

FLPD-GANS: Fake License Plate Discrimination Gans for Generalized Zero-Shot Learning

Huaiyao Zhang¹⁺, Yi Zhang² and Caixin Zhu¹

¹ School of Software Engineering, University of Science and Technology of China, China

² School of Computer Science and Technology, University of Science and Technology of China, China

Abstract. Most current generalized zero-shot learning (GZSL) methods need sufficient labels and other auxiliary information to obtain great results. In this paper, we propose Fake License Plate Discrimination GANs (FLPD-GANs) and introduce the first publicly available New Energy License Plate (NELP) image dataset named CCPONECD. Applied in the license plate (LP) image binary classification task, FLPD-GANs only need a binary label for training and can address the strong bias problem in GZSL tasks. CCPONECD contains nearly 7k unique new energy vehicles images and provides detailed LP vertex location annotations. In our work, the seen class is only real NELP image and the unseen class is manufactured fake NELP image. Trained with merely real NELP images, our FLPD-GANs can greatly discriminate between real and fake NELP images. Extensive experiments demonstrate that our FLPD-GANs model has 97.7% accuracy and performs well in NELP image discrimination for GZSL task.

Keywords: GZSL, image classification, GANs, license plate discrimination.

1. Introduction

The performance of deep convolutional neural network (DCNN) [1] is now good enough on image classification tasks. However, highly accurate DCNN model needs numerous annotated samples for training. The dependence on large-scale annotated data has become the main bottleneck of progress in the use of deep learning. Because it is expensive to obtain enough annotated data [2]. In order to tackle such an unseen image recognition issue, generalized zero-shot learning (GZSL) is now extensively researched in some applications, such as autonomous object discovery system [3]. When evaluating on both seen and unseen class with only one seen class label available, it would be hard to classify images accurately for there exists a strong bias problem [4].

Recent development of Generative Adversarial Networks (GANs) [5] has made it a powerful tool for image generation tasks. GANs include a generator (G) network that synthesizes images by learning real image features and a discriminator (D) network that distinguishes real images from fake ones. G and D play a minimax game that guides G to generate realistic images. And a balance is reached when D being difficult to distinguish between real and generated samples. GANs exploit the G to generate a new fake category which adds projected anchor points specified like the seen classes in the semantic embedding space [4].

In this work, we combine GANs with GZSL to solve the strong bias problem. We propose Fake License Plate Discrimination Generative Adversarial Networks (FLPD-GANs), which classify LP images by using a DCNN training on the adversarial way and consider the LP image authenticity discrimination as a binary classification task. Image authenticity is how much the image looks like a real LP image (like Figure 1 (b)). To the best of our knowledge, FLPD-GANs are the first solution in a GANs framework that is applied in the field of LP image discrimination with only one seen class for training. The proposed FLPD-GANs adopt the

⁺ Corresponding author.
E-mail address: sa517489@mail.ustc.edu.cn.

arithmetic squares root loss function in the discriminator and can be trained end-to-end by back-propagation. Trained in minimizing the objective function, FLPD-GANs can overcome the problem of vanishing gradients. We use three assessment methods to enhance the robustness of the recognition results.

We evaluate FLPD-GANs on CCPONECD containing 6909 unique NELP images. The experimental results show our approach outperforms some conventional binary classification networks which are trained by both real and fake NELP images. It also has better performance than the state-of-the-art GZSL and Zero-shot learning (ZSL) approaches in a zero-shot learning scenario. Our experiments also demonstrate that FLPD-GANs can greatly distinguish the authenticity of the NELP and has good discrimination effect on our CCPONECD.

Our main contributions of this paper are as follows:

- We propose FLPD-GANs, which do not need large numbers of manual annotations. FLPD-GANs exploit the generator to solve the strong bias problem in GZSL tasks and train the discriminator to LP images binary classification tasks.
- We introduce CCPONECD, the first publicly available dataset for NELP detection and recognition to date. CCPONECD provides nearly 7k unique real new energy car images with detailed annotations.
- By evaluating conventional binary classifier with better condition and the state-of-the-art GZSL and ZSL approaches on CCPONECD, we demonstrate our FLPD-GANs outperform other approaches.

2. Related Work

2.1. Generative Adversarial Networks

Generative Adversarial Networks (GANs) exploit network parameters to learn the distribution features of the input data. Generator uses these parameters for re-generation and discriminator uses them for classification. We treat the image generated during the training as a new class to deal with the strong bias problem, and then use discriminator to classify. However, the loss function of discriminator may lead to the matter of vanishing gradients when updating the generator using the fake samples which are on the correct side of the decision boundary [6].

Therefore, many improved variants of GANs have been proposed, such as DCGANs [7], WGANs [8], WGANs-GP [9], LSGANs [6]. The LSGANs adopt the least squares loss function for the discriminator to generate more realistic sample and perform better training stability during the learning process. Due rapid development, it has significantly applied in various types of image synthesis task, like face synthesis [10], image translation [11] [12] [13] and estimating the 3D pose [14]. In this paper, comprehensively considering the ability to generation and discrimination, we expend LSGANs adversarial model in our methods for generalized zero-shot learning.

2.2. Zero-shot Learning

Zero-shot learning (ZSL) [3] [15] [16] tries to recognize objects whose instances may unseen during training by making use of some extra information that may be available for seen/unseen categories. The more challenging setting where the classifier needs to discriminate both seen and unseen categories labels at test time is known as generalized zero-shot learning (GZSL) [17] [18] [19], which is considered as a more formidable problem setting.

In order to improve the performance of GZSL, there are many studies that associate GANs with GZSL. [17] and [20] utilize semantic representation synthesizes visual features of unseen classes. [3] leverages generated samples from unseen classes to train the conventional classification model. These methods use GANs to generate images or features from semantic descriptions, and then exploit these generated samples to train a conventional classifier. However, these methods cannot be trained end-to-end. In this paper, we propose an end-to-end GANs model, which generates images just from noise and fed into D throughout its training phase. Thus, the distribution of generated samples is not concentrated, which alleviates the problem of strong bias.

3. CCPONECD Overview

3.1. Dataset Description

In this section, we introduce Chinese City Parking of New Energy Car Dataset (CCPONECD) - a diverse and carefully annotated new energy car dataset. CCPONECD includes 6909 real car images and each image contains only an identifiable NELP. CCPONECD gathers images from a city roadside parking management company in one provincial capital in China where new energy vehicles owners surpass 10k. The company hires over 800 parking fee collectors (PFCs) and each of them collects the parking fee on a specific street. Each parking toll order not only records LP number, expense, parking time and so on, but also requires PFC to take a picture of the car from the front or the tail as a proof. We select the new energy car pictures to constitute our CCPONECD. Each picture contains manually labelled NELP bounding box, NELP number, horizontal tilt degree, and some other information.

In order to avoid leaking out residents' privacy, CCPONECD removes records other than the LP number and selects images from scattered days and in a different place. In addition, all image metadata including device and GPS information, etc. Use the "Body text" style for all paragraphs. The following is an example of the "Bullet" style, which you may want to use for lists.

3.2. Data Preprocessing

The resolution of each image is 720 (Width) \times 1160 (Height) \times 3 (Channels). In practice, We only need smaller image which include the NELP part and some of the new energy vehicle edges that can prove the NELP lying on the body, so we cut the original picture to 300 \times 300 \times 3, the schematic proportion is shown in Figure 1 (a). The processed image with a size of 300 \times 300 \times 3 is used in our experiments, and the result proves better in some performance.

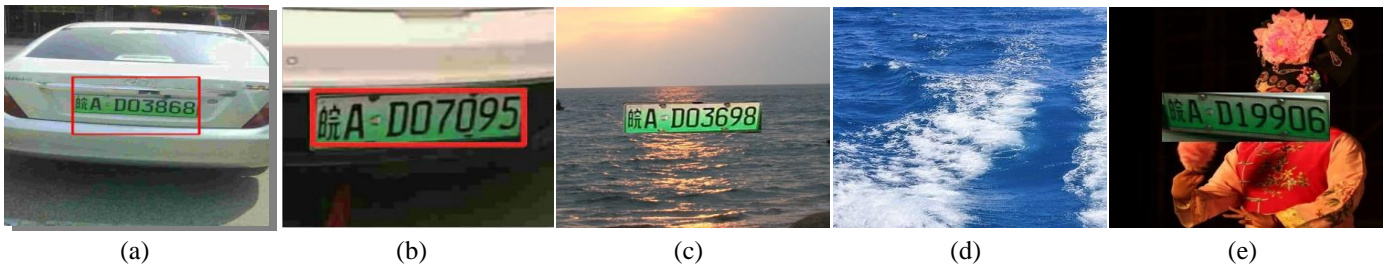


Fig. 1: Examples of sample image.

We cut the image based on the annotated vertex coordinates of the license plate provided in CCPONECD. First calculate the width and height of the license plate according to bounding box, and then separately compute the gap between the width, height and 300, evenly distributed to both sides. The bounding box of license plate area in 300 \times 300 image is shown in Figure 1 (b).

We made training and testing datasets that labelled with real or fake. Then 4195 real NELP images are used for training, but the testing images include real and fake NELP images. Testing images includes various of composite fake images of different scenes images as the background with NELP objects. We collected images by querying the image search engines of Baidu for the background of the fake NELP images.

we take the 300 \times 300 area in the middle of the picture as the background of the fake image. Then we cut out the real NELP image of distinct sizes and cover them with the same size in the middle of the background image. Consequently, artificial manufactured images involve NELP but without any other information concerning cars. Other fake data is produced by sampling images with the same size from some landscape pictures. Fake NELP images sample is shown in Figure 1 (c) (d)(e), respectively. For convenience of description, henceforth CCPONECD indicates the dataset we obtained based on the original dataset.

4. Methods

In this section, we introduce our proposed adversarial learning framework, called FLPD-GANs, and discuss the associated training methodology.

4.1. Fake License Plate Discrimination GANs

Our FLPD-GANs extended from LSGANs are composed of a generator and a discriminator. The model architecture we have utilized is shown in Figure 2. The generator G is a neural network using convolutional transpose layers, ReLU activation and batch-norm together with a tanh layer at the end. G progressively upscales the latent variables z to generate the image samples x' , represents as $x' = G(z)$, where z is a 50-dimensional random noise. The discriminator D consists of seven convolutional layers with ReLU activation and batch normalization and one fully connected layer. Resized input RGB images to the size of $256 \times 256 \times 3$, in a single forward computation, D downscales images by compressing it to a batch $\times 1024 \times 2 \times 2$ vector, then a feature vector with batch $\times 4096$ is obtained and fed it to subsequent fully connected classifier. Finally classifier outputs predicted score with the size of batch $\times 1$.

The inherent essence of FLPD-GANs is to utilize adversarial network to learn the feature of fake LP images, and to use supervised learning to understand the distribution of real LP images. The final discriminator model can identify real and fake LP images simultaneously.

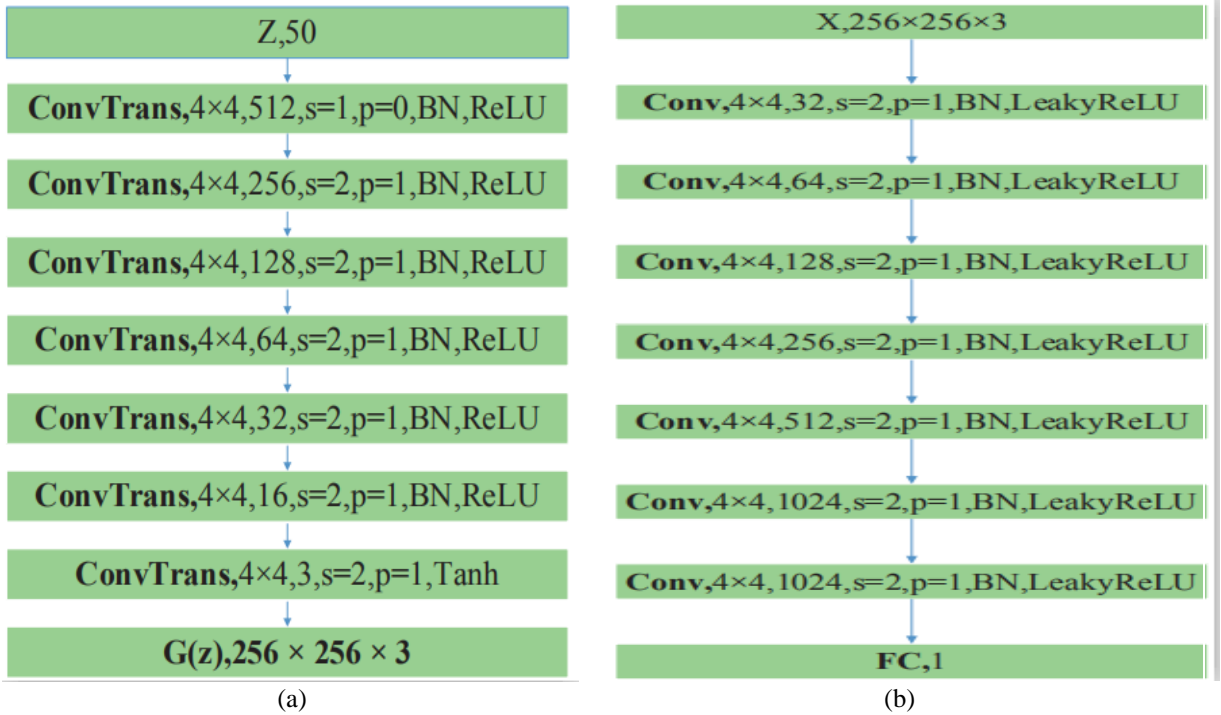


Fig. 2: Model architecture. “ConvTrans/Conv, $K \times K$, C, $s=S$, $p=P$, BN, ReLU/Tanh” denotes a deconvolutional/convolutional layer with $K \times K$ kernel, C output filters and stride = S, padding = P. The layer with BN means that the layer is followed by a batch normalization layer and FC, N denotes a fully connected layer with N output nodes. The activation layers are denoting as ReLU or Tanh. (a): The generator. (b): The discriminator.

4.2. Loss Function

The discriminator is considered as a two-class classifier. The common GANs employ a sigmoid cross-entropy loss function which can easily lead to problems like the disappearance of the gradient. LSGANs have a good performance for solving these problems. Through experiment, we found that using the arithmetic square root function between labels and predicted results achieve better effect. It has an average 1.8% boost. Thus, we use the arithmetic square root (ASR) function as the objective loss function of the discriminator. We assume R and F represent the labels of real data and fake data, respectively. Then minimizing the objective function of FLPD-GANs can be defined as follows:

$$\min \mathbf{L}_D = E_{x \in P_{data}(x)} \left[\sqrt{(D(x) - R)^2} \right] + E_{z \in P_z(z)} \left[\sqrt{(D(G(z)) - F)^2} \right] \quad (1)$$

$$\min \mathbf{L}_G = E_{z \in P_z(z)} \left[\sqrt{(D(G(z)) - F)^2} \right] \quad (2)$$

5. Experiments

5.1. Experiment Environment and Setting

In this section, we first present the details of the experiment environment, setting and evaluation methods. Next, we assess the performance of FLPD-GANs by presenting the results on CCPONECD in several aspects. Then we compare our approach with some common classification methods and the state-of-the-art GZSL and ZSL approaches.

All our experiments are accomplished on a GPU server. Detailed experimental environment is as follows:

CPU: Intel(R) Xeon(R) Silver 4108 CPU @ 1.80GHz; Memory: 125 GB RAM; GPU: Nvidia Tesla P100 with 12GB of memory. Our code was implemented under the open source Pytorch framework with CUDA version 9.1.85 accompanying cuDNN.

During training, we initialize the number of training epoch to 800 and the batch size is 32. Both G and D were training on the setting of the learning rate of 0.0002, momentums factor β_1 of 0.5, β_2 of 0.999, using the Adaptive Moment Estimation (Adam) optimization function as an optimizer and performing a single optimization step. We use the training and testing data both given in CCPONECD. In the FLPDGANs training dataset, there are only real NELP images with 4195 samples, while the rest of the samples are used to make the testing dataset.

5.2. Evaluation Methods

In the test stage, the images put into D and output the authenticity degree of these images. We use the authenticity score S_a to describe this property. T is denoting test dataset of NELP images. Hence, for a test sample x , S_a is defined as follows:

$$S_a = D_{x \in T}(x) \quad (3)$$

First, S_a is used directly to compare with a threshold for judging real or fake. After evaluating all images, we obtain the test accuracy regarding as acc1. To reduce the effect of test method on the recognition results, we also use other two methods to evaluate the accuracy of the experimental results. After getting the S_a of entire test image set T , we get a score set S , and then the maximum and minimum values of the test score are obtained which can be denoted as $\max(S)$ and $\min(S)$. According to these two parameters, all predicted scores are normalized to the range of $[a, b]$ according to the following formula:

$$K = \frac{(b - a)}{\max(S) - \min(S)} \quad (4)$$

$$N_s = a + k * (S_a - \min(S)) \quad (5)$$

where k represents the normalization coefficient and N_s represents the normalized score. In practice, we set $a=0$, $b=1$. In addition, a threshold is used to compare with N_s as a basis for judging real or fake. After evaluations of all images, we can get another test accuracy considered as acc2. The other way utilizes the results based on the area under the precision and recall curve (AUC) to determine model performance. The acc1 is suitable for any situation, but the acc2 is only suitable for predictive tasks when the number of image samples is large. AUC is used for classification tasks which contain positive and negative image samples. Both acc1 and acc2 are average per-class top-1 accuracy in GZSL task.

In our experiment, we set the evaluation threshold to 0.5, which we set according to the ratio of the real and the fake NELP images. In order to seek out the best trained model and preventing the undulation of adversarial training, our approach tests after each training session. In this process, we use acc1 as the test standard and show acc2 and AUC as the evaluation parameters of the final testing results.

5.3. Results on CCPONECD

We tried different input image size and latent variables size to find out the best import setting. The results obtained are shown in Table 1. For different image sizes, we do not alter the number of layers in the network

and only adjust the step size s and the size k of the convolution kernel to keep the size of the discriminator output unchanged.

From Table 1, we can find that models trained in 50 latent variables sizes can reach maximum acc2 and AUC in all NELP size. Experimental results also indicate that all the measures become better with a bigger size of input image. Because the smaller NELP image contains the less NELP information. Our FLPD-GANs can achieve an accuracy greater than 0.88 with a small image size 64. Although a bigger size of image yields better function, it is also associated with greater computational costs.

Table 1: Accuracy of different hyperparameter on CCPONECD

Measures		Acc2			AUC		
Dim		25	50	100	25	50	100
Size	64	0.883	0.909	0.905	0.931	0.952	0.892
	128	0.916	0.932	0.923	0.940	0.966	0.954
	256	0.940	0.944	0.937	0.966	0.977	0.965

We resize input images to 256×256 and set latent variables size to 50. Then we draw a scatter plot of the model predicted score S_a , as shown in Figure 3. The size of our total test dataset is 3000. The first 1500 instances represent fake NELP images (unseen class) and the latter are real NELP images (seen class). From 1 to 1000 are fake images with seascape and people as backgrounds. It can be observed that most of their authenticity scores are below the threshold, indicating that they can be well recognized. 1001 to 1500 are the background images without NELP information. In addition, we can find that they are all below the threshold. This result shows that our model can distinguish fake images from real NELP images. The acc1 of real NELP test images nearly reaches 95% and the total test images acc1 attains 94%.

From Figure 3, we can simply find that although do not provide any fake images for FLPD-GANs training, our FLPD-GANs model can also discriminate fake images well while ensuring a certain degree of recognition rate for real images.

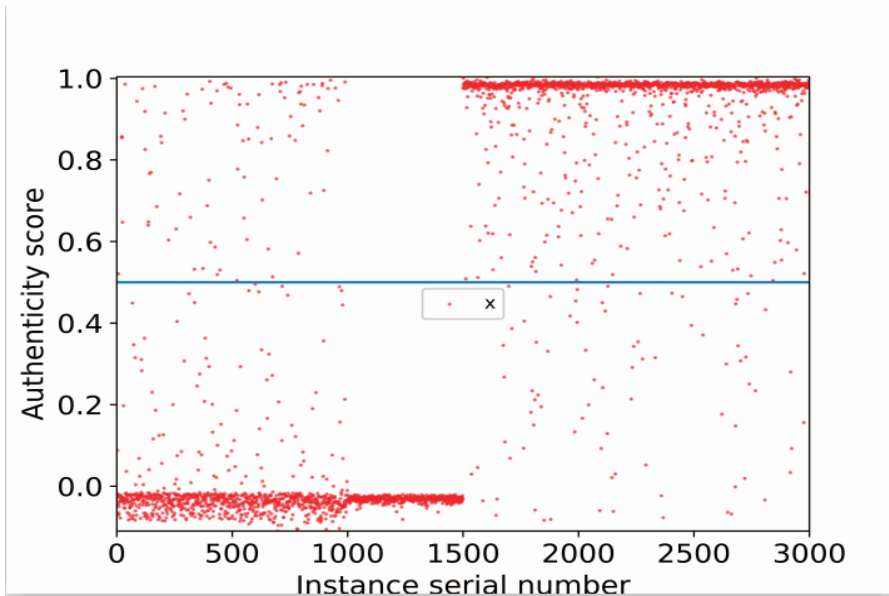


Fig. 3: Results on CCPONECD.

5.4. Different Classifiers Results on CCPONECD

We compare our model with a series of classifier models. For fair comparisons, the maximum number of training times is consistent and we also use the method that tests after each training epoch to seek the maximum AUC. We find no fake NELP image will make conventional two-class classifiers appear strong bias problem. The accuracy rate is mostly around 0.5, which indicates that they can only remember the training images, but have no ability to recognize fake images. we use both real and fake NELP images to

train traditional classifiers and compare them with FLPD-GANs trained only by real NELP images. Their training data include 2500 real NELP images and 1500 fake NELP images, which is much more diverse than FLPD-GANs’ training data. Except for adding a sigmoid function at the end of the fully connected layer, we apply the same network structure as the FLPD-GANs discriminator in the DCNN. Besides, DCNN utilizes the BCELoss and the Adam optimizer as well. In experiments we utilize images generated by the training process of FLPD-GANs as fake images for training another GZSL method named Relation Net [18]. This method integrating GANs into Relation Net is another FLPD-GANs, for which turns Relation Net into D. For the sake of convenience, we call this method Relation Net-GANs.

As shown in Table 2, our FLPD-GANs has a great AUC of 97.7%, which is over 89.8% of KNN and 90.3% of SVM. FLPD-GANs’ AUC is also close to DCNN. Furthermore, our methods are also better than the state-of-the-art GZSL and ZSL approaches on CCPONECD. Most ZSL and GZSL methods use feature vector and attribute vector for training. We utilize DCNN model trained previously to extract visual feature vector of images and each sample only has one binary label. For attribute vector, we use randomly generated standard normal distributions to substitute. Through experiments, we find that DeepEm [15], GAZSL [21] can achieve relatively high accuracy of 94.4%, 93.1%, respectively. But it is a little less AUC than our FLPD-GANs. Experiment result also proves that the AUC of Relation Net-GANs is as high as 99.9%, which is greatly higher than Relation Net and a little higher than FLPD-GANs. However, training Relation Net-GANs needs realistic fake images. Relation Net-GANs are cumbersome in practical applications. We mention here just to demonstrate that the effect of integrating GANs into GZSL tasks will be great.

Table 2: Performance of different classifiers on CCPONECD

model	AUC	model	AUC
SVM	0.898	Relation [18]	0.500
KNN	0.903	GAZSL [21]	0.931
DCNN	0.985	Relation Net-GANS	0.999
DeepEm [15]	0.944	FLPD-GANS	0.977

6. Conclusions

In this paper, we proposed FLPD-GANs for LP image authenticity discrimination. FLPD-GANs can discriminate between real and fake NELP images, regard less of the fact that only real NELP images were used for training. We also contributed a large-scale image dataset named CCPONECD, the first publicly available dataset for NELP detection and recognition to date. Experimental results on CCPONECD demonstrated a comparable accuracy of our FLPD-GANs over other traditional binary classifiers trained by real and fake NELP images. We also showed that FLPD-GANs outperformed state-of-the-art GZSL and ZSL approaches by experiments on CCPONECD. In the future, we will improve the accuracy of binary classification task. Additionally, it is possible to extend the proposed method to other multi-category tasks.

7. References

- [1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in neural information processing systems, 2012, pp. 1097 – 1105.
- [2] Arghya Pal and Vineeth N Balasubramanian, “Adversarial data programming: Using gans to relax the bottleneck of curated labeled data,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1556 – 1565.
- [3] Vinay Kumar Verma, Gundeep Arora, Ashish Mishra, and Piyush Rai, “Generalized zero-shot learning via synthesized examples,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, vol. 1, p. 3.
- [4] Jie Song, Chengchao Shen, Yezhou Yang, Yang Liu, and Mingli Song, “Transductive unbiased embedding for zero-shot learning,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1024 – 1033.

- [5] 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*, 2014, pp. 2672 – 2680.
- [6] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley, “Least squares generative adversarial networks,” in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2813 – 2821.
- [7] Alec Radford, Luke Metz, and Soumith Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *arXiv preprint arXiv:1511.06434*, 2015.
- [8] Martin Arjovsky, Soumith Chintala, and Leon Bottou, “Wasserstein gan,” *arXiv preprint arXiv:1701.07875*, 2017.
- [9] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville, “Improved training of wasserstein gans,” in *Advances in Neural Information Processing Systems*, 2017, pp. 5767 – 5777.
- [10] Yujun Shen, Ping Luo, Junjie Yan, Xiaogang Wang, and Xiaoou Tang, “Faceid-gan: Learning a symmetry threeplyer gan for identity-preserving face synthesis,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 821 – 830.
- [11] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim, “Learning to discover cross domain relations with generative adversarial networks,” in *International Conference on Machine Learning*, 2017, pp. 1857 – 1865.
- [12] 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*, 2017.
- [13] Shuang Ma, Jianlong Fu, Chang Wen Chen, and Tao Mei, ‘Da-gan: Instance-level image translation by deep attention generative adversarial networks,’ in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5657 – 5666.
- [14] Franziska Mueller, Florian Bernard, Oleksandr Sotny-chenko, Dushyant Mehta, Srinath Sridhar, Dan Casas and Christian Theobalt, “Ganenerated hands for real-time 3d hand tracking from monocular rgb,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 49 – 59.
- [15] Li Zhang, Tao Xiang, and Shaogang Gong, “Learning a deep embedding model for zero-shot learning,” in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 3010 – 3019.
- [16] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling, “Attribute-based classification for zero-shot visual object categorization,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 453 – 465, 2014.
- [17] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata, “Feature generating networks for zero-shot learning,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.
- [18] 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 *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA*, 2018.
- [19] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata, “Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly,” *IEEE trans actions on pattern analysis and machine intelligence*, 2018.
- [20] Rafael Felix, BG Vijay Kumar, Ian Reid, and Gustavo Carneiro, “Multi-modal cycle-consistent generalized zero-shot learning,” in *European Conference on Computer Vision*. Springer, 2018, pp. 21 – 37.
- [21] Yizhe Zhu, Mohamed Elhoseiny, Bingchen Liu, Xi Peng, and Ahmed Elgammal, “A generative adversarial approach for zero-shot learning from noisy texts,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.