BOOTSTRAPPING DIFFUSION MODELS: ITERATIVE SYNTHETIC DATA GENERATION FOR SELF-SUPERVISED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Assembling training data for deep generative models requires extensive effort, and practical datasets often suffer from imbalanced or biased data collection. Drawing inspiration from self-supervised learning, we introduce a novel bootstrapping approach for training generative models. Specifically, we construct synthetic datasets by combining generated samples from previous iterations with real data. Our empirical results demonstrate the potential for performance improvement through bootstrapping diffusion models. By recycling samples over successive generations, this technique reduces the dependence on large curated datasets while producing varied outputs. Our proposed bootstrapping regimen presents a promising direction for efficient and effective training of deep generative models.

1 INTRODUCTION

Generative modeling has become a pivotal area within computer vision research. These powerful models can synthesize highly realistic images, video, and other media that closely match the statistics of real-world data. As generative techniques continue to advance, they are unlocking new capabilities in content creation, image editing, and more. In particular GAN models (Goodfellow et al., 2014; Karras et al., 2018; Pan et al., 2023) can generate high-resolution photographic images with an extraordinary level of detail. Video synthesis models like MoCoGAN (Tulyakov et al., 2017) and TGANv2 (Saito et al., 2020) create flowing, believable video sequences that mimic complex spatial and temporal patterns. Diffusion models such as DDPM (Ho et al., 2016; 2020) and Latent Diffusion (Rombach et al., 2022) rapidly produce images rivaling GANs in quality. Transformers like DALL-E (Ramesh et al., 2021) and Imagen (Saharia et al., 2022) demonstrate remarkable proficiency at text-to-image generation.

Despite the success of deep generative models, they all suffer from one important limitation: gathering high-quality data for training generative model can be a tedious process. As we discovered during our last lab when building a dataset, collecting diverse images at sufficient resolution and appropriately sized for the model input requires considerable time and effort. This challenge of assembling robust training data also extends to real-world dataset construction. Carefully compiling a varied and representative dataset demands meticulous attention to detail (Russakovsky et al., 2015; Dittmer et al., 2023). And many datasets are often subjected to imbalanced or biased collection due to various constraints (Melhem et al., 2023; Utyamishev & Partin-Vaisband, 2019).

If we want to train the deep generative model in a sample-efficient manner without harming the performance, then one challenge is to ensure the training data contains enough diversity. This naturally gives rise to a class techniques called *Data Augmentation*. Traditional solutions include adding rotations (Gidaris et al., 2018), colorizations (Zhang et al., 2016) and distortions (Dosovitskiy et al., 2015) etc. These techniques have been widely integrated in the training process of modern deep generative models (Radford et al., 2016; Ho et al., 2016; 2020) to add diversity in the model training.

However, one limitation of these traditional data aggregation methods is their lack of generalizability. In theory, deep generative models assume training data is independently sampled from the real-world distribution. But regardless of rotations, colorization, or distortions applied, all aggregated data still originates from the same source images and remains highly correlated. As a result,

Algorithm 1 Bootstrap Diffusion

1: **Input:** Original dataset $\mathcal{D}^1$, bootstrap iterations $\mathcal{I}$.
2: **for** $i = 1$ in $\mathcal{I}$ **do**
3: $\quad \mathcal{P} \leftarrow$DIFFUSION($\mathcal{D}^i$).
4: $\quad \mathcal{D}^{i+1} \leftarrow \mathcal{D}^i \cup \mathcal{P}$.
5: **end for**

the underlying problems of imbalanced or inadequate training data persist. Though data augmentation expands the number of samples, it does not truly enhance the diversity or breadth of the training set. These standard aggregation techniques alone cannot solve the core issue of limited real-world data coverage.

Stepping back, we can view these traditional data augmentation techniques from a high-level perspective. The ultimate goal in training a generative model is to approximate the true real-world data distribution (Goodfellow et al., 2014; Kingma & Welling, 2022; Ho et al., 2020). This requires generalizing beyond the training data - which is precisely where augmentation struggles. For instance, when training an image diffusion model, we can assume the model has attained some generalization capability after initial training. We can then feed generated images back into the original dataset to create a synthetic dataset for further training. Critically, this new dataset should exhibit greater diversity, as its samples are drawn independently from the model's distinct distribution. The entire process forms a *bootstrapping/self-supervised* loop, as depicted in `BootstrapDiffusion`. By recycling samples over successive generations, we continually expand the breadth and variety of our training data. And inspired by this observation, we raise the following research question:

Can we train Generative models in a bootstrap manner as self-supervised learning?

Fortunately, our paper provides a promising answer to this question through empirical experiments. Finally, we summarize the main results of our paper: **1)** We provide a thorough explanation of the theory and model-training of the state-of-the-art (SOTA) diffusion model. **2)** We train both diffusion model and traditional DCGAN model from large dataset. We compare the result of model trained by these two prevailing models. **3)** We leverage synthetic data generated by the model training and train our Diffusion/GAN model in a bootstrap manner. We compare the results with the model trained with pure real-world data.

2 Related Work

Generative Adversarial Nets First proposed in 2014 (Goodfellow et al., 2014), the Generative Adversarial Nets (GAN) has emerged as a powerful framework for generating synthetic data that exhibits high fidelity and realism. An extension of GAN is Deep Convolutional Generative Adversarial Networks (DCGAN) (Radford et al., 2016), which applies deep convolutional neural networks (DCNN). It turns out that DCGAN is effective in various image synthesis tasks. Later video synthesis models like MoCoGAN (Tulyakov et al., 2017) and TGANv2 (Saito et al., 2020) are introduced to create flowing, believable video sequences that mimic complex spatial and temporal patterns. Recently, Pan et al. (2023) further enpowers interactive manipulation on the generative images.

Diffusion Models The concept of diffusion models was introduced by Ho et al. (2016) in their paper. The application was extended in another paper (Ho et al., 2020) from diffusion models to image generation, resulting in new models called "Denoising Diffusion Probabilistic Models" (DDPM). Rombach et al. (2022) runs the diffusion process in the latent space instead of pixel space, making training cost lower and inference speed faster. Nichol & Dhariwal (2021) enables generate samples conditioned on class labels or a piece of descriptive text. Xu et al. (2022) introduces a physical process called "Poisson flow" to substitute the original theomdynamic process and becomes the more prevailing diffusion model recent days.

Self-supervised Learning The idea of bootstrapping is related to self-supervised learning. The concept of self-supervised learning first discussed back in 1989 in Schmidhuber (1990), and this term is used to cover a related approach, which is using one modality as labels for another, as

in de Sa (1994) which uses audio data as labels, and video data as predictors. In general, self-supervised learning uses raw data to automatically generated labelled data, rather than relies on external labels provided by humans. In computer vision, successful approaches include colorization by Zhang et al. (2016); Larsson et al. (2017); Vondrick et al. (2018), placing image patches in the right place by Noroozi & Favaro (2017); Doersch et al. (2016), inpainting by Pathak et al. (2016), and so on.

Extending the deterministic algorithm to label data in self-supervised learning into machine learning algorithm with the goal of data augmentation remaining the same results in the idea of bootstrapping.

Synthetic Dataset Using Generative Models Data augmentation through generative AI has become widespread lately as deep generative models like Diffusion (Ho et al., 2020; Nichol & Dhariwal, 2021; Rombach et al., 2022; Xu et al., 2022) and GANs (Goodfellow et al., 2014; Radford et al., 2016) have advanced. Saragih & Tyrrell (2023); Bowles et al. (2018) applied augmented data generated by GAN to facilitate the training of image segmentation and observed improvement in performance. Voetman et al. (2023) leveraged a Diffusion model to synthesize training data that improved sample efficiency for detection tasks. Azizi et al. (2023); Nguyen et al. (2023) used text-to-image stable diffusion model to generate synthetic data for semantic segmentation. They managed to eliminate the need for labor-intensive pixel-wise annotation and obtained superior performance. Zhu et al. (2023) proposed multiple techniques with diffusion models that enable more sample efficient training in reinforcement learning settings. Unlike previous work, we focus more closely on the fundamental question of whether synthetic datasets can actually enhance the training process for generative AI itself.

3 Background

3.1 Synthetic data

Synthetic data is information that's artificially generated rather than produced by real-world events. Data is one of the fundamental elements in machine learning, and thus advances machine learning make in recent years are largely due to availability of big data. The problem is that finding enough data for machine learning algorithms in some domains or situations is difficult. Even though data are currently generated in huge amounts than before, in many cases data may fall short due to privacy, national security, or sarcity, which hinders our ability to benefit from using machine learning algorithms.

Traditional synthetic method includes stimulation by combining deterministic algorithm and random noises. In recent years, as the generative models become available and mature, using generative models to synthesize data becomes possible.

3.2 Denoising Diffusion Probabilistic Models

DDPMs (Ho et al., 2020), a category of generative models, define latent variables $x_1, \ldots, x_N$ in the same dimension as the data $x_0 \sim q(x_0)$. Diffusion models consist of both a forward diffusion process and a backward denoising process. The forward process corrupts the original data x_0 by iteratively adding random Gaussian noise, following a predetermined noise schedule for N diffusion steps, until the distribution of x_T is likely to resemble random Gaussian noise:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I) q(x_{1:N}|x_0) = \prod_{t=1}^{N} q(x_t|x_{t-1}) \tag{1}$$

where $\beta_i, i \in [N]$ is the random noise produced by a scheduler that converts the data distribution x_0 into latent x_N. The choise of the noise scheduler has been shown to have important effects an sampling efficiency and quality. In our experiment, we choose to use linear scheduler for the sake of image sizes. When increasing the image size, the optimal noise scheduling shifts towards a noisier one due to increased redundancy in pixels. Simply scaling the input data by a factor of b while keeping the noise schedule function fixed is a good strategy (equivalent to shifting the $\log \mathrm{SNR}$ by $\log b$).

Algorithm 1 Training

1: **repeat**
2: $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $t \sim \text{Uniform}(\{1, \dots, T\})$
4: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: Take gradient descent step on
$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
6: **until** converged

Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \dots, 1$ **do**
3: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

Figure 1: Algorithm for Diffusion model

Some previous work proposes the improved DDPM (Nichol & Dhariwal, 2021), and show that it can also achieve competitive logarithmic likelihood while maintaining high sample quality. Thus we adopt this model as we basic framework. In addition, using such an improved DDPM, the learning variance of the reverse diffusion process allows for sampling with orders of magnitude less, while the difference in sample quality is negligible, which is important for the practical deployment of these models. Furthermore, the sample mass and likelihood of these models scale smoothly with model capacity and training calculations, making them easy to scale.

3.3 RESULT

We trained Diffusion models on 22000 images from ffhq-dataset [Acknowledgements-https://github.com/NVlabs/ffhq-dataset] and got quite impressive results5.2

4 METHOD

Data synthesis using generative models has two types: bootstrapping and ability transfer. Bootstrapping uses generated images by the generative model to augment itself, while ability transfer uses one generative model to help another.

Ability transfer uses a well-trained and large-scale model to high-quality generate images to augment a rather small data set used by another small-scale model. This method is intuitively positive as the augmented data set will bring the model higher convergence rate and better performance. We uses a toy example "Diffusion for GAN" in 5.3

Bootstrapping may have positive or negative results, since augmentation of the data set from generated images of the same model results in improvement in data quantity and deterioration in data quality. Intuitively, this method is negative in case of small-scale models and small data set, since the quality of generated images is so worse than quality of real images, that decrease in quality dominates increase in quantity. We will use a toy example "GAN for GAN" in 5.1 to illustrate this intuition.

However, when using boostrapping in in well-trained and large-scale models, the result will need real experiments to validate since the quality of generated images is so good that the increase in quantity becomes worth trading-off with the decrease in quanlity. We implement a large-scale and well-trained diffusion model "Diffusion for Diffusion" with the most advanced arithecture and optimization skills to find the answer.

4.1 TOY EXAMPLE FOR BOOTSTRAPPING: GAN FOR GAN

We show a toy example for bootstrapping by applying this technique to a GAN model, where the original GAN is trained upon 100 real bottle images, while the augmented GAN is train upon 100 real bottle image plus 64 fake bottle images. Each of the models is trained for a 450-epoch period.

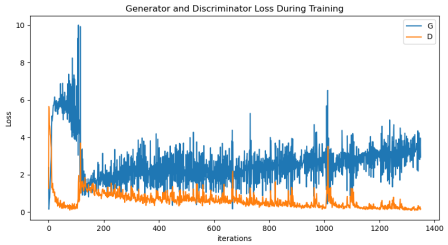

Figure 2: loss function of original GAN

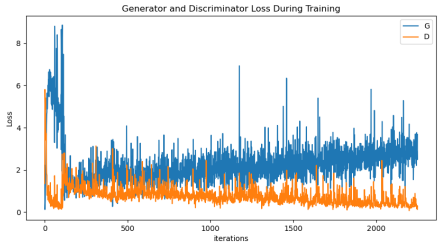

Figure 3: loss function of augmented GAN

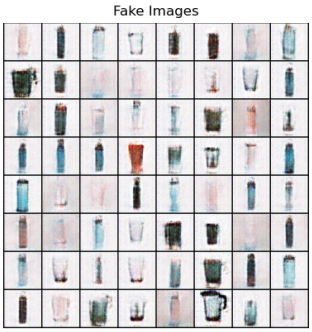

Figure 4: images by original GAN

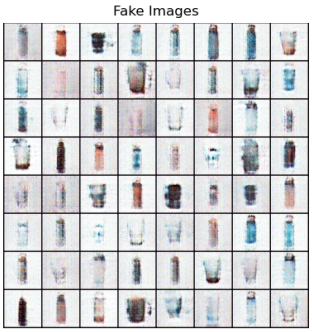

Figure 5: images by augmented GAN

As we can see in the local function, the model takes a faster leap out of the local optima, hence the time of convergence is reduced. This indicates that the bootstrapping technique has produced a positive impact on the efficiency of the model.

4.2 Real implementation of bootstrapping: Diffusion for Diffusion

As we all know, data is crucially important for training a model in the computer vision area. Therefore, we came up with the idea of using fake pictures to enlarge our dataset. However, assessing the quality of pictures generated by different models poses a challenge. One approach is to rely on humans to judge and rate the images based on various criteria, such as realism, diversity, and relevance. However, achieving fairness in human evaluations is difficult, and finding a sufficient number of people to rate our pictures is also a challenge.

Although many researchers have proposed methods for quantitatively and scientifically assessing the quality of generated images (Liu et al. (2017), Bianco et al. (2017)), to better align with our course, we decided to use a GAN model for evaluation.

Full details of experiments are showed in Section 6.

4.3 Toy example for ability transfer: Diffusion for GAN

Regarding the current imbalance of computing power, ability transfer from well-trained and large scale models to the less-trained and small scale models is of great value. One direct is to use generative power of well-trained models to strengthen the data set of individual trainers. In this way, we can exploit the knowledge of well-trained models while preserving its private parameters secure.

In some application scenarios, the data maybe highly sensitive and related to privacy, and thus hard to collect for individual model trainers, while some certain companies may be granted for access to

large data set and have high-quality well-trained generative models. The generated data from these well-trained generative models are normally irrelevant to personal information, and thus can be used as a good resource for individual model trainers.

We use 100 images generated by the well-trained diffusion model to augment the original data set which consists of 100 real human face images.

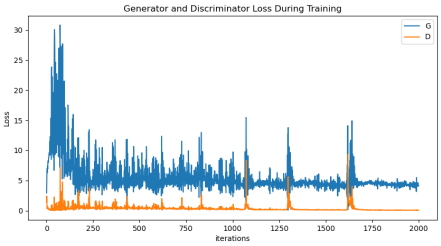

Figure 6: loss function of original GAN

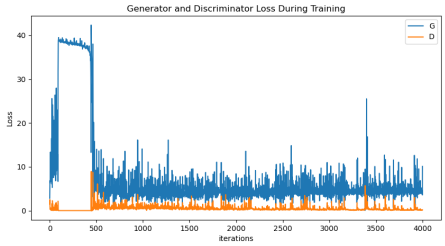

Figure 7: loss function of augmented GAN

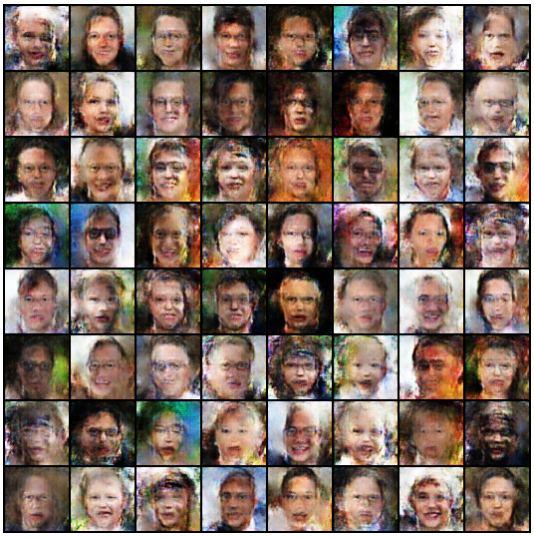

Figure 8: images by original GAN

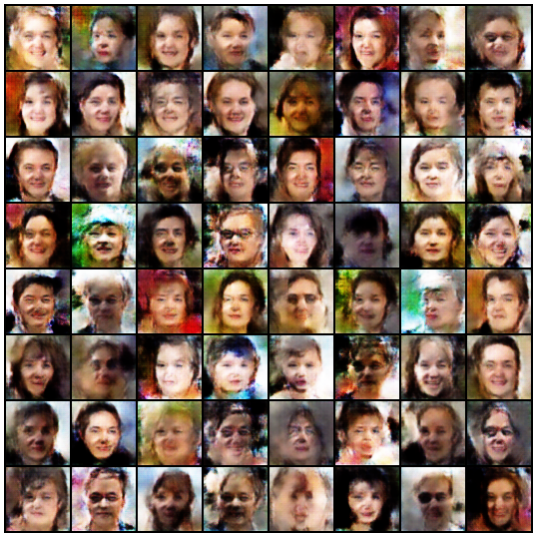

Figure 9: images by augmented GAN

The loss functions show that though the augmented model takes slightly longer time to converge, the loss of augmented generator remains lower and more stable than the original generator, which also corresponds to the sample generated images.

5 Experiments of Diffusion for Diffusion

5.1 Dataset

We use the open-source dataset ffhq-dataset [Acknowledgements-https://github.com/NVlabs/ffhq-dataset]. To be more precisely, we extract 22000 images with image size 128x128 to be our baseline dataset. After training the baseline diffusion, we sample 30% of the original dataset, i.e., 6600 images. Then we construct our new datasets as follows:

1. 70% of original dataset
2. 70% of original dataset + 30% of diffusion samples
3. Original dataset + 30% of diffusion samples

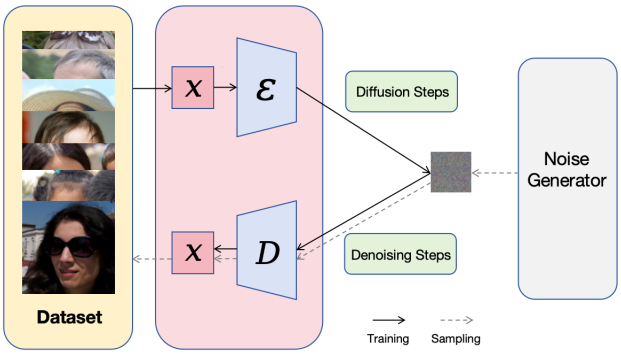

Figure 10: Framework.

5.2 Implementation Details

The code was tested on an RTX4090ti 24GB, with 1000 diffusion steps, 64 channels, 3 residual blocks, linear noise scheduler, and 1e-4 learning rate. To get a better convergence, we trained 210000 steps for each checkpoint respectively. While the GAN model was trained on 100% of the original dataset with 100 epochs. The results of the training results on different datasets are shown as follows:

Figure 11: 70% of original dataset

Figure 12: 100% of original dataset

Figure 13: 70% original+30% samples

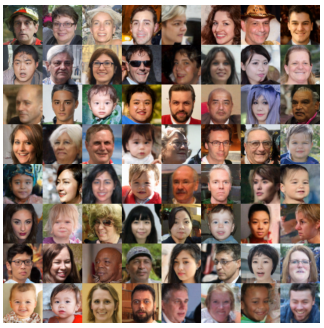

Figure 14: 100%original+30%samples

Figure 15: real images from dataset

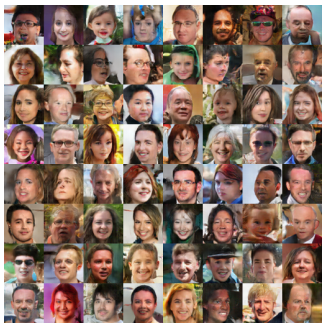

Figure 16: GAN's result from original dataset

5.3 EVALUATION

Here, we use only the discriminator from the GAN model as the assessment method. For each input picture, the discriminator outputs a number in the range of $[0, 1]$ as the probability of the image being a real picture. In practice, we have each model generate 100 pictures as inputs for the discriminator. Subsequently, we obtain a tensor with size 100×1. The assessment of the model's performance is determined by taking the average of these values.

For a more comprehensive comparison, we also include the evaluation of real images alongside those generated by the GAN. However, since the Discriminator is trained simultaneously with the Generator, solely using the Generator to produce images might lead to bias. To mitigate this issue, we use the generator from another GAN model to address this problem, more specifically, we trained another GAN model for 100 epochs with exactly the same hyperparameters, except for a different random seed. The results are shown as follows:

training data	**70% original**	**70% original+30% samples**	**100% original**
evaluating	0.0971 ± 0.1527	$\mathbf{0.1066} \pm 0.1619$	0.1107 ± 0.1656
training data	**100%original+30%samples**	**real images**	**GAN on original**
evaluating	$\mathbf{0.1205} \pm 0.1794$	0.9894 ± 0.0219	0.0481 ± 0.1073

Table 1: Assessment of different models

5.4 RESULT

Our findings indicate that incorporating images generated by the Diffusion model can indeed enhance the model's performance. Combining 70% original data with 30% 'fake' images results in improved performance compared to using only 70% original data. Similarly, combining 100% original data with 30% 'fake' images demonstrates better performance than using 100% original data alone.

6 CONCLUSION AND DISCUSSION

6.1 CONCLUSION

We give an affirmative answer to the question raised at the beginning: generative model can bootstrap itself to deepen its understanding of the world. In a broader sense, we validate the power of generative models in data synthesis using three experiments.

6.2 FURTHER WORK

While we have demonstrated the potential utility of Bootstrapping in expanding the dataset, it is crucial to acknowledge the limitations of our evaluation process. Relying solely on GANs for assessment may not be the most comprehensive or convincing approach. Moreover, determining the boundaries of Bootstrapping remains an unresolved challenge. Further exploration and refinement of evaluation methods are necessary to provide a more robust understanding of the technique's effectiveness.

REFERENCES

Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification, 2023.

Simone Bianco, Luigi Celona, Paolo Napoletano, and Raimondo Schettini. On the use of deep learning for blind image quality assessment. *Signal, Image and Video Processing*, 12(2):355–362, August 2017. ISSN 1863-1711. doi: 10.1007/s11760-017-1166-8. URL http://dx.doi.org/10.1007/s11760-017-1166-8.

Christopher Bowles, Liang Chen, Ricardo Guerrero, Paul Bentley, Roger Gunn, Alexander Hammers, David Alexander Dickie, Maria Valdés Hernández, Joanna Wardlaw, and Daniel Rueckert. Gan augmentation: Augmenting training data using generative adversarial networks, 2018.

Virginia R de Sa. Learning classification with unlabeled data. *Advances in neural information processing systems*, pp. 112–112, 1994.

Sören Dittmer, Michael Roberts, Julian Gilbey, Ander Biguri, Ian Selby, Anna Breger, Matthew Thorpe, Jonathan R. Weir-McCall, Effrossyni Gkrania-Klotsas, Anna Korhonen, Emily Jefferson, Georg Langs, Guang Yang, Helmut Prosch, Jan Stanczuk, Jing Tang, Judith Babar, Lorena Escudero Sánchez, Philip Teare, Mishal Patel, Marcel Wassin, Markus Holzer, Nicholas Walton, Pietro Lió, Tolou Shadbahr, Evis Sala, Jacobus Preller, James H. F. Rudd, John A. D. Aston, and Carola-Bibiane Schönlieb. Navigating the development challenges in creating complex data systems. *Nature Machine Intelligence*, 5(7):681–686, June 2023. ISSN 2522-5839. doi: 10.1038/s42256-023-00665-x. URL http://dx.doi.org/10.1038/s42256-023-00665-x.

Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction, 2016.

Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks, 2015.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations, 2018.

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

Jonathan Ho, Xi Chen, Arun Srinivas, Yanping Duan, and Pieter Abbeel. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems*, pp. 4743–4751, 2016.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation, 2018.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.

Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization, 2017.

Xialei Liu, Joost van de Weijer, and Andrew D. Bagdanov. Rankiqa: Learning from rankings for no-reference image quality assessment, 2017.

Rawad Melhem, Assef Jafar, and Oumayma Al Dakkak. Towards solving cocktail-party: The first method to build a realistic dataset with ground truths for speech separation, 2023.

Quang Nguyen, Truong Vu, Anh Tran, and Khoi Nguyen. Dataset diffusion: Diffusion-based synthetic dataset generation for pixel-level semantic segmentation, 2023.

Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models, 2021.

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles, 2017.

Xingang Pan, Ayush Tewari, Thomas Leimkühler, Lingjie Liu, Abhimitra Meka, and Christian Theobalt. Drag your gan: Interactive point-based manipulation on the generative image manifold, 2023.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting, 2016.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2015.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022.

Masaki Saito, Shunta Saito, Masanori Koyama, and Sosuke Kobayashi. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal gan. *International Journal of Computer Vision*, 128(10–11):2586–2606, May 2020. ISSN 1573-1405. doi: 10.1007/s11263-020-01333-y. URL http://dx.doi.org/10.1007/s11263-020-01333-y.

Daniel Saragih and Pascal Tyrrell. Using diffusion models to generate synthetic labelled data for medical image segmentation, 2023.

J. Schmidhuber. *Making the world differentiable: on using self supervised fully recurrent neural networks for dynamic reinforcement learning and planning in non-stationary environments*. Forschungsberichte Künstliche Intelligenz. Inst. für Informatik, 1990. URL https://books.google.com/books?id=9c2sHAAACAAJ.

Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation, 2017.

Dmitry Utyamishev and Inna Partin-Vaisband. Progressive vae training on highly sparse and imbalanced data, 2019.

Roy Voetman, Maya Aghaei, and Klaas Dijkstra. The big data myth: Using diffusion models for dataset generation to train deep detection models, 2023.

Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos, 2018.

Yilun Xu, Ziming Liu, Max Tegmark, and Tommi Jaakkola. Poisson flow generative models, 2022.

Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization, 2016.

Zhengbang Zhu, Hanye Zhao, Haoran He, Yichao Zhong, Shenyu Zhang, Yong Yu, and Weinan Zhang. Diffusion models for reinforcement learning: A survey, 2023.