ACE: Zero-Shot Image to Image Translation via Pretrained Auto-Contrastive-Encoder

Sihan XU
University of Michigan
Ann Arbor

Zelong Jiang
University of Michigan
Ann Arbor

Ruisi Liu
University of Illinois
Urbana-Champaign

Kaikai Yang
YanShan University
Qinhuangdao, China

Zhijie Huang
ShanghaiTech University
Shanghai, China

Abstract

Image-to-image translation is a fundamental task in computer vision. It transforms images from one domain to images in another domain so that they have particular domain-specific characteristics. Most prior works train a generative model to learn the mapping from a source domain to a target domain. However, learning such mapping between domains is challenging because data from different domains can be highly unbalanced in terms of both quality and quantity. To address this problem, we propose a new approach to extract image features by learning the similarities and differences of samples within the same data distribution via a novel contrastive learning framework, which we call Auto-Contrastive-Encoder (ACE). ACE learns the content code as the similarity between samples with the same content information and different style perturbations. The design of ACE enables us to achieve zero-shot image-to-image translation with no training on image translation tasks for the first time.

Moreover, our learning method can learn the style features of images on different domains effectively. Consequently, our model achieves competitive results on multi-modal image translation tasks with zero-shot learning as well. Additionally, we demonstrate the potential of our method in transfer learning. With fine-tuning, the quality of translated images improves in unseen domains. Even though we use contrastive learning, all of our training can be performed on a single GPU with the batch size of 8. Our code is available at github.com/SihanXU/ACE.

1. Introduction

In the field of computer vision, image-to-image translation has been well-established and achieved promising re-
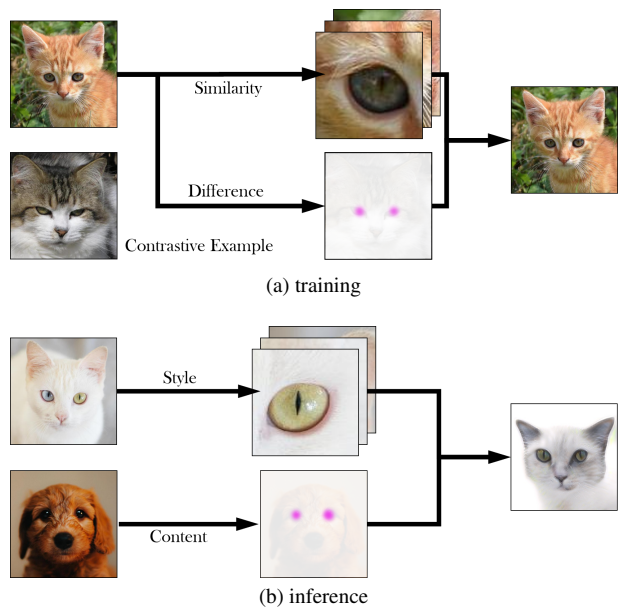

Figure 1. **The Main Idea of Auto-Contrastive-Encoder.** When training the ACE model, the encoder learns how to get the similarity and difference from the same domain by contrastive learning. And the decoder uses the similarity features as style and the difference as content to reconstruct the image in both training and inference stage.

sults on various related tasks such as image colorization, style transfer. These existing works are usually achieved by learning the mapping between the source and target domain [18, 20, 23, 40]. Some new training methods have emerged afterwards [1, 2, 7, 24, 30, 31], but they are still trained for learning the mapping relationship, which keeps them from focusing on the distribution of samples. However, such works are inherently fastidious in data distributions. For example, pix2pix requires the data of two do-
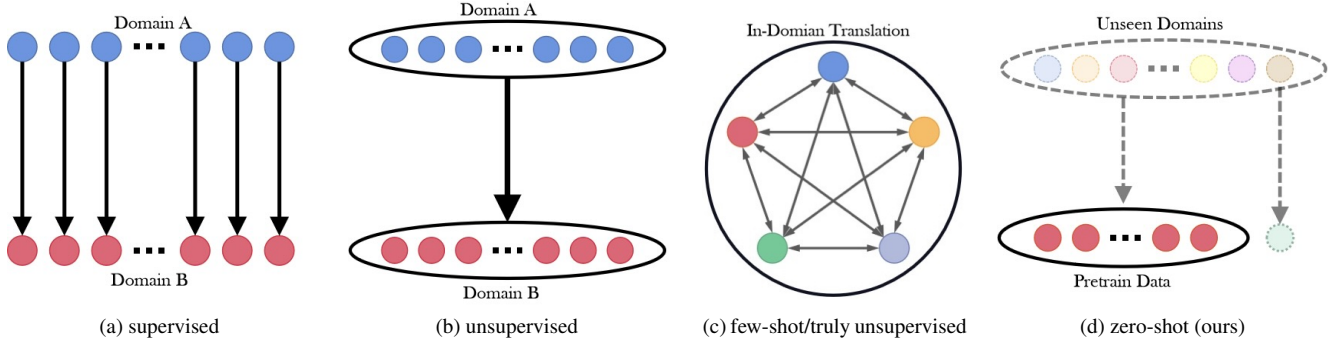
Figure 2. **Different kinds of image to image translation.** (a) Supervised image to image translation like pix2pix [20] requires paired datas and (b) unsupervised image to image translation like cycleGAN [40] requires paired dataset for training. (c) Few-shot [24] or Truly unsupervised [1] image to image translation can only translate images in the training domain. Our method, (d) zero-shot image translation is capable to translate images from unseen domains to both of training domain and unseen domains.

mains to come in pairs [20], while cycleGAN-like methods rely on the joint distribution of two domains [18, 23, 40]. Some new methods, such as [1, 7, 24], still have strict requirements for data. Meanwhile, one-shot image translation (OST) [2] has only achieved limited breakthroughs with the same idea of learning the mapping. On the other hand, most other works have focused more on how to improve the quality of generated images [30, 31]. As a consequence, they still fail to move the eyes off the mapping relations to other approaches. Conversely, we believe that we can solve the problem of strict data requirements if there is a training method that can achieve the image translation task without learning the mapping between different distributions.

In this paper, we propose a novel learning task (Section 3) that implements image translation by learning similar and different features of images within a data distribution (Fig. 1). We note that the features that are similar before and after image translation are precisely the features that need to be retained, while the features that are different under the same distribution are the features that need to be translated. In this way, our model can recognize the features to be retained or transformed by learning the similarities and differences within the distribution. Moreover, such a training method without learning the mapping relation can also translate the samples in unseen domains without being trained on image translation tasks, thereby achieving zero-shot learning as Fig. 1b, Fig. 2d . Despite several previous works claiming to attain zero-shot image translation [6, 22], they merely perform style transformation within the features in the specific domain.

Based on these ideas, we propose the Auto-Contrastive-Encoder (ACE), an Auto-Encoder structure that incorporates contrastive learning (Section 3). Contrastive learning provides us with the effectiveness of learning similarities between positive samples and differences between negative samples, which encourages our model to learn the similar

and different features in the same distribution. In this paper, we use a structure similar to Simple Siamese Representation Learning (SimSiam) [5] to keep the model simple while allowing the model to be trained with a small batch size (Section 4). For contrastive learning, we propose Adaptive Instance Augmentation and perform contrastive learning directly on the encoded image features. Proven by experiments, our method is able to capture the similarity and difference in image features effectively.

It is worth noting that ACE is a framework applicable to any model instead of a specific model. Our model in this article uses the VGG [32] model for the encoder, while our decoder is a convolutional network (CNN) [29] using ResNet [16]. We perform zero-shot image translation and achieve satisfactory results on the Summer⇔Winter [18], Orange⇔Apple [8] and Animal Face [24] datasets. Furthermore, our experiment on Animal Face [24] shows that our method also has the potential for transfer learning by pre-training on large datasets. The experiments in this paper can all be completed on a single GPU with batchsize of 8.

2. Related Work

Image-to-image translation. Image-to-image translation, a prevalent problem for computer vision, aims to convert an input image into another output image. It has been applied to style transfer [11, 12, 21], image denoising [10, 34, 37], and colorization [26, 38, 39]. Many methods for image translation tasks have been proposed since [20]. However, these methods are largely limited to learning the mapping between images and rely on the pairs of data in datasets for training [1, 7, 18, 23, 24, 30, 40], and thus can never realize zero-shot learning. Although there is similar research [6, 22] about zero-shot learning methods before, they only do translations in the same domain rather than implement real image translation.

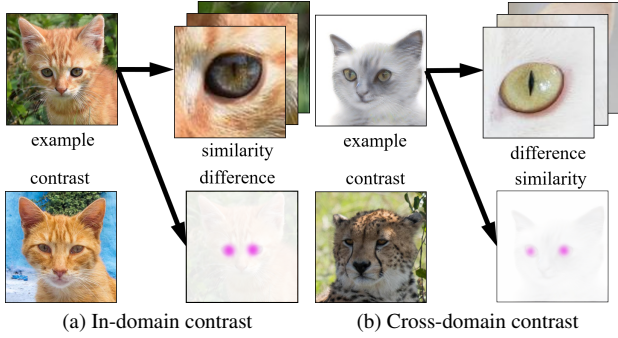(a) In-domain contrast   (b) Cross-domain contrast

Figure 3. **Assumption of zero-shot translation.** (a) For the examples in the same domain, they should have same style but different content. The similarity should be style and the difference should be the content. (b) For the paired examples before and after translated, they should have same content with different style. Thus, the difference is their style and the similarity is their content.

Figure 4. **Image translation task**. In order to realize the image translation, content code need to be obtained from the pixel space of domain A. And then the output, which is in the pixel space of domain B, should be generated from the content. Additionally, style code is required for multimodal image translation tasks.

This paper presents a new method of image translation by learning the similarities and differences of samples within the same distribution. Our ACE approach demonstrates the feasibility of this idea. Our method is not only able to achieve competitive results on zero-shot image-to-image translation but also applicable to various image translation tasks like multimodal translation task. Furthermore, our model has the potential for transfer learning to improve the quality of image translation tasks by pretrain and finetune.

Contrastive learning Contrastive learning is an efficient method for unsupervised learning. Its key idea is to learn similar features between positive samples and different features between negative samples [36]. Based on this idea, several subsequent works with great influence have come into being, such as [3–5, 14, 15]. There are some methods like [3,5,14] can learn similar features between positive samples even without negative samples.

We use a similar structure to SimSiam [5] in this paper, but the difference is that we augment the features of images instead of augmenting the image itself. We first use adaptive instance augmentation to augment the features of the images and implement contrastive learning subsequently. Unlike SimSiam [5], we also include a predictor in the process of using the encoder. Experiments demonstrate that our method is effective in learning similarities and difference within a distribution.

3. Method

3.1. Assumption

Fig. 3 shows the fundamental assumption of our method. Based on the effect of in-domain contrast and cross-domain contrast, we can model the content information as the in-domain difference and cross-domain similarity, and the
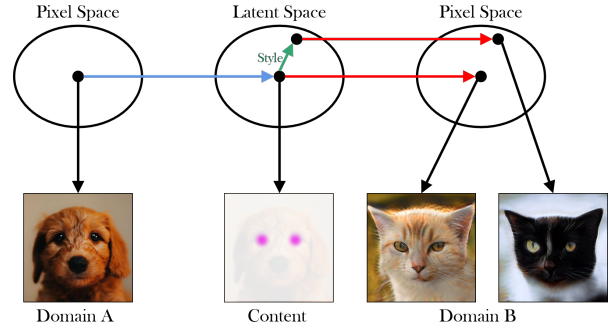
style information as the in-domain similarity and cross-domain difference. Then, we can follow the assumption in MUNIT [18] like Fig. 4 that each image $x_i \in \mathcal{X}_i$ is generated from a content latent code $c \in \mathcal{C}$ that is shared by both domains, and a style latent code $s_i \in S_i$ that is specific to the individual domain. For each image, our objective is to find a pair of underlying encoders E_s and E_c to disentangle the two latent codes and a generator G to reconstruct images with these two types of codes. Suppose we have a pair of image (x_1, x_2). We are able to generate a translated image $x_{1\to2}$ by applying the encoders and the generator, namely $x_{1\to2} = G(E_c(x_1), E_s(x_2))$. Note that now $x_{1\to2}$ is also a sample in domain $\mathcal{X}_2$ with the same content as x_1. Then, since the content encoder generates domain-invariant representations, ideally we have $E_c(x_1) = E_c(x_{1\to2})$. Similarly, considering $x_{1\to2}$ and x_2 are from the same domain, the constraint $E_s(x_2) = E_s(x_{1\to2})$ should also hold.

3.2. Model

Fig. 5 is an overview of our model, which is similar to MUNIT [18]. Our model consists of the encoder, content encoder, style encoder, and decoder. The content encoder contains residual blocks [16] and a predictor, forming a contrastive learning framework. In the process of training (Fig. 5a), first, we obtain the content and style features respectively through content encoder and style encoder. Finally, the image can be restored by the decoder. In the fine-tuning and inference process (Fig. 5b), the encoder obtains features of style and content images so that the style and content images can be learned by the content encoder and style encoder. After these steps, we use the decoder to restore the desired image.

We use VGG [32] as the encoder of the model as in the previous works [17, 18, 24]. Fig. 5c presents the learning process of content code. After acquiring the features of input images, we augment the image features with the Adap-
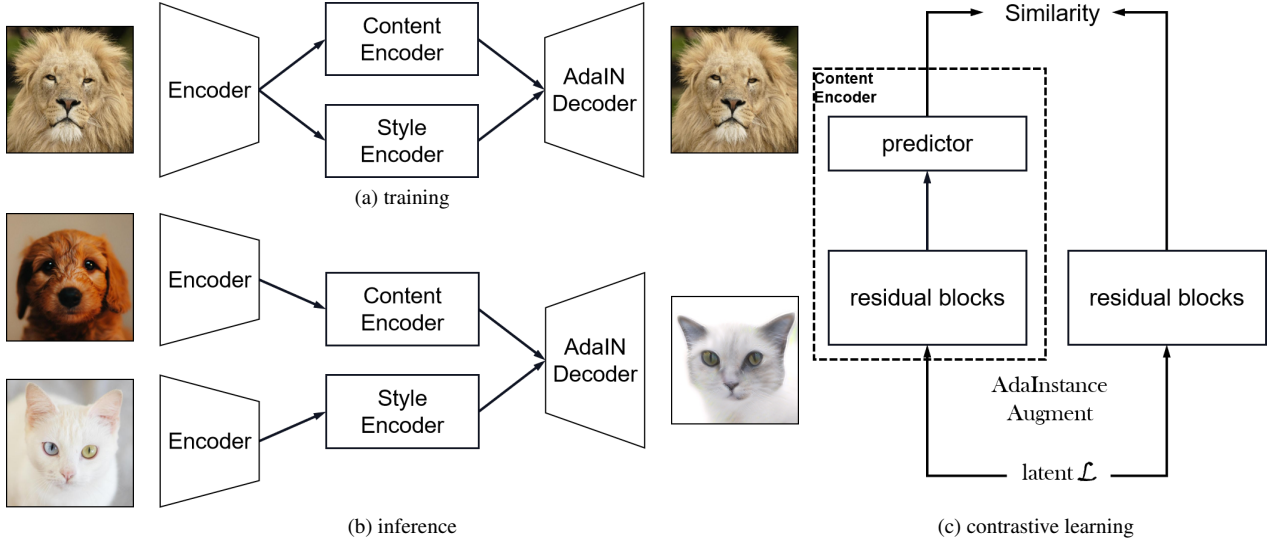
Figure 5. **Model overview**. In training, we use (a) auto-encoder and (c) contrastive learning to reconstruct images from the training domain. In inference (b), we use two-stream like architecture to encode content and style, and AdaIn decoder to reconstruct the image.

tive Instance Augmentation. Subsequently, we use a Sim-Siam similar structure [5] to implement contrastive learning and obtain the images' content features. Our style encoder is a single layer CNN with Adaptive Pooling [25], which is able to preserve the global feature information of the images. Following MUNIT [18], we use an MLP to learn the AdaIN parameters from the style codes.

The content encoder includes two parts as shown in Fig. 5c. The residual blocks are CNN with skip connection [16] and BatchNorm [19]. And the predictor is a 2-layer-MLP with a bottleneck, and it has BatchNorm between hidden layers. We don't use the BatchNorm at the output layer.

Our decoder is composed of residual networks and a convolutional network [29] with adaptive instance normalization [17]. As stated in [18], instance normalization [35] and batch normalization would destroy the style features of the image. Therefore, we exclude these two types of normalization in our decoder.

3.3. Adaptive Instance Augmentation

According to the experimental results of using the same adaptive instance normalization (AdaIN) [17], the instance norm would affect the style of images. Following this, [18] proposes to use MLP to dynamically produce the parameters for Instance Normalization layers from style codes. Inspired by these practices, we propose Adaptive Instance Augmentation, where we replace the parameters of AdaIN with Gaussian noises:

$$AdaIN(z, \gamma, \beta) = \gamma \left(\frac{z - \mu(z)}{\sigma(z)} \right) + \beta, \qquad (1)$$

$$\gamma, \beta \sim \mathcal{N}(0, 1) \qquad (2)$$

Note that this is an augmentation in the latent space. The procedure to augment a sample x is

$$Aug(x) = G_{\gamma, \beta}(E_c(x)), \qquad (3)$$

where $G_{\gamma, \beta}$ means that instead of using style encoder, we use the variables γ and β for the AdaIN layers. We first map x into its content code and then use the randomized AdaIN decoder to reconstruct the augmented sample. This method enables us to modify the style of images while ensuring the same image content. Based on this feature augmentation, our contrastive learning method can make the content encoder insensitive to style features, so that the content feature can be effectively preserved.

3.4. Loss Function

Originating from our assumptions, we first design our loss function to capture the similarity between the content codes with different style features. Let c_1 and c_2 be two latent codes extracted by content encoder and augmented with Adaptive Instance Augmentation. We define the loss for contrastive learning as the SimSiam loss from [5]:

$$\mathcal{L}_{contrast}(c_1, c_2) = \mathcal{C}ontrastiveLoss(p(c_1), stopgrad(c_2)) \qquad (4)$$

where p is the predictor layer and $\mathcal{C}ontrastiveLoss$ can be any distance measurement such as negative consine similarity. The content consistency is also forced by minimizing the distance between content codes extracted from each pair of original sample x and reconstructed sample x':

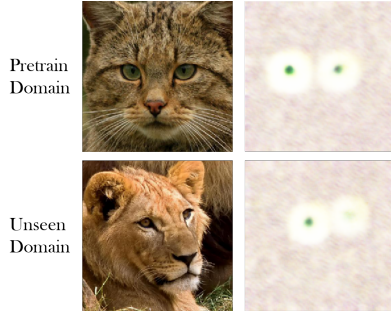$$\mathcal{L}^c_{consist}(x) = \|E_c(x) - E_c(x')\|_1. \qquad (5)$$

Figure 6. **Visualization of content code on two domains.** The content code only contains the information of the location of animal's eyes, which is the difference between the in-domain examples. And the content encoder is also capable to encode the content from the unseen domain.

Similarly, for style codes we have

$$\mathcal{L}^s_{consist}(x) = \|E_s(x) - E_s(x')\|_1. \quad (6)$$

Next, to train the auto-encoder, we adopt a reconstruction loss and a GAN loss to ensure that the reconstructed images follow the distribution of target domain.

$$\mathcal{L}_{recon}(x) = \|x - x'\|_1 \quad (7)$$

$$\mathcal{L}_{GAN} = \|D(x) - 1\|_2^2 + \|D(x')\|_2^2 \quad (8)$$

We train our model with the total objective as the weighted sum of all loss functions mentioned above.

3.5. Stop gradient

Since Auto-Encoder updates encoder in the training, it will greatly influence the effect of contrastive learning. Therefore, when we are training the Auto-Encoder, we freeze the content encoder. Which means

$$x' = G(stopgrad(E_c(x)), E_s(x)) \quad (9)$$

3.6. Discriminator

We use an approach similar to Generative Adversarial Network (GAN) [13] to train our ACE to improve the quality of the images. In our experiments, we use a loss function similar to [27] to make the training more stable; and use SpectralNorm [28] to enable the model to generate images with higher quality.

4. Experiments

4.1. Implementation Details

Our framework is comprised of a VGG encoder, a content encoder, a style encoder and a decoder. The content encoder consists of four residual blocks and the style encoder contains a global pooling layer and a fully connected
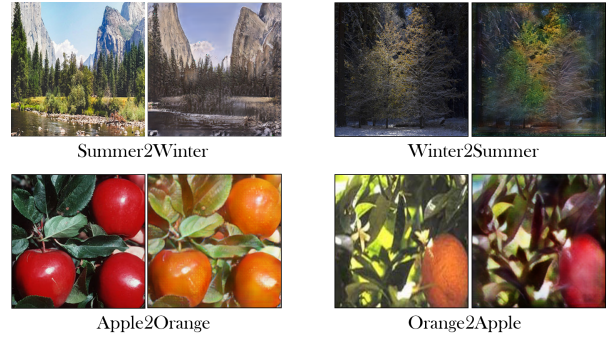

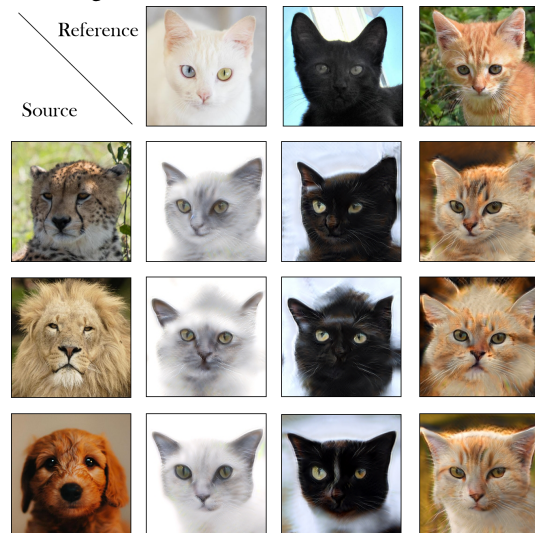Figure 7. Translation results on Yosemite summer⇔winter and Apple⇔Orange.

Figure 8. Multimodal Translation results on bigcat⇔cat and dog⇔cat.

layer. For the decoder, we have several residual blocks, each followed by up-sampling layers. We also use Adaptive Instance Normalization layers to dynamically generate parameters of Instance Normalization. However, to accelerate the convergence of style encoder, we propose to use two different style codes to respectively represent the global style in the domain and the individual style of each sample. The domain style code is a learnable tensor which is shared by all data in the pretrain domain, while the individual style code is output by the style encoder. Then we sum these two style codes up before applying them to the AdaIN layers.

4.2. Datasets

We conduct the evaluation on the same datasets as [18, 40]. Our method achieves satisfying results on Yosemite summer⇔winter, apple⇔orange and Animal face translation (including data of bigcats, cats and dogs).

Input	ACE	OST	FUNIT	MUNIT	Target

Figure 9. Comparison of zero-shot translation results with one-shot [2], few-shot [24] and unsupervised [18] methods.

4.3. Visualization

To better understand whether our designed models work as we expect, we adopt some tools of visualization for our extracted content codes. Fig. 6 shows the information in the content codes from the pretrain domain and the unseen domain. Both representations indicate the animal's eyes. It verifies that our contrastive learning based encoder is able to extract similar content codes regardless of the domains.

4.4. Effectiveness of Zero-shot Learning

Our experiment covers the zero-shot learning on datasets Summer⇔Winter [18] and Orange⇔Apple [8]. The final results are shown in Fig. 7.

At the same time, we conduct multimodal translation on the Animal Image Translation Dataset [24]. We trained our model on cat dataset and applied to bigcat2cat task and dog2cat task. With the images in Fig. 8, we can see that our method translates images to different styles while maintaining the original content features.

The focus of our work is not about producing high quality images. For this reason, we only compare our method with OST [2], FUNIT [24], MUNIT [18]. During training, our model shares the same settings as MUNIT, while OST and FUNIT keep their settings as mentioned in OST and FUNIT. The comparison is presented in Fig. 9. It is apparent that our framework obtains outstanding translating results in a zero-shot manner.

4.5. Transfer Learning

pretrain	zero-shot	fine-tuned

Figure 10. **Fine-tune result.** Even though ACE is capable of doing the zero-shot translation to an unseen target domain, but the translation result contains some features from the pretrain domain and thus affects the translation image's quality. After fine-tuning on the target domain, ACE can achieve better translation results, which shows the potential of ACE model in transfer learning.

Here we discuss whether our model is suitable for transfer learning. We pre-train our model on a specific domain and test on one or two unseen domains. If we are able to get the training data of the test domains, we can conduct fine-tuning to improve the generated image quality. The process of fine-tuning is quite similar to the pre-training. We use the images in the source domain to generate the content codes and use target domain to get style codes. The fine-

tuning loss will be consisted of the latent consistency losses $\mathcal{L}_{consist}^c, \mathcal{L}_{consist}^s$ and the GAN loss $\mathcal{L}_{GAN}$. We don't use the contrastive loss in fine-tuning.

As shown in Fig. 10, in experiments of multimodal image translation, we pre-train our model on the cat domain and it achieves satisfactory performance when applied to the task of dog2cat translation. Next, our model still performs well if we change the target domain to the bigcats. Therefore, we believe with pre-training, our model is capable of translating the images from one unseen domain to another unseen domain. If we continue to fine-tune on the dataset of bigcats, we can see that there is much room for improvement of generated image quality. As a result, we believe our model has a great potential for transfer learning based on large datasets like ImageNet [8].

4.6. Failure Cases

Our method fails on some cases where the context is complicated such as Fig 11. For instance, on the dataset of horse2zebra, our model sometimes erroneously puts the zebra's stripes on the background.

Figure 11. **Failure Cases.** The context is so complicated that our model cannot integrate style and content information well.

5. Discussion

The method proposed in this paper has achieved satisfactory results in the task of image-to-image translation, but we have not conducted experiments on other types of translation tasks. For example, we believe our method can work on language processing as well. Furthermore, due to limited resources, we only tested the potential of our method on transfer learning with small datasets. If we can pre-train on a larger data set, our model may be able to achieve better results in image translation. In this article, we just use a very simple model structure, but our approach is also applicable to other models, such as ResNet [16], Vision Transformer [9] and Diffusion Model [33]. We believe that better results can be achieved if our methods are combined with these further efforts.

6. Conclusions

In order to conquer the challenge coming from learning the mapping relationship in image-to-image translation, we propose a new objective, which is to translate images by learning the similarities and differences without learning any mappings or joint distributions. Additionally, we propose a simple model structure called Auto-Contrastive-Encoder to solve this problem, and it has achieved satisfactory results. We have also shown the potential of our model in transfer learning. It is promising that our method can make the task of image-to-image translation move forward.

References

[1] Kyungjune Baek, Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Hyunjung Shim. Rethinking the truly unsupervised image-to-image translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14154–14163, October 2021. 1, 2

[2] Sagie Benaim and Lior Wolf. One-shot unsupervised cross domain translation, 2018. 1, 2, 6

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9650–9660, October 2021. 3

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020. 3

[5] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15750–15758, June 2021. 2, 3, 4

[6] Yuanqi Chen, Xiaoming Yu, Shan Liu, and Ge Li. Toward zero-shot unsupervised image-to-image translation, 2020. 2

[7] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 2, 6, 7

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020. 7

[10] Linwei Fan, Fan Zhang, Hui Fan, and Caiming Zhang. Brief review of image denoising techniques. *Visual Computing for Industry, Biomedicine, and Art*, 2(1):1–12, 2019. 2

[11] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style, 2015. 2

[12] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016. 2

[13] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. 5

[14] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning, 2020. 3

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2, 3, 4, 7

[17] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 3, 4

[18] 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)*, September 2018. 1, 2, 3, 4, 5, 6

[19] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015. 4

[20] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, 2017. 1, 2

[21] Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Yizhou Yu, and Mingli Song. Neural style transfer: A review. *IEEE Transactions on Visualization and Computer Graphics*, 26(11):3365–3385, 2020. 2

[22] Jianxin Lin, Yingce Xia, Sen Liu, Shuqin Zhao, and Zhibo Chen. Zstgan: An adversarial approach for unsupervised zero-shot image-to-image translation, 2019. 2

[23] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks, 2017. 1, 2

[24] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10550–10559, 2019. 1, 2, 3, 6

[25] Yinglu Liu, Yan-Ming Zhang, Xu-Yao Zhang, and Cheng-Lin Liu. Adaptive spatial pooling for image classification. *Pattern Recognition*, 55:58–67, 2016. 4

[26] Qing Luan, Fang Wen, Daniel Cohen-Or, Lin Liang, Ying-Qing Xu, and Heung-Yeung Shum. Natural image coloriza-tion. In *Proceedings of the 18th Eurographics conference on Rendering Techniques*, pages 309–320, 2007. 2

[27] Xudong Mao, Qing Li, Haoran Xie, Raymond Y.K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 5

[28] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks, 2018. 5

[29] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks, 2015. 2, 4

[30] Taesung Park, Alexei A. Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for conditional image synthesis. In *ECCV*, 2020. 1, 2

[31] Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei Efros, and Richard Zhang. Swapping autoencoder for deep image manipulation. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 7198–7211. Curran Associates, Inc., 2020. 1, 2

[32] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014. 2, 3

[33] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015. 7

[34] Chunwei Tian, Lunke Fei, Wenxian Zheng, Yong Xu, Wangmeng Zuo, and Chia-Wen Lin. Deep learning on image denoising: An overview. *Neural Networks*, 131:251–275, 2020. 2

[35] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization, 2016. 4

[36] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[37] Junyuan Xie, Linli Xu, and Enhong Chen. Image denoising and inpainting with deep neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. 2

[38] Lvmin Zhang, Yi Ji, and Xin Lin. Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier gan, 2017. 2

[39] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 649–666, Cham, 2016. Springer International Publishing. 2

[40] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017. 1, 2, 5