PAGANDA: An Adaptive Task-Independent Automatic Data Augmentation

Boli Fang [*1] **Miao Jiang** [*1] **Jerry Shen** [2]

Abstract

Data augmentation is a useful technique to enlarge the size of the training set and prevent overfitting for different machine learning tasks when training data is scarce. However, current data augmentation techniques rely heavily on human design and domain knowledge, and existing automated approaches are yet to fully exploit the latent features in the training dataset. In this paper we propose *Parallel Adaptive GAN Data Augmentation*(PAGANDA), where the training set adaptively enriches itself with sample images automatically constructed from Generative Adversarial Networks (GANs) trained in parallel. We demonstrate by experiments that our data augmentation strategy, with little model-specific considerations, can be easily adapted to cross-domain deep learning/machine learning tasks such as image classification and image inpainting, while significantly improving model performance in both tasks. Our source code and experimental details are available at `https://github.com/miaojiang1987/k-folder-data-augmentation-gan/`.

1. Introduction

Deep learning and machine learning models produce highly successful results when given sufficient training data. However, when training data is scarce, overfitting will occur and the resulting model will generalize poorly. Data augmentation(DA) ameliorates such issues by enlarging the original data set and making more effective use of the information in existing data. Much prior work has centered on data augmentation strategies based on human design, including heuristic data augmentation strategies such as crop, mirror, rotation and distortion (Krizhevsky et al., 2012; Simard et al., 2003), interpolating through labeled data points in feature spaces (DeVries & Taylor, 2017), and adversarial data augmentation strategies based on (Teo et al., 2008; Fawzi et al., 2016). These methods have greatly aided many deep learning tasks across several domains such as classification (Krizhevsky et al., 2012), image segmentation (Yang et al., 2017) and image reconstruction/inpainting (Alvarez-Gila et al., 2017).

Despite their success, these DA methods generally require domain-specific expert knowledge, manual operations and extensive amount of tuning depending on actual contexts (Ciresan et al.; Dosovitskiy et al., 2016). In particular, the need to directly operate on existing data with domain knowledge prevents many previous data augmentation strategies from being applicable to more general settings. To circumvent the need for specific domain knowledge in data augmentation, more recent work (Antoniou et al., 2017) utilizes generative adversarial networks(GANs) (Goodfellow et al., 2014) to produce images that better encode features in the latent space of training data. By alternatively optimizing the generator G and the discriminator D in the GAN, the GAN is able to produce images similar to the original data and effectively complement the training set. It has been shown in experiments (Antoniou et al., 2017) that GAN-based methods have indeed significantly boosted the performance of classifiers under limited data through automatic augmentation, but applications into other tasks are yet to be explored. Furthermore, given the computational complexity of GANs, a natural way to reduce runtime is to consider parallelism (Intrator et al., 2018; Durugkar et al., 2016).

In view of these considerations, we propose in this paper *Parallel Adaptive Generative Adversarial Network Data Augmentation*(PAGANDA), where the training set adaptively enriches itself with sample images automatically constructed from Generative Adversarial Networks (GANs) trained in parallel. Our contributions can be summarized as follows:

- We propose a general adaptive black-box data augmentation strategy to diversify enhance training data, with no task-specific requirements.
- We also include in our model a novel K-fold parallel framework, which helps make the most use of the

[*]Equal contribution [1]School of Informatics Computing and Enginnering, Indiana University, Bloomington,Indiana, USA [2]Sol Price School of Public Policy, University of Southern California, Los Angelos, California, USA. Correspondence to: Boli Fang <bfang@iu.edu>, Miao Jiang <miajiang@iu.edu>.

Proceedings of the 1st Adaptive & Multitask Learning Workshop, Long Beach, California, 2019. Copyright 2019 by the author(s).

existing data.

- Experiments over various datasets and tasks demonstrate the effectiveness of our method in different context.

2. Related Work

Data Augmentation(DA) Previous work on data augmentation can be classified into several groups. Traditional Heuristic DA strategies such as crop, mirror, rotation and distortion (Krizhevsky et al., 2012; Simard et al., 2003) have found their way in many deep classification tasks, but these method generally require domain-specific expert knowledge, manual operations and extensive amount of tuning depending on actual contexts (Ciresan et al.; Dosovitskiy et al., 2016). Other DA methods used interpolation through labeled data points in feature spaces (DeVries & Taylor, 2017), but their dependence on class labels makes them inapplicable for tasks with weak or no supervision. Adversarial Data Augmentation strategies (Teo et al., 2008; Fawzi et al., 2016) choose from a select number of transformation operations to maximize the loss function of the end classification model involved in the task. While good motivations for our methods, these methods make strong assumptions over the types of augmentation and are difficult to generalize. (Ratner et al., 2017; Cubuk et al., 2018) transform the problem of choosing data augmentation strategies into a reinforcement learning policy search problems, but the choice of augmentation methods are still limited and the reinforcement learning algorithms have non-trivial computation overhead in addition to the main task.

ML problems with limited data For classfication with limited samples, (Salamon & Bello, 2017) proposed a convolutional neural network(CNN) to classify environmental sounds with limited samples. Other algorithms have been proposed in (Frid-Adar et al., 2018; Zhu et al., 2017), yet many of them have assumptions/constraints that hurts their capacity for generalization. For unsupervised learning models, recent research on sample complexity reduction in GAN training seeks to reparametrize the input noise using variational inference (Gurumurthy et al., 2017; Nowozin et al., 2016), but this method has severe mathematical limitation that prevents further generalization. (Wang et al., 2018) adopts transfer learning techniques to train a new GAN for limited data from a pre-trained GAN network. While effective, this approach requires a pre-trained network in the first place and doesn't apply to the cases when data is scarce.

Parallel/Distributed GANs (Intrator et al., 2018; Durugkar et al., 2016) proposed the first distributed multi-discriminator generative adversarial models, yet these models require large datasets to train and have great computational complexity. Moreover, these models are trained on fixed given datasets, meaning they are still susceptible to the inherent biases of the training data.

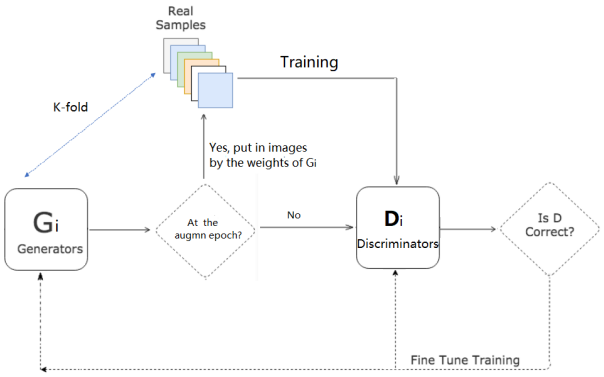


Figure 1. Generative Data Augmentation

3. Proposed method

In this section we describe the details of Parallel Adaptive Generative Adversarial Network Data Augmentation (PAGANDA). Our method consists of three interrelated components: generative data augmentation, parallel image generation with fold division, and adaptive weight adjustment.

3.1. Generative Data Augmentation

To ensure that make full use of the information contained in the existing images, the first part of our method involves *generative data augmentation*, which constructs varied images given the training set by repeatedly generating samples from and adding samples to the training set using a generative adversarial net.

We start off with a limited training set, and consecutively run the generative adversarial net using the set. After running a fixed number t of regular training epochs, we proceed to the augmentation epoch where the augmentation is conducted. During the augmentation epoch, we extract a number of sample images from the generator G using standard procedures of sample image generation as described in (Radford et al., 2015; Gulrajani et al., 2017). For this batch of samples, we calculate the Inception Score(IS) as defined by (Salimans et al., 2016) to measure the authenticity of the images generated, which we denote as w. Here the Inception Score provides a metric of the power of generator to produce realistic images: the higher the value of w, the more power the corresponding generator G. This batch of images are then added back into the original training set for subsequent augmentation epochs. We alternate running t regular training epochs and the augmentation epoch for a

fixed number of times or until convergence. Figure 1 is a flow-chart of our procedure.

Notice that our procedure is agnostic to the specific architecture of generative adversarial net used to augment the training data. Since GANs capture the information in the latent feature space of the images and translate such information into generated images, our method has the capacity to reveal the potential features that are possibly not visually evident in the original training images. Moreover, compared with many other data augmentation strategies which require one to pre-define the operations to be carried on the images, our method automatically enriches the training set and does not require human intervention.

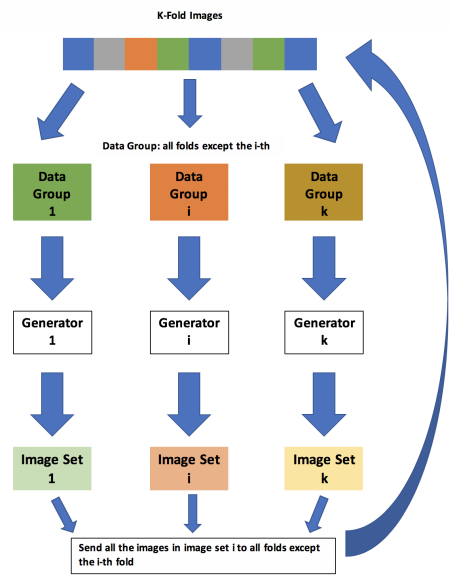


Figure 2. Parallel Image Generation

3.2. Parallel Image Generation with Fold Division

The second part of our method consists of a parallel data generation strategy, inspired by K-fold cross validation in machine learning (Bishop, 2006). Dividing the training data into K folds at the beginning, we run in parallel K independent generators $\{G_i\}_{i=1}^K$. Each generator G_i is trained on one of data groups, and each data group i consists of $K-1$ folds of the training set, except for the i-th fold. After images are generated in each generator G_i in the training epochs, the sample images produced by each generator during the augmentation epoch are fed back into the respective training data groups. To allow for maximal usage of each generated image, we insert the images in a way such that the images generated by one generator G_i are sent to the training data groups corresponding to all other $K-1$ generators except for that corresponding to G_i. This is to insure that the different generators in parallel have access to as many varied data pieces as possible in subsequent steps of training, so as to prevent overfitting and bolster the robustness of our strategy. Figure 2 demonstrates our algorithm.

3.3. Adaptive Generator Weighting

Furthermore, to determine which generators are the most effective in generating authentic images, we introduce *adaptive generator weighting* at each augmentation epoch. At the initial stage, all the generators are treated equally. Before the batch of sample images generated by one generator G_i are sent to the data group corresponding to other $K-1$ generators, we collect the inception scores $\{w_i\}_{i=1}^K$ computed in section 3.1. Since higher inception scores imply better performance of the generator, we define *the generator weight* p_i of a generator G_i as

$$p_i = \frac{w_i}{\sum_{j=1}^K w_j},$$

and use this weight to determine how many images should be sampled from generator G_i to be sent to other data groups for subsequent training in the very next augmentation epoch. When the total number of samples to be collected from generators are fixed, this method enables generators with better realistic image generation power to contribute more to the future training data groups. More realistic training sets thus augmented, in turn, exert more positive influence on the images to be generated.

Note that all three strategies introduced go hand in hand, with no need for model specific considerations. As demonstrated by our experiments Section 4, training different GANs in parallel from different folds of data substantially boosts the quality of the training set and that of the generated images.

4. Experiments

4.1. Multitask Experimental Settings

To illustrate the effectiveness of PAGANDA for multiple machine learning tasks, we have applied our data augmentation method to two tasks: image classification and image inpainting. For image classification we constructed our dataset from Imagenet and Cifar-10 by randomly drawing 5000 images from each dataset respectively and applied PAGANDA on these reduced datasets. The augmented datasets are then used to train an AlexNet CNN classifier, and the classification results are compared with the results obtained from an AlexNet trained on the corresponding original unaugmented datasets.

For image inpainting, we constructed our datasets from Places dataset. We chose images from the Ocean subset from Places to obtain the reduced Places Dataset.

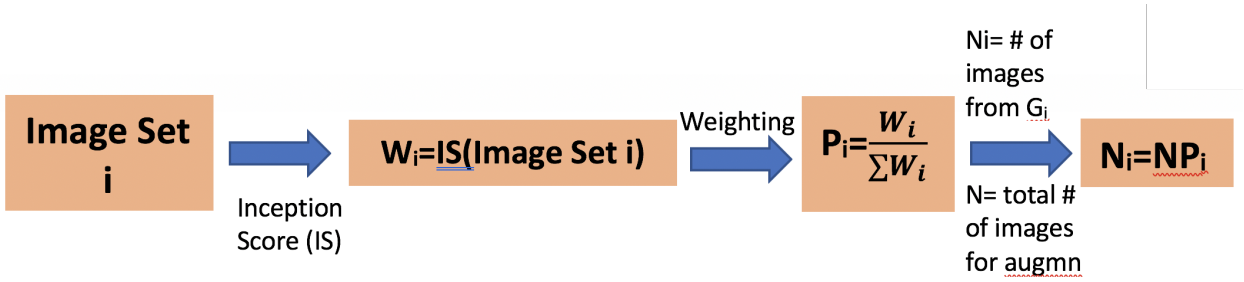
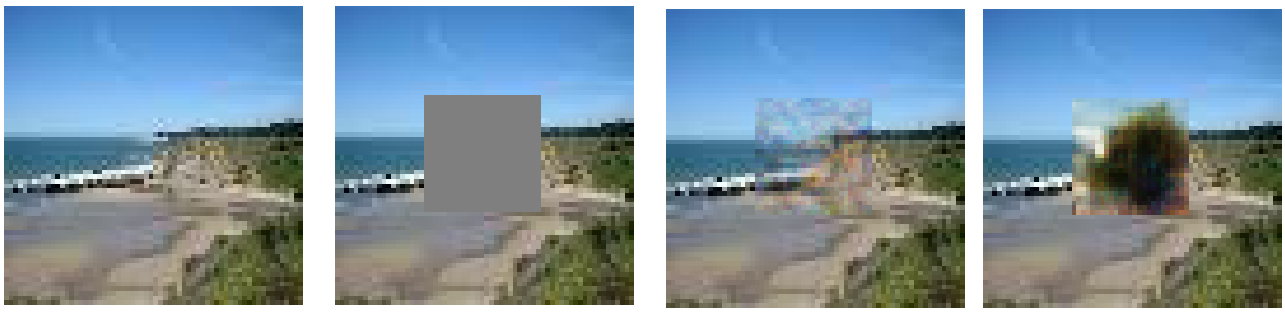


Figure 3. Adaptive Generator Weighting



original image | masked image | with PAGANDA | w/o PAGANDA

Figure 4. Inpainting results on reduced datasets

To ensure the parallelism of the experiments, we trained our model in a multi-threaded environment to make simultaneously training. Under such a setting, all the data groups are trained at the same time, and each GAN model corresponding to each data group is trained in a separate thread. All of our experiments are conducted on a server with Tesla-V GPU (32GB RAM, 7.8 TeraFLOPS) and Intel Xeon Processor E5 (2.00 GHz).

4.2. Evaluation on Classification

For our experiments on classification, we first augment the reduced Cifar-10 and reduced Imagenet datasets, and then train the CNN classifier with the augmented dataset. The classifier accuracies with and without augmentation are listed in Table 1 below.

Table 1. Image Classification Accuracy, with/without Augmentation

Datasets	w/o PAGANDA	with PAGANDA
Cifar-10	74.7	**81.3**
Imagenet	82.7	**88.1**

4.3. Evaluation on Inpainting

For the task of inpainting, we augment the reduced dataset constructed in the experiment. Without loss of generality, we train a WGAN-GP model for inpainting from the augmented dataset. We then select testing images that are not selected in the training set, and add to them gray masks covering the center part of these images. We then applied our trained WGAN-GP to generate patches that cover the masked portion of the inpainting image.

Figure 4 lists a couple of generated images with and without augmentation. Visual comparisons demonstrate the effectiveness of our method.

5. Conclusion and Future Work

In sum, our paper shows that PAGANDA effectively improves the performances for different machine learning tasks with little task-specific considerations. Our strategy is not only simple to implement, but also demonstrates capability to generate onto different settings since it does not require specific information about the task being analyzed.

As a further step, we are investigating the relationship between our proposed approach and other established methods. We hope to apply our idea to other generative models such as VAE (Kingma & Welling, 2014) and further optimize our strategy using recent theoretical advances, and wish to investigate the scenarios where the tasks involved are interrelated. Application wise, we are aiming to apply our parallel GAN model to multi-modal image synthesis/generation where training data is limited.

Acknowledgements

We thank the Digital Science Center of Indiana University for supporting our research with GPUs.

References

Alvarez-Gila, A., Van De Weijer, J., and Garrote, E. Adversarial networks for spatial context-aware spectral image reconstruction from rgb. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 480–490, 2017.

Antoniou, A., Storkey, A., and Edwards, H. Data augmentation generative adversarial networks. *arXiv preprint arXiv:1711.04340*, 2017.

Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer, 2006.

Ciresan, D., Meier, U., Gambardella, L., and Schmidhuber, J. Deep big simple neural nets excel on handwritten digit recognition. corr abs/1003.0358 (2010).

Cubuk, E. D., Zoph, B., Mané, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation policies from data. *CoRR*, abs/1805.09501, 2018.

DeVries, T. and Taylor, G. W. Dataset augmentation in feature space. *arXiv preprint arXiv:1702.05538*, 2017.

Dosovitskiy, A., Fischer, P., Springenberg, J. T., Riedmiller, M., and Brox, T. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(9):1734–1747, 2016.

Durugkar, I. P., Gemp, I., and Mahadevan, S. Generative multi-adversarial networks. *ICLR*, 2016.

Fawzi, A., Samulowitz, H., Turaga, D., and Frossard, P. Adaptive data augmentation for image classification. In *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3688–3692. Ieee, 2016.

Frid-Adar, M., Klang, E., Amitai, M., Goldberger, J., and Greenspan, H. Synthetic data augmentation using gan for improved liver lesion classification. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 289–293. IEEE, 2018.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems 30*. 2017.

Gurumurthy, S., Kiran Sarvadevabhatla, R., and Venkatesh Babu, R. Deligan : Generative adversarial networks for diverse and limited data. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Intrator, Y., Katz, G., and Shabtai, A. MDGAN: boosting anomaly detection using multi-discriminator generative adversarial networks. abs/1810.05221, 2018.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *ICLR*, volume abs/1312.6114, 2014.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*. 2012.

Nowozin, S., Cseke, B., and Tomioka, R. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems 29*. 2016.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015.

Ratner, A. J., Ehrenberg, H., Hussain, Z., Dunnmon, J., and Ré, C. Learning to compose domain-specific transformations for data augmentation. In *Advances in neural information processing systems*, pp. 3236–3246, 2017.

Salamon, J. and Bello, J. P. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24(3): 279–283, 2017.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X., and Chen, X. Improved techniques for training gans. In *Advances in Neural Information Processing Systems 29*. 2016.

Simard, P. Y., Steinkraus, D., and Platt, J. C. Best practices for convolutional neural networks applied to visual document analysis. In *ICDAR*, 2003.

Teo, C. H., Globerson, A., Roweis, S. T., and Smola, A. J. Convex learning with invariances. In *Advances in neural information processing systems*, pp. 1489–1496, 2008.

Wang, Y., Wu, C., Herranz, L., van de Weijer, J., Gonzalez-Garcia, A., and Raducanu, B. Transferring gans: generating images from limited data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 218–234, 2018.

Yang, D., Xu, D., Zhou, S. K., Georgescu, B., Chen, M., Grbic, S., Metaxas, D., and Comaniciu, D. Automatic

liver segmentation using an adversarial image-to-image network. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 507–515. Springer, 2017.

Zhu, X., Liu, Y., Qin, Z., and Li, J. Data augmentation in emotion classification using generative adversarial networks. *arXiv preprint arXiv:1711.00648*, 2017.