Project Document: Deep(FAMS)

Mohammad Alyetama
Ali Al-Ramini
Fangyi Li

Contents

Abstract

This project will attempt to optimize generative adversarial networks to generate good results with limited data using a recently proposed mechanism, the adaptive discriminator augmentation (ADA). This project aims to evaluate the reproducibility and generalizability of the results generated by this proposed mechanism. Since the ADA does not require changes to the network architectures, we will build models based on the StyleGAN2 and test it to evaluate the reported results' reproducibility in the original paper. Using several small datasets, we plan to demonstrate that the ADA produces good results using limited training images.

Chapter 1

Milestone 1: Project Ideas

1.1 Introduction

1.1.1 Project Idea 1

Generative adversarial networks always seek to increase the dataset size to generate desirable results as an approach to generative modeling. The massive demand for data is challenging to lots of areas, and thus placing constraints on more widely being used of this practical approach. However, the dataset size reduction can cause overfitting issues in checking step by discriminator sub-model. Applying dataset augmentation can generate the augmented distribution, which is highly undesirable to generate samples. An attempt to apply a wide range of augmentation by Kerras et al. is trying to demonstrate GAN can use a smaller dataset size but prevent the augmentation 'leaking' at the same time. In this project, we will follow Karras et al. methods of adaptive discriminator augmentation mechanism [1] to try to reproduce their result.

1.1.2 Project Idea 2

There is a great interest in improving the cycling infrastructure in Nebraska by analyzing cyclic data. Toward this goal, we acquired the Strava Metro data covering Nebraska's state from January 2017 to December 2019. Strava provides data into four categories edges, nodes, origin-destination polygons, and shapefiles that contain spatial attributes to create maps using GIS software. Edges are street segments between nodes. In other words, a set of edges form a street. Nodes are the intersections between edges, and origin-destination polygons divide a space into smaller areas. Every category of Strava data is divided into yearly, monthly, and hourly data. The Strava data, made available by the social fitness network company Strava, includes raw data for the hour-by-hour counts for bicycle trips that have been mostly incorporated into existing maps for better visualization. In this project, we propose building a deep learning regression model that utilizes Strava data in addition to weather data to predict the number of cycling trips in Nebraska.

1.2 Project Idea 1: Evaluating the Reproducibility of Training GAN With Limited Data

Generative adversarial networks (GAN) is a generative modeling approach that uses deep learning techniques to automatically discover and learn the input patterns to generate an output that would plausibly appear as if it was sampled from the original input data [2]. For example, a GAN that uses convolutional neural networks (CNNs) can take pictures of humans as input data and generate new pictures of humans with plausible characteristics that look superficially genuine. This approach gained popularity after it was introduced by Goodfellow and his colleagues in 2014 [2]. In their paper, they describe GAN as a structure with two essential sub-models: (1) a generator model that learns to generate superficially plausible data, which the discriminator takes as negative training examples, and (2) a discriminator model that learns to discriminate between generated and real data, and penalizes the generator if an implausible result is detected [3]. The generator and the discriminator are neural networks that are directly connected. This connection allows the discriminator to send a signal, through backpropagation, that the generator uses to update its weights. Specifically, the generator samples a vector that is randomly drawn from a Gaussian distribution and use it to seed its generative process and match it to a distribution of interest, creating a 'latent space.' With sufficient training, the generator model can learn the input data's statistical latent space and create output data similar to what is observed in the input data [4]. An example of such data is then processed by the discriminator model and attempt to distinguish it from the real distribution of the data (i.e., a binary class of fake/real). The generator model's goal here is to maximize the error of the discriminator model. In other words, the generator model becomes more effective the more the discriminator process fake data as real data. The GAN approach has been used in fashion, science, and video games with impressive results [5].

However, a significant challenge in this area is the large number of data required to build a good GAN model, which is in some cases not available for researchers interested in applying GAN to their research question. GAN typically requires a large dataset because, with smaller datasets, the discriminator model ends up overfitting to training data examples, and the training eventually diverges [6]. While dataset augmentation is typically applied in such situations to solve overfitting [7], it cannot achieve this in GAN models. The inability to solve this problem with dataset augmentation stems from GAN's ability to employ this technique without learning the augmented distribution and leaking these results to the model [6] causing undesirable outcomes. Therefore, the challenge is to demonstrate that GAN can be used with smaller datasets without the pitfalls mentioned above.

A recent attempt to solve this problem was proposed by Kerras et al. [1], demonstrating that it is possible to obtain good results using limited data. The critical point in their proposed approach is that we can prevent overfitting and augmentations leak by applying a wide variety of augmentation methods. Their work demonstrates the validity of their approach by describing a set of conditions that allows controlling the augmentations leak problem and then proposes an adaptive discriminator augmentation pipeline that can dynamically control the strength of the augmentation. This is a novel approach they propose contrasting the convention of tuning the augmentation strength manually, a resources-consuming process. The process of building an adaptive discriminator augmentation mechanism, as described by the authors, is achieved by (a) Declare an overfitting heuristic, r, in which a value of zero represents no overfitting and a value of 1 means perfect overfitting. (b) Adjust augmentation probability (p) until the heuristic reaches a target value, which in turn can be processed by: (1) initializing the augmentation strength to zero, then adjusting p every four mini-batches based on an overfitting heuristic (rf). (2) if rf shows too much overfitting, it is countered by a fixed increment of p (or fixed decrement if rf shows too little overfitting). (3) The adjustment size is then set in a way that allows p to quickly rise from 0 to 1 while clamping p after every step from below to zero (adaptive). (4) The results from adaptive versus fixed p are then compared. Using such mechanisms on limited data, the authors demonstrated that their adaptive discriminator augmentation approach improved the quality of the result and stabilized training with a minimal effect on resources consumption, showing that their strategy is both viable and cost-efficient [1].

In this project, we will attempt to reproduce the results reported by Karras et al. using their adaptive discriminator augmentation mechanism. We expect our reproduced work to support Kerras et al.'s claim made in the paper; that is, good results can be obtained in a GAN model with only a few thousand training images. The original article's hypothesis was tested using five small datasets (METFACES, BRECAHAD, AFHQ, and CIFAR-10). Here, we plan to test adaptive discriminator augmentation on additional datasets to find out whether the scope of Kerras et al. paper is generalizable. This would be a pivotal point in our project because the authors of the article claim that their model's strength stems from its ability to work despite variations between datasets in content and size. This is a central argument to their approach. They experimentally demonstrate that a set of fixed augmentation parameters (as opposed to adaptive parameters) will miss the utmost advantage a GAN model can achieve. Overall, our results will allow us to evaluate the reproducibility of the results, the readability of the source code, and experiment with the ability to generalize the approach described in the paper with different datasets.

1.3 Project Idea 2: Using App Data to Model Bicycling Patterns in Nebraska

Across the United States, cycling is becoming increasingly popular as users shift travel modes amid concerns of health, physical activity, air, and environmental quality, and to escape roadway congestion. Unfortunately, the infrastructure in the U.S. traditionally caters to automobile traffic creating impediments for bikers and impacting their safety. To accommodate cycling, a major challenge is the lack of machine learning model representation of the available data to assess the attributes of present assets accurately and to inform additional investments to integrate bicycles into our transportation system. Toward that end, this project uses citywide bicycle travel data (i.e., Strava Metro Data) to provide a comprehensive description of daily cycling in a mid-size American state as a proof-of-concept approach to planning for cycling.

Various governments and organizations utilize big data to evaluate their cycling infrastructure [8]. Big cycling data are usually collected using live point data, journey data, Bike-Share Programs (BSP), and GPS. Live point data are collected on intersections using cameras on traffic lights, counting stations, or even sensors. While the journey data provide information about the origin and the destination of the trip, it does not provide the trip details. This set of data could be collected from BSP or by other sources like online questionnaires. BSP data are complete and in real-time. However, these data only give information within the area of its location [9, 10]. Strava is considered a GPS program that is made available by the social fitness network company Strava. Strava utilizes the Open Street Map (OSM) to deliver its data. These GPS data are very detailed and historical but represent a small sample of the cyclists' total population. Strava app data contains a vast amount of spatial and temporal details to predict cycling activity patterns. It provides a good approximation of the most-used routes and the peak months and hours. To protect privacy, the Strava data set is combined into population datasets. While a small portion of cyclic may use Strava to log their trips or the app might track trips for users using other transportation methods [8, 9, 11], several studies showed that there is a strong correlation between the Strava data and the ground-truth data obtained from counting stations [12, 13]. Cycling is affected by several factors such as weather, time of the day, infrastructure, congestion, environment, income, public transportation, health, population density, the slope of the street, and cultural view towards cycling [14–16]. But traditionally, access to high-quality data has limited our understanding of cycling behavior and route choice in the face of these myriad factors. In this project, we explore the weather effects and weather parameters sensitivity to cycling in Nebraska. This work aims to specify cycling behavior further as it relates to specific factors, but more importantly, to determine the quality of the specified factors in determining the number of cycling trips over a vast area like the State of Nebraska. Using data visualization techniques and Deep learning regression (e.g., ANN, RNN, LSTM, and GRU) [17,18] we study the most influential time-related factors affecting the cycling patterns in Nebraska. Moreover, cycling is usually categorized into two classes, commuting and recreational. The proposed study will take advantage of the data shared by Strava to predict the number of commutes and recreational activities across all streets.

1.4 Conclusions

1.4.1 Project Idea 1

The core of this project is to test if the methods proposed by Karras et al. can solve the problem of augmentation leak and then ease the burden of huge datasets required by GAN. Given that the proposed mechanism is relatively new and paradigm-shifting, we believe that the idea of reproducing the paper's results would be greatly valuable. Studying such an approach to solve overfitting and subsequent problems while using limited datasets in GAN allows any researcher access to cost-efficient models.

1.4.2 Project Idea 2

This project provides an exciting idea: to create a deep learning model that predicts Nebraska's cycling patterns. With the help of high-quality data sources (e.g., Strava Data and Weather Data), this project's outcomes could be essential towards understanding how cyclists react to temporal variations. However, the spatial factors affecting cycling should also be studied, adding more complexity to the problem. The spatial representation of the data could be tough to map during this short period. Additionally, this idea is essential considering the current situation with COVID-19, adding more complexity and difficulty to reproducing the analysis to agree with the latest status.

Chapter 2

Milestone 2: Project Selection

2.1 Introduction

Our group semester project is Evaluating the Reproducibility of Training GAN with Limited Data. As mentioned in the last chapter, GAN, representing Generative Adversarial Networks, is a generative modeling approach using deep learning techniques to automatically discover and learn the input patterns to generate an output that would plausibly appear as it was sampled from the original dataset [2]. The GAN approach has gained its popularity in diverse areas including but not limited to fashion, science, and video games [5]. The success of the GAN approach can be attributed to its structure: a generator model and discriminator model connecting, where the generator model learns to generate superficially plausible data, and the discriminator model takes the generated data to discriminate with the real data and penalizes the generator if there's an implausible result detected [3]. The connection between the two models allows the updating of the generator based on the discrimination of discriminator and generating the more plausible outputs [4]. However, its success is still weakened by the challenge of its need for the large dataset to ensure the GAN approach's efficiency. The applying of a small dataset makes the discriminator ended up overfitting to training data examples, and the training diverged [6]. What's more, if the augmentation of the dataset is introduced to resolve the overfitting, the augmented distribution can be caused, which is highly undesirable for sample generation. [6] Therefore, it is a challenge that applying the GAN approach with the smaller dataset and avoid the mentioned pitfalls at the same time. In the research by Kerras et al. [1], one method was come up to realize obtaining good results using a smaller dataset. The research was applying a wide variety of augmentation methods. The results demonstrated the validity by describing conditions that control the augmentations leak problem and then proposed an adaptive discriminator augmentation pipeline that can control the strength of the augmentation.

The paper provides a small dataset to obtain good results from the GAN approach; this possibility is valuable since the broke of dataset size limitation will help the GAN approach spread in more areas, and correspondingly those areas will obtain an efficient way to solve their problems. The offered methods provide us with the prospect for similar problems, which is an excellent chance for us to learn about model learning. Additionally, the decision on Evaluating the Reproducibility of Training GAN with Limited Data is also the result of considering the feasibility within the limited one-semester timeline.

2.2 Problem Specification

The work presented in this project provides a complete evaluation of the proposed generative image training models by Karras et al. in [1]. The idea is to reproduce the generative image models trained on significantly fewer data than other approaches in the past. We are going to use the adaptive discriminator augmentation mechanism proposed in the paper, following the codes provided in [1], to assess its reproducibility and ability to classify relatively small (few thousands) datasets.

In many application fields, it is challenging to collect large datasets to perform data training. For example, in medicine, there is an ongoing challenge in modeling the possible appearance of biological specimens (tissues, tumors, etc.) This is a growing body of research that seems to suffer from limited high-quality data constantly. As introduced above, GAN has a great performance in training unlimited online data [2, 19–24], as a result, it cannot be used for lots of specific applications that only collected around hundreds or thousands of samples, where collecting more samples can be very complicated and costly. If the small datasets are applied for the GAN approach, the overfitting will occur in the discriminator. The dataset augmentation always used for overfitting can lead to training divergence in this GAN approach situation, which will not get a good result in the end. The method proposed by Kerras et al. has been providing the chance to break this deadlock. To make the GAN approach more suitable for small datasets, this method makes use of a wide range of augmentations to ensure that the discriminator does not overfit while the generator does not also leak. This method's success will make the GAN approach more available for those specific applications with relatively small datasets, where's great potential to solve more problems. Thus, in this project, we evaluate the GAN augmentations proposed by Karras et al. and reproduce their approaches in terms of model architecture and code. Moreover, we test the model on several additional small datasets and compare our results with the original paper authors' results to find out if the scope of the model is generalizable. From an applied point of view, this work contributes to efficiency; by testing the GAN augmentations proposed, this work will further confirm the elimination of the barrier for applying GAN-type models in many applied fields of research.

In this project, there are some requirements mentioned in the original paper to make sure reproduce the study successfully:

- Python 3.7
- TensorFlow 1.14 to develop and train ML models.
- `Numpy>=1.14.3 for working with arrays.`
- `tensorflow_ds and pandas>=1.0 to load datasets.`
- High-end NVIDIA GPUs with at least 12 GB of GPU memory, NVIDIA drivers, CUDA 10.0 toolkit and cuDNN 7.5.

We will use several datasets that consist of a limited number of training images, including:

1. METFACES [1]
2. AFHQ [25]
3. CIFAR-10 [26]
4. StandfordDogs [27, 28]
5. Cars196 [29]
6. OxfordFlowers102 [30]

2.3 Proposed Method:

In this section, since we are attempting to reproduce an article that proposes one specific method, we provide extended details for this one method.

- **Datasets.** In our experiments, we will use three out of the six datasets used in the original paper. That is, `METFACES`, the Animal Faces-HQ (`AFHQ`) dataset, and `CIFAR-10`. In addition to these three datasets, we will test the generalizability of the original paper's findings by using three other small datasets that were not tested in Keras et al. paper. These are the `StanfordDogs` dataset, the `Cars196` dataset, and the `OxfordFlowers102` dataset. See Table 1 for more details about the datasets we plan to use in the present project. The datasets will be either downloaded from Tensorflow datasets collection (CIFAR-10, Cars196, OxfordFlowers102, and StanfordDogs) or from source (AFHQ and METFACES).
- **Preprocessing.** The `METFACES` dataset is available in both raw and process format. We will use the raw format and process it by aligning and cropping images at 1024×1024 pixels, then use various automatic filters to prune the dataset. The `AFHQ` dataset will be split into three subsets: CAT, DOG, and WILD. All datasets will be standardized to make the training faster and reduce the probability of getting stuck in local optima.

Table 2.1: Datasets Information

Dataset	No. of images	Brief Description
METFACES	1,336	The dataset contains human faces extracted from works of art. The images are aligned and cropped images at 1024×1024.
AFHQ	CAT: 5153 DOG: 4739 WILD: 4738	The dataset contains images at 512×512 resolution, with three domains of classes (CAT, DOG, and WILD).
CIFAR-10	60,000	Colour images in 10 classes, with 6000 images per class. 50,000 training images and 10,000 test images.
StanfordDogs	20,580	The dataset contains images of 120 breeds of dogs with 12,000 training images and 8,580 test images.
Cars196	16,185	The Cars dataset contains 196 classes of cars. The data is split into 8,144 training and 8,041 testing images. Classes are at the level of Make, Model, Year.
OxfordFlowers102	6,149	The dataset contains 102 classes of flowers typically found in the United Kingdom. Each class contains 40-258 images.

- **Pipeline**. The authors of the paper implemented their techniques on top of the `StyleGAN2` official TensorFlow implementation and kept most of the network architecture unchanged. For this project, we will use the baseline StyleGAN2 as illustrated in Figure 2.1.

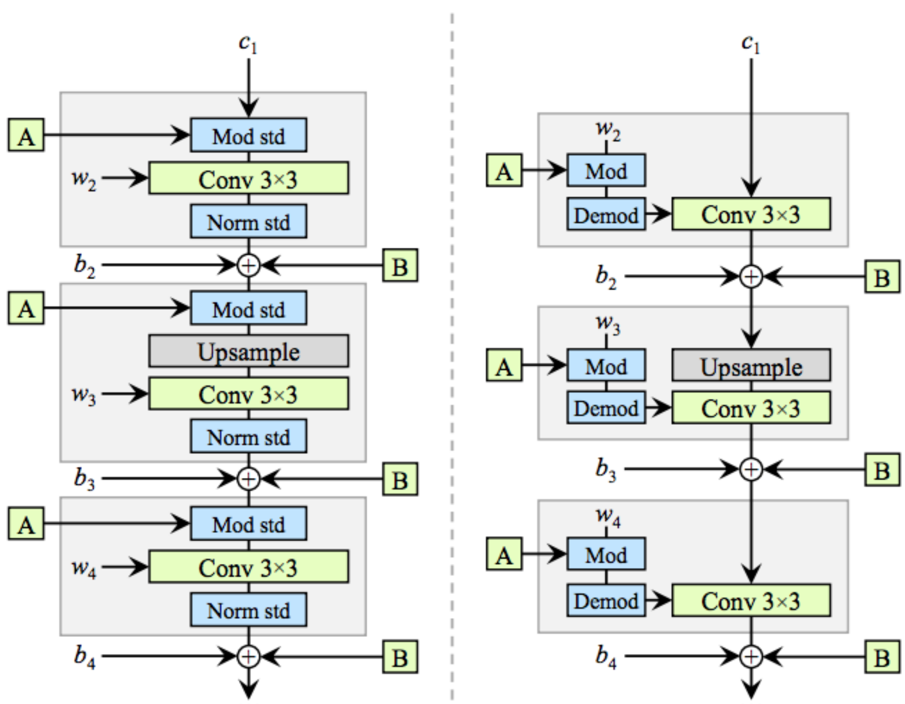

Figure 2.1: StyleGAN2 architecture

1. ***The generator model.*** The generator model starts with an input layer that takes an image as input and a vector as an output in the latent space and gets normalized. We follow this with two fully connected layers (Dense layers), then we add a 4×4 convolutional layer, upsample, a 4×4 ToRGB layer, then a UpConv layer (upsampling + Sum + Residual Unit). We repeat this six more times while increasing the kernel size to the power of 2 (8×8, 16×16, 32×32, etc.).
2. ***The discriminator model.*** For the discriminator model, following the input layer, we will add a FromRGB layer with a kernel size of 256×256, followed by a DownConv layer (convolutional layer + Residual Unit + downsampling), a skip connection. We will repeat this six times before adding a mini-batch standard deviation, a convolutional layer then a fully connected layer.
3. ***Quality metrics.*** In this project, we will use multiple metrics to evaluate the performance of our GANS.
   We will compute Frechet Inception Distance (FID) against the full dataset for each network to evaluate it. FID is a metric that calculates the distance between feature vectors calculated for real and generated images. Low FID score indicates that the images generated by the generator is similar to the real ones. We expect the FID result to be minimumal and comparable to results from the original paper (5.59 to 2.42). To calculate FID, we will perform the following:
   (a) Use the Inception v3 pre-trained model and extract the feature vectors of real and generated images.
   (b) Find the mean feature-wise of the vectors generated in the previous step.
   (c) Generate the feature vectors' covariance matrices.
   (d) Calculate the sum of the elements along the main diagonal of the square matrix.
   (e) Calculate the squared difference of the mean vectors.
   (f) Add the output from the previpous two steps.

- ***Timeline.*** The project timeline is described in Figure 2.2.

2.4 Conclusions

GAN approach is an efficient generative modeling approach, but its high demand for dataset size limits small-sized datasets applications. The experiments in Keras et al. proposed ADA as a method to increase the GAN approach's feasibility for small-size datasets. By applying an adaptive discriminator augmentation mechanism, the GAN approach can generate good results with small datasets. In our project, we will attempt to reproduce similar results to the models' performance reported in the paper, following the source code and the methodology reported in the paper, we will test ADA on three datasets from the original paper and three additional new datasets. As described above, after the preprocessing of data, we will follow the pipeline where the baseline is StyleGAN2. The paper provides good details of the methodology and well-documented source code that will allow us to test whether we can reproduce the results.

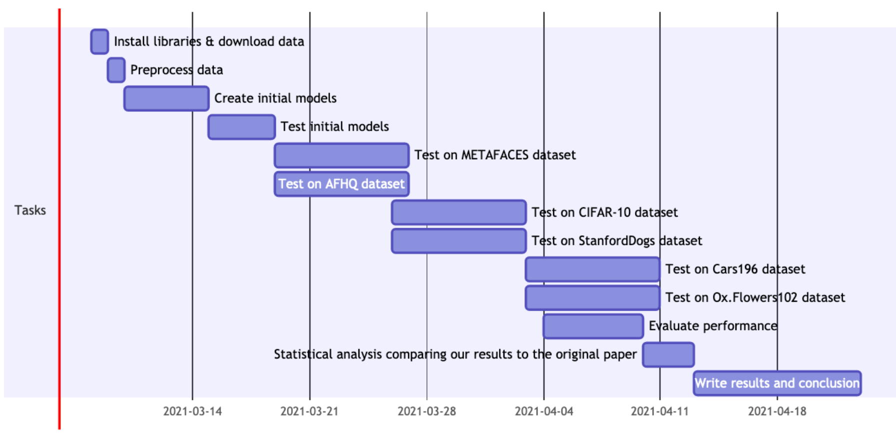

Figure 2.2: Project Timeline

Bibliography

[1] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," *arXiv preprint arXiv:2006.06676*, 2020.

[2] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv:1406.2661*, 2014.

[3] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[4] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[5] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, "A review on generative adversarial networks: Algorithms, theory, and applications," *arXiv preprint arXiv:2001.06937*, 2020.

[6] Z. Zhao, S. Singh, H. Lee, Z. Zhang, A. Odena, and H. Zhang, "Improved consistency regularization for gans," *arXiv preprint arXiv:2002.04724*, 2020.

[7] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[8] W. Hall, N. Shadbolt, T. Tiropanis, K. O'Hara, and T. Davies, "Open data and charities," 2012.

[9] G. Romanillos, M. Zaltz Austwick, D. Ettema, and J. De Kruijf, "Big data and cycling," *Transport Reviews*, vol. 36, no. 1, pp. 114–133, 2016.

[10] S. Rogers and N. P. Papanikolopoulos, "Counting bicycles using computer vision," in *ITSC2000. 2000 IEEE Intelligent Transportation Systems. Proceedings (Cat. No. 00TH8493)*, pp. 33–38, IEEE, 2000.

[11] E. Fishman, "Cycling as transport," 2016.

[12] J. Hong, D. P. McArthur, and M. Livingston, "The evaluation of large cycling infrastructure investments in glasgow using crowdsourced cycle data," *Transportation*, vol. 47, no. 6, pp. 2859–2872, 2020.

[13] B. Jestico, T. Nelson, and M. Winters, "Mapping ridership using crowdsourced cycling data," *Journal of transport geography*, vol. 52, pp. 90–97, 2016.

[14] W. Musakwa and K. M. Selala, "Mapping cycling patterns and trends using strava metro data in the city of johannesburg, south africa," *Data in brief*, vol. 9, pp. 898–905, 2016.

[15] A. Roy, T. A. Nelson, A. S. Fotheringham, and M. Winters, "Correcting bias in crowdsourced data to map bicycle ridership of all bicyclists," *Urban Science*, vol. 3, no. 2, p. 62, 2019.

[16] H. H. Hochmair, E. Bardin, and A. Ahmouda, "Estimating bicycle trip volume for miami-dade county from strava tracking data," *Journal of transport geography*, vol. 75, pp. 58–69, 2019.

[17] M. H. Hassoun *et al.*, *Fundamentals of artificial neural networks*. MIT press, 1995.

[18] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[19] T. Miyato and M. Koyama, "cgans with projection discriminator," *arXiv preprint arXiv:1802.05637*, 2018.

[20] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," *arXiv preprint arXiv:1802.05957*, 2018.

[21] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018.

[22] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[23] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

[24] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8110–8119, 2020.

[25] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "Stargan v2: Diverse image synthesis for multiple domains," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8188–8197, 2020.

[26] A. Krizhevsky, G. Hinton, *et al.*, "Learning multiple layers of features from tiny images," 2009.

[27] A. Khosla, N. Jayadevaprakash, B. Yao, and L. Fei-Fei, "Novel dataset for fine-grained image categorization," in *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, (Colorado Springs, CO), June 2011.

[28] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[29] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3d object representations for fine-grained categorization," in *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, (Sydney, Australia), 2013.

[30] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, Dec 2008.