

Few-Shot Unsupervised Image-to-Image Translation

Ming-Yu Liu¹, Xun Huang^{1,2}, Arun Mallya¹, Tero Karras¹, Timo Aila¹, Jaakko Lehtinen^{1,3}, Jan Kautz¹

¹NVIDIA, ²Cornell University, ³Aalto University

{mingyul, xunh, amallya, tkarras, taila, jlehtinen, jkautz}@nvidia.com

Abstract

Unsupervised image-to-image translation methods learn to map images in a given class to an analogous image in a different class, drawing on unstructured (non-registered) datasets of images. While remarkably successful, current methods require access to many images in both source and destination classes at training time. We argue this greatly limits their use. Drawing inspiration from the human capability of picking up the essence of a novel object from a small number of examples and generalizing from there, we seek a few-shot, unsupervised image-to-image translation algorithm that works on previously unseen target classes that are specified, at test time, only by a few example images. Our model achieves this few-shot generation capability by coupling an adversarial training scheme with a novel network design. Through extensive experimental validation and comparisons to several baseline methods on benchmark datasets, we verify the effectiveness of the proposed framework. Our implementation and datasets are available at <https://github.com/NVLabs/FUNIT>.

1. Introduction

Humans are remarkably good at generalization. When given a picture of a previously unseen exotic animal, say, we can form a vivid mental picture of the same animal in a different pose, especially when we have encountered (images of) similar but different animals in that pose before. For example, a person seeing a standing tiger for the first time will have no trouble imagining what it will look lying down, given a lifetime of experience of other animals.

While recent unsupervised image-to-image translation algorithms are remarkably successful in transferring complex appearance changes across image classes [28, 43, 26, 23, 51, 48], the capability to generalize from few samples of a new class based on prior knowledge is entirely beyond their reach. Concretely, they need large training sets over all classes of images they are to perform translation on, i.e., they do not support few-shot generalization.

As an attempt to bridge the gap between human and machine imagination capability, we propose the Few-shot UN-

supervised Image-to-image Translation (FUNIT) framework, aiming at learning an image-to-image translation model for mapping an image of a source class to an analogous image of a target class by leveraging few images of the target class given at test time. The model is never shown images of the target class during training but is asked to generate some of them at test time. To proceed, we first hypothesize that the few-shot generation capability of humans develops from their past visual experiences—a person can better imagine views of a new object if the person has seen many more different object classes in the past. Based on the hypothesis, we train our FUNIT model using a dataset containing images of many different object classes for simulating the past visual experiences. Specifically, we train the model to translate images from one class to another class by leveraging few example images of the another class. We hypothesize that by learning to extract appearance patterns from the few example images for the translation task, the model learns a generalizable appearance pattern extractor that can be applied to images of unseen classes at test time for the few-shot image-to-image translation task. In the experiment section, we give empirical evidence that the few-shot translation performance improves as the number of classes in the training set increases.

Our framework is based on Generative Adversarial Networks (GAN) [13]. We show that by coupling an adversarial training scheme with a novel network design we achieve the desired few-shot unsupervised image-to-image translation capability. Through extensive experimental validation on three datasets, including comparisons to several baseline methods using a variety of performance metrics, we verify the effectiveness of our proposed framework. In addition, we show the proposed framework can be applied to the few-shot image classification task. By training a classifier on the images generated by our model for the few-shot classes, we are able to outperform a state-of-the-art few-shot classification method that is based on feature hallucination.

2. Related Work

Unsupervised/unpaired image-to-image translation aims at learning a conditional image generation function that can map an input image of a source class to an ana-

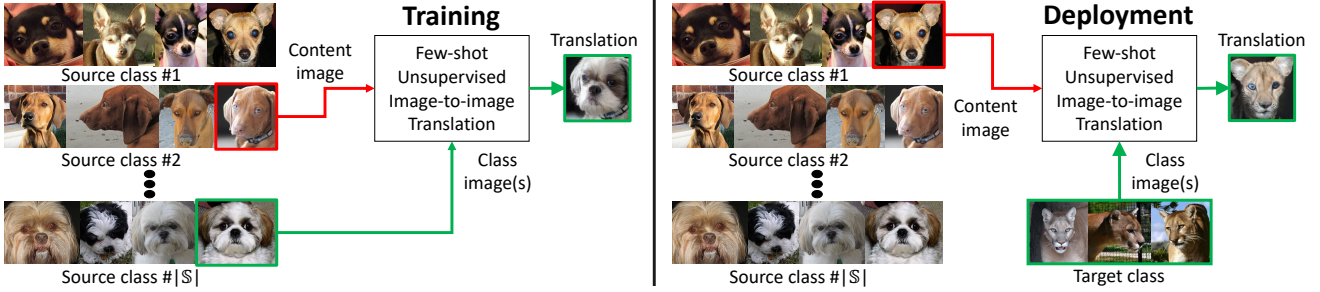


Figure 1. **Training.** The training set consists of images of various object classes (source classes). We train a model to translate images between these source object classes. **Deployment.** We show our trained model very few images of the target class, which is sufficient to translate images of source classes to analogous images of the target class even though the model has never seen a single image from the target class during training. Note that the FUNIT generator takes two inputs: 1) a content image and 2) a set of target class images. It aims to generate a translation of the input image that resembles images of the target class.

logues image of a target class without pair supervision. This problem is inherently ill-posed as it attempts to recover the joint distribution using samples from marginal distributions [26, 28]. To deal with the problem, existing works use additional constraints. For example, some works enforce the translation to preserve certain properties of the source data, such as pixel values [38], pixel gradients [5], semantic features [43], class labels [5], or pairwise sample distances [3]. There are works enforcing the cycle consistency constraint [48, 51, 23, 1, 52]. Several works use the shared/partially-shared latent space assumption [26, 28]/[18, 24]. Our work is based on the partially-shared latent space assumption but is designed for the few-shot unsupervised image-to-image translation task.

While capable of generating realistic translation outputs, existing unsupervised image-to-image translation models are limited in two aspects. First, they are sample inefficient, generating poor translation outputs if only few images are given at training time. Second, the learned models are limited for translating images between two classes. A trained model for one translation task cannot be directly reused for a new task despite similarity between the new task and the original task. For example, a husky-to-cat translation model can not be re-purposed for husky-to-tiger translation even though cat and tiger share a great similarity.

Recently, Benaim and Wolf [4] proposed an unsupervised image-to-image translation framework for partially addressing the first aspect. Specifically, they use a training dataset consisting of one source class image but many target class images to train a model for translating the *single* source class image to an analogous image of the target class. Our work differs from their work in several major ways. First, we assume many source class images but few target class images. Moreover, we assume that the few target class images are only available at test time and can be from many different object classes.

Multi-class unsupervised image-to-image translation [7,

2, 19] extends the unsupervised image-to-image translation methods to multiple classes. Our work is similar to these methods in the sense that our training dataset consists of images of multiple classes. But instead of translating images among *seen* classes, we focus on translating images of seen classes to analogous images of previously *unseen* classes.

Few-shot classification. Unlike few-shot image-to-image translation, the task of learning classifiers for novel classes using few examples is a long-studied problem. Early works use generative models of appearance that share priors across classes in a hierarchical manner [10, 36]. More recent works focus on using meta-learning to quickly adapt models to novel tasks [11, 32, 35, 31]. These methods learn better optimization strategies for training, so that the performance upon seeing only few examples is improved. Another set of works focus on learning image embeddings that are better suited for few-shot learning [45, 40, 41]. Several recent works propose augmenting the training set for the few-shot classification task by generating new feature vectors corresponding to novel classes [9, 14, 47]. Our work is designed for few-shot unsupervised image-to-image translation. However, it can be applied to few-shot classification, as shown in the experiments section.

3. Few-shot Unsupervised Image Translation

The proposed FUNIT framework aims at mapping an image of a source class to an analogous image of an unseen target class by leveraging a few target class images that are made available at test time. To train FUNIT, we use images from a set of object classes (*e.g.* images of various animal species), called the source classes. We do not assume existence of paired images between any two classes (*i.e.* no two animals of different species are at exactly the same pose). We use the source class images to train a multi-class unsupervised image-to-image translation model. During testing, we provide the model few images from a novel object class,

called the target class. The model has to leverage the few target images to translate any source class image to analogous images of the target class. When we provide the same model few images from a different novel object class, it has to translate any source class images to analogous images of the different novel object class.

Our framework consists of a conditional image generator G and a multi-task adversarial discriminator D . Unlike the conditional image generators in existing unsupervised image-to-image translation frameworks [51, 26], which take one image as input, our generator G simultaneously takes *a content image* \mathbf{x} and *a set of K class images* $\{\mathbf{y}_1, \dots, \mathbf{y}_K\}$ as input and produce the output image $\bar{\mathbf{x}}$ via

$$\bar{\mathbf{x}} = G(\mathbf{x}, \{\mathbf{y}_1, \dots, \mathbf{y}_K\}). \quad (1)$$

We assume the content image belongs to object class c_x while each of the K class images belong to object class c_y . In general, K is a small number and c_x is different from c_y . We will refer G as the few-shot image translator.

As shown in Figure 1, G maps an input content image \mathbf{x} to an output image $\bar{\mathbf{x}}$, such that $\bar{\mathbf{x}}$ looks like an image belonging to object class c_y , and $\bar{\mathbf{x}}$ and \mathbf{x} share structural similarity. Let \mathbb{S} and \mathbb{T} denote the set of source classes and the set of target classes, respectively. During training, G learns to translate images between two randomly sampled source classes $c_x, c_y \in \mathbb{S}$ with $c_x \neq c_y$. At test time, G takes a few images from an unseen target class $c \in \mathbb{T}$ as the class images, and maps an image sampled from any of the source classes to an analogous image of the target class c .

Next, we discuss the network design and learning. More details are available in our full technical paper [27].

3.1. Few-shot Image Translator

The few-shot image translator G consists of a content encoder E_x , a class encoder E_y , and a decoder F_x . The content encoder is made of several 2D convolutional layers followed by several residual blocks [15, 20]. It maps the input content image \mathbf{x} to a content latent code \mathbf{z}_x , which is a spatial feature map. The class encoder consists of several 2D convolutional layers followed by a mean operation along the sample axis. Specifically, it first maps each of the K individual class images $\{\mathbf{y}_1, \dots, \mathbf{y}_K\}$ to an intermediate latent vector and then computes the mean of the intermediate latent vectors to obtain the final class latent code \mathbf{z}_y .

The decoder consists of several adaptive instance normalization (AdaIN) residual blocks [18] followed by a couple of upscale convolutional layers. The AdaIN residual block is a residual block using the AdaIN [17] as the normalization layer. For each sample, AdaIN first normalizes the activations of a sample in each channel to have a zero mean and unit variance. It then scales the activations using a learned affine transformation consisting of a set of scalars and biases. Note that the affine transformation is spatially

invariant and hence can only be used to obtain global appearance information. The affine transformation parameters are adaptively computed using \mathbf{z}_y via a two-layer fully connected network. With E_x , E_y , and F_x , (1) becomes

$$\bar{\mathbf{x}} = F_x(\mathbf{z}_x, \mathbf{z}_y) = F_x(E_x(\mathbf{x}), E_y(\{\mathbf{y}_1, \dots, \mathbf{y}_K\})). \quad (2)$$

By using this translator design, we aim at extracting class-invariant latent representation (*e.g.*, object pose) using the content encoder and extracting class-specific latent representation (*e.g.*, object appearance) using the class encoder. By feeding the class latent code to the decoder via the AdaIN layers, we let the class images control the global look (*e.g.*, object appearance), while the content image determines the local structure (*e.g.*, locations of eyes).

At training time, the class encoder learns to extract class-specific latent representation from the images of the source classes. At test time, this generalizes to images of previously unseen classes. In the experiment section, we show that the generalization capability depends on the number of source object classes seen during training. When G is trained with more source classes (*e.g.*, more species of animals), it has a better few-shot image translation performance (*e.g.*, better in translating husky to mountain lion).

3.2. Multi-task Adversarial Discriminator

Our discriminator D is trained by solving multiple adversarial classification tasks simultaneously. Each of the tasks is a binary classification task determining whether an input image is a real image of the source class or a translation output coming from G . As there are $|\mathbb{S}|$ source classes, D produces $|\mathbb{S}|$ outputs. When updating D for a real image of source class c_x , we penalize D if its c_x th output is false. For a translation output yielding a fake image of source class c_x , we penalize D if its c_x th output is positive. We do not penalize D for not predicting false for images of other classes ($\mathbb{S} \setminus \{c_x\}$). When updating G , we only penalize G if the c_x th output of D is false. We empirically find this discriminator works better than a discriminator trained by solving a much harder $|\mathbb{S}|$ -class classification problem.

3.3. Learning

We train the proposed FUNIT framework by solving a minimax optimization problem given by

$$\min_D \max_G \mathcal{L}_{\text{GAN}}(D, G) + \lambda_R \mathcal{L}_R(G) + \lambda_F \mathcal{L}_{\text{FM}}(G) \quad (3)$$

where \mathcal{L}_{GAN} , \mathcal{L}_R , and \mathcal{L}_F are the GAN loss, the content image reconstruction loss, and the feature matching loss. The GAN loss is a conditional one given by

$$\begin{aligned} \mathcal{L}_{\text{GAN}}(G, D) = & E_{\mathbf{x}} [-\log D^{c_x}(\mathbf{x})] + \\ & E_{\mathbf{x}, \{\mathbf{y}_1, \dots, \mathbf{y}_K\}} [\log (1 - D^{c_y}(\bar{\mathbf{x}}))] \end{aligned} \quad (4)$$

	Setting	Top1-all \uparrow	Top5-all \uparrow	Top1-test \uparrow	Top5-test \uparrow	DIPD \downarrow	IS-all \uparrow	IS-test \uparrow	mFID \downarrow
Animal Faces	CycleGAN-Unfair-20	28.97	47.88	38.32	71.82	1.615	10.48	7.43	197.13
	UNIT-Unfair-20	22.78	43.55	35.73	70.89	1.504	12.14	6.86	197.13
	MUNIT-Unfair-20	38.61	62.94	53.90	84.00	1.700	10.20	7.59	158.93
	StarGAN-Unfair-1	2.56	10.50	9.07	32.55	1.311	10.49	5.17	201.58
	StarGAN-Unfair-5	12.99	35.56	25.40	60.64	1.514	7.46	6.10	204.05
	StarGAN-Unfair-10	20.26	45.51	30.26	68.78	1.559	7.39	5.83	208.60
	StarGAN-Unfair-15	20.47	46.46	34.90	71.11	1.558	7.20	5.58	204.13
	StarGAN-Unfair-20	24.71	48.92	35.23	73.75	1.549	8.57	6.21	198.07
	StarGAN-Fair-1	0.56	3.46	4.41	20.03	1.368	7.83	3.71	228.74
	StarGAN-Fair-5	0.60	3.56	4.38	20.12	1.368	7.80	3.72	235.66
	StarGAN-Fair-10	0.60	3.40	4.30	20.00	1.368	7.84	3.71	241.77
	StarGAN-Fair-15	0.62	3.49	4.28	20.24	1.368	7.82	3.72	228.42
	StarGAN-Fair-20	0.62	3.45	4.41	20.00	1.368	7.83	3.72	228.57
	FUNIT-1	17.07	54.11	46.72	82.36	1.364	22.18	10.04	93.03
	FUNIT-5	33.29	78.19	68.68	96.05	1.320	22.56	13.33	70.24
	FUNIT-10	37.00	82.20	72.18	97.37	1.311	22.49	14.12	67.35
	FUNIT-15	38.83	83.57	73.45	97.77	1.308	22.41	14.55	66.58
	FUNIT-20	39.10	84.39	73.69	97.96	1.307	22.54	14.82	66.14
North American Birds	CycleGAN-Unfair-20	9.24	22.37	19.46	42.56	1.488	25.28	7.11	215.30
	UNIT-Unfair-20	7.01	18.31	16.66	37.14	1.417	28.28	7.57	203.83
	MUNIT-Unfair-20	23.12	41.41	38.76	62.71	1.656	24.76	9.66	198.55
	StarGAN-Unfair-1	0.92	3.83	3.98	13.73	1.491	14.80	4.10	266.26
	StarGAN-Unfair-5	2.54	8.94	8.82	23.98	1.574	13.84	4.21	270.12
	StarGAN-Unfair-10	4.26	13.28	12.03	32.02	1.571	15.03	4.09	278.94
	StarGAN-Unfair-15	3.70	11.74	12.90	31.62	1.509	18.61	5.25	252.80
	StarGAN-Unfair-20	5.38	16.02	13.95	33.96	1.544	18.94	5.24	260.04
	StarGAN-Fair-1	0.24	1.17	0.97	4.84	1.423	13.73	4.83	244.65
	StarGAN-Fair-5	0.22	1.07	1.00	4.86	1.423	13.72	4.82	244.40
	StarGAN-Fair-10	0.24	1.13	1.03	4.90	1.423	13.72	4.83	244.55
	StarGAN-Fair-15	0.23	1.05	1.04	4.90	1.423	13.72	4.81	244.80
	StarGAN-Fair-20	0.23	1.08	1.00	4.86	1.423	13.75	4.82	244.71
	FUNIT-1	11.17	34.38	30.86	60.19	1.342	67.17	17.16	113.53
	FUNIT-5	20.24	51.61	45.40	75.75	1.296	74.81	22.37	99.72
	FUNIT-10	22.45	54.89	48.24	77.66	1.289	75.40	23.60	98.75
	FUNIT-15	23.18	55.63	49.01	78.70	1.287	76.44	23.86	98.16
	FUNIT-20	23.50	56.37	49.81	78.89	1.286	76.42	24.00	97.94

Table 1. Performance comparison with the fair and unfair baselines. \uparrow means larger numbers are better, \downarrow means smaller numbers are better.

The superscript attached to D denotes the object class; the loss is computed only using the corresponding binary prediction score of the class.

The content reconstruction loss helps G learn a translation model. Specifically, when using the same image for both the input content image and the input class image (in this case $K = 1$), the loss encourages G to generate an output image identical to the input

$$\mathcal{L}_R(G) = E_{\mathbf{x}} [\|\mathbf{x} - G(\mathbf{x}, \{\mathbf{x}\})\|_1^1]. \quad (5)$$

The feature matching loss regularizes the training. We first construct a feature extractor, referred to as D_f , by removing the last (prediction) layer from D . We then use D_f to extract features from the translation output $\bar{\mathbf{x}}$ and the class images $\{\mathbf{y}_1, \dots, \mathbf{y}_K\}$ and minimize

$$\mathcal{L}_F(G) = E_{\mathbf{x}, \{\mathbf{y}_1, \dots, \mathbf{y}_K\}} [\|D_f(\bar{\mathbf{x}}) - \sum_k \frac{D_f(\mathbf{y}_k)}{K}\|_1^1]. \quad (6)$$

Both of the content reconstruction loss and the feature matching loss are not new topics to image-to-image transla-

tion [26, 18, 46, 34]. Our contribution is in extending their use to the more challenging and novel few-shot unsupervised image-to-image translation setting.

4. Experiments

Implementation. We set $\lambda_R = 0.1$ and $\lambda_F = 1$. We optimize (3) using RMSProp with learning rate 0.0001. We use the hinge version of GAN loss [25, 30, 49, 6] and the real gradient penalty regularization proposed by Mescheder *et al.* [29]. The final generator is a historical average version of the intermediate generators [21] where the update weight is 0.001. We train the FUNIT model using $K = 1$ since we desire it to work well even when *only one* target class image is available at test time. In the experiments, we evaluate its performance under $K = 1, 5, 10, 15, 20$. Each training batch consists of 64 content images, which are evenly distributed on 8 V100 GPUs in an NVIDIA DGX1 machine.

Datasets. We use the following datasets for experiments.

- *Animal Faces.* We build this dataset using images from the 149 carnivorous animal classes in ImageNet [8]. We

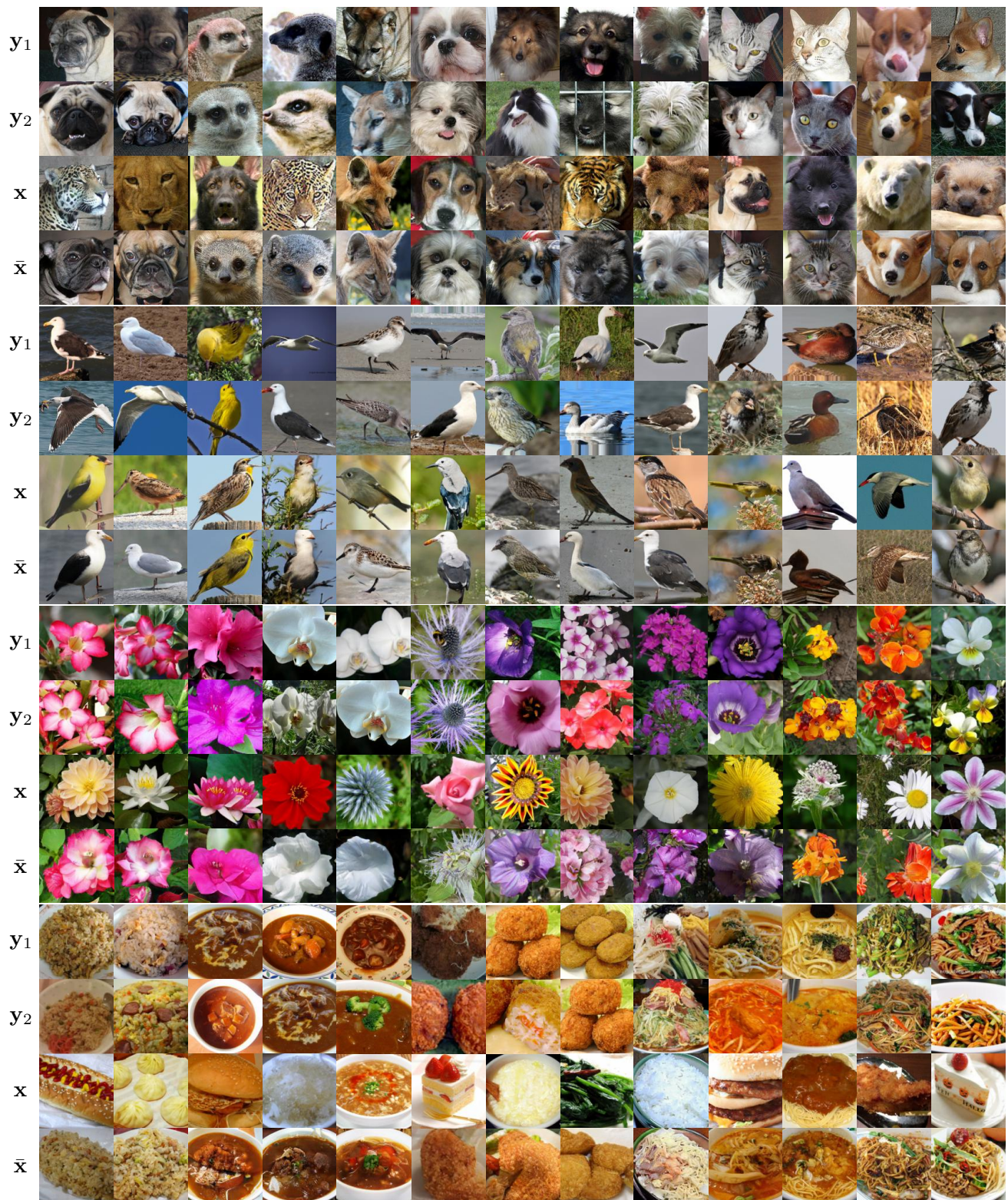


Figure 2. Visualization of the *few-shot* unsupervised image-to-image translation results. The results are computed using the FUNIT-5 model. From top to bottom, we have the results from the animal face, bird, flower, and food datasets. We train one model for each dataset. For each example, we visualize 2 out of 5 randomly sampled class images $y_1 y_2$, the input content image x , and the translation output \bar{x} . The results show that FUNIT generate plausible translation outputs under the difficult few-shot setting where the models see no images from any of the target classes during training. We note that the objects in the output images have similar poses to the inputs.

first manually label bounding boxes of 10000 carnivorous animal faces in the images. We then train a Faster RCNN [12] to detect animal faces in the images. We only use the bounding boxes with high detection scores. This renders a set of 117574 animal faces. We split the classes into a source class set and a target class set, which contains 119 and 30 animal classes, respectively.

- *Birds* [44]. 48527 images of 555 North American bird species. 444 species are used for the source class set and 111 species are used for the target class set.
- *Flowers* [33]. 8189 images from 102 species. The source and target sets have 85 and 17 species, respectively.
- *Foods* [22]. 31395 images from 256 kinds of food. The source and target set have 224 and 32 kinds, respectively.

Baselines. Depending on whether images of the target class are available during training, we define two sets of baselines: *fair* (unavailable) and *unfair* (available).

- *Fair*. This is the setting of the proposed FUNIT framework. As none of the prior unsupervised image-to-image translation methods are designed for the setting, we build a baseline by extending the StarGAN method [7], which is the state of the art for multi-class unsupervised image-to-image translation. We train a StarGAN model purely using source class images. During testing, given K images of a target class, we compute the average VGG [39] Conv5 features for the K images and compute its cosine distance to the average VGG Conv5 feature for the images of each source class. We then compute the class association vector by applying softmax to the cosine distances. We use the class association vector as input to the StarGAN model (substituting the one-hot class association vector input) for generating images of unseen target classes. The baseline method is designed with the assumption that the class association scores could encode how an unseen target object class is related to each of the source classes, which can be used for few-shot generation. We denote this baseline StarGAN-Fair- K .
- *Unfair*. These baselines include target class images in the training. We vary the number of available images (K) per target class from 1 to 20 and train various unsupervised image-to-image translation models. We denote the StarGAN model that is trained with K images per target class as StarGAN-Unfair- K . We also train several state-of-the-art two-domain translation models including CycleGAN [51], UNIT [26], and MUNIT [18]. For them, we treat images of the source classes as the first domain and images of one target class as the second domain. This results in $|\mathbb{T}|$ unsupervised image-to-image translation models per dataset per two-class baseline. We label these baselines as CycleGAN-Unfair- K , UNIT-Unfair- K , and MUNIT-Unfair- K .

For the baseline methods, we use the source code and de-



Figure 3. Visual comparison of few-shot image-to-image translation performance. From left to right, the columns are input content images x , the two input target class images y_1 y_2 , translation results from the unfair StarGAN baseline, translation results from the fair StarGAN baseline, and results from our framework.

fault parameter settings provided by the authors.

Evaluation protocol. We use a randomly sampled 25000 images from the source classes as the content images. We then translate them to each target class by randomly sample K images of the target class. This produces $|\mathbb{T}|$ sets of images for each competing approach and they are used for evaluation. We use the same K images for each content image for all the competing approaches. We test a range of K values, including 1, 5, 10, 15, and 20.

Performance metrics. We use several criteria for evaluation. First, we measure whether translations resemble images of the target class. Second, we examine whether class-invariant content are preserved during translation. Third, we quantify photorealism of output images. Finally, we measure whether the model can be used to generate the image distribution of a target class. We briefly describe our performance metrics for these criteria below and leave their details in our full technical paper [27].

- *Translation accuracy* measure whether a translation output belongs to the target class. We use two Inception-V3 [42] classifiers. One classifier is trained using both of the source and target classes (denoted as *all*), while the other is trained using the target classes along (denoted as *test*). We report both Top1 and Top5 accuracies.

- *Content preservation* is based on a variant of perceptual distance [20, 50], called the domain-invariant perceptual distance (DIPD) [18]. The distance is given by L2 distance between two normalized VGG [39] Conv5 features, which is more invariant against domain change [18].
- *Photorealism*. This is measured by the inception scores (IS) [37]. We report inception scores using the two inception classifiers trained for measuring translation accuracy, denoted by *all* and *test*, respectively.
- *Distribution matching* is based on Fréchet Inception Distance (FID) [16]. We compute FID for each of the $|\mathbb{T}|$ target object classes and report their mean FID (mFID).

Main results. As shown in Table 1, the proposed FUNIT framework outperforms the baselines for the few-shot unsupervised image-to-image translation task on all the performance metrics for both the Animal Faces and North American Birds datasets. FUNIT achieves 82.36 and 96.05 Top-5 (*test*) accuracy for the 1-shot and 5-shot settings, respectively, on the Animal Face dataset, and 60.19 and 75.75 on the North American Birds dataset. They are all significantly better than those achieved by the corresponding fair baselines. Similar trends can be found for the domain invariant perceptual distance, inception score, and Fréchet inception distance. Moreover, with just 5 shots, FUNIT outperforms all the unfair baselines under 20-shot settings. Note that for the results of CycleGAN-Unfair-20, UNIT-Unfair-20, and MUNIT-Unfair-20 are from $|\mathbb{T}|$ image-to-image translation networks, while our method is from a single translation network.

The table also shows that the performance of the proposed FUNIT model is positively correlated with number of available target images K at test time. A larger K leads to improvements across all the metrics, and the largest performance boost comes from $K = 1$ to $K = 5$. The StarGAN-Fair baseline does not exhibit a similar trend.

In Figure 2, we visualize the few-shot translation results computed by FUNIT-5. The results show that the FUNIT model can successfully translate images of source classes to analogous images of novel classes. The poses of the object in the input content image x and the corresponding output image \bar{x} remain largely the same. The output images are photorealistic and resemble images from the target classes.

In Figure 3, we provide a visual comparison. As the baselines are not designed for the few-shot image translation setting, they failed in the challenging translation task. They either generate images with a large amount of artifacts or just output the input content image. On the other hand, FUNIT generates high-quality image translation outputs.

User study. To compare the photorealism and faithfulness of the translation outputs, we perform human evaluation using the Amazon Mechanical Turk (AMT) platform. Specifically, we give the workers a target class image and two

Setting	Animal	Birds
FUNIT-5 vs. StarGAN-Fair-5	86.08	82.56
FUNIT-5 vs. StarGAN-Unfair-20	86.00	84.48
FUNIT-5 vs. CycleGAN-Unfair-20	71.68	77.76
FUNIT-5 vs. UNIT-Unfair-20	77.84	77.96
FUNIT-5 vs. MUNIT-Unfair-20	83.56	79.64

Table 2. User preference score. The numbers indicate the percentage of users favors results generated by the proposed method over those generated by the competing method.

# of generated samples N	Animal Face		North American Birds	
	S&H [14]	FUNIT	S&H [14]	FUNIT
0	38.76		30.38	
10	40.51	42.05	31.77	33.41
50	40.24	42.22	31.66	33.64
100	40.76	42.14	32.12	34.39

Table 3. Few-shot classification accuracies averaged over 5 splits.

translation outputs from different methods [46, 18] and ask them to choose the output image that resembles more the target class image. The workers are given unlimited time to make the selection. We use both the Animal Faces and North American Birds datasets. For each comparison, we randomly generate 500 questions and each question is answered by 5 different workers. For quality control, a worker must have a lifetime task approval rate greater than 98% to be able to participate in the evaluation.

According to Table 2, the human subjects consider the translation outputs generated by the proposed method under the 5-shot setting (FUNIT-5) much more similar to the target class images than those generated by the fair baseline under the same setting (StarGAN-Fair-5). Even when compared with the results of unfair baselines that have access to 20 images per target class at training time, our translation results are still considered to be much more faithful.

Number of source classes in the training set. In Figure 4, we analyze the performance versus varying number of source classes in the training set under the one-shot setting (FUNIT-1), using the animal dataset. We plot the curves by varying the number from 69 to 119 classes with an interval of 10. As shown, the performance is positively correlated with the number of object classes in terms of translation accuracy, image quality, and distribution matching. The domain-invariant perceptual distance remains flat. This shows that a FUNIT model that sees more object classes (larger diversity) during training performs better during testing. A similar trend is observed for the bird dataset, which is given in our full technical paper [27].

Comparison to AdaIN. We train an AdaIN style transfer network [17] for the few-shot animal face translation task and compare the results to ours. The visual comparison is given in the full technical report [27]. While the style transfer network can change the textures of the input animals, it

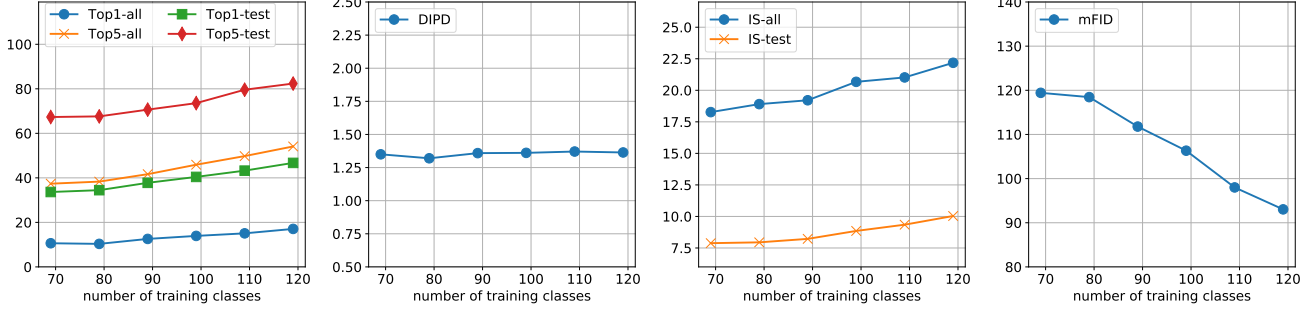


Figure 4. Few-shot image translation performance vs. number of object classes seen during training on the Animal Faces dataset. The performance is positively correlated with number of source object classes seen during training.

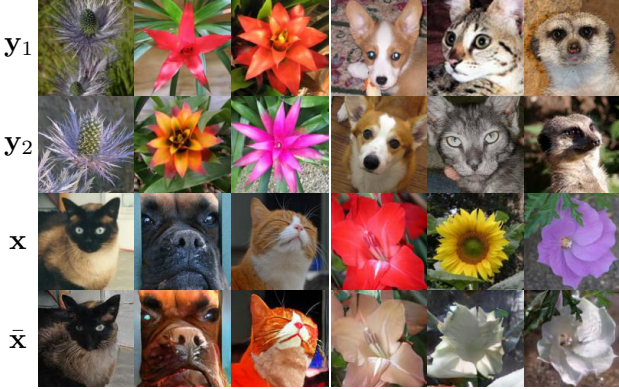


Figure 5. Limitations of the proposed framework. When the appearance of an unseen object class is dramatically different to the appearances of the source classes, (e.g. flower and animal face). The proposed FUNIT framework fails to generate meaningful translation outputs.

does not change their shapes. As a result, the translation outputs still resemble to the inputs.

Parameter analysis and ablation study. We analyze the impact of the individual terms in our objective function and find all of them are essential. Particularly, the content reconstruction loss trades translation accuracies for content preservation score. Supporting experiment results are given in our full technical paper [27].

Latent interpolation. In our full technical paper [27], we show interpolation results by keeping the content code fixed while interpolating the class code between those of two source class images. Interestingly, we find that by interpolating between two source classes (Siamese cat and Tiger) we can sometimes generate a target class (Tabby cat) that the model has never observed.

Failure cases. Several failure cases of the proposed algorithm are visualized in our full technical paper [27]. They include generating hybrid objects, ignoring input content images, and ignoring input class images.

Few-shot translation for few-shot classification. We evaluate FUNIT for few-shot classification using the animal and

bird datasets. Specifically, we use the trained FUNIT models to generate N (varying from 1, 50, to 100) images for each of the few-shot classes and use the generated images to train the classifiers. We find the classifiers trained with the FUNIT generated images consistently achieve better performance than the few-shot classification approach proposed of S&H by Hariharan *et al.* [14], which is based on feature hallucination and also has a controllable variable on sample number N . The results are shown in Table 3 and the experiment details are in our full technical paper [27].

5. Discussion and Future Work

We introduced the first few-shot unsupervised image-to-image translation framework. We showed that the few-shot generation performance is positively correlated with the number of object classes seen during training and also positively correlated with the number of target class shots provided during test time.

We provided empirical evidence that FUNIT can learn to translate an image of a source class to a corresponding image of an unseen object class by utilizing few example images of the unseen class made available at test time. Although achieving this new capability, FUNIT depends on several conditions to work: 1) whether the content encoder E_x can learn a class-invariant latent code \mathbf{z}_x , 2) whether the class encoder E_y can learn a class-specific latent code \mathbf{z}_y , and, most importantly, 3) whether the class encoder E_y can generalize to images of unseen object classes.

We observed these conditions are easy to meet when the novel classes are visually related to the source classes. However, when the appearance of novel object classes are dramatically different from those of the source classes, FUNIT fails to achieve translation as shown in Figure 5. In this case, FUNIT tends to generate color-changed versions of the input content images. This is undesirable but understandable as the appearance distribution has changed dramatically. Addressing this limitation is our future work.

References

- [1] Amjad Almahairi, Sai Rajeswar, Alessandro Sordoni, Philip Bachman, and Aaron Courville. Augmented cyclegan: Learning many-to-many mappings from unpaired data. *arXiv preprint arXiv:1802.10151*, 2018.
- [2] Asha Anooosheh, Eirikur Agustsson, Radu Timofte, and Luc Van Gool. Combogan: Unrestrained scalability for image domain translation. *arXiv preprint arXiv:1712.06909*, 2017.
- [3] Sagie Benaïm and Lior Wolf. One-sided unsupervised domain mapping. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.
- [4] Sagie Benaïm and Lior Wolf. One-shot unsupervised cross domain translation. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.
- [5] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [6] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *International Conference on Learning Representations (ICLR)*, 2019.
- [7] Yunje 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
- [9] Mandar Dixit, Roland Kwitt, Marc Niethammer, and Nuno Vasconcelos. Aga: Attribute-guided augmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [10] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2006.
- [11] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, 2017.
- [12] Ross Girshick. Fast r-cnn. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.
- [13] 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)*, 2014.
- [14] Bharath Hariharan and Ross B Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [16] 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 *Advances in Neural Information Processing Systems (NIPS)*, 2017.
- [17] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.
- [18] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. *European Conference on Computer Vision (ECCV)*, 2018.
- [19] Le Hui, Xiang Li, Jiaxin Chen, Hongliang He, and Jian Yang. Unsupervised multi-domain image translation with domain-specific encoders/decoders. *arXiv preprint arXiv:1712.02050*, 2017.
- [20] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, 2016.
- [21] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations (ICLR)*, 2018.
- [22] Yoshiyuki Kawano and Keiji Yanai. Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. In *European Conference on Computer Vision (ECCV) Workshop*, 2014.
- [23] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jungkwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *International Conference on Machine Learning (ICML)*, 2017.
- [24] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representation. In *European Conference on Computer Vision (ECCV)*, 2018.
- [25] Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017.
- [26] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.
- [27] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. *arXiv preprint arXiv:1905.01723*, 2019.
- [28] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.
- [29] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International Conference on Machine Learning (ICML)*, 2018.
- [30] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2018.
- [31] Tsendsuren Munkhdalai and Hong Yu. Meta networks. In *International Conference on Machine Learning (ICML)*, 2017.

- [32] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.
- [33] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics & Image Processing*, 2008.
- [34] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially adaptive normalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [35] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International Conference on Learning Representations (ICLR)*, 2017.
- [36] Ruslan Salakhutdinov, Joshua Tenenbaum, and Antonio Torralba. One-shot learning with a hierarchical nonparametric bayesian model. In *International Conference on Machine Learning (ICML) Workshop*, 2012.
- [37] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.
- [38] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. Learning from simulated and unsupervised images through adversarial training. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [39] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.
- [40] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.
- [41] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [42] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [43] Yaniv Taigman, Adam Polyak, and Lior Wolf. Unsupervised cross-domain image generation. In *International Conference on Learning Representations (ICLR)*, 2017.
- [44] Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [45] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.
- [46] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [47] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [48] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.
- [49] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*, 2018.
- [50] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [51] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.
- [52] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.