Samples  (b) Bad Samples

blurry regions of the images ($6^{th}$, $7^{th}$ and $8^{th}$ column) are accurately identified by these explainable approaches.

The explanation for the image classification prediction made by ResNet-101 trained on the Good & Bad face data set is illustrated in Fig. 4.14. It can be observed that Layer-CAM visualization ($2^{nd}$ and $6^{th}$ column) localizes the noses, mouths and eyes of the faces. Integrated gradient ($3^{rd}$ and $7^{th}$ column) and extremal perturbation ($4^{th}$ and $8^{th}$ column) accurately capture the hat and the entire faces. Furthermore, the implausible regions of the faces (e.g., the hat in the $1^{st}$ row and the mouth in the $2^{nd}$ row) are correctly highlighted by these explainable approaches.

The above analysis suggests that our well-trained classifiers classify a synthetic sample as Good or Bad by focusing on the discriminative regions of the objects, i.e., birds and faces. That is to say, our classification model can accurately separate implausible regions from high-quality patches and effectively discover successful synthesized samples in the space of generated images.
4.4 Conclusion

In this chapter, we created two *Good & Bad* data sets to increase the probability of drawing successful synthesized samples for the task of text-to-image generation. Specifically, to explore the semantic relationship between successful generated pictures and unsuccessful synthesized samples in the space of synthetic data, we perform the pairwise linear interpolation between a *Good* starting-point latent code and a *Bad* end-point latent vector. We empirically discover that the visual appearance and the semantics do not always change gradually with the variations of the latent codes. We therefore make the assumption that there exists a non-linear boundary separating realistic images from inadequate samples in the space of generated images. To ensure the quality of synthetic pictures, we created a *Good & Bad* data set, both for a bird and a face-image collection, which comprises high-resolution as well as implausible synthesized samples, in which the images are chosen by following strict principles. Based on the *Good & Bad* data set, we fine-tune the deep convolutional network trained on ImageNet to classify a generated image as *Good* or *Bad*. We evaluate our presented approaches on the presented DiverGAN generator that was pre-trained on two popular data sets, i.e., the *CUB bird* and the *Multi-Modal CelebA-HQ* data sets. Extensive experimental results suggest that our well-trained classifier is able to accurately predict the quality classes of the samples from the testing set, which validates the effectiveness of our proposed methods and data sets.