Hochschule der Medien

Fakultät für Druck und Medien

Computer Science and Media

Generative Data Augmentation

**Multi-Agent Diverse Generative Adversarial Networks
for Generative Data Augmentation**

**Dissertation submitted for the degree of
Master of Science**

Topic: Generative Data Aufmentation

Author: Nicolas Reinhart - nr063@hdm-stuttgart.de
                      MatNr. 44100

Version of Date: March 28, 2025

1. Advisor: Prof. Dr.-Ing. Johannes Maucher
2. Advisor: Prof. Dr.-Ing. Oliver Kretzschmar

Abstract

asdf

Contents

List of Figures

List of Tables

1 Introduction and Motivation

Generative Adversarial Networks (GANs) [GPAM+14] and their variants revolutionized the field of computer vision in the year of 2014, enabling advacements in multiple areas of generating data. From *Text to Image Synthesis* [RAY+16], *Image Translation* [IZZE18], *Super Resolution* [LTH+17], *Image Inpainting* [PKD+16], *Style Transfer* [WWR+23] to *Data Augmentation* [SK19], GANs have been used in a variety of applications.

The idea of using GANs for *Generative Data Augmentation* (GDA) has already been applied sucessfully, e.g.: in computer vision [JLR25], [BNI+23] or for creating music [JLY20]. Especially the former survey *A Comprehensive Survey of Image Generation Models Based on Deep Learning* has, along *Variational Auto Encoders* (VAEs), a dedicated focus on GANs. Despite these achievements, in practice, GANs suffer from several challenges, complicating the training and inference process:

- Mode Collapse
- Lack of inter-class Diversity
- Failure to Converge
- Vanishing Gradiants & Unstable Gradiants
- Imbalance between Generator- and Discriminator Model

This thesis investigates the potential of using GANs - specifically *Multi-Agent Diverse Generative Adversarial Networks* (MADGANs) [GKN+18] - for Generative Data Augmentation. MADGANs aim to aid the first two of the afore mentioned in particular: Mode Collapse and Loss of inter-class Diversity. They, along other modifications, "*propose to modify the objective function of the discriminator, in which, along with finding the real and the fake samples, the discriminator also has to correctly predict the generator that generated the given fake sample.*" [GKN+18]. The goal of this adjustment of the discriminator is, that the discriminator has to push the generators towards distinct identifiable modes. While various strategies have been proposed to addrese mode collapse and inter-class diversity MADGANs explicitly enfocre mode seperation by introduction of multiple generators and the adjusted discriminator objective. This makes them particularly promising for GDA, as diverse samples and clear distinction of modes is crucial for training robust classifiers. In their paper, they experimentally show, that their architectural adjustment of GANs is generally capable of giving providing assistance for the first two of the mentioned problems.

The experiments in this work are structured into three major parts.

Set 1: Training and Analysis of GANs   The first set trains and analyses GANs, explicitly MADGANs and *Conditional GANs* (cGANs). Here, the quality of the resulting images during training will be scored by the *Fréchet Inception Distance* (FID) [HRU+18] and the *Inception Score* (IS) [SGZ+16].

Set 2: Generating and Classifying Unlabeled Images   The second set uses the afore trained generative models to create images. Images without labels—images originating from MADGANs—will be classified using auxiliary classifiers trained with traditional data augmentation techniques.

Set 3: Training and Evaluating Classifiers   The third and most significant set of experiments trains classifiers using the generated data. For this, stratified classifiers with differing numbers of real and fake images are trained and evaluated on the respective validation set. Their classification performance will be assessed using standard metrics. [1]

All of the above described is executed on the following datasets:

- MNIST [LCB10]
- Fashion MNIST [XRV17]
- CIFAR10 [Kri09]

Aim of the Thesis   This thesis evaluates the effectiveness of Multi-Agent Diverse GANs for Generative Data Augmentation. First, the quality of their generated samples is compared to those produced by a Conditional GAN. Next, both sets of generated images are used to augment training datasets for classifiers, which are then assessed on their respective test sets. Classifiers trained on cGAN-augmented data and those trained with traditional augmentation techniques — such as flipping, rotation, and noise addition — serve as baselines for comparison. By doing so, this study examines the impact of MADGAN-based augmentation on classifier performance, highlighting its advantages and limitations relative to conventional methods and cGAN-based augmentation.

---

[1] The set of metrics used to assess the quality of the resulting classifiers is defined in chapter Experiments Setup 5.

2 Related Work

The effectiveness of deep learning models is instrinsically linked to the availability of large and diverse datasets for training. Models with deep and complex architectures require extensive exposure to a wide range of data to learn underlying patterns and generalize well to unseen instances. Insufficient training data can lead to a phenomena called *overfitting*, where a model becomes too specializied to the training data, failing to perfom accurately on previously unencountered data [Yin19].

To mitigate the problem of data scarcity and improve generalization capabilities of deep learning models, data augmentation techniques became indispensable. Data augmentation artificially expands the amounts and diversity of training datasets by creating modified versions of existing data or by generating entirely new instances.

Traditional Data Augmentation
Traditional data augmentation on images typically involves applying various tranformations to existing data. For image based data, augmentations can take a variaty of forms such as [2] :

Geometric Augmentation Modifying the shape, position and perspective: Rotation, Scaling, Flipping, Cropping, Shearing, Perspective Transform.

Photometric Augmentation Altering the pixel values while keeping the spatial structure: Brightness, Contrast, Hue Shift, Noise Injection, Blurring.

Noise-Corruption Augmentation Imitating real-world degradations and distorsions of cameras and sensors: Gaussion Noise, Speckle Noise, Salt-and-Pepper Noise.

The sucess of the above mentioned augmentation techniques is established in many papers [PW17], [KSH12], [Yin19], [SK19], [WZZ+13].

Generative Data Augmentation using Deep Convolutional GANs
The basic GAN framework introduced by Goodfellow and collegues offers a high degree of flexibility and can be adapted for specific augmentation tasks. It can be applied to generate music [DHYY17], speech [LMWN22], text [YZWY17], images [GPAM+14] or other instances of data, e.g. tabular data [XSCIV19].

Especially for image data, *Deep Convolutional GANs* (DCGANs) represent a significant advacement in applying GANs to iamge data augmentation [HFM22]. Their architecture specifically utilizes *Convolutional Neural Network* (CNNs) [LBD+89] in both, the generator and the discriminator. The use of CNNs allows DCGANs to learn hierarchical features from the input images and capture the spatial relationship and structure inherent in images. This leads to the generation of more realistic and coherent synthetic images. A study from Zhau et al. [ZCWD23] applied DCGANs, along their

---

[2]More traditional data augmentation techniques exists, such as Occlusion-Based, Composition-Based, Domain-Specific or Adversarial Augmentation. For the purpose of this work, soley the afore mentioned are discussed in greater detail.

adjusted version of those on multiple dataset, including *Fashion MNIST* and *Cifar10*. With their experimental setup, they achieved consistent significant improvements over multiple datasets using the DCGAN-architechture, compared to their baseline.

Inherently in the vanilla version of GANs or the DCGANs realization of using convoltional layers, the generators role is soley to learn the underlying data distribution of the training samples and produce instances of close resemblance to instances from the training data. This however results in unlabeled samples, not to be beneficial to expand data for a supervised classification task.

Generative Data Augmentation using Conditional GANs

The introduction of *Conditional Generative Adversarial Networks* (cGANs) [MO14] allows to condition the generative process by additional information, such as class labels or other modalities. The conditioning acts on both the generator and the discriminator, which means that both models have access to the same conditional information. The generator combines the random vector input and the conditioning information into a joint hidden representation. The discriminator, on the other hand, evaluates the created data from generator, given context of the conditining information, i.e. the class label passed. This approach enables the generator to create data that adheres to specific inputs, like creating specific digits from the MNIST dataset 1. Multiple papers were able to utilize the advantages of cGANs, to e.g. unify class distributions for a stratified classifier training or generatively increase the number of images and augmenting the training data[JPB22][ZCWD23][RCF25][WM21].

Generative Data Augmentation using MADGANs

Regardless of the mentioned sucesses using GANs (DCGANs or cGANs) for GDA 2 2, GANs in general have proven to be notoriously hard to train. "*Among them, mode collapse stands out as one of the most daunting ones.*" [DCLK20], which limits the GANs ability to generate diverse samples, able to be assigned to all classes trained on. Another prominent problem with GANs is the Lack of inter-class diversity between generated samples.

MADGANs [GKN$^+$18] emphasis on diversity, achieved through its multi-agent architecture and the modified discriminator objective function, directly addresses these limitations. By encouraging multiple generators to specialize in different modes of the data distribution, MADGAN aims to generate a more comprehensive and diverse set of synthetic samples compared to traditional GANs and potentially other generative data augmentation techniques that might be susceptible to mode collapse. The ability of MADGAN to disentangle different modalities i.e. classes, as suggested by experiments involving diverse-class datasets, indicates its potential to generate augmented data that effectively covers both intra-class and inter-class variations. This comprehensive coverage is crucial for training robust image classifiers that can generalize well to a wide range of real-world scenarios.

3 Theoretical Background

This chapter serves as a reference for the theoretical background necessary to understand the insights gained in the following experimental chapters. Section 3.1 discusses classification models used to train on the extended dataset resulting from the generative augmentation process. In it, *Neural Networks* (NNs) for image classification are introduced and the baselines for later comparisons are examined. Sections 3.2 and 3.3 establish the foundation for data augmentation and generative data augmentation. Following sections 3.4, 3.5, 3.6, and 3.7 provide theoretical knowledge necessary to understand the GAN architectures and their differences. The narrative follows their increasing complexity, starting from vanilla GANs, moving through deep convolutional GANs and conditional GANs, before diving into the background of multi-agent diverse GANs. The final section (3.8) explains the theory behind the Inception Score and Fréchet Inception Distance, concluding with an examination of the state-of-the-art *InceptionV3* model used to compute them.

3.1 Image Classification Models

3.1.1 Neural Networks for Classification

Convolutional Neural Networks (CNNs) have become the dominant architecture for image classification tasks due to their inherent ability to automatically learn hierarchical features from raw pixel data. At their core, CNNs are build up by a sequence of convolutional-, pooling- and fully connected layers to extract hierarchical features, and funneling the information, typically into the N classes defined by the training data. Convolutional layers employ learnable filters to detect local patterns in the two-dimensional information - two dimensional in the case of images specifically. Pooling layers reduce the spatial dimensions to small translations. Fully connected layers then map the extracted information into class probabilities, utilizing the *Softmax* activation function. The afore mentioned layers are discussed in greater detail, in the following subsections.

Convolutional Layers These layers compute the output from the local regions of the input. Let $(r \times c)$ be the two-dimensional input, e.g., a grayscale image, where r represents the X-coordinate and c the Y-coordinate of a pixel. Thus, $r \cdot c$ denotes the size of the image. Furthermore, let $(a \times b)$ be a filter with kernel size $a \cdot b$, where the filter is smaller than the input. This filter is moved from the top-left to the bottom-right over the input.

In each iteration, the dot product between the respective coefficients of the input region and the coefficients of the filter is computed. This dot product is then processed by the activation function g, which determines how much of the feature is present. If

the activation function is ReLU, for example, only positive values are retained, meaning negative responses are set to zero. The result is written to the subsequent layer.

The stride determines how far the filter is moved after each operation. For a stride of $s = 1$, the filter can be placed in $(r - a + 1)$ positions along the height and $(c - b + 1)$ positions along the width, leading to an output size of $(r - a + 1) \times (c - b + 1)$. In general, the output size in the two-dimensional case is given by:

$$\left(\frac{r - a + s}{s}\right) \times \left(\frac{c - b + s}{s}\right).$$

An image of this process can be found in Figure X in the Appendix (7). This image shows an input of size $[10 \times 10]$ and a filter of size $[3 \times 3]$. With a stride of $s = 1$, the resulting layer has a size of $[8 \times 8]$ (computed as $(10 - 3 + 1) \times (10 - 3 + 1)$).

The stride of the filter can also be greater than 1. Additionally, there is the option to apply padding to the image. There are different ways to implement padding. When padding of size p is applied, the output size for a square input and filter is calculated as follows:

$$\textbf{Output size} = \left\lfloor \frac{r - a + 2p}{s} \right\rfloor + 1$$

where $\lfloor \cdot \rfloor$ denotes the floor function, which ensures that the output size is an integer.

Pooling Layers A pooling layer compresses the data along the spatial axes to reduce its dimensionality. Similar to the convolutional layer, a pooling layer uses a filter that moves by the stride value. However, instead of summing the covered elements, the pooling operation applies the Max operator, selecting the maximum value within the filter's region.

For example, starting with an input of size $[32 \times 32 \times 10]$, applying a pooling operation with a $[2 \times 2]$ filter and a stride of 2 results in an output size of $[16 \times 16 \times 10]$. This operation reduces the spatial dimensions by half while keeping the depth unchanged.

Max pooling helps retain the most important features, providing some invariance to small translations or distortions in the input, which is crucial for tasks like object recognition in convolutional neural networks (CNNs).

Fully-Connected Layers This layer computes the scores for the respective classes. In the case of ten classes, the result is a volume of size $[1 \times 1 \times 10]^3$. By this stage, all spatial information has been transformed, leaving a quasi-one-dimensional vector containing the ten class scores for the CIFAR-10 dataset.

In a fully-connected layer, each input is connected to each output, meaning every neuron in the previous layer is connected to each neuron in the fully-connected layer.

---

[3]Typically, the output from the layer before the Fully-Connected one is "flattened" into a one-dimensional vector, preserving all information but removing spatial structure. For example, a $[2 \times 2]$ layer would become a vector of size $[1 \times 4]$.

The output is the weighted sum of all inputs, followed by an activation function, leading to the final classification scores that represent the likelihood of the input belonging to each class. The spatial dimensions are collapsed into a single vector of class scores, which are then used for classification.

Typical Activation Functions for CNNs

- **ReLU (Rectified Linear Unit)**:

$$g(x) = \max(0, x)$$

  ReLU is the most widely used activation function in CNNs. It introduces non-linearity while maintaining efficiency by outputting zero for negative values and passing positive values unchanged.

- **Leaky ReLU**:

$$f(x) = \{\, 0 \,, \mathit{if}\ x < 0x, \mathit{otherwise} \tag{1}$$

  A variant of ReLU, Leaky ReLU allows small negative values to flow through, addressing the "dying ReLU" problem where neurons can become inactive.

- **Sigmoid (Logistic)**:

$$g(x) = \frac{1}{1 + e^{-x}}$$

  The sigmoid function squashes values between 0 and 1, commonly used for binary classification tasks. However, it can suffer from vanishing gradients for very large or small inputs.

- **Tanh (Hyperbolic Tangent)**:

$$g(x) = \frac{2}{1 + e^{-2x}} - 1$$

  Tanh outputs values between -1 and 1 and is similar to the sigmoid but with a wider output range, making it more effective in many scenarios compared to sigmoid.

- **Softmax**:

$$g(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

  Softmax is typically used in the output layer of CNNs for multi-class classification. It converts logits into probabilities, ensuring that the sum of the outputs is 1.

3.1.2 Classification Models for augmented Training

3.2 Data Augmentation

3.3 Generative Data Augmentation

3.4 Generative Adversarial Network

3.4.1 Challenges in GAN Training

3.4.2 Mode Collapse

3.4.3 Training Instability

3.4.4 Evaluation Difficulties

3.5 Deep Convolutional Generative Adversarial Network

3.6 Conditional Generative Adversarial Network

3.7 Multi-Agent Diverse Generative Adversarial Network

3.8 Image Scores

3.8.1 Inception Score

3.8.2 Fréchet Inception Distance

3.8.3 InceptionV3 for Image Evaluation

4 Preliminary Remarks

5 Experiments Setup

6 Experiments Results

Motivation

7 Outlook

repair networks [TFNL22]

8 Conclusion

Data generated by a GAN, may it be a cGAN or a MADGAN, may not fully capturethe distribution characteristics of its training data. Though, generated images do visually appear realistic, they may only partially reflect the statistical characteristics of the original data. This can lead to synthetic images that appear *good* to a human inspector, but may contain amounts of noice that may interfere with a subsequent classifier.

from an information theoratival standpoint, a generative model G trained on data X, distilling knowledge into a classifier C should not offer more information that what was already present in X.

Future research could focus on directly evaluating the impact of using MAD-GAN generated samples for augmenting various image classification datasets across different domains and comparing the resulting performance gains with those achieved by traditional and other generative augmentation techniques. Exploring methods to exert more control over the types of variations generated by MAD-GAN to specifically target weaknesses or improve the robustness of classifiers against particular types of noise or adversarial attacks would also be a valuable direction. Additionally, investigating the computational efficiency and scalability of training MAD-GAN for very large and complex datasets in the context of practical data augmentation pipelines would be crucial for its wider adoption. Finally, exploring the applicability of the MAD-GAN framework to generate diverse augmented data for other computer vision tasks beyond image classification, as well as for other data modalities such as natural language processing or audio processing, could further broaden its impact. The work by Ghosh et al. on Multi-Agent Diverse GANs represents a promising step towards leveraging the power of generative models for more effective and robust data augmentation in image classification and beyond.

List of References

[BNI+23]   BISWAS, Angona ; NASIM, MD Abdullah A. ; IMRAN, Al ; SEJUTY,
           Anika T. ; FAIROOZ, Fabliha ; PUPPALA, Sai ; TALUKDER, Sajedul: *Gen-
           erative Adversarial Networks for Data Augmentation.* https://arxiv.
           org/abs/2306.02019. Version: 2023

[DCLK20]   DURALL, Ricard ; CHATZIMICHAILIDIS, Avraam ; LABUS, Peter ; KE-
           UPER, Janis: *Combating Mode Collapse in GAN training: An Empiri-
           cal Analysis using Hessian Eigenvalues.* https://arxiv.org/abs/2012.
           09673. Version: 2020

[DHYY17]   DONG, Hao-Wen ; HSIAO, Wen-Yi ; YANG, Li-Chia ; YANG, Yi-Hsuan:
           *MuseGAN: Multi-track Sequential Generative Adversarial Networks for
           Symbolic Music Generation and Accompaniment.* https://arxiv.org/
           abs/1709.06298. Version: 2017

[GKN+18]   GHOSH, Arnab ; KULHARIA, Viveka ; NAMBOODIRI, Vinay ; TORR,
           Philip H. S. ; DOKANIA, Puneet K.: *Multi-Agent Diverse Generative Ad-
           versarial Networks.* https://arxiv.org/abs/1704.02906. Version: 2018

[GPAM+14] GOODFELLOW, Ian J. ; POUGET-ABADIE, Jean ; MIRZA, Mehdi ; XU,
           Bing ; WARDE-FARLEY, David ; OZAIR, Sherjil ; COURVILLE, Aaron ;
           BENGIO, Yoshua: *Generative Adversarial Networks.* https://arxiv.
           org/abs/1406.2661. Version: 2014

[HFM22]   HUANG, Y. ; FIELDS, K. G. ; MA, Y.: A Tutorial on Generative Adver-
           sarial Networks with Application to Classification of Imbalanced Data.
           In: *Statistical Analysis and Data Mining* 15 (2022), Nr. 5, S. 543–552.
           http://dx.doi.org/10.1002/sam.11570. – DOI 10.1002/sam.11570

[HRU+18]   HEUSEL, Martin ; RAMSAUER, Hubert ; UNTERTHINER, Thomas ;
           NESSLER, Bernhard ; HOCHREITER, Sepp: *GANs Trained by a Two
           Time-Scale Update Rule Converge to a Local Nash Equilibrium.* https:
           //arxiv.org/abs/1706.08500. Version: 2018

[IZZE18]   ISOLA, Phillip ; ZHU, Jun-Yan ; ZHOU, Tinghui ; EFROS, Alexei A.:
           Image-to-Image Translation with Conditional Adversarial Networks.
           https://arxiv.org/abs/1611.07004. Version: 2018

[JLR25]   JUN LI, Wei Z. Chenyang Zhang Z. Chenyang Zhang ; REN, Yawei: A
           Comprehensive Survey of Image Generation Models Based on Deep Learn-
           ing. In: *Annals of Data Science* 12 (2025), February, 141–170. http:
           //dx.doi.org/10.1007/s40745-024-00544-1. – DOI 10.1007/s40745–
           024–00544–1

[JLY20]   JI, Shulei ; LUO, Jing ; YANG, Xinyu: *A Comprehensive Survey on
           Deep Music Generation: Multi-level Representations, Algorithms, Eval-
           uations, and Future Directions.* https://arxiv.org/abs/2011.06801.
           Version: 2020

[JPB22]   JEONG, Jason ; PATEL, B. ; BANERJEE, I.: GAN augmentation for mul-
           ticlass image classification using hemorrhage detection as a case-study.

In: *Journal of Medical Imaging (Bellingham, Wash.)* 9 (2022), Nr. 3, S. 035504. `http://dx.doi.org/10.1117/1.JMI.9.3.035504`. – DOI 10.1117/1.JMI.9.3.035504

[Kri09]   KRIZHEVSKY, Alex: *Learning Multiple Layers of Features from Tiny Images.* Technical Report, University of Toronto. `https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf`. Version: 2009

[KSH12]   KRIZHEVSKY, Alex ; SUTSKEVER, Ilya ; HINTON, Geoffrey E. ; PEREIRA, F. (Hrsg.) ; BURGES, C.J. (Hrsg.) ; BOTTOU, L. (Hrsg.) ; WEINBERGER, K.Q. (Hrsg.): *ImageNet Classification with Deep Convolutional Neural Networks.* `https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf`. Version: 2012

[LBD+89]   LECUN, Y. ; BOSER, B. ; DENKER, J. S. ; HENDERSON, D. ; HOWARD, R. E. ; HUBBARD, W. ; JACKEL, L. D.: Backpropagation Applied to Handwritten Zip Code Recognition. In: *Neural Computation* 1 (1989), Nr. 4, S. 541–551. `http://dx.doi.org/10.1162/neco.1989.1.4.541`. – DOI 10.1162/neco.1989.1.4.541

[LCB10]   LECUN, Yann ; CORTES, Corinna ; BURGES, CJ: MNIST handwritten digit database. In: *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist* 2 (2010)

[LMWN22]   LI, Xiaomin ; METSIS, Vangelis ; WANG, Huangyingrui ; NGU, Anne Hee H.: *TTS-GAN: A Transformer-based Time-Series Generative Adversarial Network.* `https://arxiv.org/abs/2202.02691`. Version: 2022

[LTH+17]   LEDIG, Christian ; THEIS, Lucas ; HUSZAR, Ferenc ; CABALLERO, Jose ; CUNNINGHAM, Andrew ; ACOSTA, Alejandro ; AITKEN, Andrew ; TEJANI, Alykhan ; TOTZ, Johannes ; WANG, Zehan ; SHI, Wenzhe: *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network.* `https://arxiv.org/abs/1609.04802`. Version: 2017

[MO14]   MIRZA, Mehdi ; OSINDERO, Simon: *Conditional Generative Adversarial Nets.* `https://arxiv.org/abs/1411.1784`. Version: 2014

[PKD+16]   PATHAK, Deepak ; KRAHENBUHL, Philipp ; DONAHUE, Jeff ; DARRELL, Trevor ; EFROS, Alexei A.: *Context Encoders: Feature Learning by Inpainting.* `https://arxiv.org/abs/1604.07379`. Version: 2016

[PW17]   PEREZ, Luis ; WANG, Jason: *The Effectiveness of Data Augmentation in Image Classification using Deep Learning.* `https://arxiv.org/abs/1712.04621`. Version: 2017

[RAY+16]   REED, Scott ; AKATA, Zeynep ; YAN, Xinchen ; LOGESWARAN, Lajanugen ; SCHIELE, Bernt ; LEE, Honglak: *Generative Adversarial Text to Image Synthesis.* `https://arxiv.org/abs/1605.05396`. Version: 2016. – arXiv:1605.05396

[RCF25]   RIBAS, Lucas C. ; CASACA, Wallace ; FARES, Ricardo T.: Conditional
          Generative Adversarial Networks and Deep Learning Data Augmenta-
          tion: A Multi-Perspective Data-Driven Survey Across Multiple Appli-
          cation Fields and Classification Architectures. In: *AI* 6 (2025), Nr. 2.
          `http://dx.doi.org/10.3390/ai6020032`. – DOI 10.3390/ai6020032. –
          ISSN 2673–2688

[SGZ+16] SALIMANS, Tim ; GOODFELLOW, Ian ; ZAREMBA, Wojciech ; CHEUNG,
          Vicki ; RADFORD, Alec ; CHEN, Xi: *Improved Techniques for Training
          GANs*. `https://arxiv.org/abs/1606.03498`. Version: 2016

[SK19]   SHORTEN, Connor ; KHOSHGOFTAAR, Taghi M.: A survey on Image Data
          Augmentation for Deep Learning. In: *Journal of Big Data* 6 (2019), July,
          Nr. 1, 60. `http://dx.doi.org/10.1186/s40537-019-0197-0`. – DOI
          10.1186/s40537–019–0197–0. – ISSN 2196–1115

[TFNL22] TANNO, Ryutaro ; F.PRADIER, Melanie ; NORI, Aditya ; LI, Yingzhen:
          Repairing Neural Networks by Leaving the Right Past Behind. (2022),
          07, S. 19. `http://dx.doi.org/10.48550/arXiv.2207.04806`. – DOI
          10.48550/arXiv.2207.04806

[WM21]   WICKRAMARATNE, S. D. ; MAHMUD, M. S.: Conditional-GAN Based
          Data Augmentation for Deep Learning Task Classifier Improvement Us-
          ing fNIRS Data. In: *Frontiers in Big Data* 4 (2021), S. 659146.
          `http://dx.doi.org/10.3389/fdata.2021.659146`. – DOI 10.3389/f-
          data.2021.659146

[WWR+23] WANG, Hanyu ; WU, Pengxiang ; ROSA, Kevin D. ; WANG, Chen ;
          SHRIVASTAVA, Abhinav: *Multimodality-guided Image Style Transfer us-
          ing Cross-modal GAN Inversion*. `https://arxiv.org/abs/2312.01671`.
          Version: 2023

[WZZ+13] WAN, Li ; ZEILER, Matthew ; ZHANG, Sixin ; LE CUN, Yann ;
          FERGUS, Rob ; DASGUPTA, Sanjoy (Hrsg.) ; MCALLESTER, David
          (Hrsg.): *Regularization of Neural Networks using DropConnect*. `https:`
          `//proceedings.mlr.press/v28/wan13.html`. Version: 17–19 Jun 2013
          (Proceedings of Machine Learning Research)

[XRV17]   XIAO, Han ; RASUL, Kashif ; VOLLGRAF, Roland: *Fashion-MNIST:
          a Novel Image Dataset for Benchmarking Machine Learning Algorithms*.
          2017

[XSCIV19] XU, Lei ; SKOULARIDOU, Maria ; CUESTA-INFANTE, Alfredo ; VEERA-
          MACHANENI, Kalyan: *Modeling Tabular data using Conditional GAN*.
          `https://arxiv.org/abs/1907.00503`. Version: 2019

[Yin19]   YING, Xue: An Overview of Overfitting and its Solutions. In: *Journal
          of Physics: Conference Series* 1168 (2019), feb, Nr. 2, 022022. `http://`
          `dx.doi.org/10.1088/1742-6596/1168/2/022022`. – DOI 10.1088/1742–
          6596/1168/2/022022

[YZWY17] YU, Lantao ; ZHANG, Weinan ; WANG, Jun ; YU, Yong: *SeqGAN:*
Sequence Generative Adversarial Nets with Policy Gradient. `https://`
`arxiv.org/abs/1609.05473`. Version: 2017

[ZCWD23] ZHAO, Gaochang ; CAI, Zhao ; WANG, Xin ; DANG, Xiaohu: GAN Data
Augmentation Methods in Rock Classification. In: *Applied Sciences* 13
(2023), Nr. 9, S. 5316. `http://dx.doi.org/10.3390/app13095316`. –
DOI 10.3390/app13095316

Appendix

Declaration of Academic Integrity

Generative Data Augmentation

Multi-Agent Diverse Generative Adversarial Networks for Generative Data Augmentation.

I hereby declare that I have written this thesis independently. I have properly cited all passages that are taken verbatim or in essence from published or unpublished works of others. All sources and aids used in the preparation of this thesis have been fully acknowledged. Furthermore, this thesis has not been submitted, in whole or in substantial part, to any other examination authority for academic credit.

Signiture :   *Place, Date* :