Deep Representation Decomposition for Feature Disentanglement

Shang-Fu Chen
National Taiwan University
f07942144@ntu.edu.tw

Jia-Wei Yan
National Taiwan University
r06942033@ntu.edu.tw

Ya-Fan Su
Chunghwa Telecom
sutony@cht.com.tw

Yu-Chiang Frank Wang
National Taiwan University
ycwang@ntu.edu.tw

Abstract

Representation disentanglement aims at learning interpretable features, so that the output can be recovered or manipulated accordingly. While existing works like infoGAN [4] and AC-GAN [24] exist, they choose to derive disjoint attribute code for feature disentanglement, which is not applicable for existing/trained generative models. In this paper, we propose a decomposition-GAN (dec-GAN), which is able to achieve the decomposition of an existing latent representation into content and attribute features. Guided by the classifier pre-trained on the attributes of interest, our dec-GAN decomposes the attributes of interest from the latent representation, while data recovery and feature consistency objectives enforce the learning of our proposed method. Our experiments on multiple image datasets confirm the effectiveness and robustness of our dec-GAN over recent representation disentanglement models.

1. Introduction

Recent developments of Generative Adversarial Network (GAN) [7] models result in promising progresses and achievements in image generation, which is realized by sampling latent codes from predetermined distributions, so that the recovered output images would be sufficiently realistic. In order to produce image outputs with desirable attributes (e.g., gender, expression, etc.), feature disentanglement aims at decomposing the above latent representation into distinct parts, each corresponding to particular properties. For example, infoGAN [4] and AC-GAN [24] are well-known GAN-based models for determining attribute of interests in the latent representation, learned in unsupervised and supervised ways, respectively.

Despite promising performances, these works are not able to be directly applied on existing/trained generative models, since they need additional dimension for describing attributes, and the attributes should be pre-determined. Thus, their disentanglement mechanisms must be trained together with generative models, and are not able to take advantage from existing well-trained generative models directly. With the scale of generative models growing, training of such models become more time- and resource-consuming. Moreover, the need of deciding attributes before training process makes their feature disentanglement less flexible, because generative models have to be trained from scratch again once the attributes are changed. Additionally, these works are not designed to handle image inputs, and thus cannot be easily applied to disentangle specific attributes from input images.

Image-to-image translation can be viewed the task of manipulating attributes of image styles [20, 22]. Typically, pairwise network modules are trained for converting images across domains, and thus only a pair of image attributes can be manipulated. Recently, a number of models [21, 5] are developed to handle multiple image attributes in a unified framework. However, these works are still not applicable for existing GANs, since they need additional dimensions for describing pre-determined attributes. On the other hand, variational autoencoder (VAE) based models like [18, 1] have been applied for describe images with specific attributes. For example, it has been shown that attribute vectors representing blond hair can be derived by comparing face images with/without blond hair, and thus the resulting vector can be applied for image manipulation.

Instead of explicitly decomposing latent representation into disjoint parts, we propose a unique decomposition-GAN (dec-GAN) for disentanglement. As depicted in Figure 1, our disentanglement mechanism focuses on extracting attributes of interest (e.g., pose, expression, etc.) from latent representation, while the generator is fixed. Depending on the attribute of interest, dec-GAN is guided by an
attribute classifier trained for distinguishing the attribute. Together with image recovery objectives, dec-GAN decomposes visual features from a joint latent representation into separate ones associated with content and attribute of interest. With the above disentangled features, our dec-GAN is able to utilize existing generative models for describing each type of disentangled features, which allows improved and interpretable feature representations, along with additional flexibility in determining the attributes of interest after the generator is trained.

We now highlight the contributions of work as follows:

- We propose a learning framework for representation disentanglement, which uniquely decomposes features of existing GAN-based models into interpretable representations.
- Our learning framework does not require predetermined or disjoint latent representations to describe attributes in advance, and thus exhibits additional flexibility in determining the attributes of interest.
- Our experiments confirm that our model successfully decomposes latent features derived by existing GANs for image manipulation and classification.

2. Related Work

Image Generation via GAN

Generative adversarial network (GAN) \[7, 12, 13, 8\] has brought remarkable progresses for image generation. The main contribution of GAN is the introduction of the adversarial loss that enforces distribution matching of generated images and the realistic ones. As an extension, VAE-GAN \[18\] combines VAE \[15\] with a GAN which takes images instead of latent codes as inputs, while the recovered outputs are with improved quality when comparing to VAE.

However, since GAN-based models for image generation generally take latent code sampled from prior distributions as inputs, the lack of ability in representing particular images would be a concern. As a result, recent research works \[10, 20, 5\] turn their attention into representation disentanglement, which focuses on separating latent representations into disjoint parts modeling visual content and attribute information.

Images Manipulation via Image Translation

Image-to-Image translation aims to learn the mapping between images in different domains, which requires disentanglement and transfer of images of particular attributes. Models like pix2pix and its extension \[11, 6\] require pairwise training image data across domains during learning. With only collection of images of the same attribute in each domain, DRIT \[20\] and MUNIT \[10\] advance the architecture of UNIT \[22\] with adversarial learning for extracting domain-invariant features. Recently, UFDN \[21\] purposes a unified network architecture for image translation across multiple domains, which derives domain-invariant and domain-specific representation of images from different domains.

Representation Disentanglement

Representation disentanglement \[4, 9, 14, 16, 17, 23, 24, 28, 31, 25\] aims at learning an interpretable representation from image variants, which can be realized in unsupervised or supervised settings. For example, with supervision of labeled data, AC-GAN \[24\] factorizes representations into disjoint parts describing visual content and attribute information, respectively (e.g., image \[11\], text \[26, 30\]) during training. On the other hand, if training data are unlabeled, infoGAN \[4\] performs representations disentanglement by maximizing the mutual information between latent variables and data variation. Nevertheless, with label supervision or not, both AC-GAN and infoGAN choose to disentangle a distinct code from the latent representation during training. As a result, none of them are able to apply disentanglement to pre-trained existing GAN models.

To take advantage from existing well-trained generative models directly, our dec-GAN aims to apply disentanglement mechanisms on existing generative models. \[29\] and \[27\] deal with similar tasks that learns disentangled features based on existing generative models. While \[29\] performs unsupervised feature disentanglement to find interpretable directions in latent space, their method is not able to disentangle and manipulate particular attributes of interest. On the other hand, \[27\] explores the GAN-based latent space for producing face images with specific attributes. Their model generates the output images from random vectors sampled at the latent space. That is, their model is not designed for input face images. To conduct real image manipulation, an additional GAN inversion technique must be included to find the best latent codes for target images. To address the above problems, our dec-GAN is guided by the classifier pre-trained on the attributes of interest, and thus be able to manipulate particular attributes for input face images.

3. Decomposition-GAN (dec-GAN) for Disentanglement

The use of GAN-based deep learning models for image generation is to produce realistic images, typically given a latent code \(z \sim \mathcal{N}(0, 1)\) as the input. While works like infoGAN \[4\] and AC-GAN \[24\] utilize particular latent codes for describing particular image attributes, such code and attributes need to be specified in advance. To address this problem, we advance GAN-based models and decomposition techniques, and propose decomposition-GAN (dec-}
GAN) for representation disentanglement.

As illustrated in Figure 1, our dec-GAN decomposes the latent code $z$ into content code $z_c$ and attribute code $z_a$ while satisfying $z = z_c + z_a$. In other words, based on existing latent feature $z$, our goal is to decompose it into content and attribute representations $z_c$ and $z_a$, respectively. It is worth noting that, as verified in Section 4, our dec-GAN can utilize existing state-of-the-art generative models. Compared to prior disentanglement models which result in disjoint hard latent codes describing attributes of interest, our model allows extracting and determining continuous attribute features. As depicted in Figure 1, we utilize two separate encoders $E_c$ and $E_a$ for extracting $z_c$ and $z_a$, respectively. The reconstruction output is denoted as $G(z_c, z_a) = G(z | z = z_c + z_a)$. We now discuss the design of our dec-GAN.

### 3.1. Auxiliary Attribute Classification and Disentanglement

In our dec-GAN, we first utilize the idea of data recovery to encourage generated images to be sufficiently realistic. For this reconstruction loss, we consider the L1 distance between the reconstructed and input images:

$$L_{rec} = |G(z_c, z_a) - x|.$$ (1)

Following VAE [15], VAE-GAN [18] and DRIT [20], we fit the distributions of encoded content and attribute features to normal distributions, which allow improved/continuous data representation ability. This can be achieved by minimizing the Kullback–Leibler divergence (KLD) between each distribution and $\mathcal{N}(0, 1)$. However, since the disentangled content and attribute features describe distinct information, we do not expect them to fit the same normal distribution. Therefore, we calculate the KLD loss for each feature as follows,

$$L_{KL,c} = \mathbb{E}[KL(P(z'_c) || \mathcal{N}(0, 1))], z_c = E^f_c(z'_c),$$  (2)

$$L_{KL,a} = \mathbb{E}[KL(P(z'_a) || \mathcal{N}(0, 1))], z_a = E^f_a(z'_a),$$  (3)

where $E^f_c$ denotes the final fully connected layer of content encoder $E_c$, and $E^f_a$ denotes the final fully connected layer of attribute encoder $E_a$.

To ensure the encoded $z_c$ and $z_a$, describing content and attribute information, respectively, we apply a classifier $C$ pre-trained on the attribute of interest, and observe the resulting loss to guide the learning of $E_a$. Thus, this guided loss is calculated as:

$$L_{guide} = |C(x) - C(G(\tilde{z}_c, z_a))|,$$  (4)

where $C(\cdot)$ indicates the classifier. We note that, $\tilde{z}_c$ denotes a randomly sampled content feature, $\tilde{z}$ is sampled from $\mathcal{N}(0, 1)$ and then is passed through the final fully connected layer of $E_c$. Thus, we have $\tilde{z}_c = E^f_c(\tilde{z})$, and the image with identical attribute but random content can be produced as $G(\tilde{z}_c, z_a)$.

From 4, we see that the enforcement of classification output similarity between an input image $x$ and a synthesized one with the same $z_a$ yet with a random content $\tilde{z}_c$, would ensure our $E_c$ and $E_a$ to extract attribute-invariant and attribute-dependent representations, respectively. That is, the deployment of the classifier $C(\cdot)$ in Fig. 1 would guide the attribute encoder $E_a$ to extract attribute-dependent information by equation 4. With equation 4 ensuring the quality of reconstruction, attribute-invariant information would be encoded by content encoder $E_c$ for fair reconstruction.
3.2. Feature Consistency for Disentanglement

With the above guidance of the attribute classifier and the use of generative models, we have $E_a$ extract latent attribute features. With this classifier to be replaced by those pre-trained on preferable attributes of interests, one can easily extend the above architecture to disentangle the corresponding attributes. To further ensure our decomposed $z_c$ and $z_a$ from $z$ contain only content and attribute information, respectively, we advance feature consistency losses during the training of our dec-GAN. As illustrated in Figure 1, this is achieved by minimizing the content and attribute feature consistency loss defined as follows:

$$L_{c}^{\text{const}} = |E_c(G(z_c, \tilde{z}_a)) - z_c|, \quad (5)$$
$$L_{a}^{\text{const}} = |E_a(G(z_c, z_a)) - z_a|. \quad (6)$$

Note that $G(z_c, \tilde{z}_a)$ indicates the synthesized image with the same content as that of input $x$ but with different attributes $\tilde{z}_a = E_a^{\text{fc}}(\tilde{z})$. Similarly, we have $G(z_c, z_a)$ denote the generated image with the same attributes as those of $x$ but with different content information via $z_c$. By observing the above feature consistency, both $E_c$ and $E_a$ would extract associated content and attribute features, realizing the decomposition of $z$ into $z_c$ and $z_a$, respectively.

4. Experiment

We consider image datasets of MNIST [19] and CMU Multi-PIE [8] for our experiments. The former consists of 60,000/10,000 training/test digit images of 10 classes, while the latter contains face images with multiple viewpoint, illumination and expression variations. We only use a subset of CMU Multi-PIE with 5 viewpoints and smiling expression variation, which consists of 68,810 images.

4.1. Implementation Details

Since our proposed architecture does not limit the use of particular GAN models, we first follow the backbone of VAE-GAN [13]. VAE-GAN has the encoder consists of three convolution layers with filter size 5x5 followed by two fully connected layers. The generator $G$ consists of a fully connected layer followed by four convolution layers. The discriminator in the pre-trained generative model follows BEGAN [2] as an autoencoder structure, which is a combination the same structure for both encoder and decoder. The attribute classifier $C$ has 3 convolution layers followed by two fully connected layers.

In addition, we consider a second generative model with a deeper backbone [20] and refer Res-GAN to this generative model. The encoder of Res-GAN consist of three convolution layers, four residual blocks and two fully connected layers. The generator $G$ consists of a fully connected layer, four residual blocks and three fractional layers. The discriminator is a multiscale structure, which is composed of five convolution layers.

For $E_c$ and $E_a$ in our dec-GAN, we simply utilize the same encoder structure of the model to be decomposed. To be more precise, $E_c$ and $E_a$ are both composed of three convolution layers followed by two fully connected layers when decomposing VAE-GAN; $E_c$ and $E_a$ both consist of three convolution layers, four residual blocks and two fully connected layers when decomposing Res-GAN.

The pseudo code summarizing the training of our dec-
outputs using derived $z$ put image pairs, and the second row depicts reconstructed and attribute features. The first row in Figure 2(a) show in-content features. As shown in Figure 2(a), we demonstrate the visual appearance like stroke thickness or angle as the digit categories, which are viewed as the attributes while MNIST

4.2. Image Generation and Manipulation

MNIST

For MNIST, the classifier $C$ is pre-trained to identify the digit categories, which are viewed as the attributes while the visual appearance like stroke thickness or angle as the content features. As shown in Figure 2(a), we demonstrate our image generation results using different pairs of content and attribute features. The first row in Figure 2(a) show input image pairs, and the second row depicts reconstructed outputs using derived $z_c$ and $z_a$ features. Image outputs by swapping $z_c$ and $z_a$ are shown in the third row. From this row, we see that the synthesized image would preserve the same content as those in the first two rows, while the attribute (digit category) would match the other one in the input image pair. This confirms the effectiveness of our dec-GAN in disentangling content and attribute features, while the latter is guided by a digit classifier in this case. To further verify $z$, $z_c$, and $z_a$ capture different visual information, we conduct t-SNE visualization on such features using MNIST data. Due to space limitation, such visualization results are presented in the supplementary material.

To demonstrate the ability our dec-GAN in disentangling and describing input images, we further sample random $z_c$ or $z_a$ and show the manipulated images in part (b) and (c) of Figure 2. Given input images, Figure 2(b) shows generated images using different and sampled $z_a$, while $z_a$ matches that of the input image. From this figure, we see that the thickness and angle of the strokes were altered, while the attribute (digit class) remained unchanged. On the other hand, Figure 2(c), shows image manipulation using the same $z_c$ but randomly sampled $z_a$. Since the attributes in MNIST indicate digit labels, we do not expect randomly sampled $z_a$ would fit particular digit category. For instance, the second sample in the second row in Figure 2(c) shows output digit images resembling ‘5’ and ‘6’; the fourth sample in the last row is visually similar to the digits of ‘7’ and ‘9’. This confirms that our dec-GAN is able to properly disentangle and derive continuous attribute representations.

CMU Multi-PIE

We take both VAE-GAN and Res-GAN as the backbone of our dec-GAN, and consider pose and expression (smile) as two distinct attributes of interest. We demonstrate image generation results when taking pose categories as attribute of interest in Figure 2(a). The first row in Figure 2(a) show input facial image pairs, and the second row depict reconstructed image outputs using derived $z_c$ and $z_a$. Image outputs by swapping $z_c$ and $z_a$ are shown in the third row. Comparing this row and the first two rows, we see that the manipulated facial images remained the same identity with pose information altered and matched the other one in the input image pair. Again, this confirms the effectiveness of our dec-GAN in disentangling content and attribute features; moreover compared to discrete categorical attributes in MNIST, we are able to handle continuous attributes such as poses.

In addition, we show the results when taking smiling expression as attribute of interest in Figure 2(b). Similarly, by
comparing the last row and the first two rows, we see that the manipulated images remained same facial information with only smiling expression altered. This confirms that smiling expression is able to be decomposed from the original latent feature. It is worth noting that, we decompose pose and smiling attributes from the same pre-trained generative models, confirming the flexibility of our dec-GAN in extracting the attributes of interest.

4.3. Representation Continuity

As described in Section 3.1, for each latent representation \(z_c\) and \(z_a\), we fit their distributions to prior Gaussians. To confirm that this allows our dec-GAN to derive and manipulate continuous latent features, we now conduct additional image synthesis tasks by manipulating such representations.

Content continuity

Given two images with the same attribute but with distinct content information, we interpolate the content codes \(z_c\) extracted from input images and then generate outputs by these interpolated content with same \(z_a\). The results on MNIST dataset are depicted in Figure 4(a), where the attribute indicates the digit label. We see that the thickness or angle of the strokes change continuously, which verified the ability of our \(z_c\) for representing continuous content information. The results on CMU Multi-PIE dataset are demonstrated in Figure 4(b), where the attribute denotes the pose information of faces. Thus, facial content information like gender, illumination, color of skin, etc. would be encoded in the content feature. We see that facial content alters smoothly, while the pose reserved. It is worth noting that, the digit/pose categories remain unchanged while we manipulate/interpolate the associated content features \(z_c\). This once again verifies our dec-GAN is able to disentangle content and attributes into \(z_c\) and \(z_a\), respectively.

Attribute continuity

We evaluate the attribute continuity using CMU Multi-PIE only, where continuous facial poses are viewed as attributes. Note that the attributes in the MNIST dataset are the digit categories. While we can conduct the same experiments, but it would be difficult to explicitly interpret the manipulated outputs. Given two face images of different identities and poses as inputs, we compare image outputs produced by VAE-GAN \[18\], UFDN \[21\] and our dec-GAN in Figure 5. More precisely, the output images of VAE-GAN were simply recovered by interpolating the resulting latent representation \(z\). As shown in the first row in Figure 5, the produced image outputs were generally interpolation of the two input images. For the interpolated images, both identity and pose information are blended, and one cannot easily identify the attribute (i.e., pose) of interest. Comparing the second and third rows (with outputs produced by UFDN and dec-GAN), our model was able to properly preserve the identity of one of the inputs, while the attribute (pose) can be manipulated accordingly.

Additionally, we calculate the Fréchet Inception Distance (FID scores) to evaluate the quality of images produced by VAE-GAN, UFDN and our dec-GAN. For each method, we sample 500 pairs of images and generate images with interpolated attributes, and then compare the Fréchet Inception Distance between the output images and the whole CMU Multi-PIE dataset. As shown in Table 1 our dec-GAN yielded the best FID score, which shows that dec-GAN successfully decomposes visual features from a joint latent representation into separate content and attribute of interest. On the other hand, for VAE-GAN, identity and pose information are blended, so the produced images are less realistic. As for UFDN, they used discrete latent code to represent attribute information, and failed to generate satisfactory images with interpolated attributes. Thus, the ability of our dec-GAN in disentangling and describing continuous attributes can be confirmed.

4.4. Quantitative Results

Quantitative Evaluation of \(z_c\) and \(z_a\)

We conduct quantitative experiments to examine the effectiveness of our dec-GAN in disentangling content and attribute features. With the use of CMU Multi-PIE, \(z_a\) derived by our model would be expected to contain pose information only, while \(z_c\) represents pose-invariant identity features. We then take these two types of features and perform pose and ID classification tasks, and compare the results to the uses of latent representations \(z\) derived by VAE-GAN \[18\] and UFDN \[21\].

Table 2 lists and compares classification results of different tasks using \(z\), \(z_c\) and \(z_a\). We simply apply a two-layer classifier (i.e., 2 fully-connected layers with ReLU activation, followed by a softmax layer) for comparison purposes. We do not apply additional or complex classifiers, which can possibly further improve the recognition performances. The number of classes is 5 for pose classification and 249 for identity classification. From this table, we observe that \(z_a\) yielded the best result in pose classification, while \(z_c\) resulted in the highest performances for identity classifica-

| Method | FID |
|--------|-----|
| VAE-GAN \[18\] | 56.91 |
| UFDN \[21\] | 122.81 |
| Ours | 47.88 |

Table 1. FID scores on CMU Multi-PIE. The number shows the Fréchet Inception Distance between 500 produced images with interpolated attribute and CMU Multi-PIE. Smaller FID score means the output images are more similar to the real data.
Figure 4. Image manipulation by interpolating $z_c$ from different images of (a) MNIST and (b) CMU Multi-PIE. Given two images $x_1$ and $x_2$ of the same attribute $z_a$ (i.e., digit categories for MNIST and poses for CMU Multi-PIE), we extract and interpolate their $z_c$ for recovering the intermediate outputs. Note that attributes can be preserved in each row, while changes in visual appearances correspond to the manipulation of $z_c$.

Figure 5. Visualization of continuous attribute representations on CMU Multi-PIE. Given two input images $x_1$ and $x_2$, the first row shows images recovered by interpolated latent codes of VAE-GAN [18]. The second and third rows show output images of UFDN [21] and our dec-GAN using the content features ($z_c$) as that of the corresponding input and with interpolated $z_a$ from $x_1$ and $x_2$.

Training Progress

As noted earlier, a major advantage of our dec-GAN is the applicability to existing GAN-based models without the need of pre-defined attributes. In Table 3 we compare the numbers of training iterations and computation times of dec-GAN and ACGAN [24] with the same backbone structures for generators and discriminators. Note that all experiments were conducted on single NVIDIA GTX 1080.
Figure 6. t-SNE visualization and ablation studies using MNIST. Visualization of latent representation $z$ of VAE-GAN is shown in (a). Distributions of $z_c$ and $z_a$ of our dec-GAN are shown in (b) and (c), respectively.

| Method   | Pose | ID  |
|----------|------|-----|
| VAE-GAN  | 97.44 ($z$) | 96.94 ($z$) |
| UFDN     | N/A  | 94.31 ($z_c$) |
| Ours     | 99.74 ($z_a$) | 98.59 ($z_c$) |

Table 2. Classification performances on CMU Multi-PIE. Note that our method decomposes attribute feature $z_a$ and content feature $z_c$ from latent representation $z$ of VAE-GAN [18]. Since the attribute feature of UFDN is a hand-crafted one-hot vector, it cannot be applied for pose classification.

| Method   | Iters (k) | Time (min) |
|----------|-----------|-------------|
| dec-GAN (s) | 3          | 6            |
| dec-GAN (p) | 8.8        | 13           |
| AC-GAN [24] (s) | 20.3       | 26           |
| AC-GAN [24] (p) | 65.1       | 89           |

Table 3. Comparisons of training time on CMU-MultiPIE. Note that s/p denote the smile/pose attributes. The number of iterations is counted by thousands, and the running time is by minutes.

4.5. t-SNE Visualization

We now visualize latent representations derived by our with VAE-GAN backbone using MNIST in Figure 6. We first show t-SNE results of latent representation $z$ of VAE-GAN in Figure 6(a), where images of different categories were visualized into different yet overlapping groups. Distributions of $z_c$ and $z_a$ of our dec-GAN are shown in (b) and (c), respectively. Note that the attribute of interest indicates digit categories for MNIST dataset. Since $z_c$ does not contain any attribute information, we only expect sufficient separation between different digit classes in Figure 6(c) but not in Figure 6(b).

5. Conclusion

In this paper, we proposed a unique decomposition-GAN (dec-GAN) to perform feature disentanglement, which jointly extracts content and attribute representations from the latent feature observed from existing GAN-based models. Different from prior disentanglement works which typically derive disjoint latent representations describing desirable features, our dec-GAN performs feature decomposition, which separates latent representation into separate features describing the properties/attributes of interest. The attribute disentanglement of our dec-GAN is driven by classifiers pre-trained on the attribute of interest. Followed by the design of generative network modules, this allows disentanglement of continuous content and attributes, while exhibiting additional flexibility in determining the attributes of interest (i.e., by replacing such classifiers based on the desirable attribute categories). We performed qualitative
and quantitative evaluation using multiple image datasets, with attributes ranging from digit categories to pose angles. The effectiveness and robustness of our dec-GAN can be successfully confirmed, while its superiority over existing models can also be verified.

References

[1] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. Cvae-gan: fine-grained image generation through asymmetric training. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 2745–2754, 2017.

[2] David Berthelot, Thomas Schumm, and Luke Metz. BeGAN: Boundary equilibrium generative adversarial networks. arXiv preprint arXiv:1703.10717, 2017.

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In Proceedings of the International Conference on Learning Representations (ICLR), 2019.

[4] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In Advances in Neural Information Processing Systems (NIPS), 2016.

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

[6] Abel Gonzalez-Garcia, Joost van de Weijer, and Yoshua Bengio. Image-to-image translation for cross-domain disentanglement. In Advances in Neural Information Processing Systems (NIPS), 2018.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in Neural Information Processing Systems (NIPS), pages 2672–2680, 2014.

[8] Ralph Gross, Iain Matthews, Jeffrey Cohn, Takeo Kanade, and Simon Baker. Multi-pie. Image and Vision Computing, 28(5):807–813, 2010.

[9] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In Proceedings of the International Conference on Learning Representations (ICLR), 2017.

[10] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[12] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In Proceedings of the International Conference on Learning Representations (ICLR), 2018.

[13] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[14] Diederik P. Kingma, Danilo J. Rezende, Shakir Mohamed, and Max Welling. Semi-supervised learning with deep generative models. In Advances in Neural Information Processing Systems (NIPS), 2014.

[15] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114, 2013.

[16] Tejas D. Kulkarni, Will Whitney, Pushmeet Kohli, and Joshua B. Tenenbaum. Deep convolutional inverse graphics network. In Advances in Neural Information Processing Systems (NIPS), 2015.

[17] Tejas D Kulkarni, William F Whitney, Pushmeet Kohli, and Josh Tenenbaum. Deep convolutional inverse graphics network. In Advances in neural information processing systems, pages 2539–2547, 2015.

[18] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. arXiv preprint arXiv:1512.09300, 2015.

[19] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998.

[20] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[21] Alexander H. Liu, Yen-Cheng Liu, Yu-Ying Yeh, and Yu-Chiang Frank Wang. A unified feature disentangler for multi-domain image translation and manipulation. In Advances in Neural Information Processing Systems (NIPS), 2018.

[22] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In Advances in Neural Information Processing Systems (NIPS), 2017.

[23] Yen-Cheng Liu, Yu-Ying Yeh, Tzu-Chien Fu, Sheng-De Wang, Wei-Chien Chiu, and Yu-Chiang Frank Wang. Detach and adapt: Learning cross-domain disentangled deep representation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[24] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In Proceedings of the International Conference on Machine Learning (ICML), 2017.

[25] Ori Press, Tomer Galanti, Sagie Benaim, and Lior Wolf. Emerging disentanglement in auto-encoder based unsupervised image content transfer. arXiv preprint arXiv:2001.05017, 2020.

[26] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In Proceedings of the International Conference on Machine Learning (ICML), 2016.
[27] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. Interfacegan: Interpreting the disentangled face representation learned by gans. arXiv preprint arXiv:2005.09635, 2020.

[28] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In Advances in Neural Information Processing Systems (NIPS), 2015.

[29] Andrey Voynov and Artem Babenko. Unsupervised discovery of interpretable directions in the gan latent space. arXiv preprint arXiv:2002.03754, 2020.

[30] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaoolei Huang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.

[31] Zhilin Zheng and Li Sun. Disentangling latent space for vae by label relevant/irrelevant dimensions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 12192–12201, 2019.