Portfolio Project - Working with a Generative Adversarial Network

Karl Estes

Colorado State University Global

CSC 580: Applying Machine Learning and Neural Networks

Dr. Brian Holbert

April 10th, 2022

Portfolio Project - Working with a Generative Adversarial Network

The following discussion will explore generative adversarial networks (GANs) across four specific use cases. Section 1 covers an overview of GANs, their advantages and disadvantages, and a few widely used architectures. Section 2 covers the following four use cases: medical image synthesis, controlled image generation, super-resolution image scaling, and the generation of synthetic network flow traffic. The relevant subsections will discuss specific GAN architectures pertinent to each use case.

1 Generative Adversarial Networks

1.1 Overview

Generative adversarial networks were introduced by Goodfellow et al. in 2014 as a powerful class of generative models. While singularly referred to as a GAN, each model is actually comprised of two primary components: the Generator and the Discriminator. The Generator is the component of the GAN tasked with creating synthetic examples, while the Discriminator is responsible for learning to identify real data from the Generator's creations. In essence, the Generator and Discriminator play a game where they continuously try to optimize themselves to improve their generation and discrimination ability, respectively (Wang et al., 2017). In fact, game theory plays a role in the GAN training process through the concept of the Nash Equilibrium. A Nash Equilibrium denotes a point in a game where the game's optimal outcome can be achieved without any player being incentivized to deviate from their current strategy after considering their opponent's strategy. In terms of GANs, reaching the Nash Equilibrium means that the Generator and Discriminator have both achieved maximal performance, and any further updates will only degrade generation ability or discrimination accuracy.

1.2 Training and Learning

In order to find an equilibrium point, though, a GAN must be trained appropriately. As with any machine learning model, data is required to train a GAN, and this data takes the form of what is considered real samples. Real samples are any data known to be true and stand in contrast to synthetic samples created by the Generator. For the Generator to create realistic-looking synthetic samples, though, it has to try and capture the potential distribution of the real samples. It, however, is given no direct access to real images. Instead, it only learns through interactions with the discriminator (Creswell et al., 2018).

On the other hand, the Discriminator has access to both synthetic and real samples. The Discriminator, which is often implemented as a binary classifier, is provided an error signal through a simple ground truth of knowing whether the image came from the real samples or the synthetic samples (Creswell et al., 2018). During the training process, it is common for the parameters of one model to update while the parameters of the other are fixed. Ideally, the Discriminator will be trained until it is optimal against the current Generator, but in practice, the Discriminator may only be updated for a maximum number of iterations before Generator training switches on (Creswell et al., 2018).

Training a GAN is not without hiccups, though, and the entire process is often unstable. For starters, the Generator is trying to capture the potential probability distribution of real samples; however, the probability distribution of the synthetic samples and the probability distribution of the real samples lie in a lower-dimensional space than the space of all possible representations, causing each distribution to reside on a manifold (Arjovsky & Bottou, 2017; Creswell et al., 2018). Certain GAN architectures will allow for the creation of synthetic samples whose manifold does not overlap with the real samples. When this is the case, there exists a trivial discriminator that will correctly label real/fake samples 100% of the time (Arjovsky & Bottou, 2017). 100% accuracy will result in a Discriminator error of 0, which can cause a convergence problem for the gradients used to update the Generator's neural network. With zero error, the gradients for updating the Generator will also converge to zero and thus may no longer be helpful in the update process.

Training is also unstable due to the potential for indefinite loss during the training process. Indefinite loss means there is no guarantee of finding a local minimum during the training process, so the optimal solution instead lies in finding a saddle point. However, Lee et al. (2016) proved that random selection of initial points for a GAN optimizer does not guarantee convergence on a saddle point with 100% probability. While there do exist second-order optimizers that could better compute saddle points, these Newton-type methods have compute-time complexities that scale quadratically or cubically (Creswell et al., 2018). This scaling can cause the training process time to become unwieldy quickly.

In order to minimize the training instability, a few tricks are commonly employed. Many GANs incorporate fully-connected (FC) layers in their networks. Minimizing the number of FC layers can increase the feasibility of training deeper models, especially when recurrent and convolutional are incorporated into the network architecture (Creswell et al., 2018). Using leaky rectified linear-units (ReLUs) between intermediate layers of the discriminator has been shown to increase performance over normal ReLUs (Radford et al., 2016). Heuristic averaging can be used to penalize network parameters if they deviate from a running average of previous values (Creswell et al., 2018). Virtual batch normalization is also sometimes employed to reduce the dependency of one sample on other samples in mini-batch training (Creswell et al., 2018).

It is also possible to modify the general GAN architecture to achieve more stable training results. While the architecture proposed by Goodfellow et al. (2014) is still at the core of any generative adversarial network, the Wasserstein GAN (WGAN) proposed by Arjovsky et al. (2017) is one of the more widely used architectures today. The WGAN was created to tackle the vanishing gradient problem and can be trained successfully even without batch normalization. WGANs impose the Lipschitz Constraint on the training process, limiting how quickly a given function can change. It is an enforcement of a maximum gradient and guarantees smoother parameter updates. WGANs are only one tailored GAN architecture, though, and the various use case explored in Section 3 will demonstrate that it is common to tailor a GAN's architecture to the problem at hand.

1.3 Advantages and Disadvantages

Before training and implementing any GAN, it is essential to understand its prospective impacts. One of the primary benefits of GANs is their ability to solve the problem of generating data that humans can naturally interpret (Wang et al., 2017). Adversarial training has gained attention in academia and industry because it can help counter domain shifts and is effective in generating new image samples (Yi et al., 2019). "The ability to generate 'infinite' new samples from potential distribution has great application value in many fields such as image and vision computing, speech and language processing, and information security" (Wang et al., 2017, p. 596). In the age of big data, sample generation ability is advantageous, especially in bringing smaller and rarer datasets up to par with the data requirements for training machine learning models.

Sample generation is not guaranteed, though, as model convergence is not assured in all instances. Training instability can lead to mixed results, and there is an issue GANs face called mode collapse. There are "scenarios in which the generator makes multiple images that contain the same color or texture themes, thereby having little difference for human understanding" (Wang et al., 2017, p. 595). While this can sometimes occur as one-off instances for the Generator, complete collapse, or the repeated production of a single synthetic sample, is not uncommon. The synthetic samples must also be vetted for accuracy in their relevant domain space. While the Generator attempts to learn the distribution of real samples, there is no guarantee that synthetic samples will generate key characteristics consistent with human standards. This point will be explored in many of the use cases in the following section.

2 Use Cases

2.1 Medical Image Synthesis

Medical image synthesis can be broken down into three primary categories: unconditional synthesis, conditional synthesis, and cross-modality synthesis. Unconditional synthesis is the process of generating images from random noise without any other conditional information. Conditional synthesis places constraints on image generation, such as the inclusion of anatomical guidelines or clear pathology rules. Cross-modality synthesis involves generating anatomical images from one medical imaging modality (i.e., CT) based on images of another modality (i.e., MR). While cross-modality synthesis is technically conditional in nature, it is often regarded as its own division of medical image synthesis.

Research into these three forms of synthesis has grown with the advancement of deep learning, and GANs have been accelerating this field even further. Data scarcity is a common situation in medical datasets, and the creation of new imaging datasets is difficult due to cost, time, and rules regarding patient privacy (Welander et al., 2018). Many existing datasets currently suffer from a lack of positive cases for different pathologies, and modern medical deep learning tends only to use one imaging modality at a time (Yi et al., 2019). GANs applied to the three divisions of medical image synthesis present a potential solution to these problems. Not only are the data they produce interpretable for humans, but their capacity for infinite new sample generation could solve the rare pathology issue while drastically reducing cost and time. While there are data augmentation techniques that can be used to expand many medical datasets, traditional techniques do not account for variations resulting from different imaging protocols or sequences, nor the size, shape, location, and appearance of specific pathologies (Yi et al., 2019). Since a trained GAN should capture learn the distribution of the real samples, these concerns should not be an issue. Also, since synthetically generated samples do not belong to real people, the rule surrounding patient privacy does not prohibit the dissemination of the images.

Once learned for normal images, the Discriminator can also be used as a regularizer or detector when presented with abnormal images (Yi et al., 2019). Imaging modalities like MR are highly subject to fluctuations in patient movement, so many anatomical sections have to be reimaged if a patient moves during a scan. Minute fluctuations can be hard to detect, however, so a trained Discriminator could be a valuable tool in detecting harder-to-spot abnormalities. It is vital with any application of GAN or GAN components, though, that research is done to validate their capabilities and ensure the accuracy of generated data.

Current research has indicated that, at least in terms of quality, GANs can construct medical images. Bermudez et al. (2018) found that generated MR images were of comparable quality to real ones when examined by neuroradiologists. To achieve this realism, Bermudez et al. (2018) utilized a distance metric from the manifold space of the synthetic and real samples when calculating loss. Since “image synthesis is a form of manifold learning from one image space to another,” the manifold space distance metric was a better discriminator between brian representations (Bermudez et al., 2018, p. 409). While neuroradiologists did note some discrepancies in anatomic accuracy of the generated images, the synthetic brain MR generation stood out because the correlation of synthetic samples with real samples shows unique structural differences in various regions, suggesting the GAN was not simply reconstructing examples from the training set. The researchers also noted that anatomic accuracy could be regulated through the imposition of structural restrictions, changing the problem to a more conditional form of image synthesis.

Cross-modality synthesis has also gained empirical evidence of the benefits of GANs. In 2017, Wolterink et al. used a CycleGAN model to synthesize CT images from MR images. CycleGANs, or Cycle-Consistent Generative Adversarial Networks, are designed to learn translations of images from one manifold space to another in the absence of paired examples (Zhu et al., 2017). This technique is instrumental in the medical field because it does not require images in the training data to be aligned anatomically, which can be a time-consuming and costly procedure. Wolterink et al. (2017) demonstrated the model’s ability to synthesize realistic-looking images on unpaired and unaligned MR/CT images, which previous works had been unable to accomplish. While slight misalignments between reference MR and CT images could affect direct quantitative evaluations with current image similarity metrics, the qualitative results appeared satisfactory. The ability for this type of image synthesis “could have implications for MR-only radiotherapy treatment planning, [and] also for clinical applications where patients typically receive only one scan of a single anatomical region” (Wolterink et al., 2017, p. 22).

Continuing with cross-modality synthesis, CycleGANs and the UNIT architecture have both demonstrated equal aptitude while transforming between T1 and T2 weighted MR images (Welander et al., 2018). These models employed adversarial loss to generate more realistic synthetic MR images, but the calculations hinged on mean absolute error (MAE). The authors noted that utilizing MAE as an evaluation metric pushed generators to produce smooth synthetic brain images. While the images were successfully translated across modalities, “another loss function would probably alter the results, but since it is difficult to create mathematical expressions for assessing how realistic an image is, obtaining visually realistic results using supervised methods is a problematic task” (Welander et al., 2018, p. 3). Choosing an ideal loss function depends on the application domain, and some approaches are notably less suitable for highly detailed applications found in the medical field (Ledig et al., 2017).

While inroads are being made in this field, the adoption of GANs in medical imaging is still very much in its infancy, and there are not yet any breakthrough implementations that have been clinically adopted (Yi et al., 2019). As noted in the three research examples, while various GAN architectures can generate realistic-looking images, tweaks still need to be made to evaluation metrics and synthesis parameters to ensure the realistic images also fall within the bounds of anatomically accurate. Solving this problem could drastically improve various treatment areas and the quality of care for many patients.

2.2 Controlled Image Generation

While GANs have been employed for many unconditional image synthesis tasks, there are use cases where models can benefit from incorporating the notional of localizable objects. Reed et al. (2016b) tackled this problem by introducing the Generative Adversarial What-Where Network (GAWWN). While they trained their model to generate images of birds and humans, the key feature of the GAWWN model is its ability to use keypoints and text descriptions to generate images or to shrink, translate, and stretch objects. Keypoints and other spatial constraints were encoded either through normalized (x,y) coordinates or through spatial masking and cropping modules fed into a text-conditioned GAN implemented using spatial transformers.

In order to learn the correspondence function between the images and the text features, Reed et al. (2016b) used a text encoder model discussed some of them previously researched (Reed et al., 2016a). While "the problem of relating images and text is still far from solved," their text encoder model was trained from scratch on character and word-level representations, which outperformed other state-of-the-art encoders trained on text attributes (Reed et al., 2016a, p. 49). To improve the GAWWN's robustness further, conditional distributions for all possibilities of observed/unobserved keypoints and text descriptions were also generated. It is not user-friendly to require every keypoint for a given object to be provided for image synthesis. As Reed et al. (2016b) noted, this would require 15 keypoint locations to be provided for each bird image that the model was to synthesize. The spread of observed/unobserved possibilities allowed the network to synthesize coherent images with only a fraction of the keypoints and text description provided.

While the results of Reed et al.'s (2016b) GAWWN trained on images of birds significantly outperformed the GAWWN trained on human poses, both experiments were able to generate images that contained the proper colors and relative locations of keypoint. This was a crucial breakthrough in the field of controlled image synthesis because it represents a new pipeline for image generation. By decomposing the problem into more manageable subproblems, more realistic, higher-resolution images were created. While the complexity of the human form eluded the GAWWN in 2016, Reed et al.'s (2016b) research yielded additional ways to control image synthesis. As with the discussion in Section 3.1 and the benefits of image synthesis on expanding medical datasets, controlled image synthesis could greatly benefit researchers in the age of Big Data. In many cases, it is difficult to capture data representing an even distribution of all possible classes. While the GAWWN model does not allow for complete automation of the image creation process, it represents a vast increase in creative capability that could be used to generate specific, tailored images to fit a variety of use cases.

2.3 Image Scaling: Super Resolution

Super-resolution, the task of estimating a high-resolution (HR) image from its low-resolution (LR) counterparts, has gained significant traction in image processing in recent years. A few notable examples of this technology can be observed in the gaming industry with technologies like Nvidia's Deep Learning Super Sampling (DLSS) and AMD's FidelityFX Super-Resolution. These technologies are predicated on generating a higher-quality image based on lower-quality assets, improving graphical load times, and reducing the amount of data that needs to be stored since high-quality images can be computed on the fly.

GANs are one tool that can be used for super-resolution scaling. Ledig et al. (2017) proposed a Super-Resolution Generative Adversarial Network (SRGAN) with a focus on single-image super-resolution (SISR). SISR is focused only on upsampling one image at a time, so it is slower than some technologies like DLSS and FidelityFX SR, but Ledig et al. (2017) was primarily focused on exploring new image quality metrics with GANs. The previous optimization of SR targets typically used MSE because minimizing MSE maximizes the peak signal-to-noise ratio (PSNR), which indicates sharper images. MSE and PSNR are not good metrics for capturing perceptually relevant differences in images, however.

Ledig et al. (2017) employed a deep residual network with skip-connections to generate SR images with higher perceptual quality and defined a new loss metric. Their newly defined perceptual loss metric (see Equation 1.) was an extension of work by Bruna et al. (2016) and Johnson et al. (2016); their work proposed using features extracted from pre-trained VGG networks instead of low-level pixel-wise loss metrics.

$$l^{SR} = \underbrace{\underbrace{l_X^{SR}}_{content\ loss} + \underbrace{10^{-3} l_{Gen}^{SR}}_{adversarial\ loss}}_{Perceptual\ Loss} \tag{1}$$

Mean Opinion Scoring (MOS) was performed on SR images generated by SRGAN and other state-of-the-art models that relied on more traditional error metrics. While SRGAN failed to achieve the highest PSNR, MOS scores were consistently higher for SRGAN generated images. Higher MOS indicates a perceptual win-out, meaning PSNR and other traditional metrics fail to accurately assess image quality compared to the human visual system (Toderici et al., 2017). This finding has profound implications for SR technology since pixel-perfect accuracy through statistical analysis may not correspond to a more visually appealing image. For SR technologies like DLSS and FidelixtyFX SR, which are focused on upscaling graphics for entertainment applications like video games, it might provide a better user experience to have more visually appealing images over pixel-perfect upscaling. However, perceptual loss may not be the best loss function for all SR applications. As Ledig et al. (2017) noted in their concluding remarks, approaches that hallucinate finer details may not be suited for applications like medical image synthesis or surveillance due to the requirements of those domains. GANs should be trained to produce content that adheres to the constraints of the problem space in which they are being implemented.

2.4 Generation of Adversarial Malware Examples

Ring et al. (2019) proposed a unique use of GANs with the generation of synthetic network traffic. More specifically, their research examined the creation of realistic network flows, which are aggregations of transmitted network packets that share certain properties; Flows contain header information about network connections between any two end-point devices. Synthetically generating realistic network flows would be a boon for increasing network-based intrusion detection (NIDS) datasets. Very few NID publicly available labeled datasets contain behavior with up-to-date attack scenarios. Increasing the availability of these datasets could be useful since “large training data sets with high variance can increase robustness of anomaly-based intrusion detection methods” (Ring et al., 2019, p. 157).

A few aspects of network flows complicated the application of GANs, however. Attributes like IP addresses and ports (while comprised of a range of numerical values) are categorical, and GANs can only process continuous input attributes. Previous research has shown that Wasserstein GANs (WGANs) are capable of modeling discrete distributions (those that would arise from categorical data) over continuous latent spaces (Gulrajani et al., 2017). WGANs are not without their training drawbacks, though, so the Improved WGAN architecture proposed by Gulrajani et al. (2017) was selected as the base model for network flow generation. The Improved WGAN (WGAN-GP) model addresses optimization difficulties arising from weight clipping by using gradient penalty (GP) as an alternative method of Lipschitz constraint enforcement to improve training speed and generated sample quality. The WGAN-GP model was extended with three types of data preprocessing, forming a Numeric-Based WGAN-GP (N-WGAN-GP), a Binary-Based WGAN-GP (B-WGAN-GP), and an Embedding-Based WGAN-GP (E-WGAN-GP).

The N-WGAN-GP transformed each octet of an IP address to the interval [0,1] and applied a similar procedure to port numbers. The B-WGAN-GP took each octet of an IP address and mapped it to an 8-bit binary representation, while ports were converted to a 16-bit binary representation. The E-WGAN-GP extended the IP2Vec framework proposed by Ring et al. (2017) to learn embeddings for IP addresses, ports, transport protocols, duration, bytes, and packets. Each model also applied the two-time scale update rule (TTUR), which has been proven to force convergence of a GAN to a stationary local nash equilibrium during the training process (Heusel et al., 2017; Prasad et al., 2015).

Since there are no standards for evaluating network traffic flow generators in the IT security domain (Molnár et al., 2013), violing plots were utilized to assess the visual data distributions, and the Euclidean distances were computed between generated and actual flow-based network data. Ring et al. (2019) captured the distribution of network flow traffic well with both the B-WGAN-GP and the E-WGAN-GP models. The N-WGAN-GP architecture could not represent all data distributions properly, and other baseline network traffic flow generators lagged significantly in capturing the distribution of source to destination IP addresses in the simulated domain. The B-WGAN-GP and the E-WGAN-GP models were not identical in their distribution capture, though.

The E-WGAN-GP architecture achieved better results on average with the smallest Euclidean distances from the actual network-flow data. However, using embeddings reduces a model's ability to produce values outside the previously seen data space. B-WGAN-GP, on the other hand, was slightly more error-prone than the embedding counterpart, but it was able to generate data outside of the embedding space. Both of these models could thus be used to bolster NID datasets in their own way. As Ring et al. (2019) noted, if there were a need to generate previously unseen IP addresses, E-WGAN-GP would not be a suitable model; however, if very few network flow examples were present, the B-WGAN-GP model could potentially expand the variance of the data samples which would, in turn, increase the robustness of anomaly-based intrusion detection methods trained with that data. The ability to generate synthetic flows could also allow for datasets tailored to specific network configurations. It would also be possible to take a clean flow and augment the dataset with a virtual malware attack rather than having to establish physical systems from which to retrieve data.

3 Conclusion

Generative adversarial networks are becoming a vital tool in the machine learning space. While four use-case-specific examples were discussed throughout Section 3, each example highlighted how GANs could transform some aspect of that domain for the better. GANs are not limited only to these domains, though, and each use case should highlight how a GAN architecture can be refined to address a more narrow problem. GANs are modular in many ways, and the limits of their potential have yet to be reached.

It is important to note, though, that are not yet clear metrics in every field for analyzing the performance of generative models. As discussed in Section 3.3, Ledig et al. (2017) designed a new loss metric to push GANs to generate more realistic-looking images because standardized approaches were not cutting it. The various investigations into cross-modality research in Section 3.1 touched on the differences in perceptual ratings of the images versus scored image similarities. As GANs continue to develop and are further implemented in various industries, it will be important to ensure proper metrics are researched to accurately grade synthetic sample quality and ensure that computer-generated content meets the same data requirements and integrity that we already expect from our data.

References

Arjovsky, M., & Bottou, L. (2017). Towards principled methods for training generative adversarial networks. *ArXiv:1701.04862 [Cs, Stat]*. http://arxiv.org/abs/1701.04862

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. *Proceedings of the 34th International Conference on Machine Learning*, 214–223. https://proceedings.mlr.press/v70/arjovsky17a.html

Bermudez, C., Plassard, A. J., Davis, L. T., Newton, A. T., Resnick, S. M., & Landman, B. A. (2018). Learning implicit brain MRI manifolds with deep learning. *Medical Imaging 2018: Image Processing*, *10574*, 408–414. https://doi.org/10.1117/12.2293515

Bruna, J., Sprechmann, P., & LeCun, Y. (2016). Super-Resolution with deep convolutional sufficient statistics. *ArXiv:1511.05666 [Cs]*. http://arxiv.org/abs/1511.05666

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A. (2018). Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, *35*(1), 53–65. https://doi.org/10.1109/MSP.2017.2765202

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, *27*. https://proceedings.neurips.cc/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. C. (2017). Improved training of Wasserstein GANs. *Advances in Neural Information Processing Systems*, *30*. https://proceedings.neurips.cc/paper/2017/hash/892c3b1c6dccd52936e27cbd0ff683d6-Abstract.html

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). GANs trained by a two time-scale update rule converge to a local Nash Equilibrium. *Advances in Neural Information Processing Systems*, *30*. https://proceedings.neurips.cc/paper/2017/hash/8a1d694707eb0fefe65871369074926d-Abstract.html

Johnson, J., Alahi, A., & Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer Vision – ECCV 2016* (pp. 694–711). Springer International Publishing. https://doi.org/10.1007/978-3-319-46475-6_43

Ledig, Christian, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, et al. “Photo-realistic single image super-resolution using a generative adversarial network.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4681–90. Honolulu, HI: IEEE, 2017. https://openaccess.thecvf.com/content_cvpr_2017/html/Ledig_Photo-Realistic_Single_Image_CVPR_2017_paper.html.

Lee, J. D., Simchowitz, M., Jordan, M. I., & Recht, B. (2016). Gradient descent only converges to minimizers. *JMLR: Workshop and Conference Proceedings*, *49*, 1–12.

Molnár, S., Megyesi, P., & Szabó, G. (2013). How to validate traffic generators? *2013 IEEE International Conference on Communications Workshops (ICC)*, 1340–1344. https://doi.org/10.1109/ICCW.2013.6649445

Prasad, H. L., Prashanth, L., & Bhatnagar, S. (2015). Two-timescale algorithms for learning Nash equilibria in general-sum stochastic games. *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, 1371–1379.

Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised representation learning with deep convolutional Generative Adversarial Networks. *ArXiv:1511.06434 [Cs]*. http://arxiv.org/abs/1511.06434

Reed, Scott, Zeynep Akata, Honglak Lee, and Bernt Schiele. "Learning deep representations of fine-grained visual descriptions." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 49–58. Las Vegas, NV: IEEE, 2016a. https://openaccess.thecvf.com/content_cvpr_2016/html/Reed_Learning_Deep_Representations_CVPR_2016_paper.html.

Reed, S. E., Akata, Z., Mohan, S., Tenka, S., Schiele, B., & Lee, H. (2016b). Learning what and where to draw. *Advances in Neural Information Processing Systems*, *29*. https://proceedings.neurips.cc/paper/2016/hash/a8f15eda80c50adb0e71943adc8015cf-Abstract.html

Ring, M., Dallmann, A., Landes, D., & Hotho, A. (2017). IP2Vec: Learning similarities between IP addresses. *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, 657–666. https://doi.org/10.1109/ICDMW.2017.93

Ring, M., Schlör, D., Landes, D., & Hotho, A. (2019). Flow-based network traffic generation using Generative Adversarial Networks. *Computers & Security*, *82*, 156–172. https://doi.org/10.1016/j.cose.2018.12.012

Toderici, George, Damien Vincent, Nick Johnston, Sung Jin Hwang, David Minnen, Joel Shor, and Michele Covell. "Full resolution image compression with recurrent neural networks." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5306–14. Honolulu, HI: IEEE, 2017. https://openaccess.thecvf.com/content_cvpr_2017/html/Toderici_Full_Resolution_Image_CVPR_2017_paper.html.

Wang, K., Gou, C., Duan, Y., Lin, Y., Zheng, X., & Wang, F.-Y. (2017). Generative adversarial networks: Introduction and outlook. *IEEE/CAA Journal of Automatica Sinica*, *4*(4), 588–598. https://doi.org/10.1109/JAS.2017.7510583

Welander, P., Karlsson, S., & Eklund, A. (2018). Generative adversarial networks for image-to-image translation on multi-contrast MR images - A comparison of CycleGAN and UNIT. *ArXiv:1806.07777 [Cs]*. http://arxiv.org/abs/1806.07777

Wolterink, J. M., Dinkla, A. M., Savenije, M. H. F., Seevinck, P. R., van den Berg, C. A. T., & Išgum, I. (2017). Deep MR to CT synthesis using unpaired data. In S. A. Tsaftaris, A. Gooya, A. F. Frangi, & J. L. Prince (Eds.), *Simulation and Synthesis in Medical Imaging* (pp. 14–23). Springer International Publishing. https://doi.org/10.1007/978-3-319-68127-6_2

Yi, X., Walia, E., & Babyn, P. (2019). Generative adversarial network in medical imaging: A review. *Medical Image Analysis*, *58*. https://doi.org/10.1016/j.media.2019.101552

Zhu, Jun-Yan, Taesung Park, Phillip Isola, and Alexei A. Efros. “Unpaired image-to-image translation using cycle-consistent adversarial networks.” In *Proceedings of the IEEE International Conference on Computer Vision*, 2223–32. Honolulu, HI: IEEE, 2017. https://openaccess.thecvf.com/content_iccv_2017/html/Zhu_Unpaired_Image-To-Image_Translation_ICCV_2017_paper.html.