BF-GAN: Few-shot Generation With Decoupling Background and Foreground

Jun Lv Yunlong Cheng Yunkai Zheng

Abstract

1. Introduction

Deep generative models, including Variational Auto-Encoder (VAE) [12] and Generatative Adversarial Network (GAN) [7], are currently one of the most promising directions in Machine Learning. Although the models have a good performance, they are notoriously hard to train and their training processes require large amounts of data, especially for high-dimensional data like images. And once a model is trained, it cannot be adopted to new categories without complete retraining. Hence, few-shot image generation [3] is proposed to generate images for new categories with only a few images, which draws more attention in image generation.

In this paper, we mainly focus on the challenge of the few-shot image generation problem, which is studied only on quite few works. These works can be roughly divided into three categories: optimization-based, fusion-based and transformation-based. Particularly, optimization-based methods [3, 16] use some meta-learning techniques such as Reptile [19] and MAML [5]. Fusion-based methods use some fusion techniques, such as MatchingGAN [10] and F2GAN [11], to fuse the features of conditional images in order to increase the diverse of generated images. And transformation-based methods [1, 9] adopt transformation technique. Though previous works achieve impressive result on few-shot generation, there still have a key drawback, the background of the generated image usually lack of diversity.

To solve the issue, We follow the idea in [11, 15] and propose a novel Background-Foreground GAN (BFGAN). The high-level idea of our model is to make use of the mixed features learned from a few conditional images within the same category and separate the background from the foreground. Because for few-shot tasks, we only have little images in our testing set, that's why we want to generate more combination of features of image's foreground in the same category. We can combine them with randomly chosen and noise blended background features yielded by the given conditional images, which can increase the diversity of generated images. At length our model contains a background branch and a foreground branch. The foreground branch can be further divided into two stages: pose/shape stage and texture stage.Background branch is to generate background of new images while two stages of foreground branch aim to generate shape, pose and texture information of foreground object.

In the background branch, we randomly choose a background feature vector generated by a decoder among the few-shot images, then add a noise factor, afterwards a decoder is used to generate new background image.

In the foreground branch, we use a few decoders to generate pose,shape and texture feature vectors for the few-shot images, then use random interpolation coefficients to mix each feature vector, e.g. texture vector. Afterwards, We use a decoder to decode the combined features to generate images and use a discriminator to ensure the diversity of generated images.

The paper is organized as follows. First, in section 2 we mention the necessary background in GAN and few-shot image generation. In section 3, we propose our BF-GAN in detail. Section 4 briefly describes experiment setting and evaluation metric of our proposed model.Section 5 concludes our contribution and propose some thought about future work.

2. Related Works

Data Augmentation. This concept [13] is proposed to amplify the data samples in the training set, which is usually insufficient in the case of few-shot scenario. Traditional data augmentation methods, for example: revolvement , crop and color jittering will not generate enough diversity of pictures. So some premium methods [26][27] are raised, however the pictures generated by these methods are not real enough. Moreover, there exist deep generative models which is able to produce samples with more diversity and reality, using the distribution of the training data. These methods can augment data in both feature augmentation[4][22] and image augmentation[1]. We develop a method of the type of image augmentation, which means to generate more samples to enrich the training set.

Generative Adversarial Network. Generative Adversarial Network (GAN) [7][24] is on the grounds of adversarial learning, which trains both generative and discriminate model to generate more accurate pictures and distinguish more subtle pictures. In before, unconditional GANs [18] produce pictures with the help of randomly generated vectors, which is learned by the distribution of training pictures. Afterwards, GANS which is conditioned on one image are raised to alter this picture into a target picture. Recently, some conditional GANS are trying to achieve more difficult tasks conditioned on different pictures, for example, few-shot picture translation[17] and few-shot picture generation[2][3].In our paper, we will concentrate on the few-shot picture generation problem.

Few-shot Image Generation. This is a difficult problem because we need to generate more pictures using only a few known pictures.Previously, this few-shot picture generation work is only applicable within limited cases. In[14][21], Bayesian learning is used to learn small ideas such as pen stroke and unite the ideas hierarchically to produce new pictures. More recently, FIGR [3] was raised to unite adversarial learning with optimization-based few-shot learning method Reptile [19] to generate new images. Like FIGR , DAWSON used meta-learning [6] for generative models based on GAN to accomplish domain adaptation within seen and unseen types. Metric-based few-shot learning method Matching Network [23] was incorporated with Variational Auto-Encoder [20] in GMN [2] to produce more pictures without fine-tuning during the testing stage. Matching GAN [10] tried to utilize learned metric to produce pictures based on a fewhttps://www.overleaf.com/project/5fae7ec0e890a12dac49a3bc conditional pictures. In this work, we raise a idea for few-shot picture generation, which can generate images with more diversity and reality.

3. Method

3.1. Overview

The proposed BF-GAN mainly utilize the MixNMatch [15] and F2GAN [11] structure, and consists of two components, namely background branch and foreground branch. Given a few conditional images $\mathcal{X} = \{x_i \in \mathbb{R}^{H\times W\times 3}\}_{i=1}^{k}$, first we generate 4 vectors b, z, p, c, b representing background code, z and p stands for the object's shape and pose while c is on behalf of texture of the objects. In the background branch, we randomly choose a picture's background vector b, add some random noise on it and send it into background generator G_b. In the foreground branch, this branch still consists of 2 stages, shape stage and texture stage. Before both stages, we first interpolate among z, p, c vectors generated former, using interpolation coefficients $\alpha_1, \alpha_2 ... \alpha_k$ ($a \in \mathbb{R}^k$, k is the number of conditional images) for each image$x_1, x_2 ... x_k$. We get a lot of mixed z', p', c' afterwards. Then in shape stage, we link the continuous vector z' and shape vector p' , send them into shape generator and we will receive a mask of a new object's shape. In texture stage, similarly, we use c' to generate new object's texture information, with shape stage's mask and texture stage's texture, we could generate this new image on the background image.

3.2. Background Branch

The background branch is proposed to produce features representing background information for few-shot image generation. To obtain background branch network, we need to pre-tain a background generation GAN. Then the encoder of the background generator will be used as background branch.Although actually this GAN simultaneously generate background and foreground features(including pose, shape and texture features).

For a few conditional images

$$x_i \in \mathcal{X} = \{x_i \in \mathbb{R}^{H\times W\times 3}\}_{i=1}^{k}$$

after we generate background vectors b_i from a few conditional images.

$$b_i = \mathcal{G}_b^{enc}(x_i) \tag{1}$$

we will randomly choose one vector as our base background vector b, then we add a random noise n on it and get b'

$$b' = b + n \tag{2}$$

Afterwards, background image $x_b \in \mathbb{R}^{H\times W\times 3}$ is generated from b' through a generator decoder $\mathcal{G}_b^{dec}(\cdot)$

$$x_b = \mathcal{G}_b^{dec}(b') \tag{3}$$

To supervise the performence of the background generator $\mathcal{G}_b = [\mathcal{G}_b^{enc}, \mathcal{G}_b^{dec}]$, there is a discriminator $\mathcal{D}_b(\cdot)$ to estimate the domain of generated image and real image $c = [real, fake]$.

$$c = \mathcal{D}_b(x_b) \tag{4}$$

Loss Term. We train the background branch in generative adversarial manner. in this branch, we use a generator G_b and a discriminator pair, D_b and D_{aux}. G_b is conditioned on latent background code b, which controls the different background types, like sky, ocean, dessert. We use an object bounding box detector instead of labeled box to separate the object from the background. We train G_b and D_b using two objectives:

$$\mathcal{L}_b = \mathcal{L}_{bg_adv} + \mathcal{L}_{bg_aux} \tag{5}$$

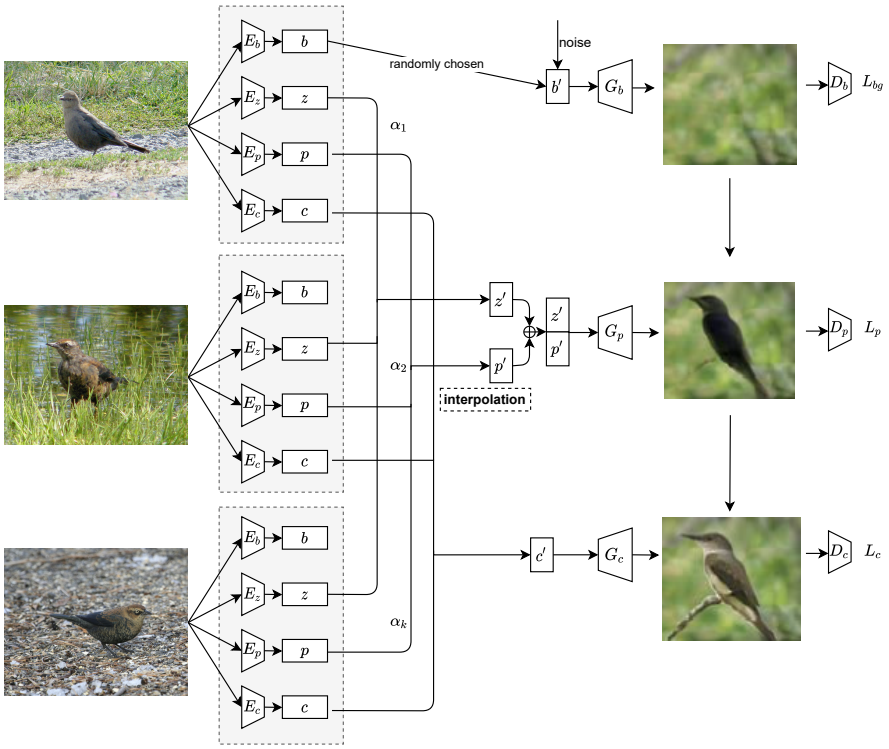


Figure 1. method overview

where $\mathcal{L}_{bg\_aux}$ is the auxiliary background classification loss and $\mathcal{L}_{bg\_adv}$ is the adversarial loss.

For the adversarial loss $\mathcal{L}_{bg\_adv}$, we employ the discriminator on a patch level.

$$\mathcal{L}_{bg\_adv} = \min_{G_b} \max_{D_b} \mathbb{E}_x[log(D_b(x))] + \mathbb{E}_{z,b}[log(1 - D_b(G_b(z, b)))]$$

then use D_{aux} to train the generator G_b:

$$\mathcal{L}_{bg_aux} = \min_{G_b} \mathbb{E}_{z,b}[log(1 - D_{aux}(G_b(z, b)))] \quad (6)$$

this loss updates G_b so that D_{aux} assigns a high background probability to the generated background patches.

3.3. Foreground Branch

The foreground branch is responsible to generate the foreground feature z, p, c from input image, and then create new image.

Given a few input foreground images $x_1, x_2, ...x_k$, we first utilize encoder to yield $3k$ vectors : $z_1, z_2, ...z_k$, $p_1, p_2, ...p_k$ and $c_1, c_2, ...c_k$, representing pose, shape and textures feature respectively.

$$z_i = \mathcal{G}_z^{enc}(x_i) \quad (7)$$

$$p_i = \mathcal{G}_p^{enc}(x_i) \quad (8)$$

$$c_i = \mathcal{G}_c^{enc}(x_i) \quad (9)$$

then we use interpolation method to mix all the k vector in the same kind.

$$z' = \sum_{i=1}^{k} \alpha_i * z_i \quad (10)$$

where α_i is a interpolation coefficient of i'th image and it is randomly generated while satisfying the following property.

$$\sum_{i=1}^{k} \alpha_i = 1 \quad (11)$$

similarly, p' and c' are generated in the same mixed way.

After getting the mixed vector z', p', c', shape and pose mask $M_p \in \mathbb{R}^{H \times W \times 3}$ is generated from z', p' through a generator decoder $\mathcal{G}_p^{dec}(\cdot)$

$$M_p = \mathcal{G}_p^{dec}(z', p') \quad (12)$$

To supervise the performance of the pose/shape generator $\mathcal{G}_p = [\mathcal{G}_z^{enc}, \mathcal{G}_p^{enc}, \mathcal{G}_p^{dec}]$, there is a discriminator $\mathcal{D}_p(\cdot)$ to estimate the domain of generated image and real image $c = [real, fake]$.

$$c = \mathcal{D}_b(x_b) \quad (13)$$

Loss Term. We use D_p to induce the pose/shape code p to represent the hierarchical concept. With no supervision from image labels, we exploit information theory to discover this concept in a completely unsupervised manner. Specifically, we maximize the mutual information $I(p, \mathcal{P}_{f,m})$, with D_p approximating the posterior $P(p|\mathcal{P}_{f,m})$:

$$\mathcal{L}_p = \mathcal{L}_{p\_info} = \max_{D_p, G_{p,f}, G_{p,m}} \mathbb{E}_{z,p}[\log D_p(p|\mathcal{P}_{f,m})] \quad (14)$$

We use $\mathcal{P}_{f,m}$ instead of $\mathcal{P}$ so that D_p makes its decision solely based on the foreground object (shape) and it doesn't get influenced by the background.

then for texture code c, we use D_c and D_{adv}.The loss function can be divided into two parts:

$$\mathcal{L}_c = \mathcal{L}_{c\_adv} + \mathcal{L}_{c\_info} \quad (15)$$

where

$$\mathcal{L}_{c\_adv} = \min_{G_c} \max_{D_{adv}} \mathbb{E}_x[log(D_{adv}(x))] + \mathbb{E}_{z,b,p,c}[log(1 - D_{adv}(\mathcal{C})]$$

and the $\mathcal{L}_{c\_info}$ is as follows:

$$\mathcal{L}_{c\_info} = \max_{D_c, G_{c,f}, G_{c,m}} \mathbb{E}_{z,p,c}[log(D_c(c|\mathcal{C}_{f,m}))] \quad (16)$$

Similarly,we use $\mathcal{C}_{f,m}$ instead of $\mathcal{C}$ so that D_c makes its decision solely based on the object (color/texture and shape) and not get influenced by the background.

3.4. BF-GAN

To separate picture's background and foreground, we need to use object detecting bounding box, which is not labeled by hand but utilizing an object detector. To separate the remaining factors of change while not under any supervision, BF-GAN brings in the information theory as FineGAN does.

BF-GAN is trained with three losses, the first loss is for background branch, the next two loss are for pose/shape stage and texture stage in foreground branch, which use either adversarial training to make the generated image look real and/or mutual information maximization between the latent code and corresponding image so that each code gains control over the respective factor (background, pose, shape, texture/color). We simply denote its full loss as:

$$\mathcal{L}_{BFGAN} = \mathcal{L}_b + \mathcal{L}_p + \mathcal{L}_c \quad (17)$$

where $\mathcal{L}_b$, $\mathcal{L}_p$, $\mathcal{L}_c$ denote the losses in the background, pose/shape, texture stages.

4. Experiments

We evaluate our BF-GAN's few-shot image generation results, its ability to disentangle background and foreground features,

4.1. Experiment Setting

Dataset To conduct few-shot generation experiments on the proposed BF-GAN, we utilize only **CUB** dataset, which contains 11788 images from 200 categories of birds. Because we need to disentangle the background and foreground features, we need the bounding boxes of birds to model these features. And the bounding boxes of birds are obtained by some existing object detector.

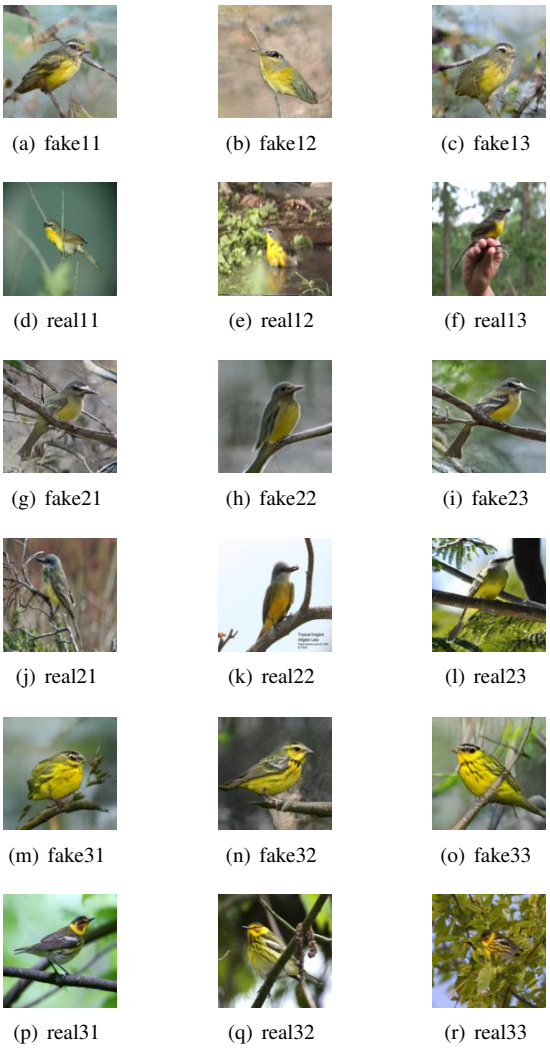
Metric We evaluate the quality of the generated images by three different metrics: Inception Scores(IS)[25] and Fréchet Inception Distance(FID)[8]. IS is related with visual quality of generated images. FID is designed to measure similarities between two sets of images. We compute IS on generated images from unseen categories and Fréchet Inception Distance between the real images and the generated images from unseen categories.

Implementation details We divide the dataset into two part. One part for training our model, and the other part is used as test set, which is never seen by our model. The training set have 150 categories of birds and it is randomly chosen from the total 200 categories. The remaining part of the dataset is used as test set. Then, for each categories which is never seen by our model, we use $K = 3$ conditional images from this category to generate new images. We use Adam optimizer with learning rate 0.0002 and beta $(0.5, 0.99)$. We train our model for 600 epochs.

4.2. Qualitative Results

Conditional image generation We show our generation results in Fig. 2. The fake images are generated from the corresponding three conditional real images. We can see that although we have only three images of a specific fine-grain category, we can generate many reasonable samples with diverse pose, shape, color, and especially background.

Linear interpolation of background and foreground We perform linear interpolation based on two conditional images x_1, x_2 to evaluate whether the space of generated images is densely populated. As can see from Fig. 3 and Fig. 4, our BF-GAN can produce more diverse images with smoother transition between two conditional images. And both the background and foreground can be transited without influencing each other.



(a) fake11 (b) fake12 (c) fake13

(d) real11 (e) real12 (f) real13

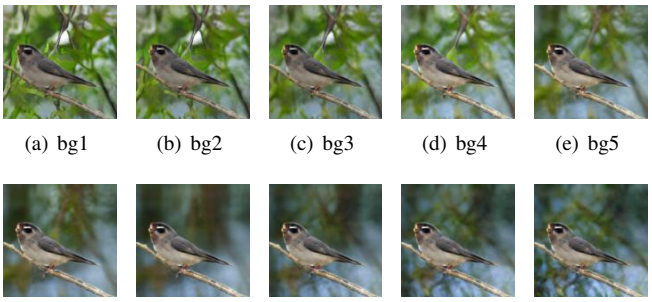
(g) fake21 (h) fake22 (i) fake23

(j) real21 (k) real22 (l) real23

(m) fake31 (n) fake32 (o) fake33

(p) real31 (q) real32 (r) real33

Figure 2. Conditional generation results



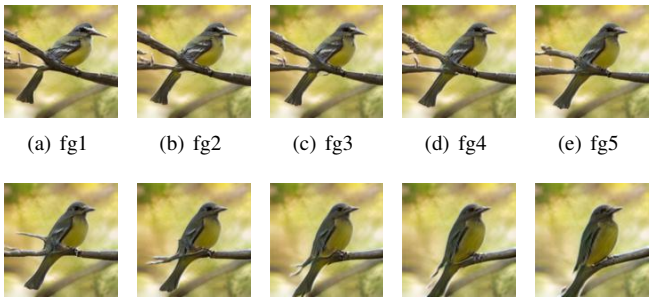
(a) bg1 (b) bg2 (c) bg3 (d) bg4 (e) bg5

(f) bg6 (g) bg7 (h) bg8 (i) bg9 (j) bg10

Figure 3. Background linear interpolation

4.3. Quantitative Results

In Tab. 1, we report the performance of the proposed BF-GAN on CUB benchmark. We mainly evaluate on two metrics, namely IS and FID. Note that, both the Simple-GAN and MixNMatch treat this task as normal generation task, have 20 samples for each fine-grain categories. Our BF-GAN perform few-shot generation on this benchmark, have only 3 samples for each fine-grain categories, which is much less. As a result, the performance of the proposed BF-GAN is naturally poor than Simple-GAN and MixNMatch.



(a) fg1 (b) fg2 (c) fg3 (d) fg4 (e) fg5

(f) fg6 (g) fg7 (h) fg8 (i) fg9 (j) fg10

Figure 4. Foreground linear interpolation

Method	IS	FID
Simple-GAN	31.85	16.69
MixNMatch[15]	50.05	9.17
Ours	26.88	71.93

Table 1. Quantitative result on CUB dataset with IS and FID metric comapring to some baseline.

5. Conclusion

We mainly followed the idea of [11, 15], and propose the thought of synthesising the mixed foreground features and few-shot generating background feature. Which is a new combination of the few-shot image generation field. For future work, we will fine-tune the models and test them on more datasets.

References

[1] Antreas Antoniou, Amos Storkey, and Harrison Edwards. Data augmentation generative adversarial networks. *arXiv preprint arXiv:1711.04340*, 2017. 1

[2] Sergey Bartunov and Dmitry Vetrov. Few-shot generative modelling with generative matching networks. In *International Conference on Artificial Intelligence and Statistics*, pages 670–678, 2018. 2

[3] Louis Clouâtre and Marc Demers. Figr: Few-shot image generation with reptile. *arXiv preprint arXiv:1901.02199*, 2019. 1, 2

[4] Yuanqiang Fang, Wengang Zhou, Yijuan Lu, Jinhui Tang, Qi Tian, and Houqiang Li. Cascaded feature augmentation with diffusion for image retrieval. In Susanne Boll, Kyoung Mu Lee, Jiebo Luo, Wenwu Zhu, Hyeran Byun,

Chang Wen Chen, Rainer Lienhart, and Tao Mei, editors, *2018 ACM Multimedia Conference on Multimedia Conference, MM 2018, Seoul, Republic of Korea, October 22-26, 2018*, pages 1644–1652. ACM, 2018. 1

[5] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*, 2017. 1

[6] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 2017. 2

[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*, pages 2672–2680, 2014. 1, 2

[8] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 6626–6637, 2017. 4

[9] Yan Hong, Li Niu, Jianfu Zhang, Jing Liang, and Liqing Zhang. Deltagan: Towards diverse few-shot image generation with sample-specific delta. *arXiv preprint arXiv:2009.08753*, 2020. 1

[10] Yan Hong, Li Niu, Jianfu Zhang, and Liqing Zhang. Matchinggan: Matching-based few-shot image generation. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2020. 1, 2

[11] Yan Hong, Li Niu, Jianfu Zhang, Weijie Zhao, Chen Fu, and Liqing Zhang. F2gan: Fusing-and-filling gan for few-shot image generation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2535–2543, 2020. 1, 2, 5

[12] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 1

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 1106–1114, 2012. 1

[14] Brenden M. Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua B. Tenenbaum. One shot learning of simple visual concepts. In Laura A. Carlson, Christoph Hölscher, and Thomas F. Shipley, editors, *Proceedings of the 33th Annual Meeting of the Cognitive Science Society, CogSci 2011, Boston, Massachusetts, USA, July 20-23, 2011*. cognitivesciencesociety.org, 2011. 2

[15] Yuheng Li, Krishna Kumar Singh, Utkarsh Ojha, and Yong Jae Lee. Mixnmatch: Multifactor disentanglement and encoding for conditional image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8039–8048, 2020. 1, 2, 5

[16] Weixin Liang, Zixuan Liu, and Can Liu. Dawson: A domain adaptive few shot generation framework. *arXiv preprint arXiv:2001.00576*, 2020. 1

[17] 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 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 10550–10559. IEEE, 2019. 2

[18] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 2

[19] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018. 1, 2

[20] Yunchen Pu, Zhe Gan, Ricardo Henao, Xin Yuan, Chunyuan Li, Andrew Stevens, and Lawrence Carin. Variational autoencoder for deep learning of images, labels and captions. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 2352–2360, 2016. 2

[21] Danilo Jimenez Rezende, Shakir Mohamed, Ivo Danihelka, Karol Gregor, and Daan Wierstra. One-shot generalization in deep generative models. In Maria-Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1521–1529. JMLR.org, 2016. 2

[22] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Abhishek Kumar, Rogério Schmidt Feris, Raja Giryes, and Alexander M. Bronstein. Delta-encoder: an effective sample synthesis method for few-shot object recognition. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 2850–2860, 2018. 1

[23] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Process-*

ing Systems 2016, December 5-10, 2016, Barcelona, Spain, pages 3630–3638, 2016. 2

[24] Han Xu, Pengwei Liang, Wei Yu, Junjun Jiang, and Jiayi Ma. Learning a generative model for fusing infrared and visible images via conditional generative adversarial network with dual discriminators. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 3954–3960. ijcai.org, 2019. 2

[25] Qiantong Xu, Gao Huang, Yang Yuan, Chuan Guo, Yu Sun, Felix Wu, and Kilian Q. Weinberger. An empirical study on evaluation metrics of generative adversarial networks. *CoRR*, abs/1806.07755, 2018. 4

[26] Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Seong Joon Oh, Youngjoon Yoo, and Junsuk Choe. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 6022–6031. IEEE, 2019. 1

[27] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 1